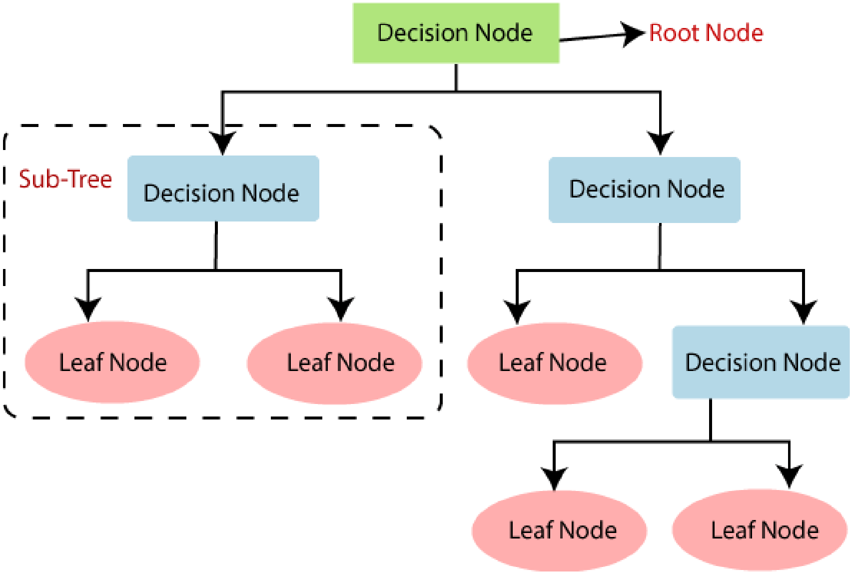
Decission Tree

The major difference between a classification tree and a regression tree is the nature of the variable to be predicted. In a regression tree, the variable is continuous rather than categorical.

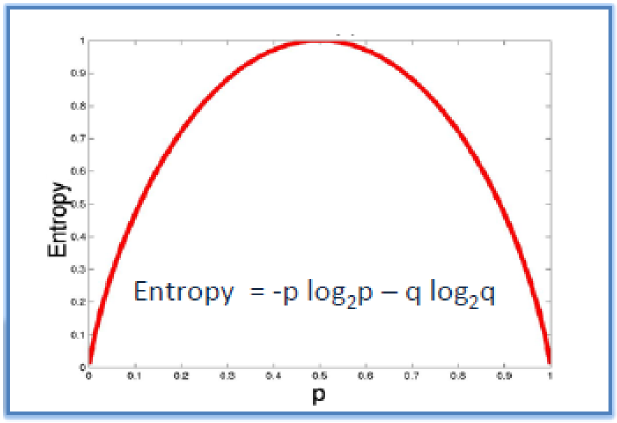
The decision tree is a type of supervised learning algorithm (having a predefined target variable) that is mostly used in classification problems.
It works for both categorical and continuous input and output variable. In this technique, we split the population or sample into two or more homogeneous sets (or sub-populations) based on most significant splitter / differentiator) in input variables.



- **Root Node:** It represents entire population or sample and this further gets divided into two or more homogeneous sets.
- **Splitting:** It is a process of dividing a node into two or more sub-nodes.
- **Decision Node:** When a sub-node splits into further sub-nodes, then it is called decision node.
- **Leaf/Terminal Node:** Nodes do not split is called Leaf or Terminal node.
- **Pruning:** When we remove sub-nodes of a decision node, this process is called pruning. You can say opposite process of splitting.
- **Branch / Sub-Tree:** A sub section of entire tree is called branch or sub-tree.
- **Parent and Child Node:** A node, which is divided into sub-nodes is called parent node of sub-nodes whereas sub-nodes are the child of parent node.

	outlook	temp	humidity	wind	decision
0	2	1	0	1	0
1	2	1	0	0	0
2	0	1	0	1	1
3	1	2	0	1	1
4	1	0	1	1	1
5	1	0	1	0	0
6	0	0	1	0	1
7	2	2	0	1	0
8	2	0	1	1	1
9	1	2	1	1	1
10	2	2	1	0	1
11	0	2	0	0	1
12	0	1	1	1	1
13	1	2	0	0	0

