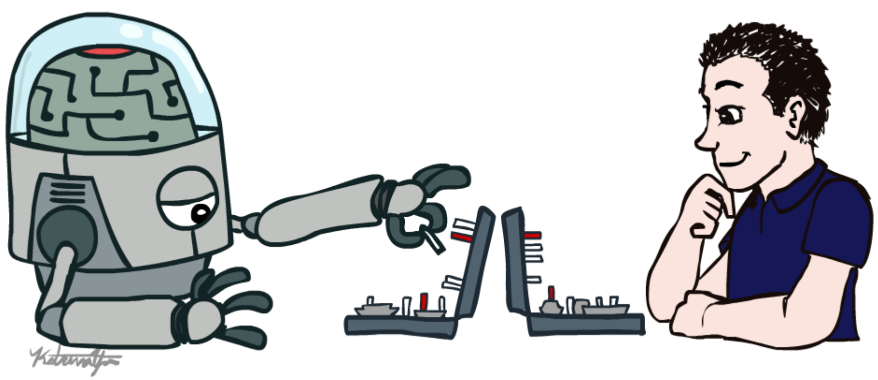
CSE 3521:
Introduction to Artificial Intelligence



[Many slides are adapted from the UC Berkeley. CS188 Intro to AI at UC Berkeley and previous CSE 3521 course at OSU.]

THE OHIO STATE UNIVERSITY

Supervised Learning: find f

- Given: Training set $\{(x_i, y_i) \mid i = 1 \ldots n\}$
- Find: A good approximation to $f : X \rightarrow Y$
- Examples: what are X, Y?
  - Spam Detection
    - Map email to {Spam, NotSpam}
  - Digit recognition
    - Map pixels to {0,1,2,3,4,5,6,7,8,9}
  - Stock Prediction
    - Map new, historic prices, etc. to $\Re$ (the real numbers)

A Supervised Learning Problem

- Consider a simple Boolean dataset:

$$f : X \rightarrow Y$$
$$X = \{0,1\}^4$$
$$Y = \{0,1\}$$

Example	x_1	x_2	x_3	x_4	y
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	1	1
4	1	0	0	1	1
5	0	1	1	0	0
6	1	1	0	0	0
7	0	1	0	1	0

- Question 1:
  - How should we pick the *hypothesis space*, the set of possible functions *f* ?
- Question 2:
  - How do we find the best *f* in the hypothesis space?

Most General Hypothesis Space

- Consider all possible Boolean functions over four input features!
  - 2^{16} possible hypotheses
  - 2^9 are consistent with our dataset
  - How do we choose the best one?

x_1	x_2	x_3	x_4	y
0	0	0	0	?
0	0	0	1	?
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	?
1	0	0	0	?
1	0	0	1	1
1	0	1	0	?
1	0	1	1	?
1	1	0	0	0
1	1	0	1	?
1	1	1	0	?
1	1	1	1	?

Example	x_1	x_2	x_3	x_4	y
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	1	1
4	1	0	0	1	1
5	0	1	1	0	0
6	1	1	0	0	0
7	0	1	0	1	0

A Restricted Hypothesis Space

- Consider all conjunctive Boolean functions.
  - 16 possible hypotheses
  - None are consistent with our dataset
  - How do we choose the best one?

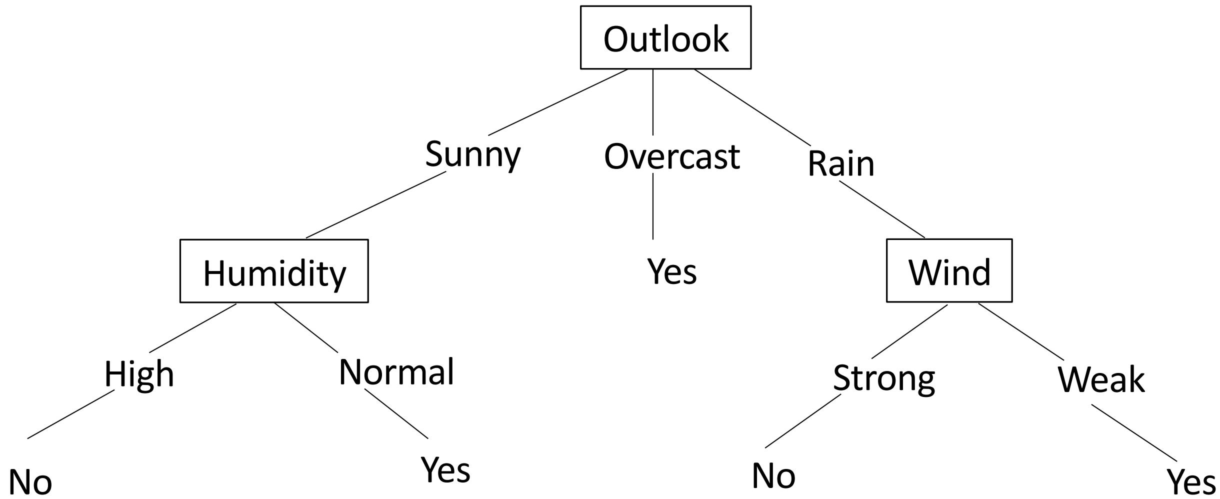
Rule	Counterexample
$\Rightarrow y$	1
$x_1 \Rightarrow y$	3
$x_2 \Rightarrow y$	2
$x_3 \Rightarrow y$	1
$x_4 \Rightarrow y$	7
$x_1 \wedge x_2 \Rightarrow y$	3
$x_1 \wedge x_3 \Rightarrow y$	3
$x_1 \wedge x_4 \Rightarrow y$	3
$x_2 \wedge x_3 \Rightarrow y$	3
$x_2 \wedge x_4 \Rightarrow y$	3
$x_3 \wedge x_4 \Rightarrow y$	4
$x_1 \wedge x_2 \wedge x_3 \Rightarrow y$	3
$x_1 \wedge x_2 \wedge x_4 \Rightarrow y$	3
$x_1 \wedge x_3 \wedge x_4 \Rightarrow y$	3
$x_2 \wedge x_3 \wedge x_4 \Rightarrow y$	3
$x_1 \wedge x_2 \wedge x_3 \wedge x_4 \Rightarrow y$	3

