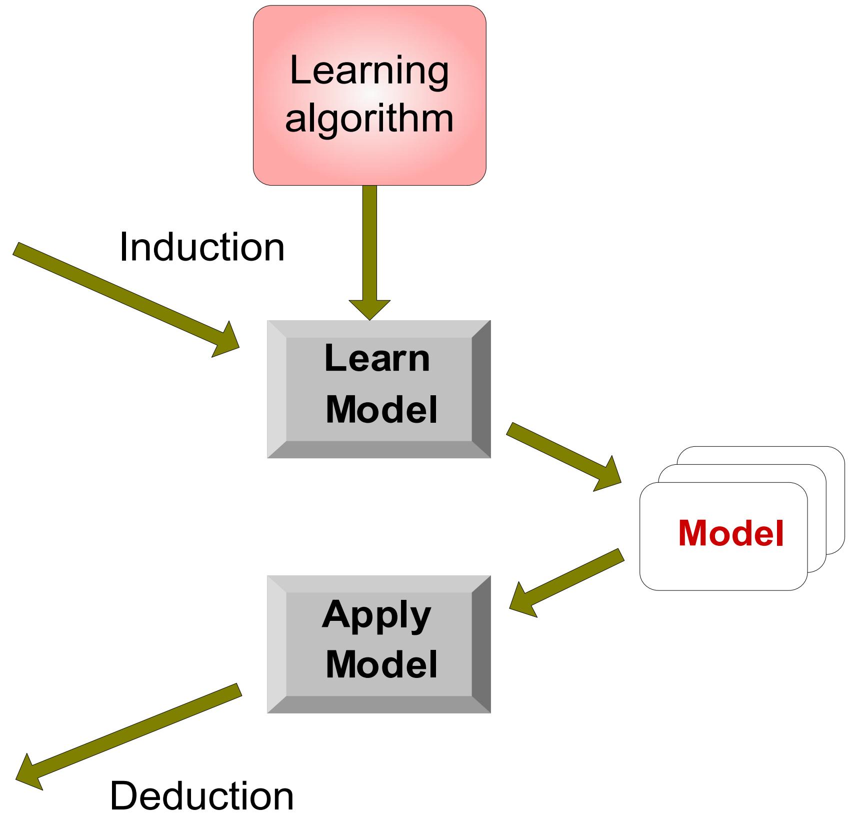
WHAT IS A DECISION TREE MODEL

SYRACUSE UNIVERSITY
School of Information Studies

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

Two steps

CLASSIFICATION TECHNIQUES

Many classification algorithms have been developed to date.

This class will introduce the details of several of the most popular algorithms:

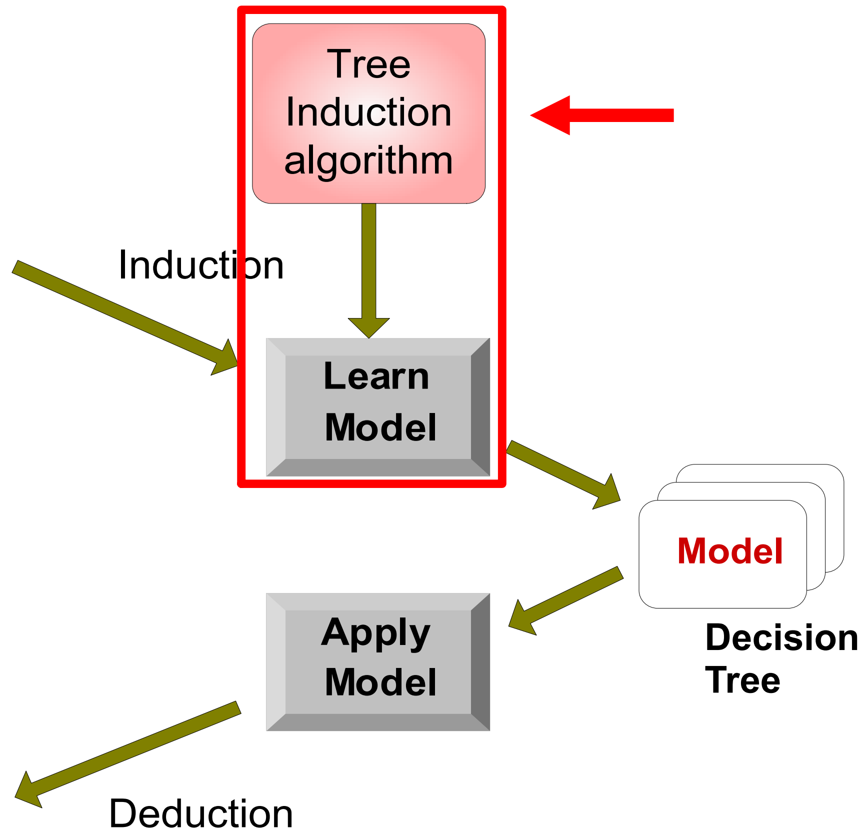
- Decision tree
- Bayesian method (naïve Bayes)
- Instance-based learning (k-nearest neighbors)
- Support vector machines (SVMs)

This week, we illustrate classification tasks using decision tree methods.