Entropy formula



Entropy $= -0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$

a) Entropy using the frequency table of one attribute:

$$E(S) = \sum_{i=1}^{c} -p_i \log_2 p_i$$

Play Golf	
Yes	No
9	5

Entropy(PlayGolf) = Entropy (5,9)

= Entropy (0.36, 0.64)

= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)

= 0.94

b) Entropy using the frequency table of two attributes:

$$E(T, X) = \sum_{c \in X} P(c)E(c)$$

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

E(PlayGolf, Outlook) = **P**(Sunny)***E**(3,2) + **P**(Overcast)***E**(4,0) + **P**(Rainy)***E**(2,3)

= (5/14)*0.971 + (4/14)*0.0 + (5/14)*0.971

= 0.693

Step 1: Calculate entropy of the target.

Entropy(PlayGolf) = Entropy (5,9)

= Entropy (0.36, 0.64)

= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)

= 0.94

Step 2: The dataset is then split on the different attributes. The entropy for each branch is calculated. The resulting entropy is subtracted from the entropy before the split. The result is the Information Gain.

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1
Gain = 0.029			

		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1
Gain = 0.152			

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3
Gain = 0.048			

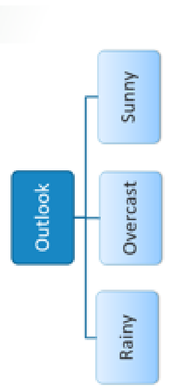
$$Gain(T, X) = Entropy(T) - Entropy(T, X)$$

G(PlayGolf, Outlook) = **E**(PlayGolf) – **E**(PlayGolf, Outlook)

= 0.940 – 0.693 = 0.247

Step 3: Choose attribute with the largest information gain as the decision node, divide the dataset by its branches and repeat the same process on every branch.

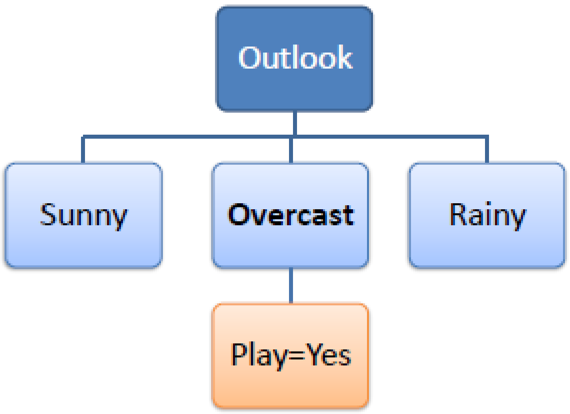
		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			



Outlook	Temp	Humidity	Windy	Play Golf
Sunny	Mild	High	FALSE	Yes
Sunny	Cool	Normal	FALSE	Yes
Sunny	Cool	Normal	TRUE	No
Sunny	Mild	Normal	FALSE	Yes
Sunny	Mild	High	TRUE	No
Overcast	Hot	High	FALSE	Yes
Overcast	Cool	Normal	TRUE	Yes
Overcast	Mild	High	TRUE	Yes
Overcast	Hot	Normal	FALSE	Yes
Rainy	Hot	High	FALSE	No
Rainy	Hot	High	TRUE	No
Rainy	Mild	High	FALSE	No
Rainy	Cool	Normal	FALSE	Yes
Rainy	Mild	Normal	TRUE	Yes

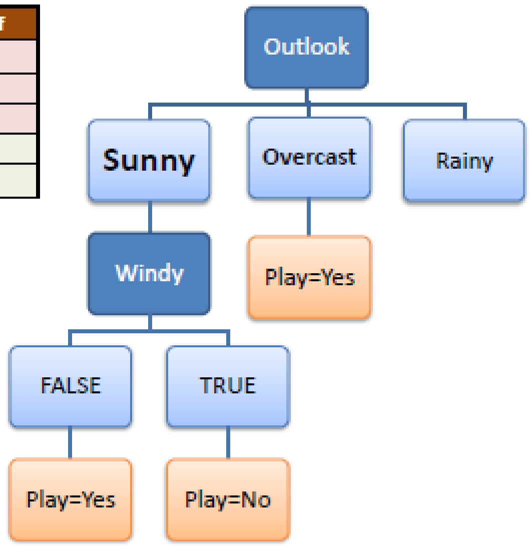
Step 4a: A branch with entropy of 0 is a leaf node.