Example	x_1	x_2	x_3	x_4	y
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	1	1
4	1	0	0	1	1
5	0	1	1	0	0
6	1	1	0	0	0
7	0	1	0	1	0

Simple Training Dataset

Day	Outlook	Temperature	Humidity	Wind	PlayGolf?
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

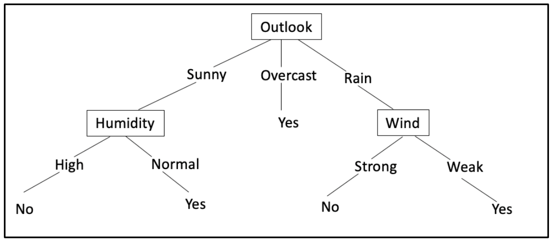
A Decision Tree for
f: <Outlook, Temperature, Humidity, Wind> → PlayGolf



- Each internal node: test one discreet valued attribute X_i
- Each branch from a node: selects one value for X_i
- Each leaf node: predict Y

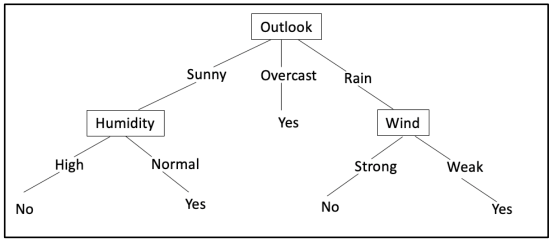
Decision Tree Learning

- Problem Setting
  - Set of possible instances X
    - Each instance *x* in *X* is a feature vector
    - e.g. <Humidity = Low; Wind = Weak; Outlook = Rain; Temperature = Hot>
  - Unknow target function $f: X \rightarrow Y$
    - Y = 1 if we play Golf on this day; else Y = 0
  - Set of function hypotheses $H = \{h \mid h : X \rightarrow Y\}$
    - Each hypotheses *h* is a decision tree
    - Trees sort *x* to leaf, which assigns *y*



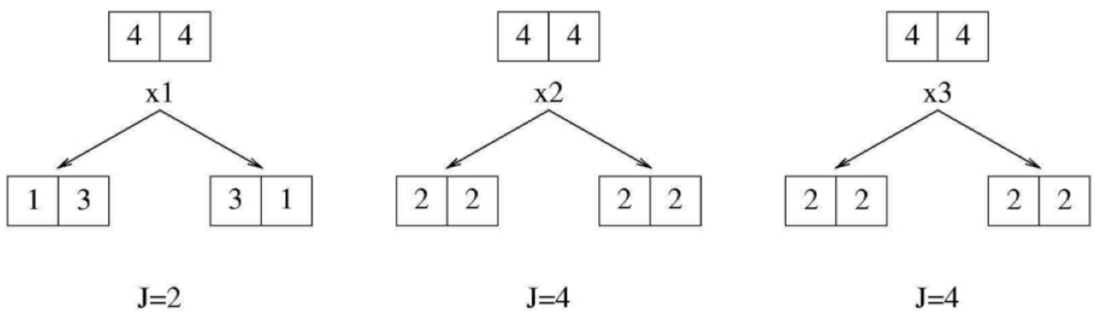
Decision Tree Learning

- Problem Setting
  - Set of possible instances X
    - Each instance *x* in *X* is a feature vector
    - e.g. <Humidity = Low; Wind = Weak; Outlook = Rain; Temperature = Hot>
  - Unknow target function $f: X \rightarrow Y$
    - Y = 1 if we play Golf on this day; else Y = 0
  - Set of function hypotheses $H = \{h \mid h : X \rightarrow Y\}$
    - Each hypotheses *h* is a decision tree
    - Trees sort *x* to leaf, which assigns *y*
- Input:
  - Training examples $\{<x^{(i)}, y^{(i)}>\}$ of unknown target function *f*
- Output
  - Hypothesis $h \in H$ that best approximates target function *f*



