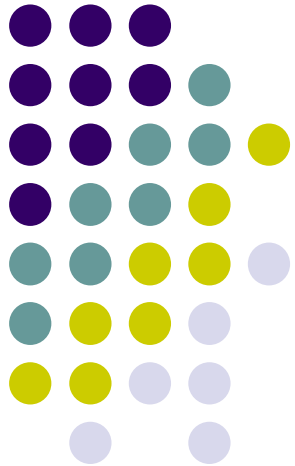
Classification by Decision Tree Induction

Prof. Dr. Songül Varlı
Yıldız Technical University
Department of Computer Engineering
svarli@yildiz.edu.tr



General Approach for Building a Classification Model

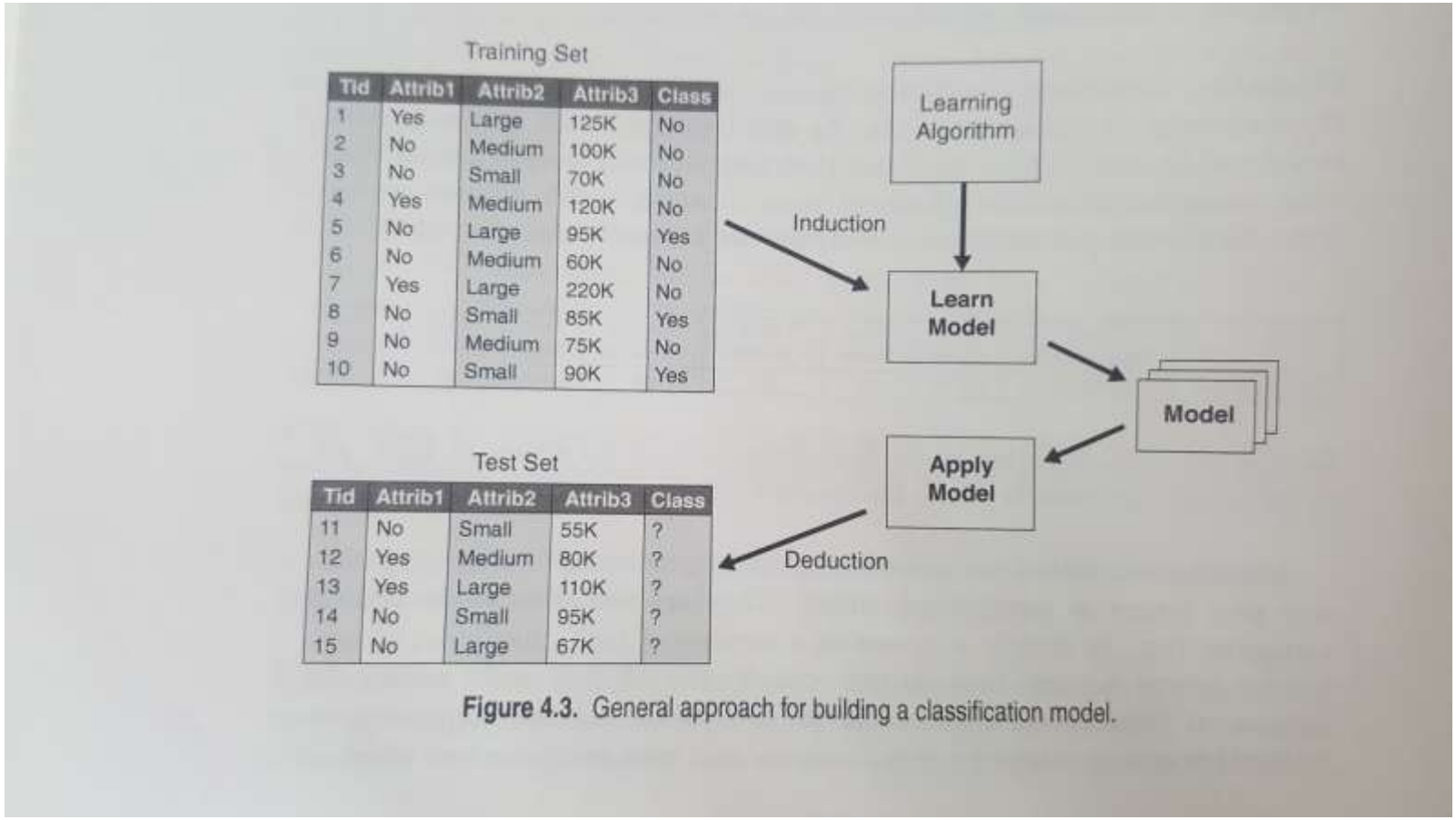
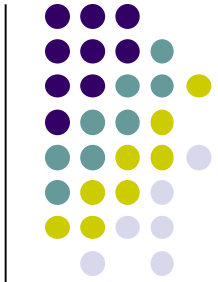


Figure 4.3. General approach for building a classification model.

