Decision Trees

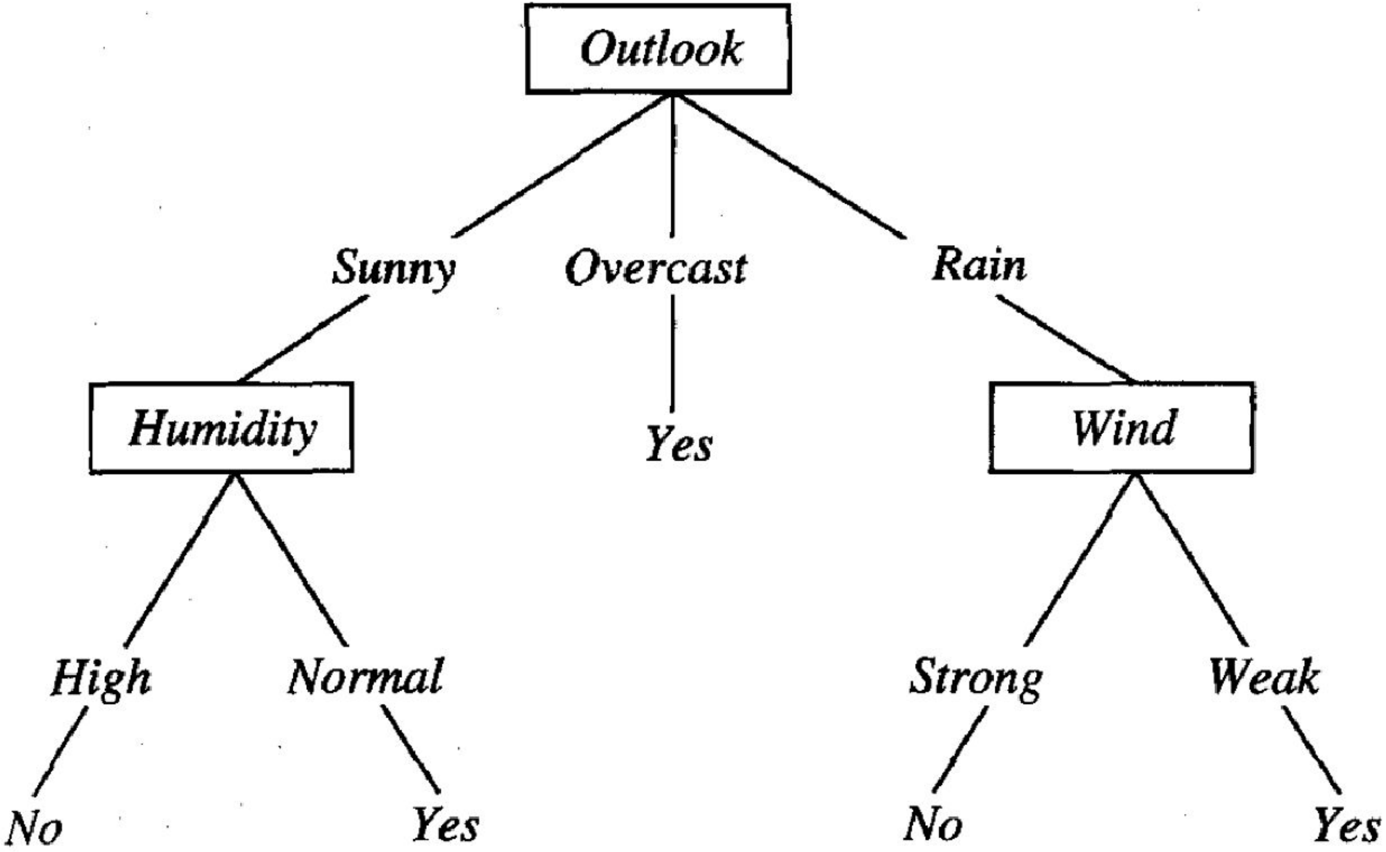
INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY
DELHI

