Decision Trees

Greg Grudic

(Notes borrowed from Thomas G. Dietterich and Tom Mitchell)

1

Outline

- Decision Tree Representations
  - ID3 and C4.5 learning algorithms (Quinlan 1986)
  - CART learning algorithm (Breiman et al. 1985)
- Entropy, Information Gain
- Overfitting

2

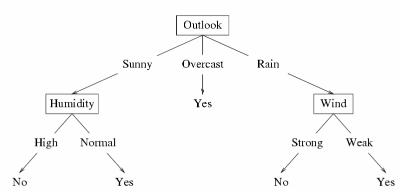
Training Example

Day	Outlook	Temp.	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

3

Decision Tree Hypothesis Space

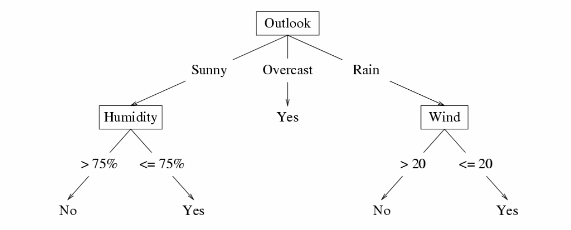
- **Internal nodes** test the value of particular features x_j and branch according to the results of the test.
- **Leaf nodes** specify the class $h(\mathbf{x})$.



Suppose the features are **Outlook** (x_1), **Temperature** (x_2), **Humidity** (x_3), and **Wind** (x_4). Then the feature vector $\mathbf{x} = (Sunny, Hot, High, Strong)$ will be classified as **No**. The **Temperature** feature is irrelevant.

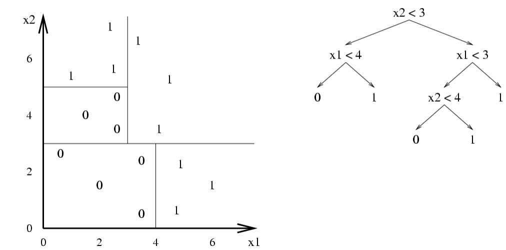
4

If the features are continuous, internal nodes may test the value of a feature against a threshold.

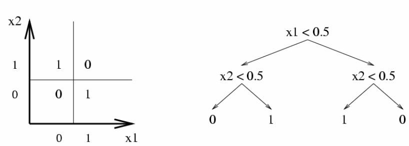


Decision Tree Decision Boundaries

Decision trees divide the feature space into axis-parallel rectangles, and label each rectangle with one of the K classes.



Decision Trees Can Represent Any Boolean Function



The tree will in the worst case require exponentially many nodes, however.

Learning Algorithm for Decision Trees

$$S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$$

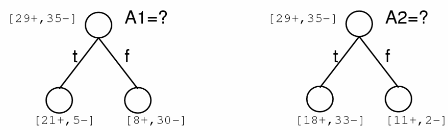
$$\mathbf{x} = (x_1, \ldots, x_d)$$
$$x_j, y \in \{0,1\}$$

$\textsc{GrowTree}(S)$
if $(y = 0$ for all $\langle \mathbf{x}, y\rangle \in S)$ **return** new leaf(0)
else if $(y = 1$ for all $\langle \mathbf{x}, y\rangle \in S)$ **return** new leaf(1)
else
choose best attribute x_j
$S_0 =$ all $\langle \mathbf{x}, y\rangle \in S$ with $x_j = 0$;
$S_1 =$ all $\langle \mathbf{x}, y\rangle \in S$ with $x_j = 1$;
return new node$(x_j, \textsc{GrowTree}(S_0), \textsc{GrowTree}(S_1))$

What happens if features are not binary? What about regression?

Choosing the Best Attribute

Which attribute is best?

