# Introduction to Decision Tree Algorithm

*Wenyan Li (Emily Li)*

*Sep. 29, 2009*

## Outline

- Introduction to Classification
- Examples of Decision Tree
- Decision Tree Induction
- Advantages of Tree-based Algorithm
- Decision Tree Algorithm in STATISTICA

## Introduction to Classification

- A classification technique (or classifier) is a systematic approach to building classification models from an input data set.
- The training data consist of pairs of input objects (typically vectors), and desired outputs.
- The output of the function can be a continuous value (called regression), or can be a categorical value (called classification).

## Introduction to Classification

Training data set:

| | input objects | | | desired output |
|---|---|---|---|---|
| **No** | **Time** | **Power Output** | **Wind Speed** | **Fault 296** |
| 1 | 4/3/09 8:40 PM | -1.48 | 4.30 | 0 |
| 2 | 4/4/09 7:00 AM | -5.40 | 5.88 | 1 |
| 3 | 4/4/09 5:05 PM | -10.07 | 6.72 | 0 |
| … | … | … | … | … |

Test data set:

| No | Time | Power Output | Wind Speed | Fault 296 |
|---|---|---|---|---|
| 1 | 4/5/09 3:10 AM | -8.40 | 11.74 | ? |
| 2 | 4/5/09 1:20 PM | -19.00 | 12.70 | ? |

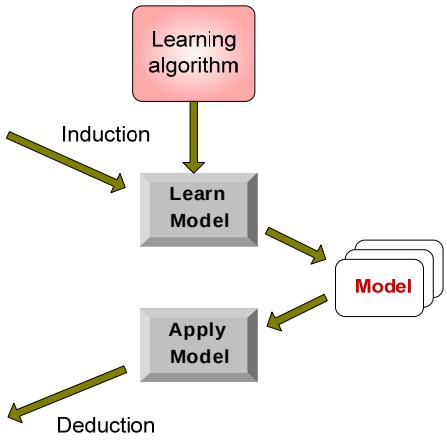
## Illustrating Classification Task

| Tid | Attrib1 | Attrib2 | Attrib3 | Class |
|---|---|---|---|---|
| 1 | Yes | Large | 125K | No |
| 2 | No | Medium | 100K | No |
| 3 | No | Small | 70K | No |
| 4 | Yes | Medium | 120K | No |
| 5 | No | Large | 95K | Yes |
| 6 | No | Medium | 60K | No |
| 7 | Yes | Large | 220K | No |
| 8 | No | Small | 85K | Yes |
| 9 | No | Medium | 75K | No |
| 10 | No | Small | 90K | Yes |

Training Set

| Tid | Attrib1 | Attrib2 | Attrib3 | Class |
|---|---|---|---|---|
| 11 | No | Small | 55K | ? |
| 12 | Yes | Medium | 80K | ? |
| 13 | Yes | Large | 110K | ? |
| 14 | No | Small | 95K | ? |
| 15 | No | Large | 67K | ? |

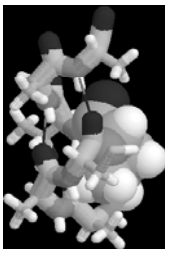
Test Set




## Examples of Classification Task

- Predicting tumor cells as benign or malignant
- Classifying credit card transactions as legitimate or fraudulent

- Classifying secondary structures of protein as alpha-helix, beta-sheet, or random coil



- Categorizing news stories as finance, weather, entertainment, sports, etc

## Classification Techniques

- Decision Tree based Methods

- Rule-based Methods
- Memory based reasoning
- Neural Networks
- Naïve Bayes and Bayesian Belief Networks
- Support Vector Machines