DECISION TREE 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

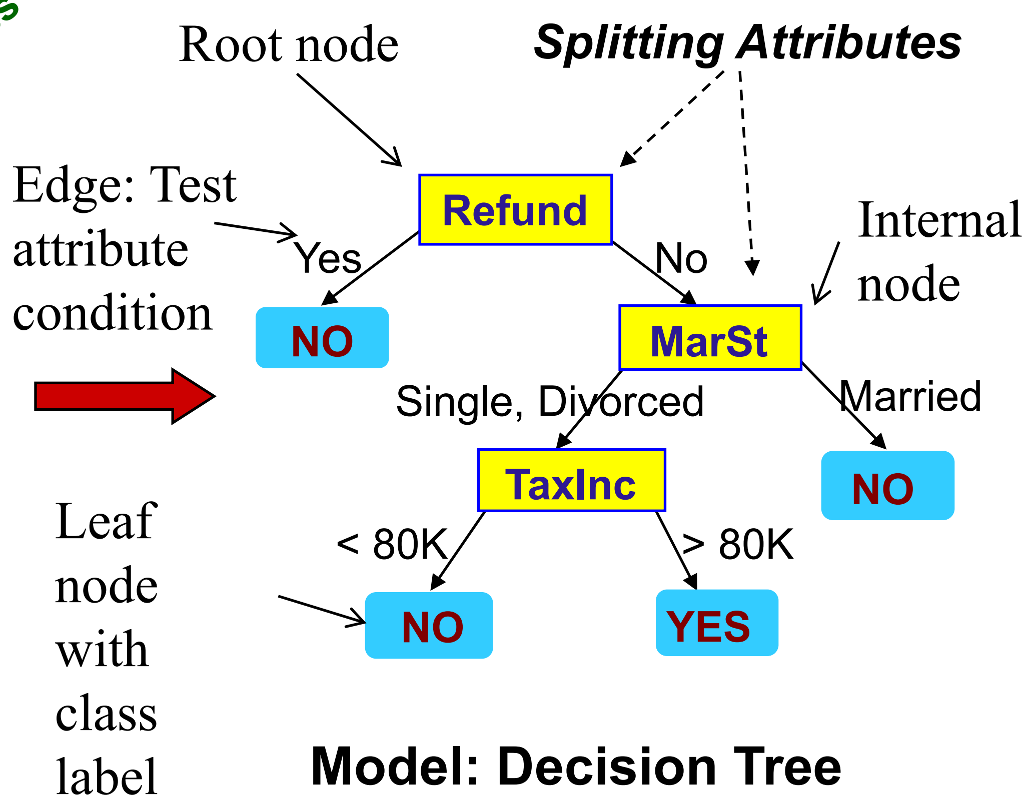
AN EXAMPLE OF DECISION TREE

Problem: To label each person as to whether they will cheat IRS

categorical categorical continuous class

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**

Training Data

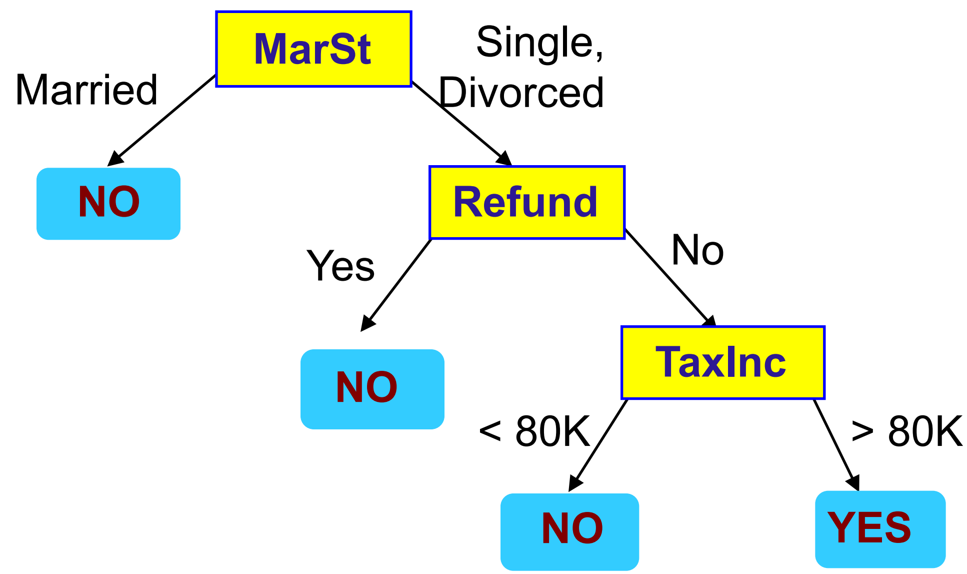


Model: Decision Tree

ANOTHER EXAMPLE OF DECISION TREE

categorical categorical continuous class

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**



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

C4.5 ALGORITHM (1) HOW TO SPLIT DATA AT A NODE

SYRACUSE UNIVERSITY
School of Information Studies

HOW TO FIND THE BEST DECISION TREE

Too many candidate trees

Manual construction takes too long

Need some machine intelligence to help

DECISION TREE INDUCTION

Many algorithms:

- Hunt's algorithm (one of the earliest)
- CART
- ID3, C4.5
- SLIQ, SPRINT

C4.5 is introduced in this class.

TREE INDUCTION

Key questions to build a decision tree model:

Which attribute to pick as internal node?

How to split the data set at a node?

HOW TO SPLIT DATA AT A NODE

How many branches?

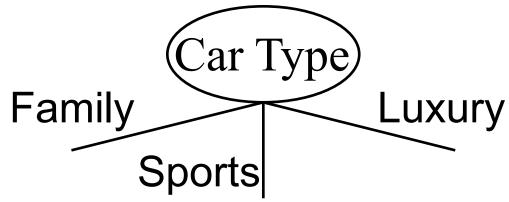
- Splitting can be:
  - Two-way split
  - Multiway split

What are the splitting values?

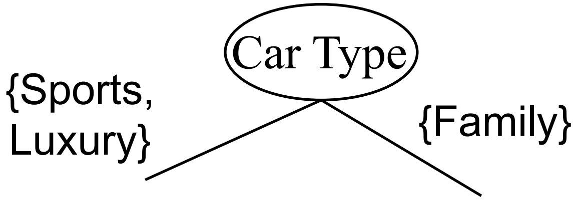
- Splitting conditions depend on attribute type:
  - Nominal or categorical
  - Ordinal
  - Continuous

SPLITTING BASED ON CATEGORICAL ATTRIBUTES

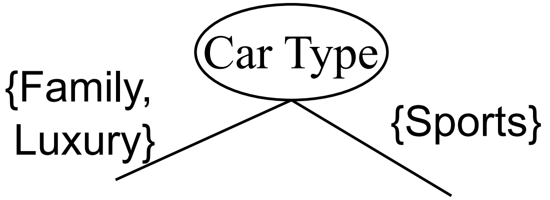
Multiway split: Use as many partitions as distinct values.