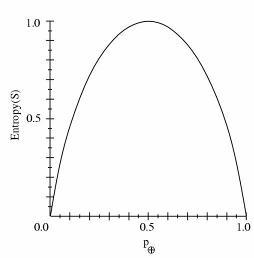


Choosing Splits Using Entropy

- $p_{\oplus}$ is the proportion of positive examples in S
- $p_{\ominus}$ is the proportion of negative examples in S
- Entropy measures the impurity of S

$$Entropy(S) \equiv -p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus}$$

Entropy



- S is a sample of training examples

Entropy

$Entropy(S)$ = expected number of bits needed to encode class ($\oplus$ or $\ominus$) of randomly drawn member of S (under the optimal, shortest-length code)

Why?

Information theory: optimal length code assigns $-\log_2 p$ bits to message having probability p.

So, expected number of bits to encode $\oplus$ or $\ominus$ of random member of S:

$$p_{\oplus}(-\log_2 p_{\oplus}) + p_{\ominus}(-\log_2 p_{\ominus})$$

$$Entropy(S) \equiv -p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus}$$

Information Gain

$Gain(S, A)$ = expected reduction in entropy due to sorting on A

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

13

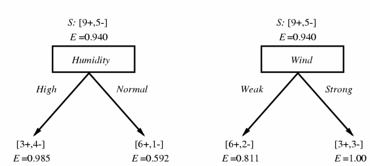
Training Example

Day	Outlook	Temp.	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

14

Selecting the Next Attribute

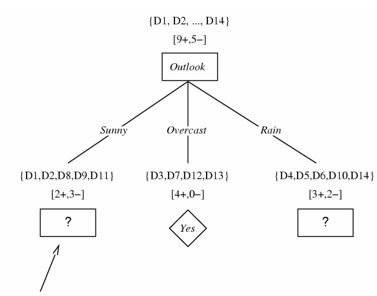
Which attribute is the best classifier?



Gain (S, Humidity)
= .940 - (7/14).985 - (7/14).592
= .151

Gain (S, Wind)
= .940 - (8/14).811 - (6/14)1.0
= .048

15



Which attribute should be tested here?

S_{sunny} = {D1,D2,D8,D9,D11}

$Gain(S_{sunny}, Humidity)$ = .970 − (3/5) 0.0 − (2/5) 0.0 = .970

$Gain(S_{sunny}, Temperature)$ = .970 − (2/5) 0.0 − (2/5) 1.0 − (1/5) 0.0 = .570

$Gain(S_{sunny}, Wind)$ = .970 − (2/5) 1.0 − (3/5) .918 = .019

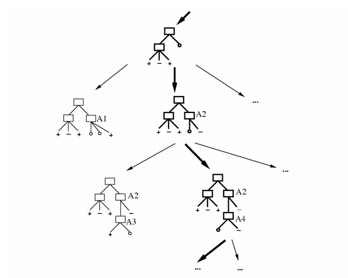
16

Non-Boolean Features

- Features with multiple discrete values
  - Multi-way splits?
  - Test for one value versus the rest?
  - Group values into disjoint sets?
- Real-valued features
  - Use thresholds?
- Regression
  - Splits based on mean squared error metric

17

Hypothesis Space Search



18

Overfitting

Consider error of hypothesis h over

- training data: $error_{train}(h)$
- entire distribution $\mathcal{D}$ of data: $error_{\mathcal{D}}(h)$

Hypothesis $h \in H$ **overfits** training data if there is an alternative hypothesis $h' \in H$ such that

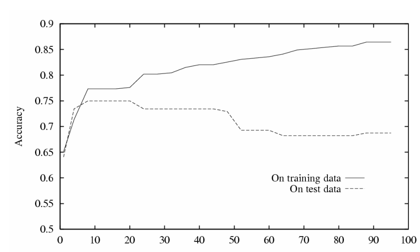
$$error_{train}(h) < error_{train}(h')$$

and

$$error_{\mathcal{D}}(h) > error_{\mathcal{D}}(h')$$

19

Overfitting in Decision Trees



20

Validation Data is Used to Control Overfitting

- Prune tree to reduce error on validation set

21