Decision Trees

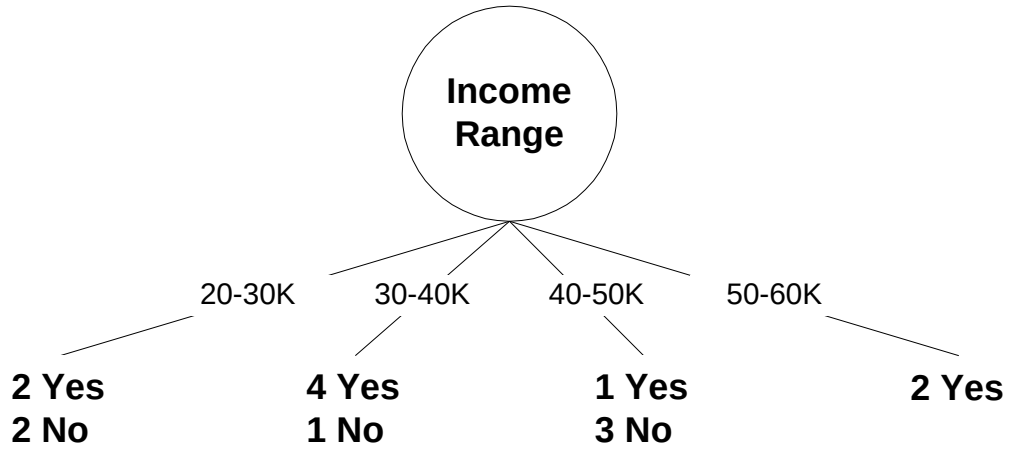


- Decision Tree induction is the learning of decision trees from class-labeled training tuples(instances).
- A decision tree is a flowchart-like tree structure, where each internal node (nonleaf node) denotes a test on attribute, each branch represents an outcome of the test, and leaf node (or terminal) holds a class label.
- The topmost node in a tree is the root node

Decision Trees

Table 3.1 • **The Credit Card Promotion Database**

Income Range	Output Variable: Life Insurance Promotion	Credit Card Insurance	Sex	Age
40–50K	No	No	Male	45
30–40K	Yes	No	Female	40
40–50K	No	No	Male	42
30–40K	Yes	Yes	Male	43
50–60K	Yes	No	Female	38
20–30K	No	No	Female	55
30–40K	Yes	Yes	Male	35
20–30K	No	No	Male	27
30–40K	No	No	Male	43
30–40K	Yes	No	Female	41
40–50K	Yes	No	Female	43
20–30K	Yes	No	Male	29
50–60K	Yes	No	Female	39
40–50K	No	No	Male	55
20–30K	Yes	Yes	Female	19