Temp	Humidity	Windy	Play Golf
Hot	High	FALSE	Yes
Cool	Normal	TRUE	Yes
Mild	High	TRUE	Yes
Hot	Normal	FALSE	Yes



Step 4b: A branch with entropy more than 0 needs further splitting.

Temp	Humidity	Windy	Play Golf
Mild	High	FALSE	Yes
Cool	Normal	FALSE	Yes
Mild	Normal	FALSE	Yes
Cool	Normal	TRUE	No
Mild	High	TRUE	No



Step 5: The ID3 algorithm is run recursively on the non-leaf branches, until all data is classified.

Decision Trees - Issues

- Working with continuous attributes (binning)
- Avoiding overfitting
- Super Attributes (attributes with many unique values)

- Working with missing values

Impurity Criterion

Gini Index

$$I_G = 1 - \sum_{j=1}^{c} p_j^2$$

p_j: proportion of the samples that belongs to class c for a particular node

Entropy

$$I_H = - \sum_{j=1}^{c} p_j log_2(p_j)$$

p_j: proportion of the samples that belongs to class c for a particular node.

*This is the the definition of entropy for all non-empty classes ($p \neq 0$). The entropy is 0 if all samples at a node belong to the same class.

Gini Index

So as the first step we will find the root node of our decision tree. For that Calculate the Gini index of the class variable

- Gini(S) = 1 - $[(9/14)^2 + (5/14)^2]$ = 0.4591

- First, consider case of Outlook

		play		
		yes	no	total
	sunny	3	2	5
Outlook	overcast	4	0	4
	rainy	2	3	5
				14

Gini(S, outlook) = (5/14)gini(3,2) + (4/14)*gini(4,0)+ (5/14)*gini(2,3) => $(5/14)(1 - (3/5)^2 - (2/5)^2)$ + (4/14)*0 + $(5/14)(1 - (2/5)^2 - (3/5)^2)$=> 0.171+0+0.171 => 0.342
Find for all columns

- Choose one that has lower Gini gain. Gini gain is lower for outlook. So we can choose it as our root node.

Than repeat the same steps:-

Decision Tree Regressor

In [2]:
```
import pandas as pd
```

In [3]:
```
df = pd.read_csv('50_Startups.csv')
df.head()
```

Out[3]:

	R&D Spend	Administration	Marketing Spend	State	Profit
0	165349.20	136897.80	471784.10	New York	192261.83
1	162597.70	151377.59	443898.53	California	191792.06
2	153441.51	101145.55	407934.54	Florida	191050.39
3	144372.41	118671.85	383199.62	New York	182901.99
4	142107.34	91391.77	366168.42	Florida	166187.94

In [4]:
```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
```

In [5]:
```
df['State']=le.fit_transform(df['State'])
```

In [6]:
```
df.head(2)
```

Out[6]:

	R&D Spend	Administration	Marketing Spend	State	Profit
0	165349.2	136897.80	471784.10	2	192261.83
1	162597.7	151377.59	443898.53	0	191792.06

In [7]:
```
x=df.iloc[::-1]
y=df.iloc[:,-1]
```

In [8]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20)
```

In [9]:
```
from sklearn.tree import DecisionTreeRegressor
model=DecisionTreeRegressor()
```

In [10]:
```
model.fit(x_train,y_train)
```
Out[10]:
```
DecisionTreeRegressor()
```

In [11]:
```
model.predict(x_test)
```
Out[11]:
```
array([ 96712.8 ,  81005.76, 110352.25,  71498.49,  99937.59, 134307.35,
        71498.49,  71498.49, 103282.38,  81005.76])
```

Decision Tree Classifer

In [13]:
```
import pandas as pd
data = pd.read_csv('gini_index.csv')
data.head()
```
Out[13]:

	outlook	temp	humidity	wind	decision
0	sunny	hot	high	weak	no
1	sunny	hot	high	strong	no
2	overcast	hot	high	weak	yes
3	rain	mild	high	weak	yes
4	rain	cool	normal	weak	yes

In [14]:
```
data.shape
```
Out[14]:
```
(14, 5)
```

In [15]:
```
data.columns
```
Out[15]:
```
Index(['outlook', 'temp', 'humidity', 'wind', 'decision'], dtype='object')
```

In [16]:
```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
```

In [15]:
```
categorical_columns = ['outlook', 'temp', 'humidity', 'wind', 'decision']
# I would recommend using columns names here if you're using pandas. If you're using numpy then stick with range(n) instead