Playing Tennis

PlayTennis: training examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
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

Playing Tennis



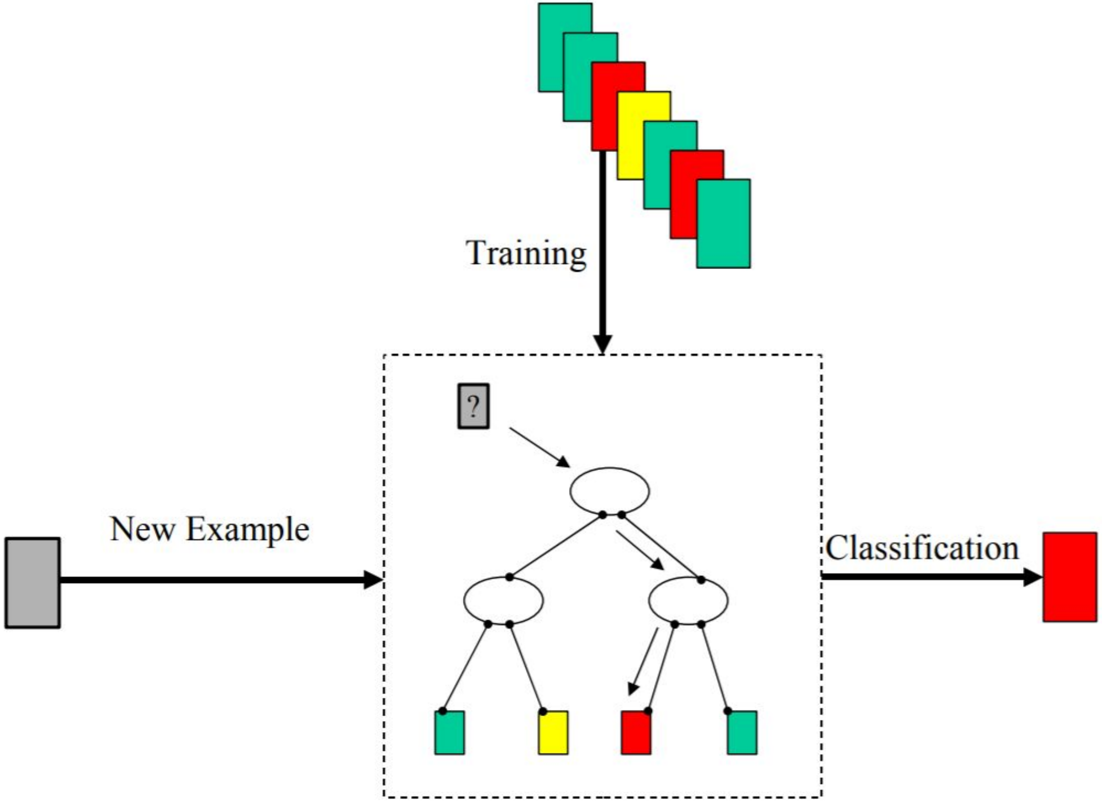
Why DT?

- Interpretable model
  - Simple and visual
- Lower computational complexity
- Exploratory analysis
  - Accuracy is not a concern
- Data is non-parametric in nature
  - Does not require any assumptions on the distribution of data

Examples

- Approve a credit card application or not
- Approve a loan application or not
- If a customer will take up a product or not
- If a transaction is fraudulent or not
- If a customer will close a mobile/telephone connection

Decision Tree Learning



Decision Tree Learning

A Decision tree for
F: <Outlook, Humidity, Wind, Temp> PlayTennis?

