An Algorithm for Building Decision Trees

- C4.5 is a computer program for inducing classification rules in the form of decision trees from a set of given instances.
- C4.5 is a software extension of the basic ID3 algorithm designed by Quinlan.
Algorithm Description

- Select one attribute from a set of training instances
- Select an initial subset of the training instances
- Use the attribute and the subset of instances to build a decision tree
- Use the rest of the training instances (those not in the subset used for construction) to test the accuracy of the constructed tree
- If all instances are correctly classified – stop
- If an instance is incorrectly classified, add it to the initial subset and construct a new tree
- Iterate until
  - A tree is built that classifies all instances correctly
  - OR
  - A tree is built from the entire training set
Simplified Algorithm

- Let T be the set of training instances
- Choose an attribute that best differentiates the instances contained in T (C4.5 uses the Gain Ratio to determine)
- 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
Example

Credit Card Promotion Data from Chapter 2
### Example – Credit Card Promotion Data Descriptions

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Value Description</th>
<th>Numeric Values</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Income Range</td>
<td>20-30K, 30-40K, 40-50K, 50-60K</td>
<td>20000, 30000, 40000, 50000</td>
<td>Salary range for an individual credit card holder</td>
</tr>
<tr>
<td>Magazine Promotion</td>
<td>Yes, No</td>
<td>1, 0</td>
<td>Did card holder participate in magazine promotion offered before?</td>
</tr>
<tr>
<td>Watch Promotion</td>
<td>Yes, No</td>
<td>1, 0</td>
<td>Did card holder participate in watch promotion offered before?</td>
</tr>
<tr>
<td>Life Ins Promotion</td>
<td>Yes, No</td>
<td>1, 0</td>
<td>Did card holder participate in life insurance promotion offered before?</td>
</tr>
<tr>
<td>Credit Card Insurance</td>
<td>Yes, No</td>
<td>1, 0</td>
<td>Does card holder have credit card insurance?</td>
</tr>
<tr>
<td>Sex</td>
<td>Male, Female</td>
<td>1, 0</td>
<td>Card holder’s gender</td>
</tr>
<tr>
<td>Age</td>
<td>Numeric</td>
<td>Numeric</td>
<td>Card holder’s age in whole years</td>
</tr>
</tbody>
</table>
Problem to be Solved from Data

- Acme Credit Card Company is going to do a life insurance promotion – sending the promo materials with billing statements. They have done a similar promotion in the past, with results as represented by the data set. They want to target the new promo materials to credit card holders similar to those who took advantage of the prior life insurance promotion.
- Use supervised learning with output attribute = life insurance promotion to develop a profile for credit card holders likely to accept the new promotion.
## Sample of Credit Card Promotion Data (from Table 2.3)

<table>
<thead>
<tr>
<th>Income Range</th>
<th>Magazine Promo</th>
<th>Watch Promo</th>
<th>Life Ins Promo</th>
<th>CC Ins</th>
<th>Sex</th>
<th>Age</th>
</tr>
</thead>
<tbody>
<tr>
<td>40-50K</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>45</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>40</td>
</tr>
<tr>
<td>40-50K</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>42</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Male</td>
<td>43</td>
</tr>
<tr>
<td>50-60K</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>38</td>
</tr>
<tr>
<td>20-30K</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Female</td>
<td>55</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Male</td>
<td>35</td>
</tr>
<tr>
<td>20-30K</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>27</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>43</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>41</td>
</tr>
</tbody>
</table>
Problem Characteristics

• Life insurance promotion is the output attribute
• Input attributes are income range, credit card insurance, sex, and age
  • Attributes related to the instance’s response to other promotions is not useful for prediction because new credit card holders will not have had a chance to take advantage of these prior offers (except for credit card insurance which is always offered immediately to new card holders)
• Therefore, magazine promo and watch promo are not relevant for solving the problem at hand – disregard – do not include this data in data mining
Apply the Simplified C4.5 Algorithm to the Credit Card Promotion Data

<table>
<thead>
<tr>
<th>Income Range</th>
<th>Magazine Promo</th>
<th>Watch Promo</th>
<th>Life Ins Promo</th>
<th>CC Ins</th>
<th>Sex</th>
<th>Age</th>
</tr>
</thead>
<tbody>
<tr>
<td>40-50K</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>45</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>40</td>
</tr>
<tr>
<td>40-50K</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>42</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Male</td>
<td>43</td>
</tr>
<tr>
<td>50-60K</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>38</td>
</tr>
<tr>
<td>20-30K</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Female</td>
<td>55</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Male</td>
<td>35</td>
</tr>
<tr>
<td>20-30K</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>27</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>43</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>41</td>
</tr>
</tbody>
</table>

Training set = 15 instances (see handout)
Apply the Simplified C4.5 Algorithm to the Credit Card Promotion Data

<table>
<thead>
<tr>
<th>Income Range</th>
<th>Magazine Promo</th>
<th>Watch Promo</th>
<th>Life Ins Promo</th>
<th>CC Ins</th>
<th>Sex</th>
<th>Age</th>
</tr>
</thead>
<tbody>
<tr>
<td>40-50K</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>45</td>
</tr>
<tr>
<td>40-50K</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>42</td>
</tr>
<tr>
<td>50-60K</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>38</td>
</tr>
<tr>
<td>20-30K</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Female</td>
<td>55</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Male</td>
<td>35</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>43</td>
</tr>
<tr>
<td>30-40K</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>41</td>
</tr>
</tbody>
</table>

Step 2: Which input attribute best differentiates the instances?
Apply Simplified C4.5