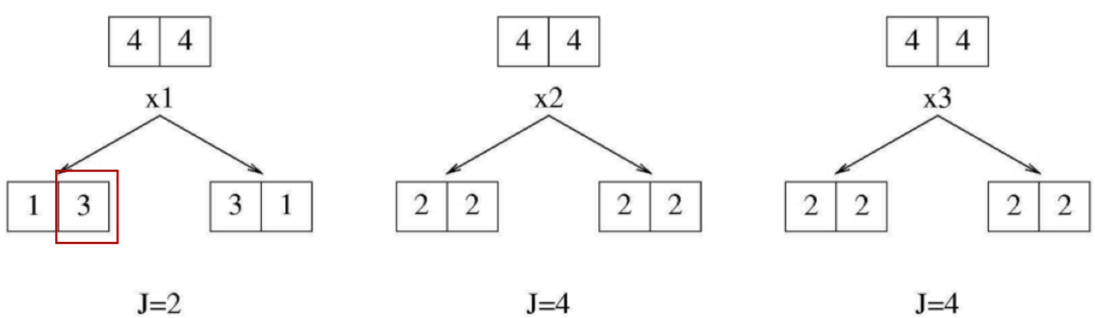
Example of Choosing the Best Attribute

x_1	x_2	x_3	y
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0



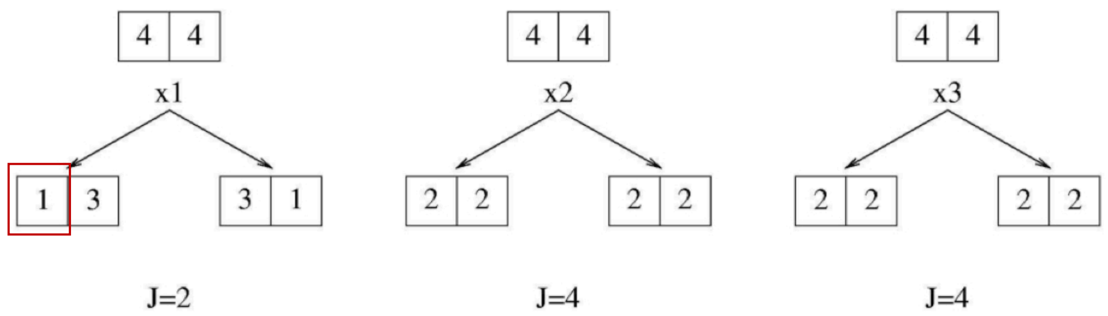
Example of Choosing the Best Attribute

x_1	x_2	x_3	y
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0



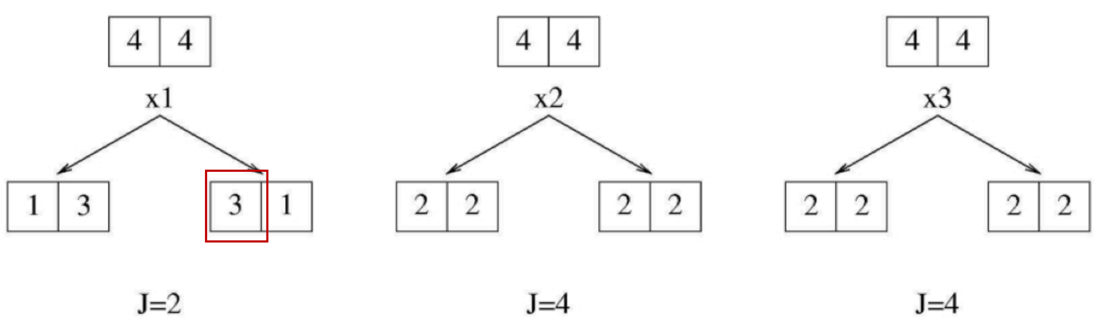
Example of Choosing the Best Attribute

x_1	x_2	x_3	y
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0



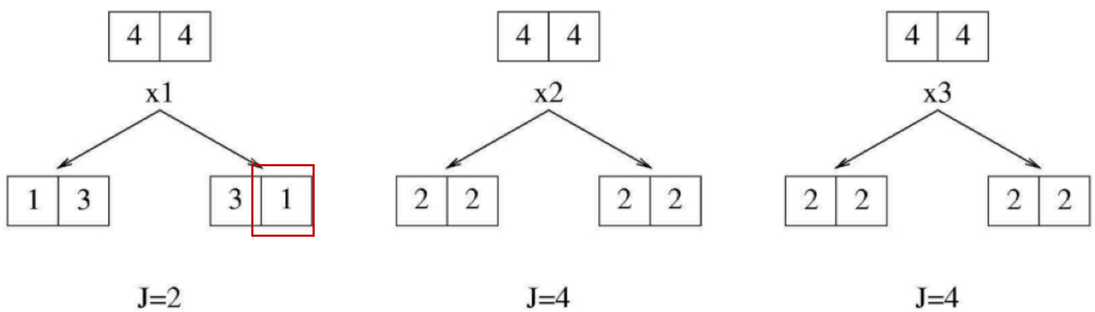
Example of Choosing the Best Attribute

x_1	x_2	x_3	y
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0



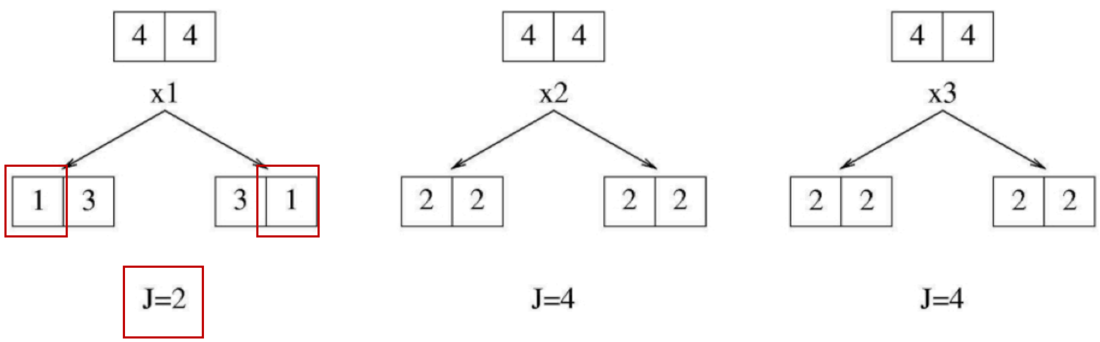
Example of Choosing the Best Attribute

x_1	x_2	x_3	y
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0



Example of Choosing the Best Attribute

x_1	x_2	x_3	y
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0



Choosing the Best Attribute

- Unfortunately this measure does not always work well
  - As it does not detect cases where we are making progress toward a good tree

Better Heuristic from Information Theory

- Let V be a random variable with the following probability distribution

$P(V=0)$	$P(V=1)$
0.2	0.8

- The surprise *S(V=v)* of each value of V is defined to be
  - $S(V=v) = -\log P(V=v)$
- An event with probability 1 gives zeo surprise
- An event with probability 0 gives infinite surprise
- Surprise equals Number of bits of information that needed to be transmitted to a recipient who knows the probabilities of the results
- This is also called the description length of *V = v*
- Fractional bits only make sense if they are part of longer message
  - e.g. describe a whole sequence of coin toss

Entropy

- The Entropy of V, denoted by *Entropy(V)* is defined as below:

$$Entropy(V) = -P(V=0)log_2P(V=0) - P(V=1)log_2P(V=1)$$

  - Here V is a binary variable with only possible values 0 and 1
- Entropy can be viewed as a measure of uncertainty

Dataset

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

Step 1: Determine the Decision Column

- Since decision trees are used for clarification, you need to determine the classes

Step 1: Determine the Decision Column

- Since decision trees are used for clarification, you need to determine the classes which are the basis for the decision.
- In this case, it it the last column, that is *Play Golf* column with classes ***Yes*** and ***No***.

Attributes | Classes

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

Step 1: Determine the Decision Column

- Since decision trees are used for clarification, you need to determine the classes
- In this case, it it the last column, that which are the basis for the decision.

  is *Play Golf* column with

  classes ***Yes*** and ***No***.