- Each internal node: test one attribute X_i
  - <Outlook, Humidity, Wind, Temp>
- Each branch from a node: selects one value for X_i
  - <Outlook: Sunny, Overcast, Rain>
- Each leaf node: predict Y (or $P(Y|X \in \text{leaf})$)
  - Given <Outlook: Rainy Wind:Week>
    - PlayTennis = Yes

Problem Setting

- Set of possible instances X
  - each instance x in X is a feature vector
  - e.g., <Humidity=low, Wind=weak, Outlook=rain, Temp=hot>
- Unknown target function f : X->Y
  - Y is discrete valued
- Set of function hypotheses H={ h | h : X->Y }
  - Each hypothesis h is a decision tree
  - Trees sorts 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

Top Down Induction of Decision Trees

Node =*Root*

Main loop:

1. A <— the "best" decision attribute for next node
2. Assign A as decision attribute for node
3. For each value of A, create new descendant of node
4. Sort training examples to leaf nodes
5. If training examples perfectly classified,
    a. Then STOP
    b. Else iterate over new leaf nodes

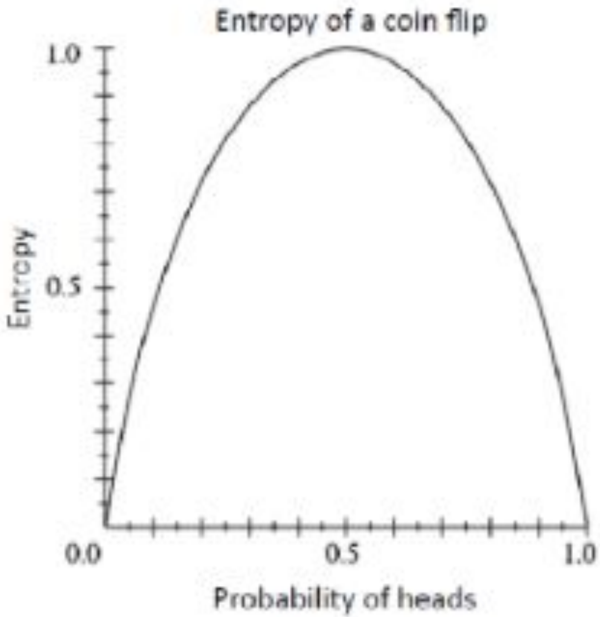
Entropy

- ***Which attribute is best?***
- The entropy -> information content
  a. More uncertainty -> More entropy [Predictability?]

$$H(X) = -\sum_{i=1}^{n} P(X=i)\log_2 P(X=i)$$

Result	Prob
H	0.5
T	0.5

Result	Prob
H	0.75
T	0.25



Conditional Entropy

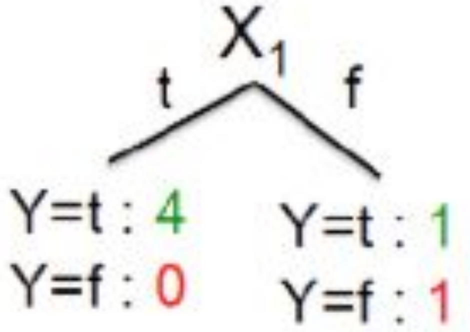
$$H(Y|X) = \sum_{j=1}^{v} P(X = x_j) \sum_{i=1}^{k} P(Y = y_i|X = x_j) log_2 P(Y = y_i|X = x_j)$$

X_1	X_2	Y
T	T	T
T	F	T
T	T	T
T	F	T
F	T	T
F	F	F

$H(Y|X_1) = ?$

$P(X_1=t) = 4/6$

$P(X_1=f) = 2/6$



$$H(Y|X_1) = -\,4/6\,(1 \log_2 1 + 0 \log_2 0)$$
$$-\,2/6\,(1/2 \log_2 1/2 + 1/2 \log_2 1/2)$$
$$= 2/6$$

Information Gain: ID3

- Purity (or impurity) is homogeneity (or heterogeneity) of the data.
    - Entropy measures the impurity of the training sample S.
- Information Gain is the expected reduction in entropy after splitting
    - IG(X) = H(Y) - H(Y|X);
    - IG(X) > 0; split is preferred

Selecting the next attribute: H, W?