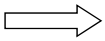
## Outline

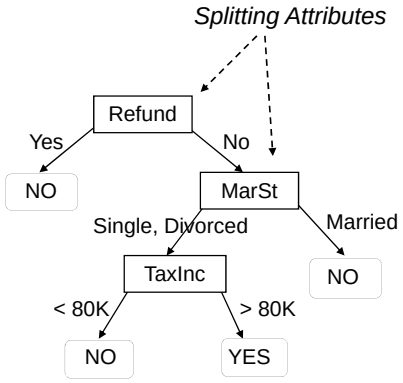
- **Introduction to Classification**
- **Examples of Decision Tree**
- **Tree Induction**
- **Advantages of Tree-based Algorithm**
- **Decision Tree Algorithm in STATISTICA**

# Example of a Decision Tree

| Tid | Refund | Marital Status | Taxable Income | Cheat |
|---|---|---|---|---|
| | | categorical | categorical | continuous | class |
| 1 | Yes | Single | 125K | No |
| 2 | No | Married | 100K | No |
| 3 | No | Single | 70K | No |
| 4 | Yes | Married | 120K | No |
| 5 | No | Divorced | 95K | Yes |
| 6 | No | Married | 60K | No |
| 7 | Yes | Divorced | 220K | No |
| 8 | No | Single | 85K | Yes |
| 9 | No | Married | 75K | No |
| 10 | No | Single | 90K | Yes |

Training Data

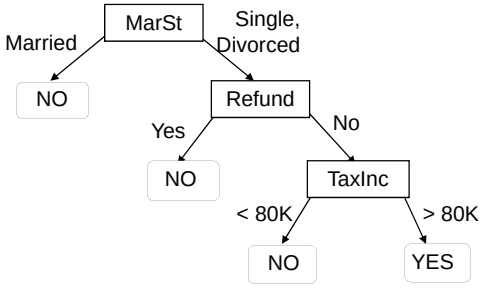





Model: Decision Tree

# Another Example of Decision Tree

| Tid | Refund | Marital Status | Taxable Income | Cheat |
|---|---|---|---|---|
| | | categorical | categorical | continuous | class |
| 1 | Yes | Single | 125K | No |
| 2 | No | Married | 100K | No |
| 3 | No | Single | 70K | No |
| 4 | Yes | Married | 120K | No |
| 5 | No | Divorced | 95K | Yes |
| 6 | No | Married | 60K | No |
| 7 | Yes | Divorced | 220K | No |
| 8 | No | Single | 85K | Yes |
| 9 | No | Married | 75K | No |
| 10 | No | Single | 90K | Yes |

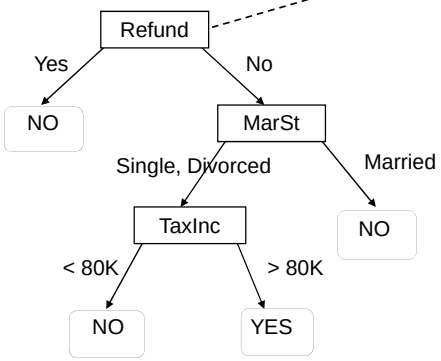



There could be more than one tree that fits the same data!

# Apply Model to Test Data

Test Data

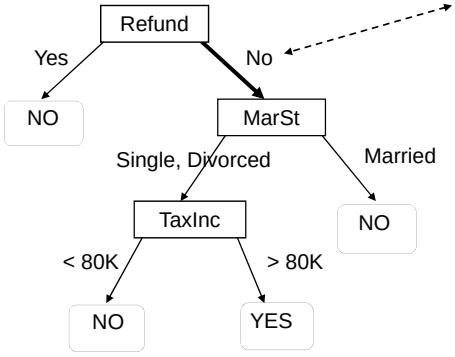
| Refund | Marital Status | Taxable Income | Cheat |
|---|---|---|---|
| No | Married | 80K | ? |




# Apply Model to Test Data

Test Data

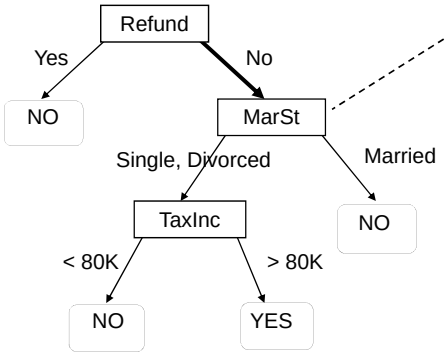
| Refund | Marital Status | Taxable Income | Cheat |
|---|---|---|---|
| No | Married | 80K | ? |



