Classification – Decision Trees

UROŠ KRČADINAC

EMAIL: uros@krcadinac.com
URL: http://www.krcadinac.com
What is classification?

• The task of defining a class which an instance belongs to
  • an instance is defined by a set of attributes;
  • a set of possible classes is given
Decision trees

Example: Deciding whether to buy a car

1. Road tested
   - no
     - DO NOT BUY
   - no

2. Mileage
   - big
     - old
     - DO NOT BUY
   - small
     - young
     - BUY
ID3 algorithm

• ID3 - Iterative Dichotomiser 3

• One of the best known algorithms for generating decision trees based on the set of examples (dataset)

• Resulting tree can be used for classifying future (unknown) instances
Example – Forecasting whether the play will be played

ToPlayOrNotToPlay.arff dataset

<table>
<thead>
<tr>
<th>Outlook</th>
<th>Temp.</th>
<th>Humidity</th>
<th>Windy</th>
<th>Play</th>
</tr>
</thead>
<tbody>
<tr>
<td>sunny</td>
<td>hot</td>
<td>high</td>
<td>false</td>
<td>no</td>
</tr>
<tr>
<td>sunny</td>
<td>hot</td>
<td>high</td>
<td>true</td>
<td>no</td>
</tr>
<tr>
<td>overcast</td>
<td>hot</td>
<td>high</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>rainy</td>
<td>mild</td>
<td>high</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>rainy</td>
<td>cool</td>
<td>normal</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>rainy</td>
<td>cool</td>
<td>normal</td>
<td>true</td>
<td>no</td>
</tr>
<tr>
<td>overcast</td>
<td>cool</td>
<td>normal</td>
<td>true</td>
<td>yes</td>
</tr>
<tr>
<td>sunny</td>
<td>mild</td>
<td>high</td>
<td>false</td>
<td>no</td>
</tr>
<tr>
<td>sunny</td>
<td>cool</td>
<td>normal</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>rainy</td>
<td>mild</td>
<td>normal</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>sunny</td>
<td>mild</td>
<td>normal</td>
<td>true</td>
<td>yes</td>
</tr>
<tr>
<td>overcast</td>
<td>mild</td>
<td>high</td>
<td>true</td>
<td>yes</td>
</tr>
<tr>
<td>overcast</td>
<td>hot</td>
<td>normal</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>rainy</td>
<td>mild</td>
<td>high</td>
<td>true</td>
<td>no</td>
</tr>
</tbody>
</table>
Top-down approach

Recursive divide-and-conquer:

- **Select** attribute for root node
  - Create branch for each possible attribute value
- **Split** instances into subsets
  - One for each branch extending from the node
- **Repeat** recursively for each branch
  - using only instances that reach the branch
- **Stop**
  - if all instances have the same class
Which attribute to select?
Which attribute to select?

- **Aim**: to get the smallest tree

- **Information theory**: measure information in bits. Founder is Claude Shannon, American mathematician and scientist 1916 - 2001

- Entropy $H(S)$ can be calculated by using the formula:

$$H(S) = -\sum_{i=1}^{N} p_i \log_2 p_i$$

where:

- $S$ – set of all instances in the dataset
- $N$ – number of distinct class values
- $p_i$ – event probability
Dataset entropy

- From the total of 14 instances we have:
  - 9 instances “yes”
  - 5 instances “no”

\[
H(S) = - \sum_{i=1}^{N} p_i \log_2 p_i
\]

\[
H(S) = - \frac{9}{14} \log_2 \frac{9}{14} - \frac{5}{14} \log_2 \frac{5}{14} = 0.940
\]
Information gain

- Information gain $\text{Gain}(A, S)$ of an attribute $A$ over the set of instances $S$ represents an amount of information we would gain by knowing the value of the attribute $A$. Information gain represents the difference between an entropy before branching and entropy after branching over the attribute $A$. 
Information gain

\[Gain(A, S) = H(S) - \sum_{j=1}^{v} \frac{|S_j|}{|S|} \cdot H(S_j) = H(S) - H(A, S)\]

where:

- \(H(S)\) – entropy of the whole dataset \(S\)
- \(|S_j|\) – number of instances with \(j\) value of an attribute \(A\)
- \(|S|\) – total number of instances in dataset \(S\)
- \(v\) – set of distinct values of an attribute \(A\)
- \(H(S_j)\) – Entropy of subset of instances for attribute \(A\)
- \(H(A, S)\) – entropy of an attribute \(A\)

Choose an attribute with highest information gain.
Information gain of attribute “windy”