PlayTennis: training examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
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

Selecting the next attribute: H, W?

$$H(Y) = -\sum_{i=1}^{k} P(Y = y_i) \log_2 P(Y = y_i)$$

$$H(Y \mid X) = -\sum_{j=1}^{v} P(X = x_j) \sum_{i=1}^{k} P(Y = y_i \mid X = x_j) \log_2 P(Y = y_i \mid X = x_j)$$

$$IG(X) = H(Y) - H(Y \mid X)$$

Selecting the next attribute: Humidity

- IG(Humidity) = H(PlayTennis) - H(PlayTennis|Humidity)
- H(PlayTennis)
  $= -(9/14)*\log_2(9/14) -(5/14)*\log_2(5/14) = 0.940$
- H(PlayTennis|Humidity)
  =
  -P(High)*[P(PlayTennis|High)*$\log_2$P(PlayTennis|High) + P(~PlayTennis|High)*$\log_2$P(~PlayTennis|High)]
  -
  P(Normal)*[P(PlayTennis|Normal)*$\log_2$P(PlayTennis|Normal) + P(~PlayTennis|Normal)*$\log_2$P(~PlayTennis|Normal)]

Selecting the next attribute: Humidity

$$H_D(Y \mid \text{high}) = -\frac{3}{7}\log_2\left(\frac{3}{7}\right) - \frac{4}{7}\log_2\left(\frac{4}{7}\right)$$
$$= 0.985$$

$$H_D(Y \mid \text{normal}) = -\frac{6}{7}\log_2\left(\frac{6}{7}\right) - \frac{1}{7}\log_2\left(\frac{1}{7}\right)$$
$$= 0.592$$

$$\text{InfoGain}(D, \text{Humidity}) = 0.940 - \left[\frac{7}{14}(0.985) + \frac{7}{14}(0.592)\right]$$
$$= 0.151$$

Selecting the next attribute: H, W?

- IG(Humidity) = 0.151
- IG(Wind) = 0.048
    - *It is better to split on humidity rather than wind as humidity has a higher information gain.*

Gini Impurity: CART (Classification And Regression Trees)

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

$$I(A) = 1 - P(A_+)^2 - P(A_-)^2$$

$$I(Al) = 1 - P(Al_+)^2 - P(Al_-)^2$$

$$I(Ar) = 1 - P(Ar_+)^2 - P(Ar_-)^2$$

$$GiniGain(A) = I(A) - p_{left} I(Al) - p_{right} I(Ar)$$

Continuous Valued Attributes: Jugaad!