Attributes | Classes

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

- Next determine the rootNode
  - we need to compute the entropy.
  - To compute the entropy, we create a frequency table for the classes

Step 1: Determine the Decision Column

Attributes

Classes

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

Play Golf(14)	
Yes	No
9	5

Step 2: Calculating Entropy for the classes (Play Golf)

- In this step, you need to calculate the entropy for the Decision Column (Play Golf)
- *Entropy(PlayGolf) =* ***E****(5-,9+)*

Play Golf(14)	
Yes	No
9	5

Step 2: Calculating Entropy for the classes (Play Golf)

- In this step, you need to calculate the entropy for the Decision Column (Play Golf)
- *Entropy(PlayGolf) = **E**(5-,9+)*

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

Play Golf(14)	
Yes	No
9	5

Step 2: Calculating Entropy for the classes (Play Golf)

- In this step, you need to calculate the entropy for the Decision Column (Play Golf)
- *Entropy(PlayGolf) = **E**(5-,9+)*

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

$$Entropy(PlayGolf) = -p_{yes} log_2(p_{yes}) - p_{no} log_2(p_{no})$$

Play Golf(14)	
Yes	No
9	5

Step 2: Calculating Entropy for the classes (Play Golf)

- In this step, you need to calculate the entropy for the Decision Column (Play Golf)
- *Entropy(PlayGolf) = **E**(5-,9+)*

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

$$Entropy(PlayGolf) = -p_{yes} log_2(p_{yes}) - p_{no} log_2(p_{no})$$

Play Golf(14)	
Yes	No
9	5

$$E(\boldsymbol{PlayGolf}) = E(5{,}9)$$

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

$$= -(0.357 \ \log_2 0.357) - (0.643 \ \log_2 0.643)$$

$$= \mathbf{0.94}$$

Step 3: Calculate Entropy for Other Attributes After Split

For the other four attributes, we need to calculate the entropy after each of the split.

- E(PlayGolf, Outlook)
- E(PlayGolf, Temperature)
- E(PlayGolf, Humidity)
- E(PlayGolf,Windy)

The entropy for two variables is calculated using the formula.

$$Entropy(\mathrm{S},\mathrm{T}) = \sum_{c \in T} P(c)E(c)$$

The easiest way to approach this calculation is to create a frequency table for the two variables

Step 3: Calculate Entropy for Other Attributes After Split

E(PlayGolf, Outlook) Calculation:

To calculate **E**(PlayGolf, Outlook), we would use the formula below:

$$\boldsymbol{E}(PlayGolf, Outlook) = \boldsymbol{P}(Sunny)\boldsymbol{E}(Sunny) + \boldsymbol{P}(Overcast)\boldsymbol{E}(Overcast) + \boldsymbol{P}(Rainy)\boldsymbol{E}(Rainy)$$

		PlayGolf(14)		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5

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

$$\boldsymbol{E}(PlayGolf, Outlook) = \frac{5}{14}\boldsymbol{E}(3,2) + \frac{4}{14}\boldsymbol{E}(4,0) + \frac{5}{14}\boldsymbol{E}(2,3)$$

Step 3: Calculate Entropy for Other Attributes After Split

E(PlayGolf, Outlook) Calculation:

		PlayGolf(14)		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5

$$\boldsymbol{E}(PlayGolf, Outlook) = \frac{5}{14}\boldsymbol{E}(3{,}2) + \frac{4}{14}\boldsymbol{E}(4{,}0) + \frac{5}{14}\boldsymbol{E}(2{,}3)$$

$$\boldsymbol{E}(\boldsymbol{Sunny}) = E(3{,}2)$$

$$= -\left(\frac{3}{5}\log_2\frac{3}{5}\right) - \left(\frac{2}{5}\log_2\frac{2}{5}\right)$$

$$= -(0.60\ \log_2 0.60)\ -(0.40 \log_2 0.40)$$

$$= -(0.60\ * 0.737)\ -(0.40 * 0.529)$$

$$= \mathbf{0.971}$$

Step 3: Calculate Entropy for Other Attributes After Split

E(PlayGolf, Outlook) Calculation:

		PlayGolf(14)		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5

$$\boldsymbol{E}(PlayGolf, Outlook) = \frac{5}{14}\boldsymbol{E}(3,2) + \frac{4}{14}\boldsymbol{E}(4,0) + \frac{5}{14}\boldsymbol{E}(2,3)$$

$$\boldsymbol{E}(\boldsymbol{Sunny}) = E(3,2)$$

$$= -\left(\frac{3}{5}\log_2\frac{3}{5}\right) - \left(\frac{2}{5}\log_2\frac{2}{5}\right)$$

$$= -(0.60\ \log_2 0.60) - (0.40 \log_2 0.40)$$

$$= -(0.60 * 0.737) - (0.40 * 0.529)$$

$$= \mathbf{0.971}$$

$$\boldsymbol{E}(\boldsymbol{Overcast}) = E(4,0)$$

$$= -\left(\frac{4}{4}\log_2\frac{4}{4}\right) - \left(\frac{0}{4}\log_2\frac{0}{4}\right)$$

$$= -(0) - (0)$$

$$= \mathbf{0}$$

Step 3: Calculate Entropy for Other Attributes After Split

E(PlayGolf, Outlook) Calculation:

		PlayGolf(14)		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5

$$\boldsymbol{E}(PlayGolf, Outlook) = \frac{5}{14}\boldsymbol{E}(3{,}2) + \frac{4}{14}\boldsymbol{E}(4{,}0) + \frac{5}{14}\boldsymbol{E}(2{,}3)$$

$\boldsymbol{E}(\boldsymbol{Sunny}) = E(3{,}2)$

$= -\left(\frac{3}{5}\log_2\frac{3}{5}\right) - \left(\frac{2}{5}\log_2\frac{2}{5}\right)$

