Decision Trees: Information Gain

These slides were assembled by Byron Boots, with grateful acknowledgement to Eric Eaton and the many others who made their course materials freely available online. Feel free to reuse or adapt these slides for your own academic purposes, provided that you include proper attribution.
**Last Time:** Basic Algorithm for Top-Down Learning of Decision Trees

[ID3, C4.5 by Quinlan]

\[\text{node} = \text{root of decision tree}\]

Main loop:
1. \(A \leftarrow \text{the “best” decision attribute for the next node}\).
2. Assign \(A\) as decision attribute for \(\text{node}\).
3. For each value of \(A\), create a new descendant of \(\text{node}\).
4. Sort training examples to leaf nodes.
5. If training examples are perfectly classified, stop. Else, recurse over new leaf nodes.

How do we choose which attribute is best?
Entropy $H(X)$ of a random variable $X$

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

$H(X)$ is the expected number of bits needed to encode a randomly drawn value of $X$ (under most efficient code)
Entropy

Entropy $H(X)$ of a random variable $X$

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

$H(X)$ is the expected number of bits needed to encode a randomly drawn value of $X$ (under most efficient code)

Why? Information theory:

- Most efficient code assigns $-\log_2 P(X=i)$ bits to encode the message $X=i$
- So, expected number of bits to code one random $X$ is:

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

Slide by Tom Mitchell
2-Class Cases:

Entropy \( H(x) = - \sum_{i=1}^{n} P(x = i) \log_2 P(x = i) \)

- What is the entropy of a group in which all examples belong to the same class?
  - entropy = - 1 \( \log_2 1 = 0 \)

- What is the entropy of a group with 50% in either class?
  - entropy = -0.5 \( \log_2 0.5 \) – 0.5 \( \log_2 0.5 \) = 1

Based on slide by Pedro Domingos
Sample Entropy

- $S$ is a sample of training examples
- $p_\oplus$ is the proportion of positive examples in $S$
- $p_\ominus$ is the proportion of negative examples in $S$
- Entropy measures the impurity of $S$

$$H(S) \equiv -p_\oplus \log_2 p_\oplus - p_\ominus \log_2 p_\ominus$$
### Information Gain

- We want to determine **which attribute** in a given set of training feature vectors is **most useful** for discriminating between the classes to be learned.

- **Information gain** tells us how important a given attribute of the feature vectors is.

- We will use it to decide the ordering of attributes in the nodes of a decision tree.
From Entropy to Information Gain

Entropy $H(X)$ of a random variable $X$

$$H(X) = - \sum_{i=1}^{n} P(X = i) \log_2 P(X = i)$$
From Entropy to Information Gain

Entropy $H(X)$ of a random variable $X$

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

Specific conditional entropy $H(X|Y=v)$ of $X$ given $Y=v$ :

$$H(X|Y = v) = - \sum_{i=1}^{n} P(X = i|Y = v) \log_2 P(X = i|Y = v)$$
From Entropy to Information Gain

Entropy $H(X)$ of a random variable $X$

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

Specific conditional entropy $H(X|Y=v)$ of $X$ given $Y=v$:

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

Conditional entropy $H(X|Y)$ of $X$ given $Y$:

$$H(X|Y) = \sum_{v \in \text{values}(Y)} P(Y = v) H(X|Y = v)$$
From Entropy to Information Gain

Entropy $H(X)$ of a random variable $X$

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

Specific conditional entropy $H(X|Y=v)$ of $X$ given $Y=v$:

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

Conditional entropy $H(X|Y)$ of $X$ given $Y$:

$$H(X|Y) = \sum_{v \in \text{values}(Y)} P(Y = v) H(X|Y = v)$$

Mutual information (aka Information Gain) of $X$ and $Y$:

$$I(X, Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$$
Information Gain is the expected reduction in entropy of target variable Y for data sample S, due to sorting
Calculating Information Gain

Information Gain = \text{entropy(parent)} - \left[ \text{average entropy(children)} \right]

\text{child entropy}

\begin{align*}
\text{parent entropy} & = -\left( \frac{14}{30} \cdot \log_2 \frac{14}{30} \right) - \left( \frac{16}{30} \cdot \log_2 \frac{16}{30} \right) = 0.996 \\
\text{impurity} & = \frac{14}{30} \cdot 0.787 + \frac{16}{30} \cdot 0.391 = 0.615
\end{align*}

(Weighted) Average Entropy of Children = \left( \frac{17}{30} \cdot 0.787 \right) + \left( \frac{13}{30} \cdot 0.391 \right) = 0.615

Information Gain = 0.996 - 0.615 = 0.38

Based on slide by Pedro Domingos
Entropy-Based Automatic Decision Tree Construction

Training Set X
x1=(f11, f12, ..., f1m)
x2=(f21, f22, ..., f2m)
  .
  .
xn=(fn1, f22, ..., f2m)

Node 1
What feature should be used?