## Apply Model to Test Data

Test Data

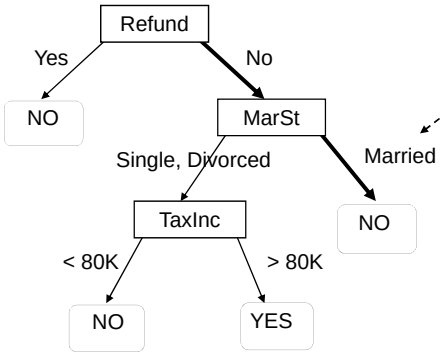
| Refund | Marital Status | Taxable Income | Cheat |
|---|---|---|---|
| No | Married | 80K | ? |




## Apply Model to Test Data

Test Data

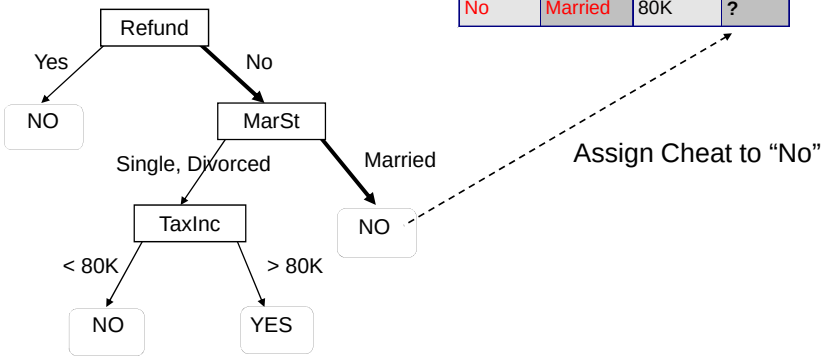
| Refund | Marital Status | Taxable Income | Cheat |
|---|---|---|---|
| No | Married | 80K | ? |



## Apply Model to Test Data

Test Data

| Refund | Marital Status | Taxable Income | Cheat |
|---|---|---|---|
| No | Married | 80K | ? |




Assign Cheat to “No”

## Outline

- Introduction to Classification
- Examples of Decision Tree
- Decision Tree Induction
- Advantages of Tree-based Algorithm
- Decision Tree Algorithm in STATISTICA

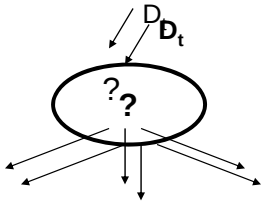
## Decision Tree Induction

- Many Algorithms:
  - Hunt's Algorithm (one of the earliest)
  - CART
  - ID3, C4.5
  - SLIQ,SPRINT

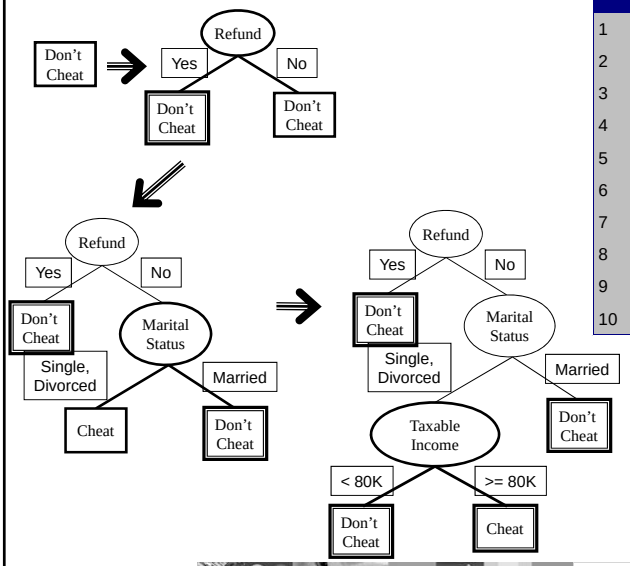
## General Structure of Hunt's Algorithm