$= -(0.60\ \log_2 0.60) - (0.40 \log_2 0.40)$

$= -(0.60 * 0.737) - (0.40 * 0.529)$

$= \mathbf{0.971}$

$\boldsymbol{E}(\boldsymbol{Overcast}) = E(4{,}0)$

$= -\left(\frac{4}{4}\log_2\frac{4}{4}\right) - \left(\frac{0}{4}\log_2\frac{0}{4}\right)$

$= -(0) - (0)$

$= \mathbf{0}$

$\boldsymbol{E}(\boldsymbol{Rainy}) = E(2{,}3)$

$= -\left(\frac{2}{5}\log_2\frac{2}{5}\right) - \left(\frac{3}{5}\log_2\frac{3}{5}\right)$

$= -(0.40\ \log_2 0.40) - (0.6 \log_2 0.60)$

$= \mathbf{0.971}$

Step 3: Calculate Entropy for Other Attributes After Split

E(PlayGolf, Outlook) Calculation:

		PlayGolf(14)		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5

$$E(PlayGolf, Outlook) = \frac{5}{14}E(3,2) + \frac{4}{14}E(4,0) + \frac{5}{14}E(2,3)$$

$$\boldsymbol{E}(\boldsymbol{Sunny}) = E(3,2)$$

$$= -\left(\frac{3}{5}\log_2\frac{3}{5}\right) - \left(\frac{2}{5}\log_2\frac{2}{5}\right)$$

$$= -(0.60\ \log_2 0.60) - (0.40 \log_2 0.40)$$

$$= -(0.60 * 0.737) - (0.40 * 0.529)$$

$$= \mathbf{0.971}$$

$$\boldsymbol{E}(\boldsymbol{Overcast}) = E(4,0)$$

$$= -\left(\frac{4}{4}\log_2\frac{4}{4}\right) - \left(\frac{0}{4}\log_2\frac{0}{4}\right)$$

$$= -(0) - (0)$$

$$= \mathbf{0}$$

$$\boldsymbol{E}(\boldsymbol{Rainy}) = E(2,3)$$

$$= -\left(\frac{2}{5}\log_2\frac{2}{5}\right) - \left(\frac{3}{5}\log_2\frac{3}{5}\right)$$

$$= -(0.40\ \log_2 0.40) - (0.6 \log_2 0.60)$$

$$= \mathbf{0.971}$$

E(4,0) = 0;

E(2,3) = E(3,2)

Step 3: Calculate Entropy for Other Attributes After Split

E(PlayGolf, Outlook) Calculation:

		PlayGolf(14)		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5

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

$$= \frac{5}{14}\boldsymbol{E}(3,2) + \frac{4}{14}\boldsymbol{E}(4,0) + \frac{5}{14}\boldsymbol{E}(2,3)$$

$$= \frac{5}{14}0.971 + \frac{4}{14}0.0 + \frac{5}{14}0.971$$

$$= 0.357 * 0.971 + 0.0 + 0.357 * 0.971$$

$$= \mathbf{0.693}$$

Step 3: Calculate Entropy for Other Attributes After Split

E(PlayGolf, Temperature) Calculation

		PlayGolf(14)		
		Yes	No	
Temperature	Hot	2	2	4
	Cold	3	1	4
	Mild	4	2	6

E(PlayGolf, Temperature) = P(Hot) E(2,2) + P(Cold) E(3,1) + P(Mild) E(4,2)

E (PlayGolf, Temperature) = **4/14 * E**(Hot) + **4/14 * E**(Cold) + **6/14 * E**(Mild)

E (PlayGolf, Temperature) = **4/14 * E**(2, 2) + **4/14 * E**(3, 1) + **6/14 * E**(4, 2)

E (PlayGolf, Temperature) = 4/14 * -(2/4 log 2/4) – (2/4 log 2/4)

+ 4/14 * -(3/4 log 3/4) – (1/4 log 1/4)

+ 6/14 * -(4/6 log 4/6) – (2/6 log 2/6)

E (PlayGolf, Temperature) = 5/14 * 1.0

+ 4/14 * 1.811

+ 5/14 * 0.918

= 0.911

Step 3: Calculate Entropy for Other Attributes After Split

E(PlayGolf, Humidity) Calculation

		PlayGolf(14)		
		Yes	No	
Humidity	High	3	4	7
	Normal	6	1	7

E (PlayGolf, Humidity) = **7/14 * E**(High) + **7/14 * E**(Normal)

E (PlayGolf, Humidity) = **7/14 *** E(3,4) + **7/14 *** E(6,1)

E (PlayGolf, Humidity) = 7/14 * -(3/7 log 3/7) – (4/7 log 4/7)

\+ 7/14 * -(6/7 log 6/7) – (1/7 log 1/7)

E (PlayGolf, Humidity) = 7/14 * 0.985

\+ 7/14 * 0.592

= 0.788

Step 3: Calculate Entropy for Other Attributes After Split

E(PlayGolf, Windy) Calculation

		PlayGolf(14)		
		Yes	No	
Windy	TRUE	3	3	6
	FALSE	6	2	8

$$\text{E (PlayGolf, Windy)} = \mathbf{6/14 * E}(\text{True}) + \mathbf{8/14 * E}(\text{False})$$

$$\text{E (PlayGolf, Windy)} = \mathbf{6/14 * E}(3, 3) + \mathbf{8/14 * E}(6, 2)$$

$$\text{E (PlayGolf, Windy)} = 6/14 * -(3/6 \log 3/6) - (3/6 \log 3/6)$$
$$+ 8/14 * -(6/8 \log 6/8) - (2/8 \log 2/8)$$

$$\text{E (PlayGolf, Windy)} = 6/14 * 1.0$$
$$+ 8/14 * 0.811$$
$$= \mathbf{0.892}$$

Step 3: Calculate Entropy for Other Attributes After Split

1. E(PlayGolf, Outlook) = **0.693**
2. E(PlayGolf, Temperature) = **0.911**
3. E(PlayGolf, Humidity) = **0.788**
4. E(PlayGolf,Windy) = **0.892**