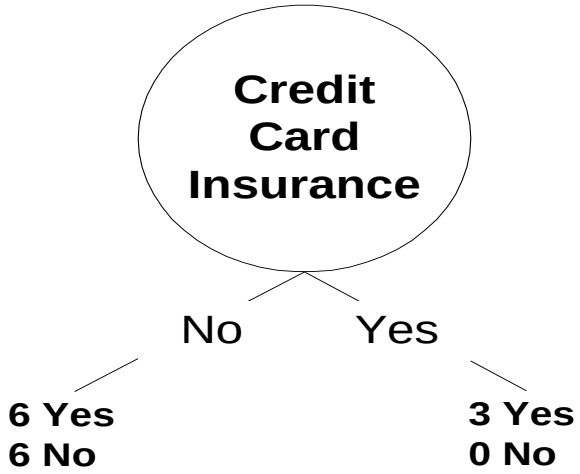


A partial decision tree with root node=income range

Decision Tree

Table 3.1 • **The Credit Card Promotion Database**

Income Range	Life Insurance Promotion	Credit Card Insurance	Sex	Age
40–50K	No	No	Male	45
30–40K	Yes	No	Female	40
40–50K	No	No	Male	42
30–40K	Yes	Yes	Male	43
50–60K	Yes	No	Female	38
20–30K	No	No	Female	55
30–40K	Yes	Yes	Male	35
20–30K	No	No	Male	27
30–40K	No	No	Male	43
30–40K	Yes	No	Female	41
40–50K	Yes	No	Female	43
20–30K	Yes	No	Male	29
50–60K	Yes	No	Female	39
40–50K	No	No	Male	55
20–30K	Yes	Yes	Female	19



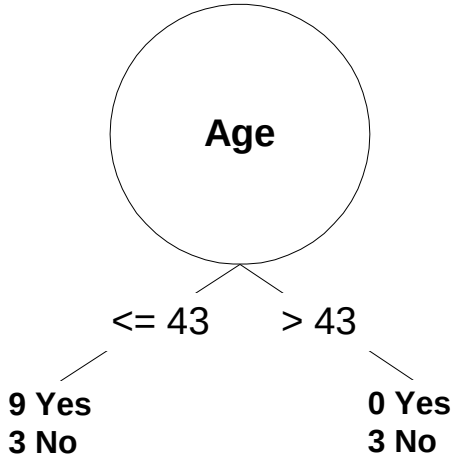
A partial decision tree with

root node=CreditCardInsurance

Decision Tree

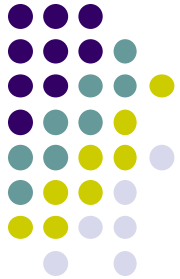
Table 3.1 • **The Credit Card Promotion Database**

Income Range	Life Insurance Promotion	Credit Card Insurance	Sex	Age
40–50K	No	No	Male	45
30–40K	Yes	No	Female	40
40–50K	No	No	Male	42
30–40K	Yes	Yes	Male	43
50–60K	Yes	No	Female	38
20–30K	No	No	Female	55
30–40K	Yes	Yes	Male	35
20–30K	No	No	Male	27
30–40K	No	No	Male	43
30–40K	Yes	No	Female	41
40–50K	Yes	No	Female	43
20–30K	Yes	No	Male	29
50–60K	Yes	No	Female	39
40–50K	No	No	Male	55
20–30K	Yes	Yes	Female	19



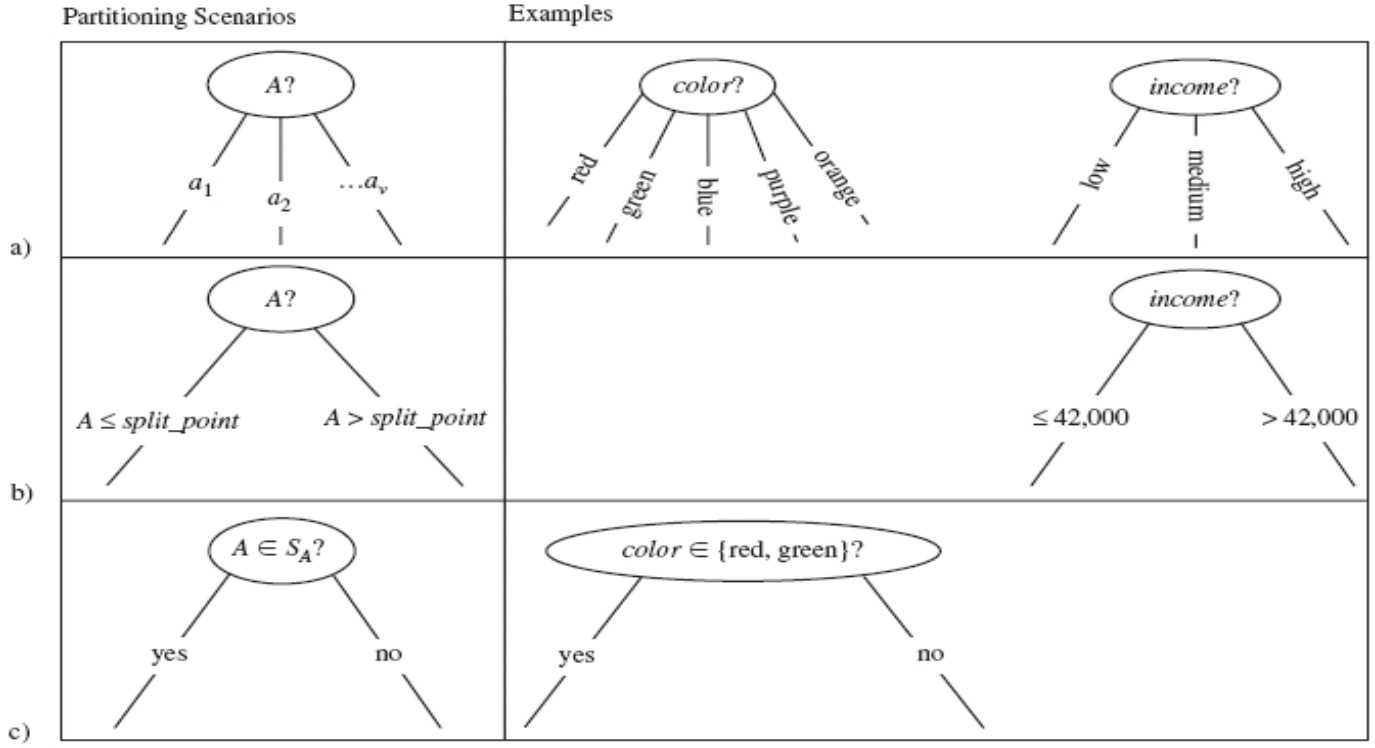
A partial decision tree with root node=age