- From the total of 14 instances we have:
  - 6 instances “true”
  - 8 instances “false”

\[
\text{Gain}(A, S) = H(S) - \sum_{j=1}^{v} \frac{|S_j|}{|S|} \cdot H(S_j)
\]

\[
\text{Gain}(A_{\text{Windy}}, S) = 0.940 - \frac{8}{14} \cdot \left( - \left( \frac{6}{8} \cdot \log_2 \frac{6}{8} + \frac{2}{8} \cdot \log_2 \frac{2}{8} \right) \right) + \frac{6}{14} \cdot \left( - \left( \frac{3}{6} \cdot \log_2 \frac{3}{6} + \frac{3}{6} \cdot \log_2 \frac{3}{6} \right) \right) = 0.048
\]
**Information gain of attribute “outlook”**

- From the total of 14 instances we have:
  - 5 instances “sunny”
  - 4 instances “overcast”
  - 5 instances “rainy”

\[
\text{Gain}(A_{\text{outlook}}, S) = 0.940 - \\
\frac{5}{14} \cdot \left(- \left( \frac{2}{5} \log_2 \frac{2}{5} + \frac{3}{5} \log_2 \frac{3}{5} \right) \right) + \\
\frac{4}{14} \cdot \left(- \left( \frac{4}{4} \log_2 \frac{4}{4} \right) \right) + \\
\frac{5}{14} \cdot \left(- \left( \frac{3}{5} \cdot \log_2 \frac{3}{5} + \frac{2}{5} \cdot \log_2 \frac{2}{5} \right) \right) = 0.247
\]
Information gain of attribute “humidity”

- From the total of 14 instances we have:
  - 7 instances “high”
  - 7 instances “normal”

\[
Gain(A_{Humidity}, S) = 0.940 - \\
\frac{7}{14} \cdot \left( - \left( \frac{3}{7} \cdot \log_2 \frac{3}{7} + \frac{4}{7} \cdot \log_2 \frac{4}{7} \right) \right) + \\
\frac{7}{14} \cdot \left( - \left( \frac{6}{7} \cdot \log_2 \frac{6}{7} + \frac{1}{7} \cdot \log_2 \frac{1}{7} \right) \right) = 0.151
\]
Information gain of attribute “temperature”

- From the total of 14 instances we have:
  - 4 instances “hot”
  - 6 instances “mild”
  - 4 instances “cool”

\[
\text{Gain}(A_{Temperature}, S) = 0.940 - \\
\frac{4}{14} \cdot \left( - \left( \frac{2}{4} \cdot \log_2 \frac{2}{4} + \frac{2}{4} \cdot \log_2 \frac{2}{4} \right) \right) + \\
\frac{6}{14} \cdot \left( - \left( \frac{4}{6} \cdot \log_2 \frac{4}{6} + \frac{2}{6} \cdot \log_2 \frac{2}{6} \right) \right) + \\
\frac{4}{14} \cdot \left( - \left( \frac{3}{4} \cdot \log_2 \frac{3}{4} + \frac{1}{4} \cdot \log_2 \frac{1}{4} \right) \right) = 0.029
\]
Which attribute to select?

0.247

outlook

sunny

yes

yes

yes

no

no

overcast

yes

yes

yes

yes

no

no

rainy

yes

yes

yes

no

no

0.048

windy

false

yes

yes

yes

yes

no

no

true

yes

yes

yes

no

no

0.151

humidity

high

yes

yes

yes

no

no

normal

yes

yes

yes

no

no

0.029

temperature

hot

yes

yes

no

no

cool

yes

yes

yes

no
Iteration 2: Repeat recursively for each branch

0.571

0.020

0.971
Iteration 2: Repeat recursively for each branch

- Outlook:
  - Sunny
  - Rainy
  - Overcast

- Humidity:
  - High
  - Normal

- Yes
- No
Iteration 2: Repeat recursively for each branch
Weka

- Software for data mining in Java
- Set of algorithms for machine learning and data mining
- Developed at the University of Waikato, New Zealand
- Open-source
- Website: http://www.cs.waikato.ac.nz/ml/weka
ARFF file

- Attribute-Relation File Format – ARFF
- Textual file

```
@relation TPONTPNom

@attribute Outlook {sunny, overcast, rainy}
@attribute Temp. {hot, mild, cool}
@attribute Humidity {high, normal}
@attribute Windy {'false', 'true'}
@attribute Play {no, yes}

@data
sunny, hot, high, 'false', no
sunny, hot, high, 'true', no
overcast, hot, high, 'false', yes
...
```

Attributes can be:
- Numerical
- Nominal
Datasets used for this class

- Datasets from the website Technology Forge:

  http://www.technologyforge.net/Datasets
Loading dataset
Dataset overview
J48 class

• Implementation of C4.5 algorithm for generating decision trees.