Step 4: Calculating Information Gain for Each Split

- The next step is to calculate the information gain for each of the attributes.
- The information gain is calculated from the split using each of the attributes.
- Then the attribute with the largest information gain is used for the split.
- The information gain is calculated using the formula:

$$\text{Gain}(S,T) = \text{Entropy}(S) - \text{Entropy}(S,T)$$

Step 4: Calculating Information Gain for Each Split

Gain(PlayGolf, Outlook) = Entropy(PlayGolf) – Entropy(PlayGolf, Outlook)

= 0.94 – 0.693 **= 0.247**

Gain(PlayGolf, Temperature) = Entropy(PlayGolf) – Entropy(PlayGolf, Temperature)

= 0.94 – 0.911 ***= 0.029***

Gain(PlayGolf, Humidity) = Entropy(PlayGolf) – Entropy(PlayGolf, Humidity)

= 0.94 – 0.788 ***= 0.152***

Gain(PlayGolf, Windy) = Entropy(PlayGolf) – Entropy(PlayGolf, Windy)

= 0.94 – 0.892 ***= 0.048***

Step 4: Calculating Information Gain for Each Split

Gain(PlayGolf, Outlook) = Entropy(PlayGolf) – Entropy(PlayGolf, Outlook)

= 0.94 – 0.693 **= 0.247**

Gain(PlayGolf, Temperature) = Entropy(PlayGolf) – Entropy(PlayGolf, Temperature)

= 0.94 – 0.911 ***= 0.029***

Gain(PlayGolf, Humidity) = Entropy(PlayGolf) – Entropy(PlayGolf, Humidity)

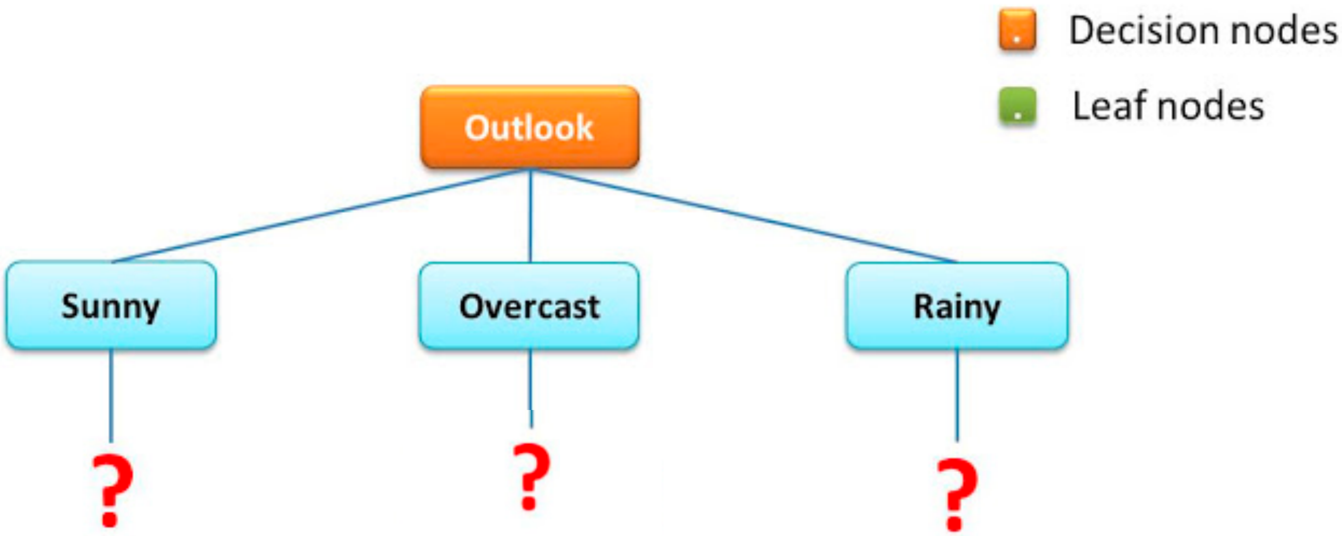
= 0.94 – 0.788 ***= 0.152***

Gain(PlayGolf, Windy) = Entropy(PlayGolf) – Entropy(PlayGolf, Windy)

= 0.94 – 0.892 ***= 0.048***

Step 5: Perform the 1st split

From our calculation, the highest information gain comes from Outlook. Therefore the split will look like this:



Step 5: Perform the 1st split

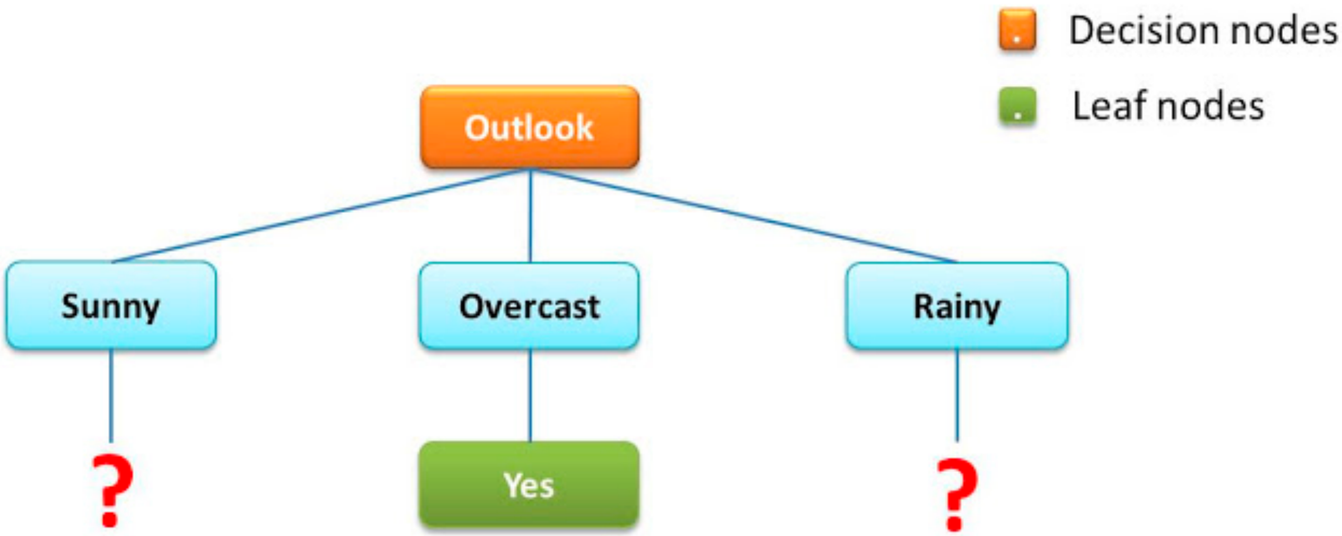
Outlook	Temperature	Humidity	Windy	Play Golf
Sunny	Mild	Normal	FALSE	Yes
Sunny	Mild	High	FALSE	Yes
Sunny	Cool	Normal	FALSE	Yes
Sunny	Cool	Normal	TRUE	No
Sunny	Mild	High	TRUE	No
Overcast	Hot	High	FALSE	Yes
Overcast	Mild	High	TRUE	Yes
Overcast	Hot	Normal	FALSE	Yes
Overcast	Cool	Normal	TRUE	Yes
Rainy	Hot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