for column in categorical_columns:
    le = LabelEncoder()
    data[column] = le.fit_transform(data[column])
# if numpy instead of pandas use X[:, column] instead
```

In [16]:
```
data
```
Out[16]:

	outlook	temp	humidity	wind	decision
0	2	1	0	1	0
1	2	1	0	0	0
2	0	1	0	1	1
3	1	2	0	1	1
4	1	0	1	1	1
5	1	0	1	0	0
6	0	0	1	0	1
7	2	2	0	1	0
8	2	0	1	1	1
9	1	2	1	1	1
10	2	2	1	0	1
11	0	2	0	0	1
12	0	1	1	1	1
13	1	2	0	0	0

In [17]:
```
x=data.iloc[:,:-1]
y=data.iloc[:,-1]
```

In [18]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20)
```

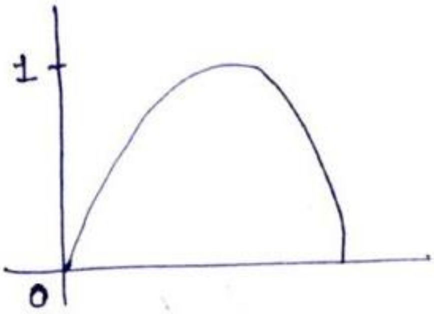
In [19]:
```
from sklearn.tree import DecisionTreeClassifier
dtc=DecisionTreeClassifier()
```

In [20]:
```
dtc.fit(x_train,y_train)
```
Out[20]:
```
DecisionTreeClassifier()
```

In [21]:
```
dtc.predict(x_test)
```
Out[21]:
```
array([0, 0, 1])
```

Decision Tree

Data set - gini_index CSV



Entropy :

Entropy lies b/w '0' to '1'.

$$Gain(S, A) = Entropy(S) - \sum Entropy(A)$$

$$Entropy(S) = -P_1(\log_2 P_1) - P_2(\log_2 P_2) - \ldots\ldots - P_n(\log_2 P_n)$$

$$E(S) = -P(\checkmark) \times \log_2 P(\checkmark) - P(x)\log_2 P(x)$$

$$\Rightarrow S = 14\ [9+\ 5-]$$

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

$$\boxed{E(S) \Rightarrow 0.94029} \leftarrow \text{Entropy of all samples}$$

- Now find the entropy of each sample.

$$Entropy(v) = -P_{\oplus}\log_2(P_+) - P_{\ominus}\log_2 P_{\ominus}$$

Sunny = 5 = [2+ 3-]
Overcast = 4 = [4+ 0-]
Rainy = 5 = [3+ 2-]

$$E_{sunny} = -\left(\frac{2}{5}\right)\log\left(\frac{2}{5}\right) - \left(\frac{3}{5}\right)\log\left(\frac{3}{5}\right)$$
$$= 0.97095$$

$E_{overcast} = 0$

$E_{rainy} = 0.97015$

<u>Now, we find the Information Gain</u>

$$Gain(S, outlook) = Entropy(S) - \left[\frac{\sum P_i N_i}{P+N} \times E(v) \cdots\right]$$

$$G(s,o) = 0.94029 - \left[\frac{5}{14} \times 0.97095 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0.97095\right]$$

$$\boxed{Gain(S, outlook) = 0.24675}$$

Now, we calculate the Gain of Temperature.