• C4.5 algorithm is an extension of the ID3 algorithm.

• Extending the ID3 algorithm by:
  • supporting continual and discrete attributes
  • supporting missing values (excludes instances with missing values when calculating entropy and information gain)
  • tree pruning

Choosing J48 classifier
Training the classifier

Classifier

Choose J48 -C 0.25 -M 2

Test options

- Use training set
- Supplied test set
- Cross-validation Folds 10
- Percentage split % 66

More options...

(Nom) Play

Start Stop
Overview of classification results

Classifier: J48 -C 0.25 -M 2

Test options:
- Cross-validation: Folds 10

Classifier output:

--- Summary ---
Correctly Classified Instances 7  50 %
Incorrectly Classified Instances 7  50 %
Kappa statistic -0.0426
Mean absolute error 0.4167
Root mean squared error 0.5984
Relative absolute error 87.5 %
Root relative squared error 121.2987 %
Total Number of Instances 14

--- Detailed Accuracy By Class ---

<table>
<thead>
<tr>
<th>Class</th>
<th>TP Rate</th>
<th>FP Rate</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>0.4</td>
<td>0.444</td>
<td>0.333</td>
<td>0.4</td>
<td>0.364</td>
</tr>
<tr>
<td>b</td>
<td>0.556</td>
<td>0.6</td>
<td>0.625</td>
<td>0.556</td>
<td>0.588</td>
</tr>
</tbody>
</table>

Weighted Avg. 0.5 0.544 0.521 0.5 0.508

--- Confusion Matrix ---

a  b  <-- classified as
2  3  | a = no
4  5  | b = yes
Confusion Matrix

<table>
<thead>
<tr>
<th>Predicted Class</th>
<th>Yes</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>TP</td>
<td>FN</td>
</tr>
<tr>
<td>No</td>
<td>FP</td>
<td>TN</td>
</tr>
</tbody>
</table>

TP = True Positive
FP = False Positive
TN = True Negative
FN = False Negative

--- Confusion Matrix ---

a b  <-- classified as
2 3  | a = no
4 5  | b = yes
Precision, Recall and F measure

=== Detailed Accuracy By Class ===

<table>
<thead>
<tr>
<th>TP Rate</th>
<th>FP Rate</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Measure</th>
<th>ROC Area</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.4</td>
<td>0.444</td>
<td>0.333</td>
<td>0.4</td>
<td>0.364</td>
<td>0.633</td>
<td>no</td>
</tr>
<tr>
<td>0.556</td>
<td>0.6</td>
<td>0.625</td>
<td>0.556</td>
<td>0.588</td>
<td>0.633</td>
<td>yes</td>
</tr>
</tbody>
</table>

Weighted Avg. 0.5 0.544 0.521 0.5 0.508 0.633

True Positives Rate

False Positives Rate

Precision = $\frac{TP}{TP + FP}$

Recall = $\frac{TP}{TP + NP}$

F measure = $\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$
Visualizing decision tree
Visualizing decision tree
Tree pruning
Tree pruning

- Pruning is the process of reducing the tree size by removing sub-trees that adds little to the efficiency of the decision tree. Sub-tree whose classification error is bigger than the error of a leaf node in its place is removed and replaced by the leaf node.
Example 2 – “Diabetes” dataset

- Dataset “Pima Indians Diabetes Database” contains data about female Pima Indians aged 21 years or higher and tested for diabetes. Dataset was donated by the Johns Hopkins University, Maryland, USA.

- There are total of 768 instances described by 8 numerical attributes about patient conditions and annotated with a class determining whether patients were positive or negative for diabetes.

- Our goal is to predict whether a new patient will be diagnosed positive or negative.
Example 3 – “Breast cancer” dataset

- “Breast cancer data” dataset contains information about patients diagnosed with breast cancer donated by Institute of Oncology, Ljubljana, Slovenia.

- This data set includes 201 instances of one class and 85 instances of another class. The instances are described by 9 attributes, some of which are linear and some are nominal.

- Our goal is to predict whether there will be recurrent events or not.
Credits

Weka Tutorials and Assignments @ The Technology Forge

- Link: http://www.technologyforge.net/WekaTutorials/

"Data Mining with Weka" and "More Data Mining with Weka": MOOCs from the University of Waikato. A self-paced session of "Data Mining with Weka" runs until 23 October June 2015.

- Link: https://www.youtube.com/user/WekaMOOC/
(Anonymous) survey for your comments ad suggestions:

http://goo.gl/cqdp3l
Questions?

UROŠ KRČADINAC

EMAIL: uros@krcadinac.com

URL: http://www.krcadinac.com