Overcast outlook requires no further split because it is just one homogeneous group.

So we have a leaf node.

Step 5: Perform the 1st split

From our calculation, the highest information gain comes from Outlook. Therefore the split will look like this:



Overcast outlook requires no further split because it is just one homogeneous group. So we have a leaf node.

Step 6: Perform further splits

The Sunny and the Rainy attributes needs to be split

The Rainy outlook can be split using either Temperature, Humidity or Windy.

Question: What attribute would best be used for this split?

- *Gain(PlayGolf, Outlook=Rainy, Temperature) =*

  Entropy(PlayGolf, Outlook=Rainy) – Entropy(PlayGolf, Outlook=Rainy, Temperature)

- *Gain(PlayGolf, Outlook=Rainy, Humidity) =*

  Entropy(PlayGolf, Outlook=Rainy) – Entropy(PlayGolf, Outlook=Rainy, Humidity)

- *Gain(PlayGolf, Outlook=Rainy, Windy) =*

  Entropy(PlayGolf, Outlook=Rainy) – Entropy(PlayGolf, Outlook=Rainy, Windy)

Step 6: Perform further splits

The Sunny and the Rainy attributes needs to be split

The Rainy outlook can be split using either Temperature, Humidity or Windy.

Question: What attribute would best be used for this split?

Humidity, produces homogenous groups.

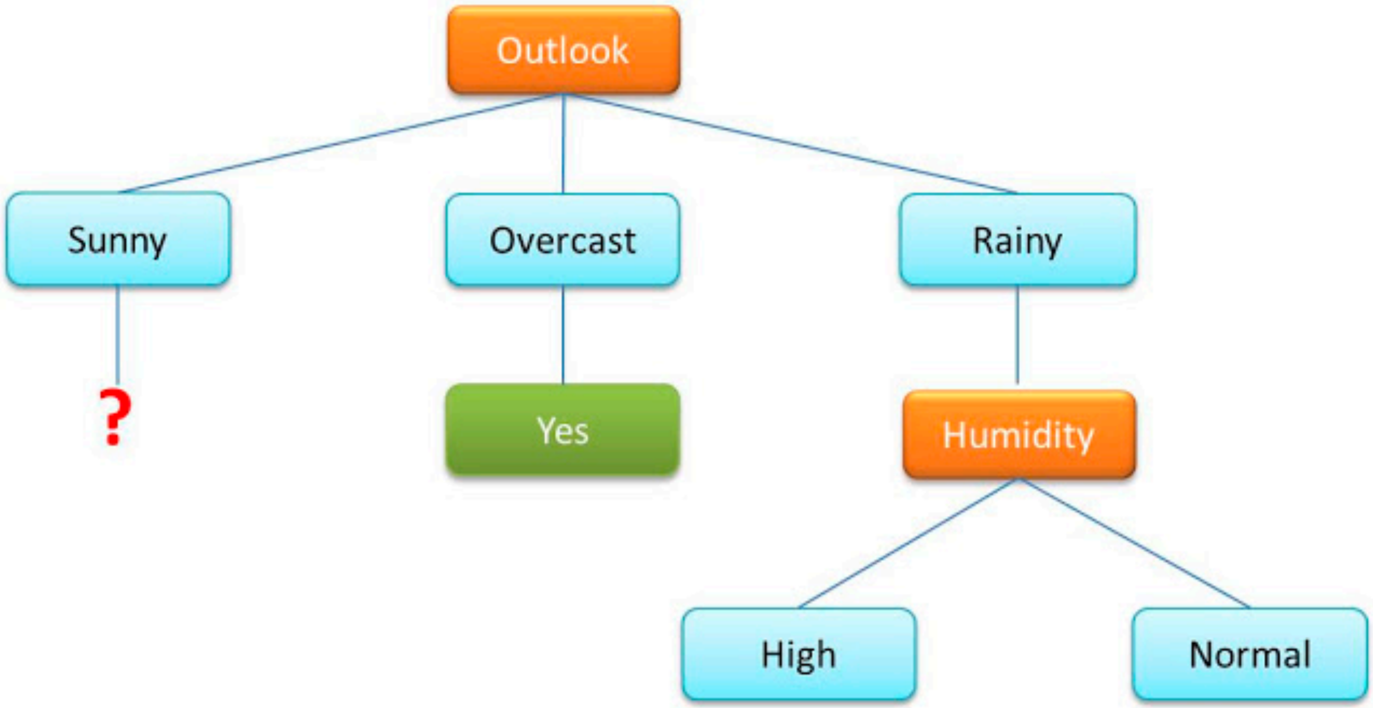
Outlook	Temperature	Humidity	Windy	Play Golf
Rainy	Hot	High	FALSE	No
Rainy	Hot	High	TRUE	No
Rainy	Mild	High	FALSE	No
Rainy	Cool	Normal	FALSE	Yes
Rainy	Mild	Normal	TRUE	Yes

- *Gain(PlayGolf, Outlook=Rainy, Humidity) =*

  Entropy(PlayGolf, Outlook=Rainy) – Entropy(PlayGolf, Outlook=Rainy, Humidity)=

  Entropy(PlayGolf, Outlook=Rainy) – 0

Step 6: Perform further splits



Step 6: Perform further splits

The Rainy outlook can be split using either Temperature, Humidity or Windy.