What values?

Quinlan suggested information gain in his ID3 system
Using Information Gain to Construct a Decision Tree

Choose the attribute $A$ with highest information gain for the full training set at the root of the tree.

Construct child nodes for each value of $A$. Each has an associated subset of vectors in which $A$ has a particular value.
Sample Dataset (was Tennis Played?)

- Columns denote features $X_i$
- Rows denote labeled instances $\langle x_i, y_i \rangle$
- Class label denotes whether a tennis game was played

<table>
<thead>
<tr>
<th>Predictors</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Outlook</strong></td>
<td><strong>Temperature</strong></td>
</tr>
<tr>
<td>Sunny</td>
<td>Hot</td>
</tr>
<tr>
<td>Sunny</td>
<td>Hot</td>
</tr>
<tr>
<td>Overcast</td>
<td>Hot</td>
</tr>
<tr>
<td>Rain</td>
<td>Mild</td>
</tr>
<tr>
<td>Rain</td>
<td>Cool</td>
</tr>
<tr>
<td>Rain</td>
<td>Cool</td>
</tr>
<tr>
<td>Overcast</td>
<td>Cool</td>
</tr>
<tr>
<td>Sunny</td>
<td>Mild</td>
</tr>
<tr>
<td>Sunny</td>
<td>Cool</td>
</tr>
<tr>
<td>Rain</td>
<td>Mild</td>
</tr>
<tr>
<td>Sunny</td>
<td>Mild</td>
</tr>
<tr>
<td>Overcast</td>
<td>Mild</td>
</tr>
<tr>
<td>Overcast</td>
<td>Hot</td>
</tr>
<tr>
<td>Rain</td>
<td>Mild</td>
</tr>
</tbody>
</table>
Selecting the Next Attribute

Which attribute is the best classifier?

- **Humidity**
  - High: [3+, 4-]
  - Normal: [6+, 1-]

- **Wind**
  - Weak: [6+, 2-]
  - Strong: [3+, 3-]

$S: [9+, 5-]$
$E = 0.940$
Selecting the Next Attribute

Which attribute is the best classifier?

\[ S: [9+, 5-] \]
\[ E = 0.940 \]

- **Humidity**
  - **High**
  - **Normal**
    - \[ [3+, 4-] \]
    - \[ E = 0.985 \]
    - \[ [6+, 1-] \]
    - \[ E = 0.592 \]

\[ Gain (S, \text{Humidity}) = .940 - (7/14).985 - (7/14).592 = .151 \]

\[ S: [9+, 5-] \]
\[ E = 0.940 \]

- **Wind**
  - **Weak**
  - **Strong**
    - \[ [6+, 2-] \]
    - \[ E = 0.811 \]
    - \[ [3+, 3-] \]
    - \[ E = 1.00 \]

\[ Gain (S, \text{Wind}) = .940 - (8/14).811 - (6/14)1.0 = .048 \]
Which attribute should be tested here?

\[ S_{\text{sunny}} = \{D_1, D_2, D_8, D_9, D_{11}\} \]

\[
\text{Gain} \left(S_{\text{sunny}}, \text{Humidity}\right) = .970 - \frac{3}{5} \cdot 0.0 - \frac{2}{5} \cdot 0.0 = .970
\]

\[
\text{Gain} \left(S_{\text{sunny}}, \text{Temperature}\right) = .970 - \frac{2}{5} \cdot 0.0 - \frac{2}{5} \cdot 1.0 - \frac{1}{5} \cdot 0.0 = .570
\]

\[
\text{Gain} \left(S_{\text{sunny}}, \text{Wind}\right) = .970 - \frac{2}{5} \cdot 1.0 - \frac{3}{5} \cdot .918 = .019
\]
Which Tree Should We Output?

• ID3 performs heuristic search through space of decision trees

• It stops at smallest acceptable tree. Why?

Slide by Tom Mitchell