$Hot = 4 = [2+\ 2-] \implies E = 1$

$Mid = 6 = [4+\ 2-] \implies E = 0.91829$

$Cool = 4 = [3+\ 1-] \implies E = 0.811278$

$$Gain(S, Temp) = 0.94029 - \left[\frac{4}{14} \times 1 + \frac{6}{14} \times 0.91829 + \frac{4}{16} \times 0.\right.$$

$$\boxed{Gain(S, Temp) = 0.02922}$$

Now, we calculate the Gain of humidity.

$$\text{High} = 7 = [3+ \; 4-] \Rightarrow E =$$
$$\text{Normal} = 7 = [6+ \; 1-] \Rightarrow E =$$

$$\text{Gain}(S, \text{humidity}) = 0.94029 - [$$

$$\boxed{\text{Gain}(S, \text{humidity}) = 0.02522}$$

Now, we calculate the Gain of Wind:

$$\text{Weak} = 8 \; [6+ \; 2-] \Rightarrow E =$$
$$\text{Strong} = 6 \; [3+ \; 3-] \Rightarrow E =$$

$$\text{Gain}(S, \text{Wind}) = 0.94029 - [$$

$$\boxed{\text{Gain}(S, \text{Wind}) = 0.093061}$$

Which value is greater, than will mak a root node

Outlook

Sunny — Overcast — Rainy

Now, we find the Entropy for Sunny -

$Sample_{Sunny} = 5 = [2+ \ 3-]$

$Entropy_{(Sunny)} = 0.97095$

Find the gain of all left columns like, Temp, windek

$Gain(Sunny, Temp) = ?$

$Hot = 2 = [0+ \ 2-] \Rightarrow E = 0$
$Mid = 2 = [1- \ 1+] \Rightarrow E = 1$
$Cool = 1 = [1+ \ 0-] \Rightarrow E = 0$

$$Gain(Sunny, Temp) = 0.97095 - \left[\frac{2}{5} \times 0 + \frac{2}{5} \times 1 + \frac{1}{5} \times 0\right]$$

$$\boxed{Gain(S, Temp) = 0.57095}$$

Now, we calculate gain, sunny for humidity.

$High = 3 = [0+ \ 3-] \Rightarrow E = 0$
$Normal = 2 = [2+ \ 0-] \Rightarrow E = 0$

$$Gain(S, hum) = 0.97095 - \left(\frac{3}{5} \times 0 + \frac{2}{5} \times 0\right)$$
$$= 0.97095$$

- This is the higher entropy for Outlook.

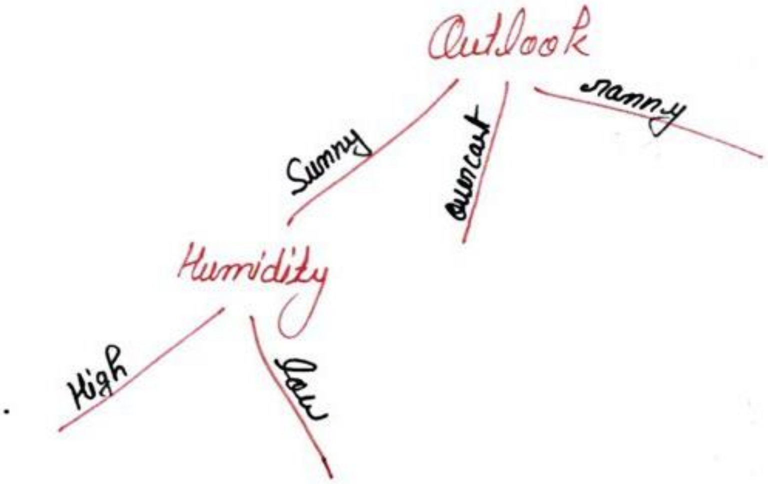
Now, we calculate gain, sunny for wind :

Weak = 3 [1+ 2+] => E = 0.918295

Strong = 2 [1+ 1-] => E = 1

$\boxed{\text{Gain}(S, \text{wind}) = 0.019973}$

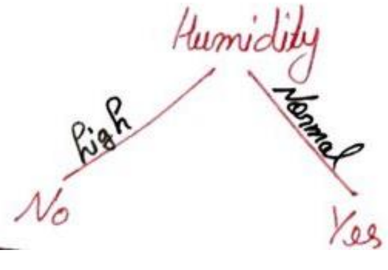
Than I will draw the next node of the tree by Check the higher Entropy value.