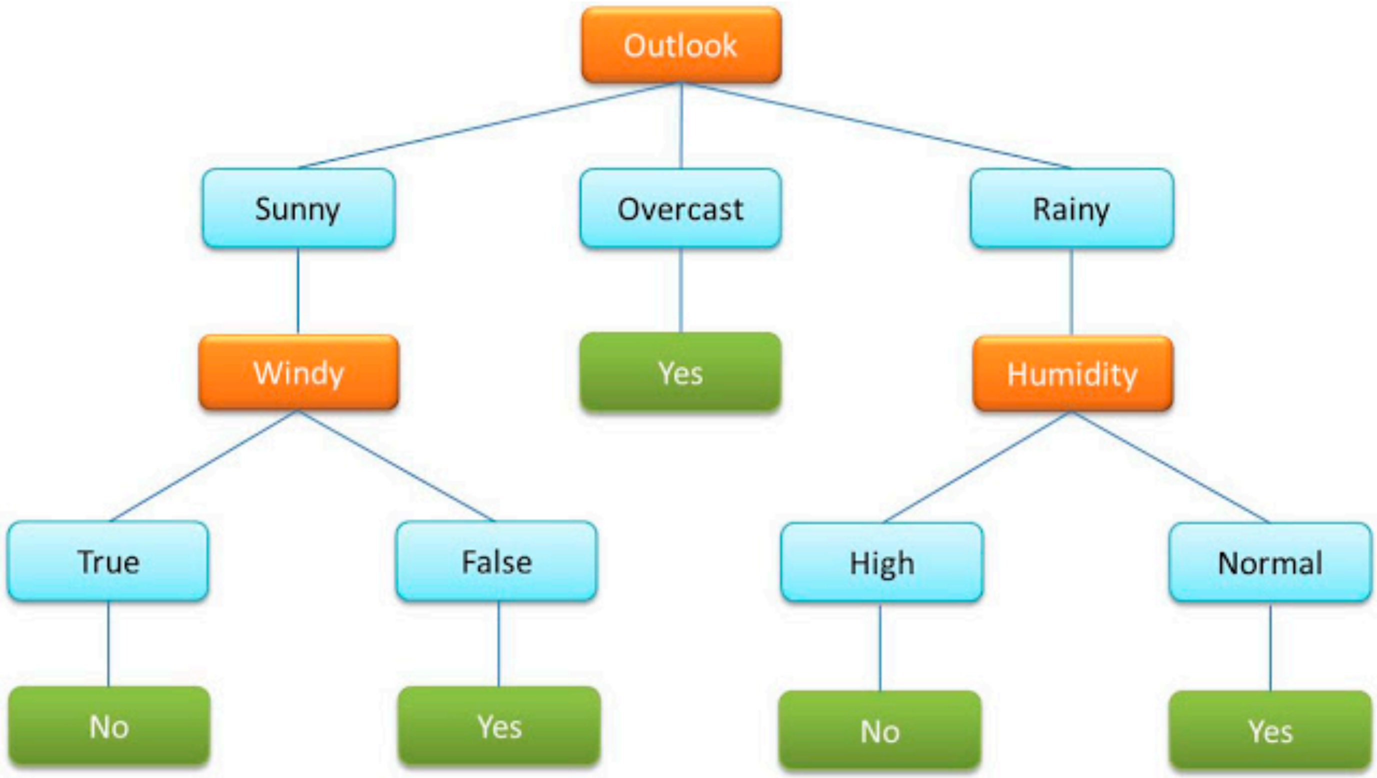
Question: What attribute would best be used for this split? Why?

Answer: **Windy** . Because it produces homogeneous groups.

Outlook	Temperature	Humidity	Windy	Play Golf
Sunny	Mild	Normal	FALSE	Yes
Sunny	Mild	High	FALSE	Yes
Sunny	Cool	Normal	FALSE	Yes

Sunny	Cool	Normal	TRUE	No
Sunny	Mild	High	TRUE	No

Step 6: Perform further splits



ID3 Algorithm

```
ID3 (S, A, V)
    Let:
        S = Learning Set
        A = Attibute Set
        V = Attribute Values

    Begin
        Load learning sets  and create decision tree root node(rootNode),
        Add learning set S into root not as its subset

        For rootNode, compute Entropy(rootNode.subset)

        If Entropy(rootNode.subset) == 0 (subset is homogeneous)
            return a leaf node

        If Entropy(rootNode.subset)!= 0 (subset is not homogeneous)
            compute Information Gain for each attribute left (not been used for spliting)
            Find attibute A with Maximum(Gain(S,A))
            Create child nodes for this root node and add to rootNode in the decision tree

        For each child of the rootNode
            Apply ID3(S,A,V)
            Continue until a node with Entropy of 0 or a leaf node is reached
    End
```

Continuous Valued Attribute

Create a discrete attribute to test continuous

- $Temperature = 82.5$
- $(Temperature > 72.3) = t, f$

Temperature:	40	48	60	72	80	90
Play Golf:	No	No	Yes	Yes	Yes	No

Unknown Attribute

Attributes | Classes

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

Unknown Attribute

Attributes　　Classes

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

Unknown Attribute

Attributes / Classes

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

#High=5
#Normal = 7

Unknown Attribute

Attributes | Classes

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

#High=5
#Normal = 7

Unknown Attribute

Attributes | Classes

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

For PlayGlolf=No
#High=3
#Normal = 1

Unknown Attribute

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

Attributes: Outlook, Temperature, Humidity, Windy. Classes: Play Golf.

For PlayGlolf=No
#High=3
#Normal = 1

Unknown Attribute Values

What if some examples are missing values of A?

Use training example anyway, sort through tree

- If node n tests A, assign most common value of A among other examples sorted to node n
- Assign most common value of A among other examples with same target value