Decision Tree Induction



- During late 1970's, Ross Quinlan, a researcher in Machine Learning, developed a decision tree algorithm as ID3 (Iterative Dichotomiser).
- This work is expanded and presented C4.5, which became a benchmark to which newer supervised learning algorithms are often compared.

Three possibilities for partitioning tubles based on the splitting criterion, shown with examples;

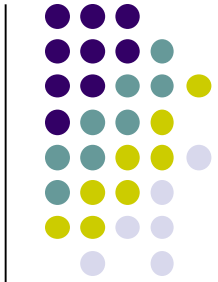


a- If A is discrete-valued

b- If A is continuous-value, then two branchs are grown

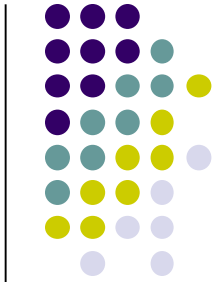
c- If A is discrete-valued and binary tree must be produced

An Algorithm for Building Decision Trees



1. Let *T* be the set of training instances.
2. Choose an attribute that best differentiates the instances in *T.*
3. Create a tree node whose value is the chosen attribute.
   - Create child links from this node where each link represents a unique value for the chosen attribute.
   - Use the child link values to further subdivide the instances into subclasses.
4. For each subclass created in step 3:
   - If the instances in the subclass satisfy predefined criteria or if the set of remaining attribute choices for this path is null, specify the classification for new instances following this decision path.
   - If the subclass does not satisfy the criteria and there is at least one attribute to further subdivide the path of the tree, let *T* be the current set of subclass instances and return to step 2.

Information Gain -Entropy

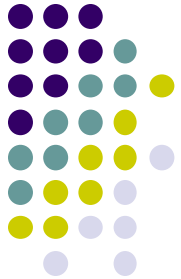


- In order to define information gain precisely, we use a measure commonly used in information theory, called entropy
- Entropy characterizes the (im)purity of an arbitrary collection of examples.

