10. Decision Trees
Informal Definition:
Telling things apart
Nominal data

• No numeric feature vector
• Just a list or properties:
  • Banana: longish, yellow
  • Apple: round, medium sized, different colors like red and green but not blue
Basic Classification Tree

```
          root
        /    \
  Color?  yellow  red
   /  \     /   \  
Size? green  Shape? Size?
    \  /     \    /  
   big  medium small  round  thin  medium  small
     \  /     \    /   \  /   \  
Watermelon Apple Grape Size? Banana Apple Taste?
   \  /     \    /   \    /   \  
   big  small  big  small  sweet  sour
     \  /     \    /   \  /   \  
Grapefruit Lemon  Grapevine
```
Advantages of Decision Trees

- Trees can be interpreted
- Possibility to include prior knowledge
Conversion to Binary Tree

Every tree can be converted to binary tree
Consider binary trees for simplicity
Issues when building a decision tree

- Structure of tree (binary?)
- What are the right questions to ask?
- When should be add a node (splitting)?
- When should we remove a node (pruning)?
Types of questions
(also for continuous data)
Questions for nominal data

(Notes!)

E.g. for text classification:

• Is key word x present?
• Is frequency of word x larger than y?
• Is a key phrase present?
• ...

• It all depends on your intuition about the task
• Usually asking for presence/absence or words is a good baseline
Application of CART to Feature Vectors

- Feature Vector \((x_1...x_N)\)
- Typ of questions: \(x_i < a\)
  \[\implies\] ordinary binary tree
Decision boundary induced by ordinary binary tree in 2 dimensions

- Decision boundaries always parallel to the axis
- Only very simple decision boundaries can be modelled
Decision boundary induced by ordinary binary tree in 3 dimensions
Difficult case for ordinary binary tree

How does the decision boundary look like?
Difficult case for ordinary binary tree

Simple questions of type $x_i < a$ are not capable of efficiently describing sophisticated decision boundaries.
Better Decision Boundary

What's the corresponding question and the resulting tree?
Corresponding Tree and Question

However, the number of possible questions of this type is huge?
Binary space partition (BSP) trees

- Type of questions:
  \[ \sum_{i=1}^{d} a_i x_i < \alpha \]

- Induced partitioning of feature space:
Sphere trees

- Questions are

\[ \sum_{i=1}^{d} (z_i - x_i)^2 < \alpha \]

- Induced partitioning of feature space:
Nonlinear generalizations

- Type of questions:
  \[ \Psi(x, a) > 0 \]
  With \( Y \) some nonlinear function
  May be different at each node
  Issue: lack of simplicity
Adding nodes: splitting
Impurity of a node

- Do all the data at a leaf belong to 1 class?
- Example:
  - At one of the present nodes, the distribution is
    - 70% Grapefruit
    - 20% Lemons
    - 10% Apples
- Introduce $P_N(\omega_i)$ at node N
- What’s a measure to define which node to split next?
Missclassification Impurity

\[ i(N) = 1 - \max_j P_N(\omega_i) \]

- Measures fraction of data that does not belong to the dominant class
- Turns out to be too greedy for multiple splits
Entropy Impurity

\[ i(N) = -\sum_j P_N(\omega_i) \log_2 P_N(\omega_i) \]

• Very popular
Gini Impurity

\[ i(N) = \frac{1}{2} \sum_j \left( 1 - P_N(\omega_i)^2 \right) \]

- Very popular as well
- No practical difference to entropy impurity
Comparison of impurity functions: Example for 2-class problem
Training Algorithms (Greedy)

• Until the tree is large enough
  – Test all nodes N:
    • Test all questions Q:
      – Calculate change in impurity
        \[
        \Delta i(N) = i(N) - P_L i(N_L(Q)) - (1 - P_L) i(N_R(Q))
        \]
      – Add node/question to achieve largest drop in impurity

\(N_L(Q)\): new left node
\(N_R(Q)\): new right node
\(P_L\): fraction of data in \(N_L(Q)\)
Training of a CART Tree:
Spam-Mail detection (From Hastie et al.)

Root node question: frequency of $ sign

Green: 10-fold cross-validation
Red: test set
Influence of Amount of Training Data: Text Classification (From Manning+Schütze)

- High variability for small amounts of training data
- Saturation for large amounts of data
Pruning a CART-tree (Text Classification)

Tune the pruning of a validation set
Example for Instability of Tree

Example data for a two class problem

<table>
<thead>
<tr>
<th>$\omega_1$ (black)</th>
<th>$\omega_2$ (red)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>$x_2$</td>
</tr>
<tr>
<td>.15</td>
<td>.83</td>
</tr>
<tr>
<td>.09</td>
<td>.55</td>
</tr>
<tr>
<td>.29</td>
<td>.35</td>
</tr>
<tr>
<td>.38</td>
<td>.70</td>
</tr>
<tr>
<td>.52</td>
<td>.48</td>
</tr>
<tr>
<td>.57</td>
<td>.73</td>
</tr>
<tr>
<td>.73</td>
<td>.75</td>
</tr>
<tr>
<td>.47</td>
<td>.06</td>
</tr>
</tbody>
</table>
Example for Instability of Tree
Example for Instability of Tree

Small changes in the data can result in large changes of the tree however, classification accuracy is usually stable.
Summary

- Decision trees are easy to understand.
- Decision trees can classify both categorical and numerical data, but the output attribute must be categorical.
- Decision tree algorithms are unstable.
- Trees created from numeric data sets can be quite complex.