Now, we check the entropy of High and normal :

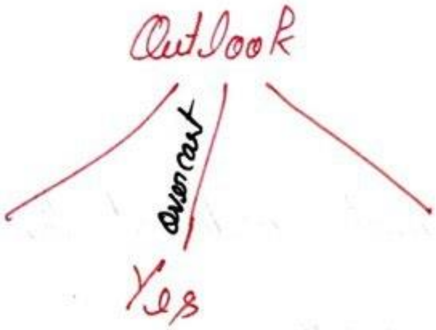
Sample (High) = 3 = [0+ 3-] => E = 0 [Stop]

Sample (Normal) = 2 = [2+ 0-] = E = 0 [Stop]



Entropy for Overcast :-

Entropy (overcast) = 0



Entropy for Rain :-

$S = 5 = [3+ \ 2-]$

Entropy (Rainy) = 0.97085

Gain (Rain, Temp) :- ?

Mild = 3 = [2+ 1-]

Cool = 2 = [1+ 1-]

$E_{(mild)} = 0.918295$

$E_{(cool)} = 1$

$$\text{Gain (Rain, Temp)} = 0.97085 - \left[\frac{3}{5} \times 0.918295 + \frac{2}{5} \times 1\right]$$

$$= 0.019873$$

$$Gain(Rain, Humidity) = ?$$

$$High = 2 = [1+ \; 1-]$$
$$Normal = 3 = [2+ \; 1-]$$

$$E_{high} = 1$$

$$E_{normal} = 0.918295$$

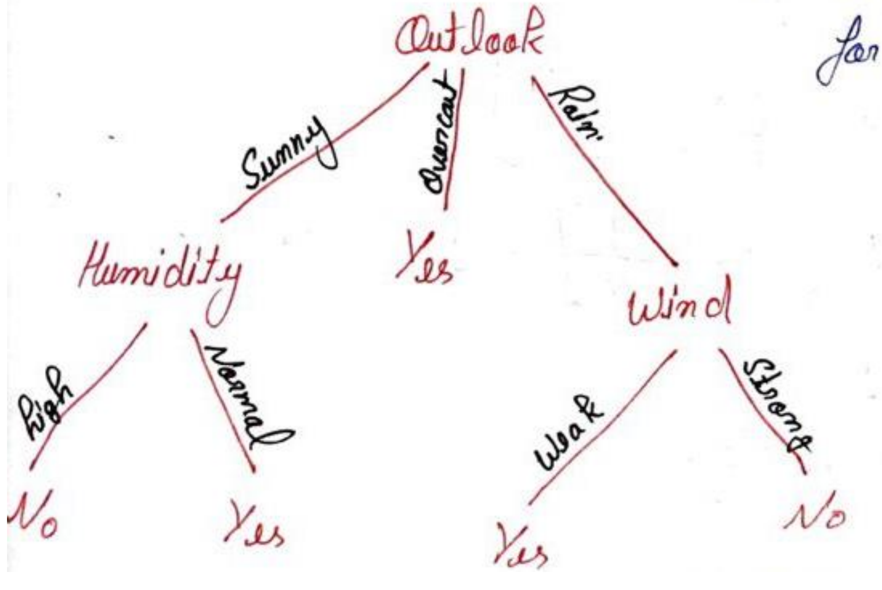
$$Gain(Rain, humidity) = 0.97085 - \left(\frac{2}{5} \times 1 + \frac{3}{5} \times 0.918295\right)$$
$$= 0.019873$$

$$Gain(Rain, Wind) = ?$$

$$Weak = 3 = [3+ \; 0-] \Rightarrow E = 0$$
$$Strong = 2 = [0+ \; 2-] \Rightarrow E = 0$$

$$\boxed{Gain(Rain, Wind) = 0.97085}$$

This is the highest Entropy for Rain



Gini Index :

$$\boxed{Gini(s) = 1 - \sum p^2}$$

For whole data set - of output data.

~~Sunny~~ $S = 14\ [9+\ 5-]$

$$Gini(S) = 1 - \left[\left(\frac{9}{14}\right)^2 + \left(\frac{5}{14}\right)^2\right]$$

$$= 1 - [0.41326 + 0.12755]$$

$$= 1 - 0.54081$$

$$\boxed{Gini(S) = 0.45919}$$

Gini index for each columns value.