Entropy

- Given a collection S, containing positive and negative examples of some target concept, the entropy of S relative to this binary classification is:
- $\text{Entropy}(S) = -(p_+ \log_2 p_+ + p_- \log_2 p_-)$
  - S is a sample of training examples
  - p+ is the proportion of positive examples
  - p- is the proportion of negative examples
- entropy is a measure of the impurity in a collection of training examples

Attribute Selection Measure: Information Gain (ID3/C4.5)



- Select the attribute with the highest information gain
- Let p_i be the probability that an arbitrary tuple in D belongs to class C_i, estimated by $|C_{i,D}|/|D|$
- Expected information (entropy) needed to classify a tuple in D:

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

- Information needed (after using A to split D into v partitions) to classify D:

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times I(D_j)$$

- Information gained by branching on attribute A

$$Gain(A) = Info(D) - Info_A(D)$$

Attribute Selection: Information Gain

- Class P: buys_computer = “yes”
- Class N: buys_computer = “no”

$$Info(D) = I(9,5) = -\frac{9}{14}\log_2(\frac{9}{14}) - \frac{5}{14}\log_2(\frac{5}{14}) = 0.940$$

$$Info_{age}(D) = \frac{5}{14} I(2,3) + \frac{4}{14} I(4,0) + \frac{5}{14} I(3,2) = 0.694$$

age	p_i	n_i	$I(p_i, n_i)$
<=30	2	3	0.971
31…40	4	0	0
>40	3	2	0.971

$\frac{5}{14} I(2,3)$ means “age <=30” has 5 out of 14 samples, with 2 yes’es and 3 no’s. Hence

$$Gain(age) = Info(D) - Info_{age}(D) = 0.246$$

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Similarly,

$$Gain(income) = 0.029$$

$$Gain(student) = 0.151$$

$$Gain(credit\_rating) = 0.048$$

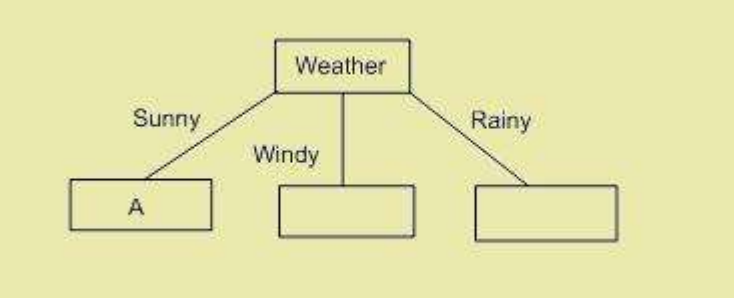
Decison Tree: Weekend example

- **1.step:** Let *T* be the set of training instances.

Weekend (Example)	Weather	Parents	Money	Decision (Category
W1	Sunny	Yes	Rich	Cinema
W2	Sunny	No	Rich	Tennis
W3	Windy	Yes	Rich	Cinema
W4	Rainy	Yes	Poor	Cinema
W5	Rainy	No	Rich	Stay in
W6	Rainy	Yes	Poor	Cinema
W7	Windy	No	Poor	Cinema
W8	Windy	No	Rich	Shopping
W9	Windy	Yes	Rich	Cinema
W10	Sunny	No	Rich	Tennis

Decison Tree: Weekend example

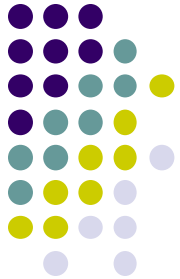
- 2. Step:Choose an attribute that best differentiates the instances in *T.*



- 3. Step: Create a tree node whose value is the chosen attribute.

Weekend (Example)	Weather	Parents	Money	Decision (Category)
W1	Sunny	Yes	Rich	Cinema
W2	Sunny	No	Rich	Tennis
W10	Sunny	No	Rich	Tennis

Decison Tree: Entropy



- For the 10 Training instances
  - 6 : Go to Cinema
  - 2 : Play Tennis
  - 1 : Stay at Home
  - 1 : Go to Shopping
- Entropy:

$H(T)= - (6/10) \log_2(6/10) - (2/10) \log_2(2/10) - (1/10) \log_2(1/10) - (1/10) \log_2(1/10)$

$H(T)= 1.571$

Decision Tree: Information Gain

- Gain(T, weather)= ?
  - Sunny=3 (1 Cinema, 2 Tennis)
  - Windy=4 (3 Cinema, 1 Shopping)
  - Rainy=3 (2 Cinema, 1 Stay in)

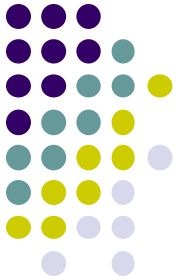
  - Entropy(T_{sunny})= - (1/3) $\log_2$ (1/3) - (2/3) $\log_2$ (2/3)=0,918
  - Entropy(T_{windy})= - (3/4) $\log_2$ (3/4) - (1/4) $\log_2$ (1/4) =0,811
  - Entropy(T_{rainy})= - (2/3) $\log_2$ (2/3) - (1/3) $\log_2$ (1/3) =0,918