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



OR



SPLITTING BASED ON CONTINUOUS ATTRIBUTES

Different ways of handling

- Discretization to form an ordinal categorical attribute
  - E.g., age: 1 1 6 7 8 9 9 9 10 10 11 11 12 13 14 15 17 18
- Equal interval: One bin for every six years [0-6][7-12][13-18]
  - 1 1 6 • 7 8 9 9 9 10 10 11 11 12 • 13 14 15 17 18
- Equal frequency: One bin for every six numbers (could have ties)
  - 1 1 6 7 8 • 9 9 9 10 10 11 11 • 12 13 14 15 17 18
- Customized discretization

SPLITTING BASED ON CONTINUOUS ATTRIBUTES

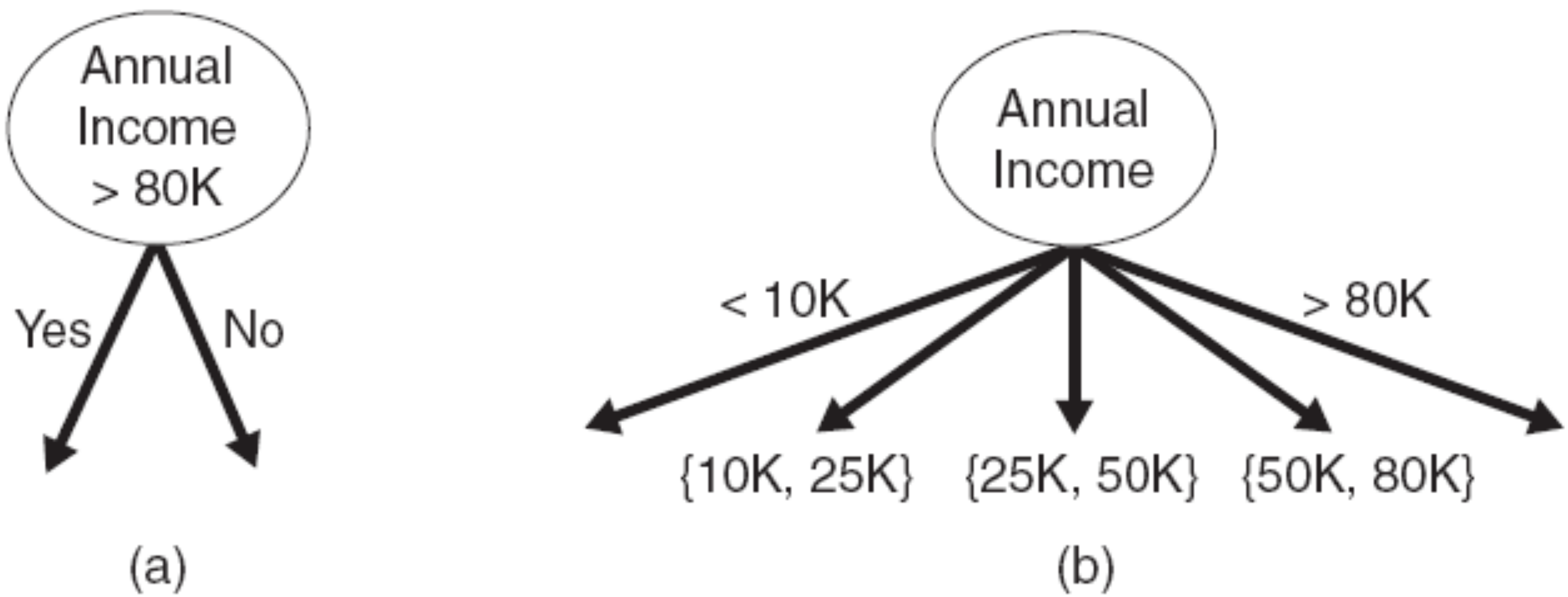


Figure 4.11. Test condition for continuous attributes.

C4.5 ALGORITHM (2) WHICH ATTRIBUTE TO CHOOSE AS A NODE

SYRACUSE UNIVERSITY
School of Information Studies

DETERMINE THE BEST ATTRIBUTE FOR SPLITTING

Information gain (IG):

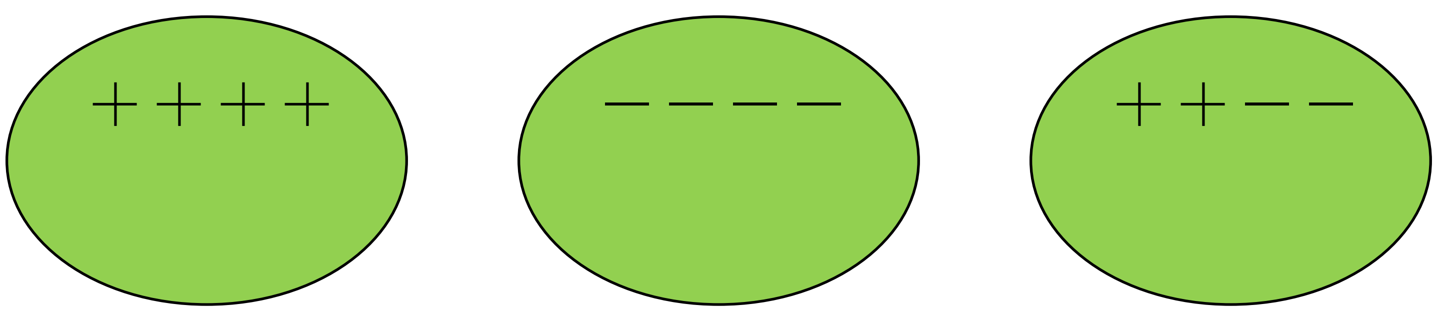
A statistical measure that measures how well a given attribute separates the training examples according to their target classification (Mitchell, 1990)

DETERMINE THE BEST ATTRIBUTE FOR SPLITTING

Entropy

- To measure the impurity of a data set
- Given a collection S, which contains positive (+) and negative (-) examples, p_i is the probability that an example belongs to Class i
- Entropy(S) = $- p_+ \log_2 p_+ - p_- \log_2 p_-$

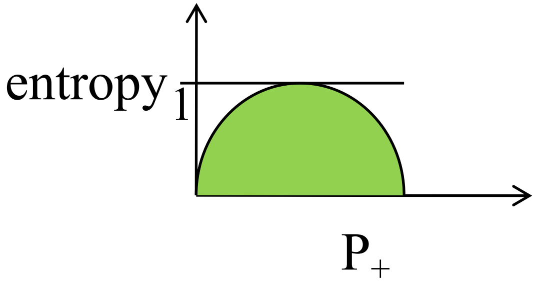
What is the entropy for each of the following collections?