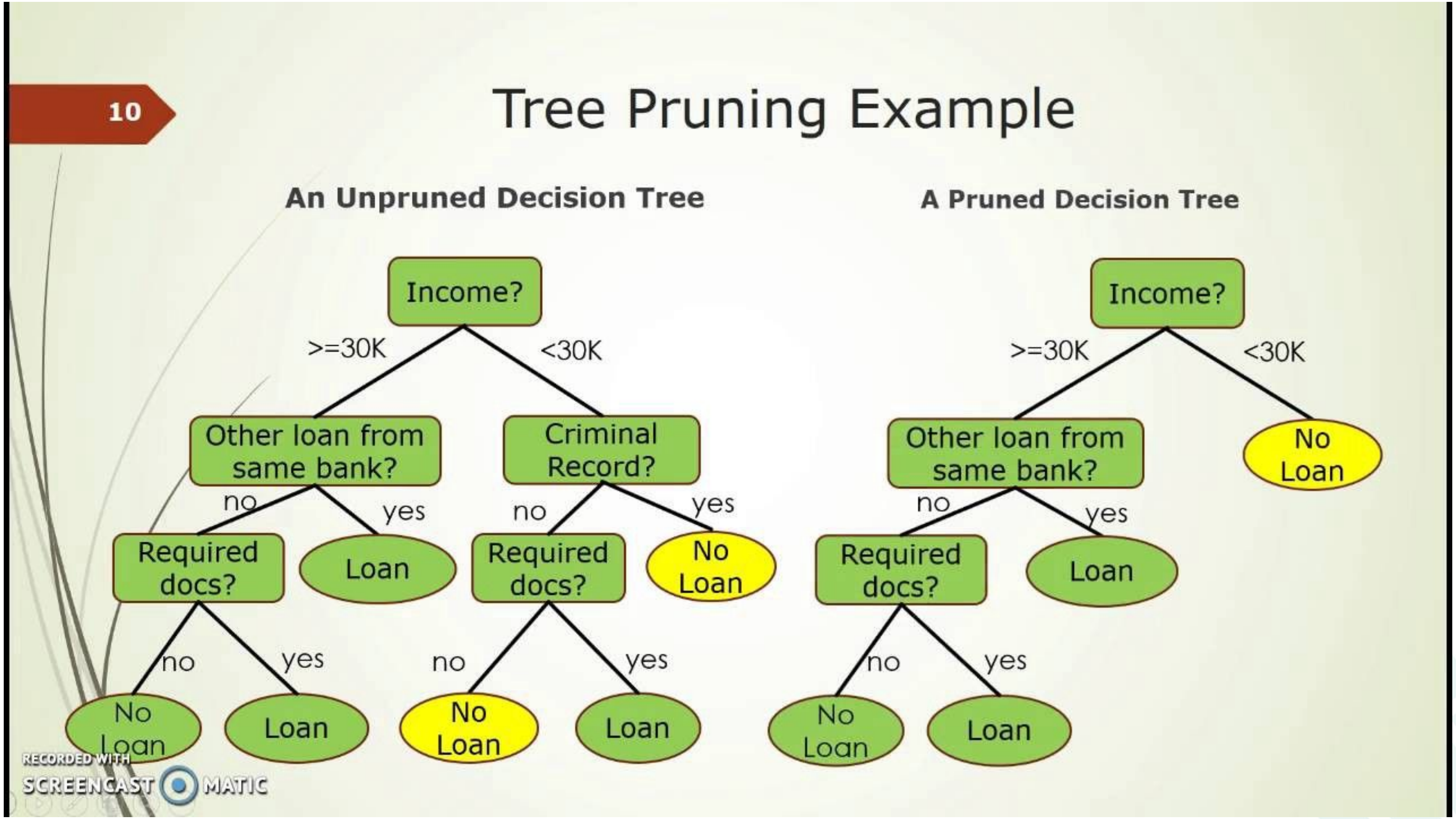
- Create a discrete attribute to test continuous!
- Temperature = 82.5
- Temperature > 70 [T, F]

Temperature	40	48	60	72	80	90
Temp_Jugaad	False	False	False	True	True	True
PlayTennis	No	No	Yes	Yes	Yes	No

Decision Trees will Overfit!

- Standard decision trees have low bias.
  - Training set error is almost zero!
  - High variance
  - Must introduce some bias towards simpler trees
- Pruning: strategies for picking simpler trees
  - Pre-pruning
    - Fixed depth
    - Fixed number of leaves
  - Post-pruning

Pruning


10
Tree Pruning Example
An Unpruned Decision Tree
A Pruned Decision Tree
Income?
>=30K
<30K
Other loan from same bank?
Criminal Record?
no
yes
Required docs?
Loan
No Loan
Income?
>=30K
<30K
Other loan from same bank?
No Loan
Required docs?
Loan
No Loan
Loan
RECORDED WITH SCREENCAST MATIC

References

1. Constructing optimal binary decision trees is NP-complete." Information Processing Letters 5.1 (1976): 15-17.
2. Entropy: https://www.cs.utexas.edu/~byoung/cs361/lecture32.pdf
3. http://www.cs.cmu.edu/~tom/10701_sp11/slides/DTreesAndOverfitting-1-11-2011_final.pdf
4. https://www.ke.tu-darmstadt.de/lehre/archiv/ws0809/mldm/dt.pdf
5. Theoretical comparison between the Gini Index and Information Gain criteria
6.