- Gain(T, weather) = Entropy(T)- ((P(sunny)Entropy(T_{sunny}) + P(windy) Entropy(T_{windy})+ P(rainy) Entropy(T_{rainy}))

=1.571- ((3/10)Entropy(T_{sunny})+(4/10)Entropy(T_{windy})+ (3/10)Entropy(T_{rainy}))

Gain(T, weather) =0.70

Decision Tree: Information Gain



- Gain(T, parents)= ?
  - Yes=5 (5 Cinema)
  - No =5 (2 Tennis, 1 Cinema, 1 Shopping, 1 Stay in)

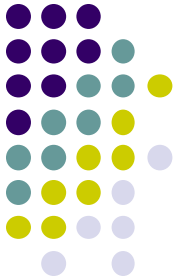
  - Entropy(T_{yes})= - (5/5) $\log_2$ (5/5) = 0
  - Entropy(T_{no})= - (2/5) $\log_2$ (2/5) - 3(1/5) $\log_2$ (1/5) =1.922

- Gain(T, parents) = Entropy(T)- ((P(yes)Entropy(T_{yes}) + P(no) Entropy(T_{no}))

  =1.571- ((5/10)Entropy(T_{yes})+(5/10)Entropy(T_{no}))

  Gain(T, parents)=0.61

Decision Tree: Information Gain



- Gain(T, money)= ?
  - Rich=7 (3 Cinema, 2 Tennis, 1 Shopping, 1 Stay in)
  - Poor=3 (3 Cinema)

  - Entropy(T_{rich})= 1.842
  - Entropy(T_{poor})= 0

- Gain(T, money) = Entropy(T)- ((P(rich)Entropy(T_{rich}) + P(poor) Entropy(T_{poor}))

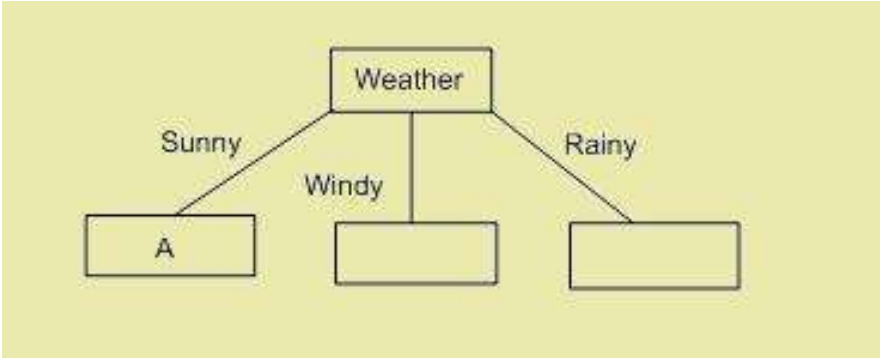
  =1.571- ((5/10)Entropy(T_{rich})+(5/10)Entropy(T_{poor}))

  Gain(T, money)=0.2816

Decision Tree: Information Gain

- Gain(T, weather) =0.70
- Gain(T, parents)=0.61
- Gain(T, money)=0.2816

Create a tree node whose value is the chosen attribute.



Decision Trees:

- Create child links from this node where each link represents a unique value for the chosen attribute.

Weekend (Example)	Weather	Parents	Money	Decision (Category)
W1	Sunny	Yes	Rich	Cinema
W2	Sunny	No	Rich	Tennis
W10	Sunny	No	Rich	Tennis