DETERMINE THE BEST ATTRIBUTE FOR SPLITTING

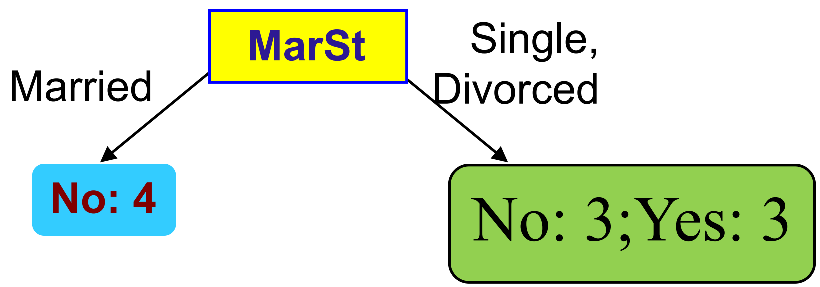
Entropy

- A measure that characterizes the impurity of a collection of examples
- Given a collection S, which contains positive (+) and negative (-) examples, p_i is the probability that an example belongs to Class i
- Entropy(S) = $- p_+ \log_2 p_+ - p_- \log_2 p_-$
- A collection of half-positive examples and half-negative examples
  - Entropy(S) = 1
- A collection of all positive examples or all negative examples
  - Entropy(S) = 0



INFORMATION GAIN: HOW MUCH IMPROVEMENT TOWARD PURITY?

Tid	Refund (categorical)	Marital Status (categorical)	Taxable Income (continuous)	Cheat (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**



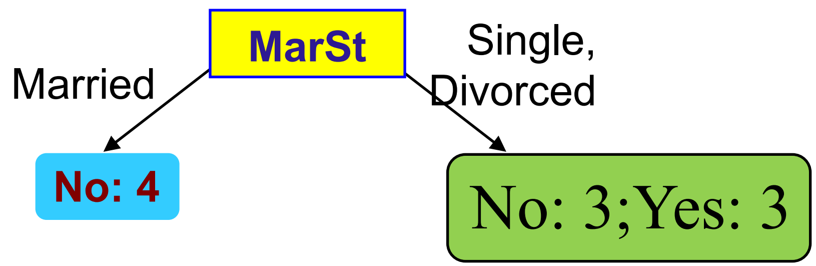
$$Gain(S,A) = Entropy(S) \qquad \underset{v \quad Values(A)}{} \frac{|S_v|}{|S|} \cdot Entropy(S_v)$$

The expected reduction in entropy caused by knowing the value of attribute A

INFORMATION GAIN: HOW MUCH IMPROVEMENT TOWARD PURITY?

categorical categorical continuous class

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**



Entropy(S) = -0.7*log2(0.7)-0.3*log2(0.3) = 0.88

Entropy(S1) = 0
Entropy(S2) = 1

IG = 0.88-(0.4*0+0.6*1) = 0.28

Repeat this calculation to find the attribute that provides the highest IG.

SYRACUSE UNIVERSITY
School of Information Studies

WHICH ATTRIBUTE SHOULD BE THE FIRST NODE?

Calculate the information gain (IG) for each attribute; choose the one with the highest IG.

WHAT'S THE NEXT STEP?

Repeat the IG calculation for every subset generated from the last step …

… until all nodes are "pure" with all positive examples or all negative examples; these are all leaf nodes

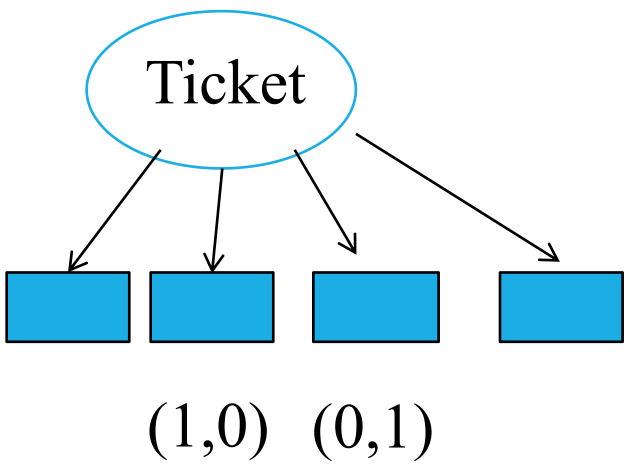
GAIN RATIO

SYRACUSE UNIVERSITY
School of Information Studies

GAIN RATIO

Impurity measures tend to favor attributes that have a large number of distinct values (textbook, p. 163).

- E.g., the "ticket" attribute in the Titanic data set means the ticket number. Assuming every passenger has a unique ticket number, the ticket attribute has many distinct values, and impurity measures such as IG favor such attributes.



GAIN RATIO

What to do?

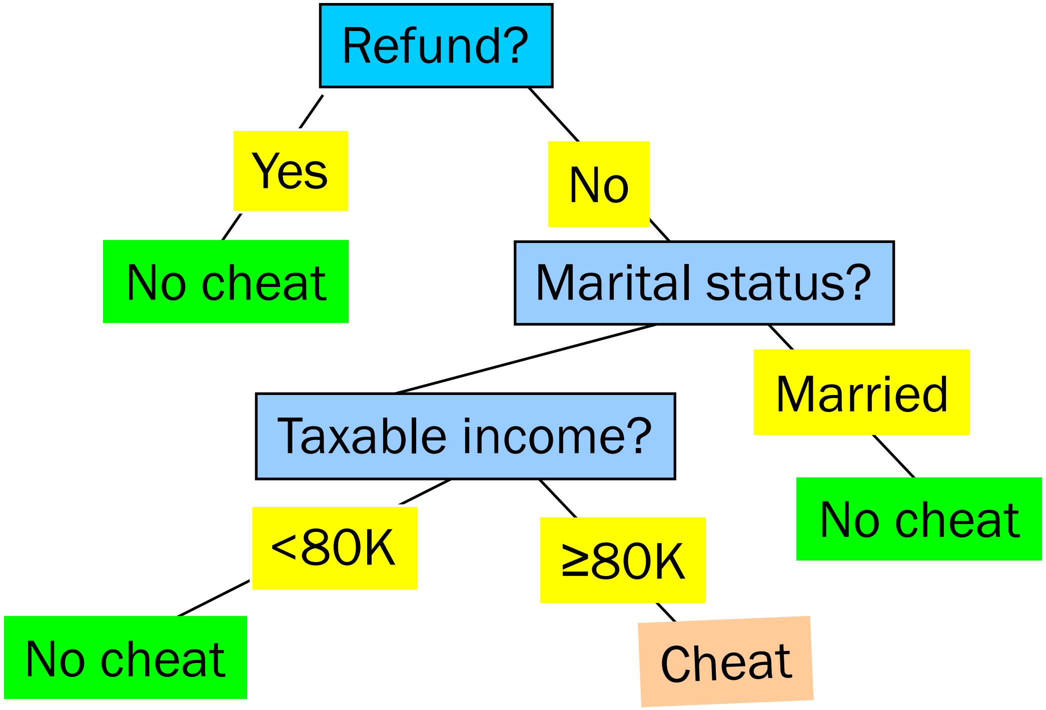
- Use domain knowledge: Does ticket number have anything to do with survival chance?
- Use gain ratio, which is IG divided by “split info.”
  - “Split info” is a penalty to a large number of splits.