Outlook :

Sunny $= 5 = [2+\ 3-]$

Overcast $= 4 = [4+\ 0-]$

Rain $= 5 = [3+\ 2-]$

$$\text{Gini index (Sunny)} = 1 - \left[\left(\frac{2}{4}\right)^2 + \left(\frac{3}{4}\right)^2\right]$$

$$= 1 - \left[\frac{4}{16} + \frac{9}{16}\right] = 1 - [0.25 + 0.562]$$

$$= 1 - 0.8125$$

$$\boxed{GI(Sunny) = 0.1875}$$

$GI(\cancel{\text{mild}}\ \text{Overcast}) = 4 = [4+\ 0-]$

$= 1 - \left[\left(\frac{4}{4}\right)^2 + \left(\frac{0}{4}\right)^2\right] \Rightarrow 1 - [0]$

$\boxed{GI(\text{Overcast}) = 1}$

$GI(\text{Rain}) = ?$

$S = 5 = [3+\ 2-]$

$= 1 - \left[\left(\frac{3}{5}\right)^2 + \left(\frac{2}{5}\right)^2\right]$

$= 1 - [0.36 + 0.16] = 1 - 0.52$

$\boxed{GI(\text{Rain}) = 0.48}$

Weighted Average Outlook :

$\boxed{GI_{Column} = \Sigma P_V \times (G_1 C_V)}$

$GI(\text{outlook}) = 0.1875 \times \left(\frac{5}{14}\right) + 1 \times \left(\frac{4}{14}\right) + 0.48 \times \left(\frac{5}{14}\right)$

$= 0.1875 \times 0.3571 + 1 \times 0.28571 + 0.48 \times 0.35714$

$= 0.06696 + 0.28571 + 0.171428$

$\boxed{GI(\text{outlook}) = 0.52409}$

Temperature :

$Hot = 4 = [2+ \; 2-]$

$Mild = 6 = [4+ \; 2-]$

$Cool = 4 = [3+ \; 1-]$

$GI(hot) = 1-[0.25+0.25] = 0.5$

$GI(Mild) = 1-[0.444+0.111] = 0.4448$

$GI(Cool) = 1-[0.562+0.062] = 0.3755$

$$Gini(Temp) = \frac{4}{14} \times 0.5 + \frac{6}{14} \times 0.444 + \frac{4}{14} \times 0.375$$

$$= 0.142 + 0.190 + 0.107$$

$$\boxed{GI(Temp) = 0.439}$$

Humidity :

$High = 3 = [0+ \; 3-] = GI = 1$

$normal = 2 = [2+ \; 0-] = GI = 1$

$$GI(Humidity) = \frac{3}{14} \times 1 + \frac{2}{14} \times 1$$

$$= 0.2142 + 0.1428$$

$$\boxed{GI(humidity) = 0.357}$$

Wind :

Weak = 8 = [6+ 2-]
Strong = 6 = [3+ 3-]

$GI(Weak) = 1 - [0.5625 + 0.0625] \Rightarrow 0.375$

$GI(Strong) = 1 - [0.25 + 0.25] \Rightarrow 0.5$

$$GI(Wind) = \frac{8}{14} \times 0.375 + \frac{6}{14} \times 0.5$$

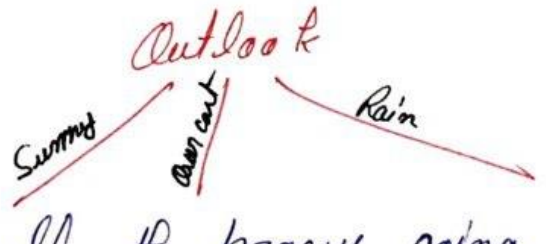
$$= 0.2142 + 0.2142$$

$$\boxed{GI(Wind) = 0.4285}$$

All Gini Index Value :

GI	Value
Outlook	= 0.34 (Correct value)
Temp	= 0.43
Humidity	= 0.35
Wind	= 0.42

And now assuming the smallest value of GI And than make G.I is root Node -



After this all the process going as well as Entropy.