- Let $D_t$ be the set of training records that reach a node t
- General Procedure:
  - If $D_t$ contains records that belong the same class $y_t$, then t is a leaf node labeled as $y_t$
  - If $D_t$ is an empty set, then t is a leaf node labeled by the default class, $y_d$
  - If $D_t$ contains records that belong to more than one class, use an attribute test to split the data into smaller subsets. Recursively apply the procedure to each subset.

| *Tid* | Refund | Marital Status | Taxable Income | Cheat |
|---|---|---|---|---|
| 1 | Yes | Single | 125K | No |
| 2 | No | Married | 100K | No |
| 3 | No | Single | 70K | No |
| 4 | Yes | Married | 120K | No |
| 5 | No | Divorced | 95K | Yes |
| 6 | No | Married | 60K | No |
| 7 | Yes | Divorced | 220K | No |
| 8 | No | Single | 85K | Yes |
| 9 | No | Married | 75K | No |
| 10 | No | Single | 90K | Yes |



## Hunt's Algorithm




| Tid | Refund | Marital Status | Taxable Income | Cheat |
|---|---|---|---|---|
| 1 | Yes | Single | 125K | No |
| 2 | No | Married | 100K | No |
| 3 | No | Single | 70K | No |
| 4 | Yes | Married | 120K | No |
| 5 | No | Divorced | 95K | Yes |
| 6 | No | Married | 60K | No |
| 7 | Yes | Divorced | 220K | No |
| 8 | No | Single | 85K | Yes |
| 9 | No | Married | 75K | No |
| 10 | No | Single | 90K | Yes |

## Tree Induction

- Greedy strategy.
  - Split the records based on an attribute test that optimizes certain criterion.
- Issues
  - Determine how to split the records
    - How to specify the attribute test condition?
    - How to determine the best split?
  - Determine when to stop splitting

## Tree Induction

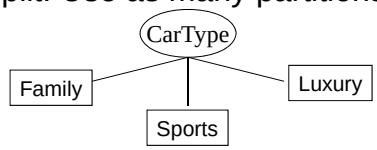
- Greedy strategy.
  - Split the records based on an attribute test that optimizes certain criterion.
- Issues
  - Determine how to split the records
    - How to specify the attribute test condition?
    - How to determine the best split?
  - Determine when to stop splitting

## How to Specify Test Condition?

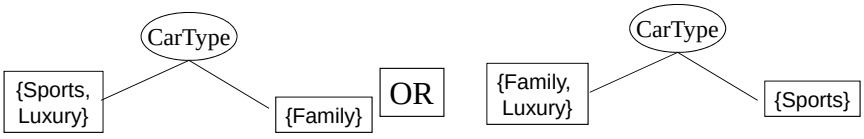
- Depends on attribute types
  - Nominal
  - Ordinal
  - Continuous
- Depends on number of ways to split
  - 2-way split
  - Multi-way split

## Splitting Based on Nominal Attributes

- Multi-way split: Use as many partitions as distinct values.

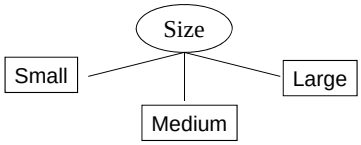



- Binary split: Divides values into two subsets.
  Need to find optimal partitioning.

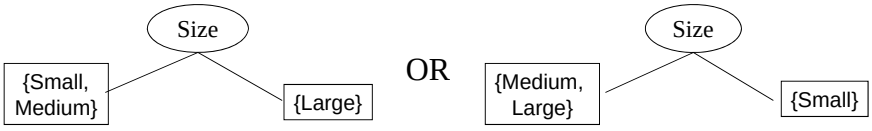



## Splitting Based on Ordinal Attributes

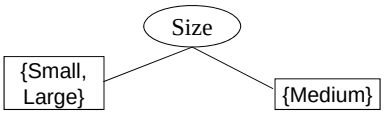
- Multi-way split: Use as many partitions as distinct values.