In J48, the information gain measure has taken steps to avoid choosing the “ticket” types of attributes.

APPLY MODEL

SYRACUSE UNIVERSITY
School of Information Studies

CONVERTING DECISION TREE TO DECISION RULES



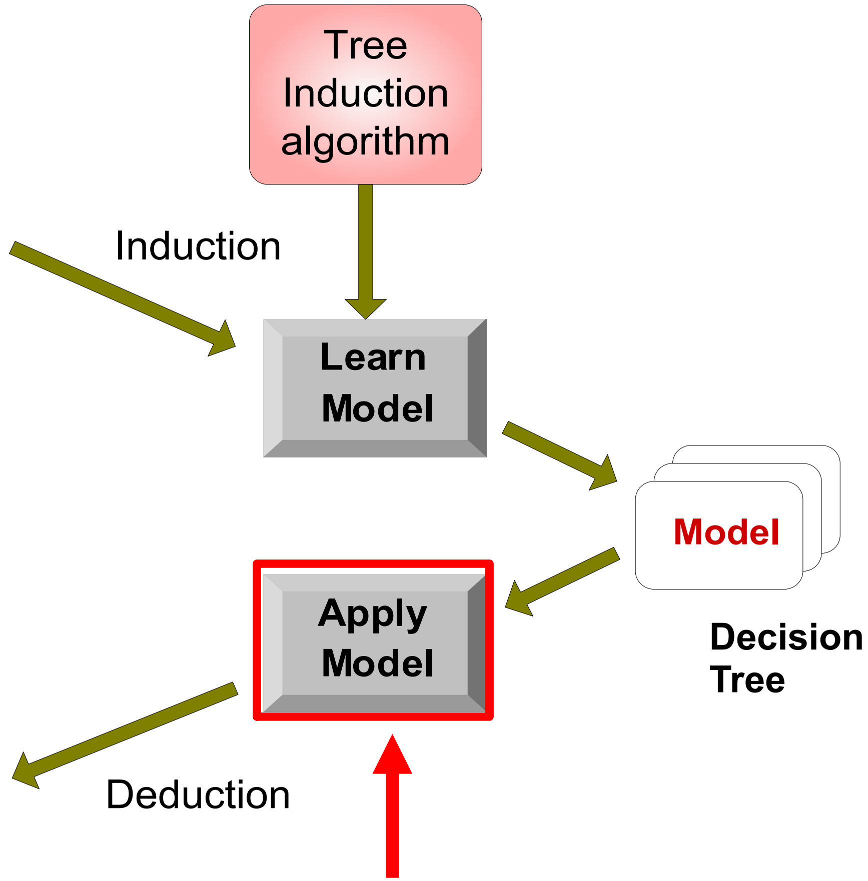
Tree can be displayed as a set of rules:

if Refund = “Yes,” then “No cheat”
else if Marital_status = “Married,” then “No cheat”
    else if Taxable_income < 80K, then “No cheat”
    else “Cheat”