For each case (attribute value), how many instances of Life Insurance Promo = Yes and Life Insurance Promo = No?
Apply Simplified C4.5

For each branch, choose the most frequently occurring decision. If there is a tie, then choose Yes, since there are more overall Yes instances (9) than No instances (6) with respect to Life Insurance Promo.
Apply Simplified C4.5

Evaluate the classification model (the tree) on the basis of accuracy. How many of the 15 training instances are classified correctly by this tree?
Apply Simplified C4.5

- Tree accuracy = 11/15 = 73.3%
- Tree cost = 4 branches for the computer program to use
- Goodness score for Income Range attribute is 11/15/4 = 0.183
- Including Tree “cost” to assess goodness lets us compare trees
Apply Simplified C4.5
Consider a Different Top-Level Node

For each case (attribute value), how many instances of Life Insurance Promo = Yes and Life Insurance Promo = No?
Apply Simplified C4.5

For each branch, choose the most frequently occurring decision. If there is a tie, then choose Yes, since there are more total Yes instances (9) than No instances (6).
Apply Simplified C4.5

Evaluate the classification model (the tree). How many of the 15 training instances are classified correctly by this tree?
Apply Simplified C4.5

- Tree accuracy = 9/15 = 60.0%
- Tree cost = 2 branches for the computer program to use
- Goodness score for Income Range attribute is 9/15/2 = 0.300
- Including Tree “cost” to assess goodness lets us compare trees
Apply Simplified C4.5

What’s problematic about this?
Apply Simplified C4.5

How many instances for each case?
A binary split requires the addition of only two branches. Why 43?
Apply Simplified C4.5

For each branch, choose the most frequently occurring decision. If there is a tie, then choose Yes, since there are more total Yes instances (9) than No instances (6).
Apply Simplified C4.5

For this data, a binary split at 43 results in the best “score”.

Life Insurance = Yes  
Life Insurance = No
Apply Simplified C4.5

- Tree accuracy = $\frac{12}{15} = 80.0\%$
- Tree cost = 2 branches for the computer program to use
- Goodness score for Income Range attribute is $\frac{12}{15} \div 2 = 0.400$
- Including Tree “cost” to assess goodness lets us compare trees
How many instances for each case?
A binary split requires the addition of only two branches. Why 43?
Apply Simplified C4.5

For each branch, choose the most frequently occurring decision. If there is a tie, then choose Yes, since there are more total Yes instances (9) than No instances (6).
Apply Simplified C4.5

Evaluate the classification model (the tree). How many of the 15 training instances are classified correctly by this tree?
Apply Simplified C4.5

- Tree accuracy = $\frac{11}{15} = 73.3\%$
- Tree cost = 2 branches for the computer program to use
- Goodness score for Income Range attribute is $\frac{11}{15}/2 = 0.367$
- Including Tree “cost” to assess goodness lets us compare trees
Apply Simplified C4.5

Model “goodness” = 0.183

Model “goodness” = 0.30

Model “goodness” = 0.40

Model “goodness” = 0.367
Apply Simplified C4.5

- Consider each branch and decide whether to terminate or add an attribute for further classification
- Different termination criteria make sense
  - If the instances following a branch satisfy a predetermined criterion, such as a certain level of accuracy, then the branch becomes a terminal path
  - No other attribute adds information
Apply Simplified C4.5

- 100% accuracy for >43 branch

```
Age

<=43
  9 Yes
  3 No

>43
  0 Yes
  3 No
```
Apply Simplified C4.5

• Production rules are generated by following to each terminal branch
Apply Simplified C4.5

If Age $\leq 43$ AND Sex = Male AND CCIns = No
Then Life Insurance Promo = No
Accuracy = 75%
Coverage = 26.7%
Apply Simplified C4.5

Simplify the Rule

If Sex = Male AND CCIns = No
Then Life Insurance Promo = No
Accuracy = 83.3%
Coverage = 40.0%
This rule is more general, more accurate
Decision Tree Algorithm Implementations

- Automate the process of rule creation
- Automate the process of rule simplification
- Choose a default rule – the one that states the classification of an instance that does not meet the preconditions of any listed rule
Example – Use WEKA
Example – Use WEKA
Example – Use WEKA

- Download CreditCardPromotion.zip from Blackboard and extract CreditCardPromotion.arff
Example – Use WEKA

- Why remove magazine promotion and watch promotion from the analysis?
Example – Use WEKA
Example – Use WEKA

See algorithm options through Choose

Choose PART under rules
Example – Use WEKA
Example – Use WEKA

Classifier output

--- Classifier model (full training set) ---

PART decision list
-------------------
Sex = Female: Yes (7.0/1.0)
Credit Card Insurance = No: No (6.0/1.0)
: Yes (2.0)
Number of Rules: 3

Time taken to build model: 0 seconds
Example – Use WEKA

• Decision tree equivalent of rules generated by PART

- Sex
  - Female
  - Male
    - Life Insurance = Yes (7/1)
    - Credit Card Insurance
      - Yes
        - Life Insurance = Yes (2/0)
      - No
        - Life Insurance = No (6/1)
Example – Use WEKA
Decision Trees – Advantages

**Pluses**
- Easy to understand
- Map readily to production rules
- No prior assumptions about the nature of the data needed
  - e.g., no assumption of normally distributed data needed
- Apply to categorical data, but numerical data can be binned for application

**Issues**
- Output attribute must be categorical
- Only one output attribute
- Sufficiently robust?
  - Change in one training set data item can change outcome
- Numerical attributes can create complex decision trees (due to split algorithms)
Decision Trees

By Susan Miertschin