- Binary split: Divides values into two subsets.
  Need to find optimal partitioning.



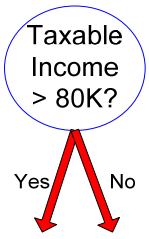

- What about this split?



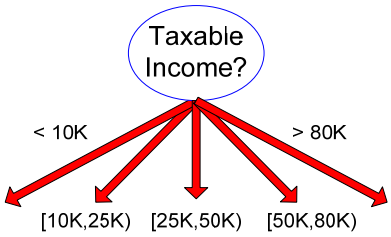
## Splitting Based on Continuous Attributes

- Different ways of handling:
  - Discretization to form an ordinal categorical attribute
    - Static – discretize once at the beginning
    - Dynamic – ranges can be found by equal interval bucketing, equal frequency bucketing (percentiles), or clustering.
  - Binary Decision: (A < v) or (A ≥ v)
    - consider all possible splits and finds the best cut
    - can be more compute intensive

## Splitting Based on Continuous Attributes




(i) Binary split



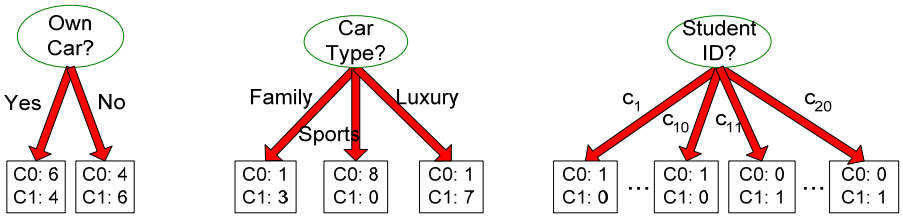

(ii) Multi-way split

## Tree Induction

- Greedy strategy.
  - Split the records based on an attribute test that optimizes certain criterion.
- Issues
  - Determine how to split the records
    - How to specify the attribute test condition?
    - How to determine the best split?
  - Determine when to stop splitting

## How to determine the Best Split

Before Splitting: 10 records of class 0,
10 records of class 1




Which test condition is the best?

## How to determine the Best Split

- Greedy approach:
  - Nodes with homogeneous class distribution are preferred
- Need a measure of node impurity:

| C0: 5<br>C1: 5 | C0: 9<br>C1: 1 |
|---|---|
| Non-homogeneous,<br>High degree of impurity | Homogeneous,<br>Low degree of impurity |

## Measures of Node Impurity

- Gini Index
- Entropy
- Misclassification error

## Splitting Criteria based on INFO

- Entropy at a given node t:

$$Entropy(t) = -\sum_j p(j \mid t) \log p(j \mid t)$$

(NOTE: $p(j \mid t)$ is the relative frequency of class j at node t).

- Measures homogeneity of a node.
  - Maximum ($\log n_c$) when records are equally distributed among all classes implying least information
  - Minimum (0.0) when all records belong to one class, implying most information
- Entropy based computations are similar to the GINI index computations

## Examples for Computing Entropy

$$Entropy(t) = -\sum_j p(j \mid t) \log_2 p(j \mid t)$$

| | |
|---|---|
| C1 | **0** |
| C2 | **6** |

| | |
|---|---|
| C1 | **1** |
| C2 | **5** |

| | |
|---|---|
| C1 | **2** |
| C2 | **4** |

## Examples for Computing Entropy

$$Entropy(t) = -\sum_j p(j \mid t) \log_2 p(j \mid t)$$

| C1 | **0** |
|---|---|
| C2 | **6** |

P(C1) = 0/6 = 0  P(C2) = 6/6 = 1

Entropy = – 0 log 0 – 1 log 1 = – 0 – 0 = 0

| C1 | **1** |
|---|---|
| C2 | **5** |

P(C1) = 1/6  P(C2) = 5/6

Entropy = – (1/6) $\log_2$ (1/6) – (5/6) $\log_2$ (1/6) = 0.65

| C1 | **2** |
|---|---|
| C2 | **4** |

P(C1) = 2/6  P(C2) = 4/6

Entropy = – (2/6) $\log_2$ (2/6) – (4/6) $\log_2$ (4/6) = 0.92

## Splitting Based on INFO...

- Information Gain:

$$GAIN_{split} = Entropy(p) - \left( \sum_{i=1}^{k} \frac{n_i}{n} Entropy(i) \right)$$

Parent Node, p is split into k partitions;

$n_i$ is number of records in partition i

  - Measures Reduction in Entropy achieved because of the split. Choose the split that achieves most reduction (maximizes GAIN)
  - Disadvantage: Tends to prefer splits that result in large number of partitions, each being small but pure.

## Splitting Based on INFO...

- Gain Ratio:

$$GainRATIO_{split} = \frac{GAIN_{Split}}{SplitINFO} \qquad SplitINFO = -\sum_{i=1}^{k} \frac{n_i}{n} \log \frac{n_i}{n}$$

Parent Node, p is split into k partitions
$n_i$ is the number of records in partition i

- Adjusts Information Gain by the entropy of the partitioning (SplitINFO). Higher entropy partitioning (large number of small partitions) is penalized!
- Designed to overcome the disadvantage of Information Gain

## Splitting Criteria based on Classification Error

- Classification error at a node t :

$$Error(t) = 1 - \max_i P(i \mid t)$$

- Measures misclassification error made by a node.
  - Maximum (1 - $1/n_c$) when records are equally distributed among all classes, implying least interesting information
  - Minimum (0.0) when all records belong to one class, implying most interesting information

## Examples for Computing Error

$$Error(t) = 1 - \max_i P(i \mid t)$$

| C1 | **0** |
|---|---|
| C2 | **6** |

| C1 | **1** |
|---|---|
| C2 | **5** |

| C1 | **2** |
|---|---|
| C2 | **4** |

## Examples for Computing Error

$$Error(t) = 1 - \max_i P(i \mid t)$$

| C1 | **0** |
|---|---|
| C2 | **6** |

P(C1) = 0/6 = 0  P(C2) = 6/6 = 1

Error = 1 – max (0, 1) = 1 – 1 = 0

| C1 | **1** |
|---|---|
| C2 | **5** |

P(C1) = 1/6  P(C2) = 5/6

Error = 1 – max (1/6, 5/6) = 1 – 5/6 = 1/6

| C1 | **2** |
|---|---|
| C2 | **4** |

P(C1) = 2/6  P(C2) = 4/6

Error = 1 – max (2/6, 4/6) = 1 – 4/6 = 1/3

## Tree Induction

- Greedy strategy.
  - Split the records based on an attribute test that optimizes certain criterion.
- Issues
  - Determine how to split the records
    - How to specify the attribute test condition?
    - How to determine the best split?
  - Determine when to stop splitting

## Stopping Criteria for Tree Induction

- Stop expanding a node when all the records belong to the same class
- Stop expanding a node when all the records have similar attribute values
- Early termination (to be discussed later)

## Outline

- Introduction to Classification
- Examples of Decision Tree
- Decision Tree Induction
- Advantages of Tree-based Algorithm
- Decision Tree Algorithm in STATISTICA

## Decision Tree Based Classification

- Advantages:
  - Inexpensive to construct
  - Extremely fast at classifying unknown records
  - Easy to interpret for small-sized trees
  - Accuracy is comparable to other classification techniques for many simple data sets

## Outline

- Introduction to Classification
- Examples of Decision Tree
- Decision Tree Induction
- Advantages of Tree-based Algorithm
- Decision Tree Algorithm in STATISTICA

Thank You