DECISION TREE 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

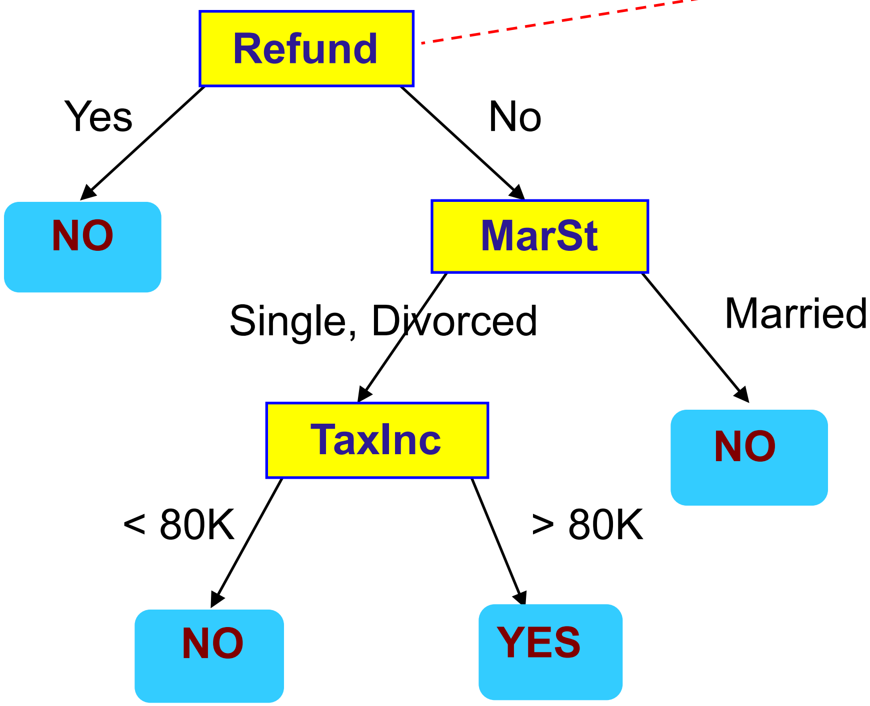
Training Set



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

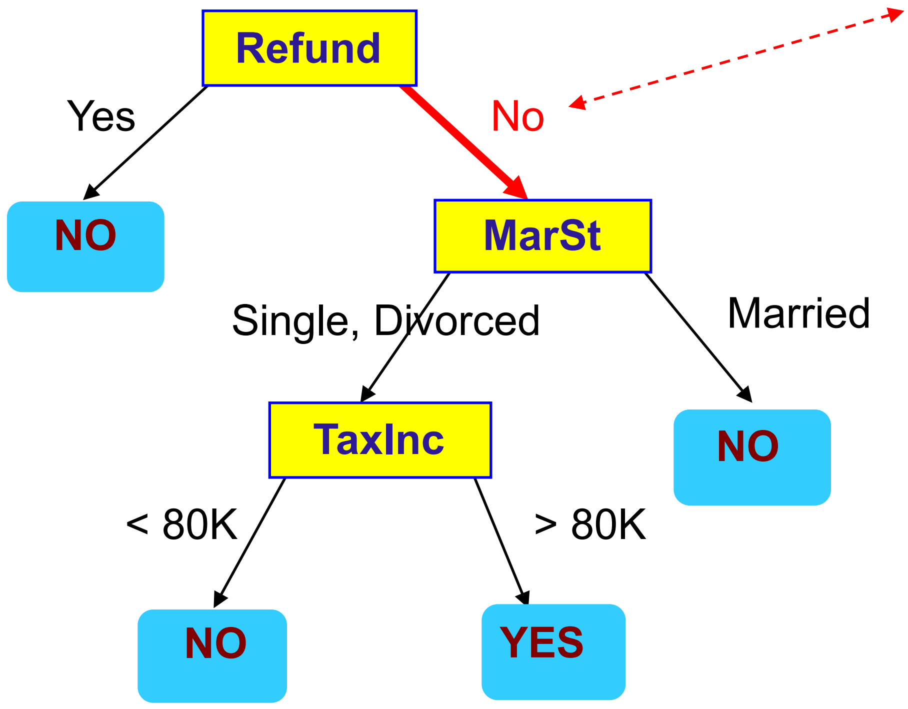
APPLY MODEL TO TEST DATA



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

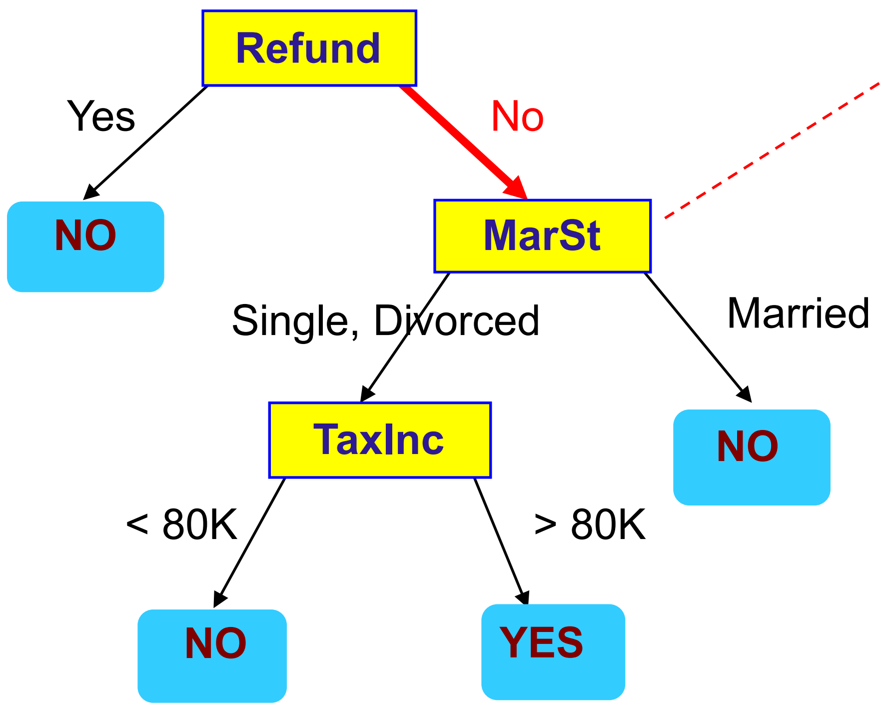
APPLY MODEL TO TEST DATA



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

APPLY MODEL TO TEST DATA



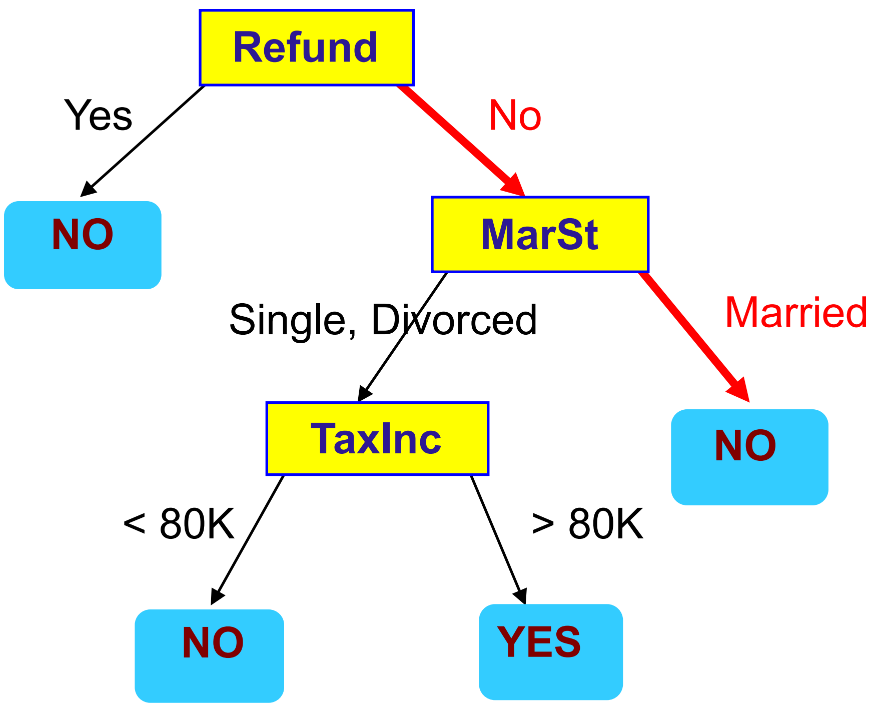
Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