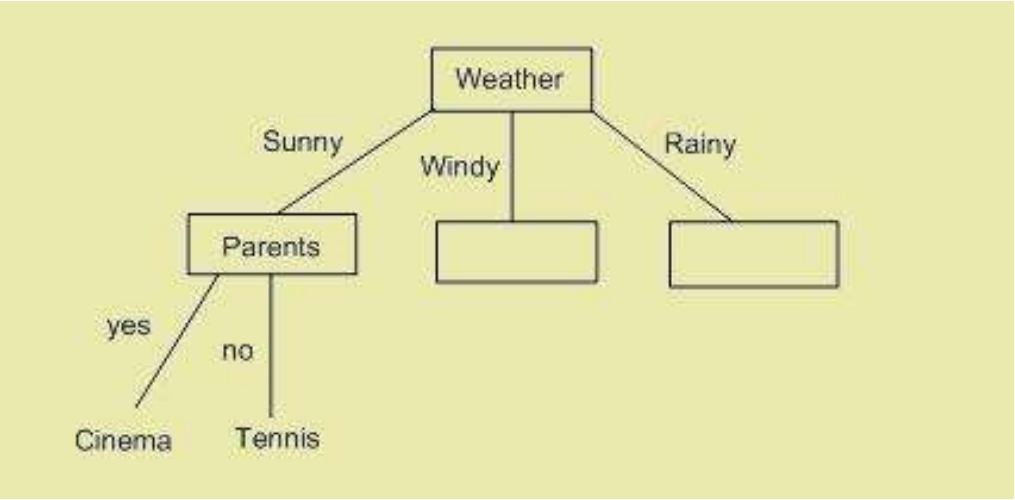
- Use the child link values to further subdivide the instances into subclasses.

$$\text{Gain}(S_{sunny}, \text{parents}) = 0.918 - (|S_{yes}|/|S|)*\text{Entropy}(S_{yes}) - (|S_{no}|/|S|)*\text{Entropy}(S_{no})$$
$$= 0.918 - (1/3)*0 - (2/3)*0 = 0.918$$

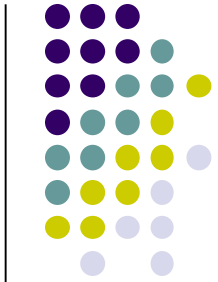
$$\text{Gain}(S_{sunny}, \text{money}) = 0.918 - (|S_{rich}|/|S|)*\text{Entropy}(S_{rich}) - (|S_{poor}|/|S|)*\text{Entropy}(S_{poor})$$
$$= 0.918 - (3/3)*0.918 - (0/3)*0 = 0.918 - 0.918 = 0$$

Decision Trees:

- If the instances in the subclass satisfy predefined criteria or if the set of remaining attribute choices for this path is null, specify the classification for new instances following this decision path.
- If the subclass does not satisfy the criteria and there is at least one attribute to further subdivide the path of the tree, let *T* be the current set of subclass instances and return to step 2.



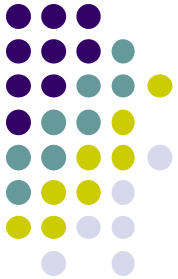
Example for Numeric Attribute



A1	A2	A3	Class
A	70	True	CLASS1
A	90	True	CLASS2
A	85	False	CLASS2
A	95	False	CLASS2
A	70	False	CLASS1
B	90	True	CLASS1
B	78	False	CLASS1
B	65	True	CLASS1
B	75	False	CLASS1
C	80	True	CLASS2
C	70	True	CLASS2
C	80	False	CLASS1
C	80	False	CLASS1
C	96	False	CLASS1

Nine samples belongs to CLASS1 and five samples to CLASS2, so entropy before splitting is
Info(T)=0.940 bits
Gain(A1)=0.940-0.694=0.246 bits
Gain(A3)=0.940-0.892=0.048 bits

Example for Numeric Attribute



Ordered Numerical attribute														
Age	65	70	70	70	75	78	80	80	80	85	90	90	95	96
Class Label	C1	C1	C1	C2	C1	C1	C2	C1	C1	C2	C2	C1	C2	C1

- To illustrate the splitting-point finding process, we could analyze the possibilities of A2 !
- After a sorting process, splitting-point can be {65, 70, 75, 78, 80, 85, 90, 95}
- Out of this eight values, the optimal splitting-point should be selected.

- For example splitting-point=80 and corresponding process of information gain computed for the test A2 (A2≤ 80 and A2>80)

- $Info_{A2}(T) = \frac{9}{14}\left(-\frac{7}{9}\log_2\frac{7}{9} - \frac{2}{9}\log_2\frac{2}{9}\right) + \frac{5}{14}\left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right)$
- $Info_{A2}(T) = 0.837$
- $Gain(A2) = 0.940 - 0.837 = 0.103\ bits$