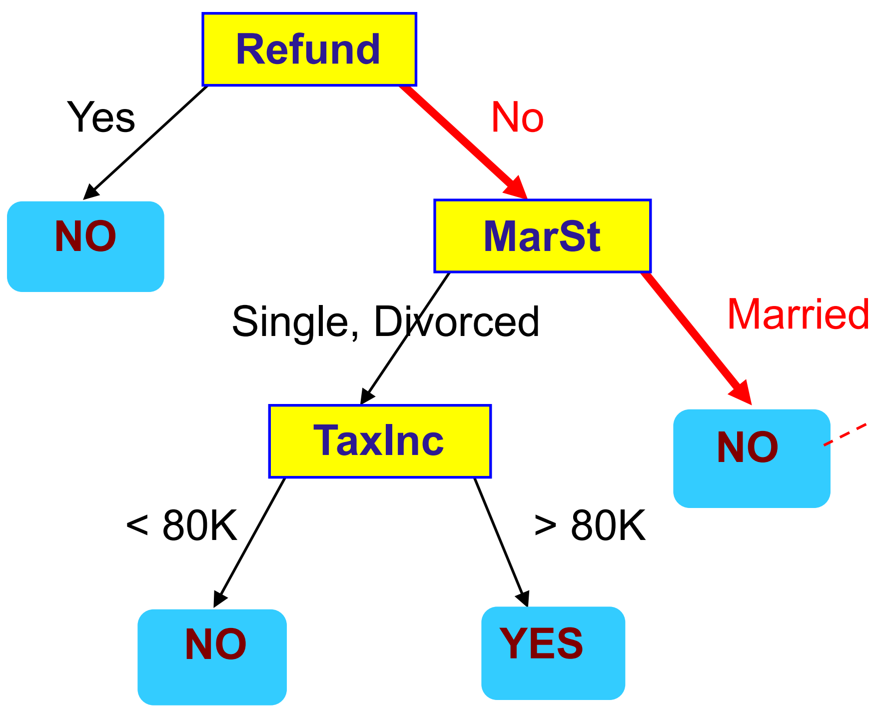
APPLY MODEL TO TEST DATA

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."

OVERFITTING AND PRUNING

SYRACUSE UNIVERSITY
School of Information Studies

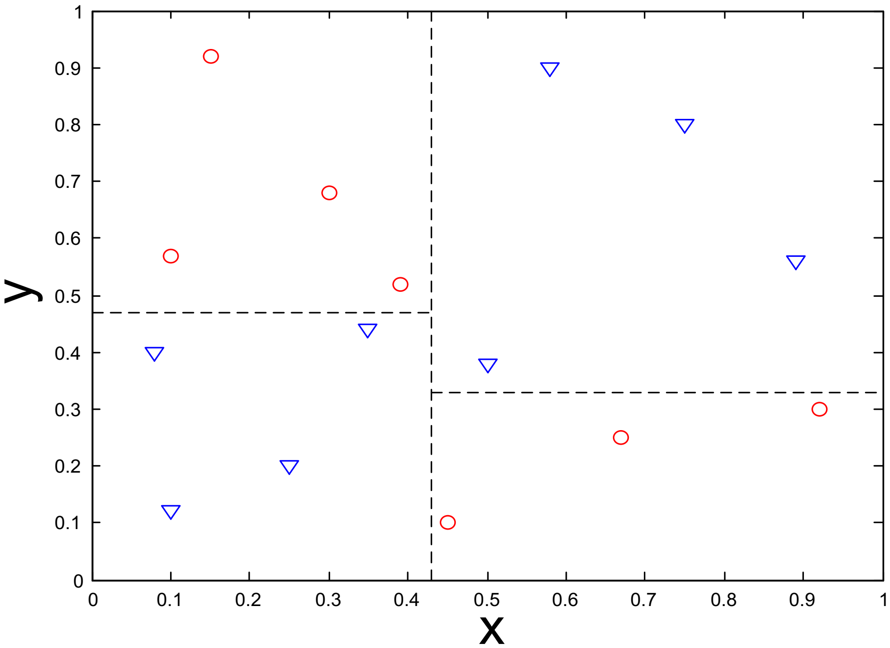
CHARACTERISTICS OF DECISION TREE INDUCTION

Decision tree (DT) is a nonparametric algorithm, meaning, it does not require any prior assumptions regarding the type of probability distributions satisfied by the class and other attributes.

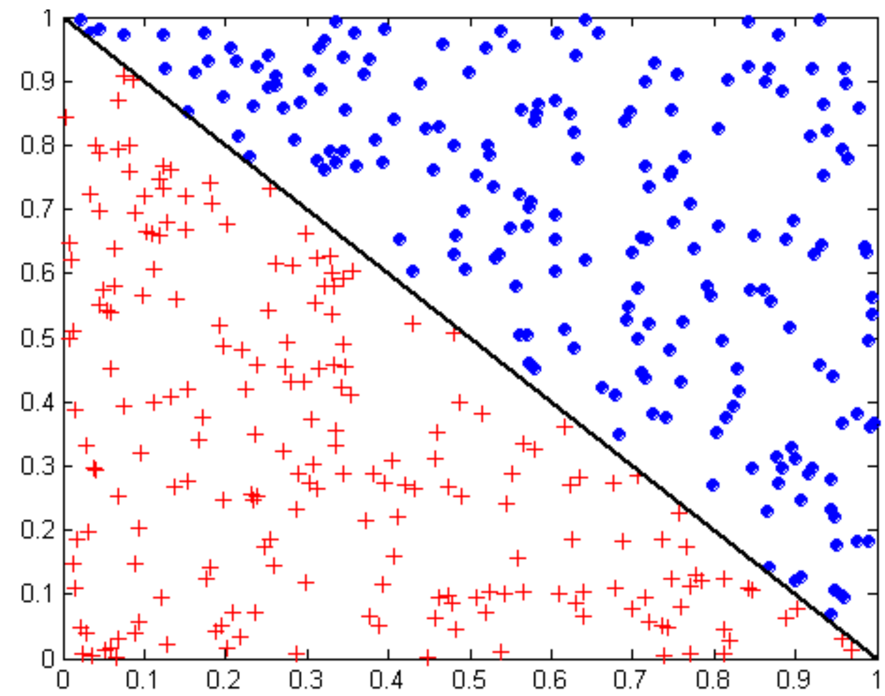
Linear classification algorithms are parametric algorithms because they assume the decision boundary is linear, such as a line in two-dimensional space.

“Decision boundary” means the border between two neighboring regions of different classes.

THERE IS NO SILVER BULLET



Nonlinear



Linear

MODEL OVERFITTING

Decision trees have the particular problem of overfitting.

There may not be enough examples to fully represent all possible cases that may arise in the future.

If decision tree is fully developed, it may be too detailed a fit to the training data and lead to more errors on the test data.

E.g., assume we are looking for patterns of buyers for a certain product. In the training data set, no women purchased a product; the DT algorithm may learn a pattern that “if women, no purchase.” But this training data set included very few women, and actually, there were women who bought this product. In such cases, the DT model overfit the training data and lost precision in future prediction.

Occam’s razor (preference of small trees)

MODEL OVERFITTING

Generally speaking, complex models are more likely to overfit than simple models.

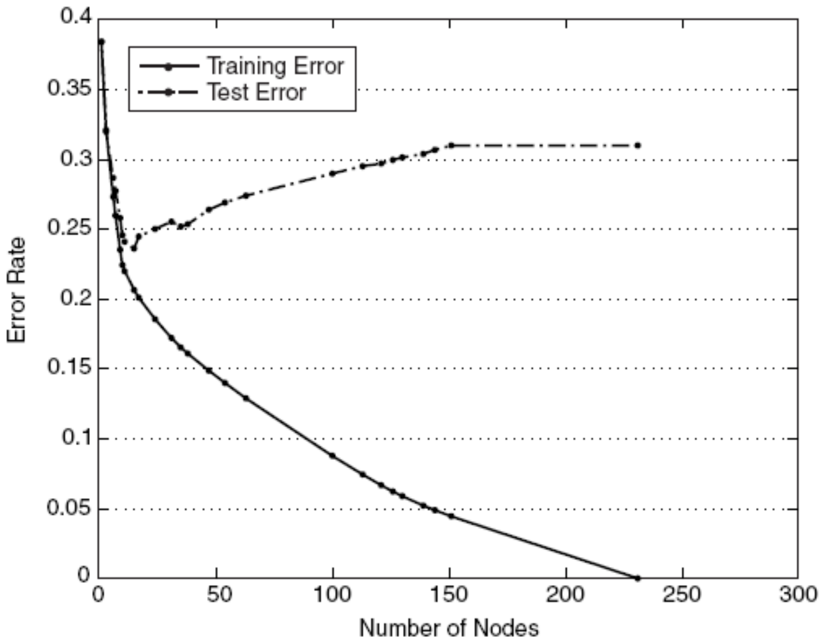


Figure 4.23. Training and test error rates.

For decision tree, number of nodes indicates model complexity.

In this figure, the higher the number of nodes, the lower the training error and the higher the test error, meaning, increasingly complex models are increasingly overfitting.

OVERFITTING AND TREE PRUNING

Two approaches to avoid overfitting:

Prepruning: Halt tree construction early—do not split a node if information gain falls below a threshold.

- Difficult to choose an appropriate threshold

Postpruning: Remove branches from a "fully grown" tree—get a sequence of progressively pruned trees.

- Use a set of data different from the training data to decide which is the "best pruned tree"

SUMMARY OF DECISION TREES

Strengths of decision trees are that they are:

- Fast in prediction
- Interpretable patterns
- Robust to noise

Weaknesses of decision trees are that they:

- Tend to overfit (pruning helps)
- Are error prone with too many classes
- Are computationally expensive in training (compared to the low cost in prediction)

WEKA J48 TUTORIAL

SYRACUSE UNIVERSITY
School of Information Studies

J48 ALGORITHM

J48 is an implementation of the famous C4.5 algorithm to construct decision tree.

J48 provides many parameters to tune.

Every time you tune a parameter, a new decision model will be created.

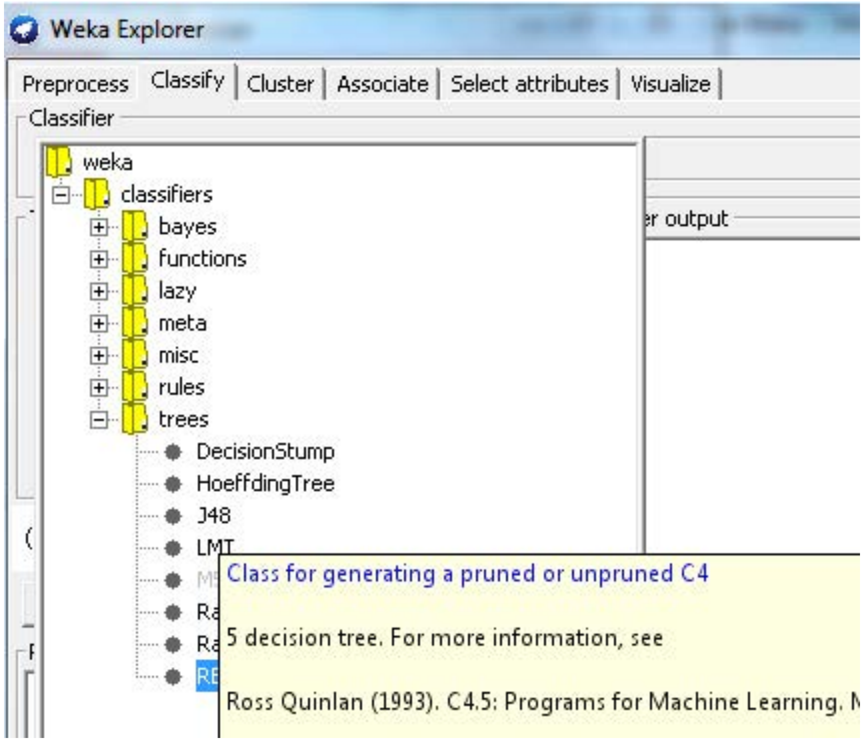
USE J48 IN WEKA

Two key questions:

How to tune parameters to get better models?

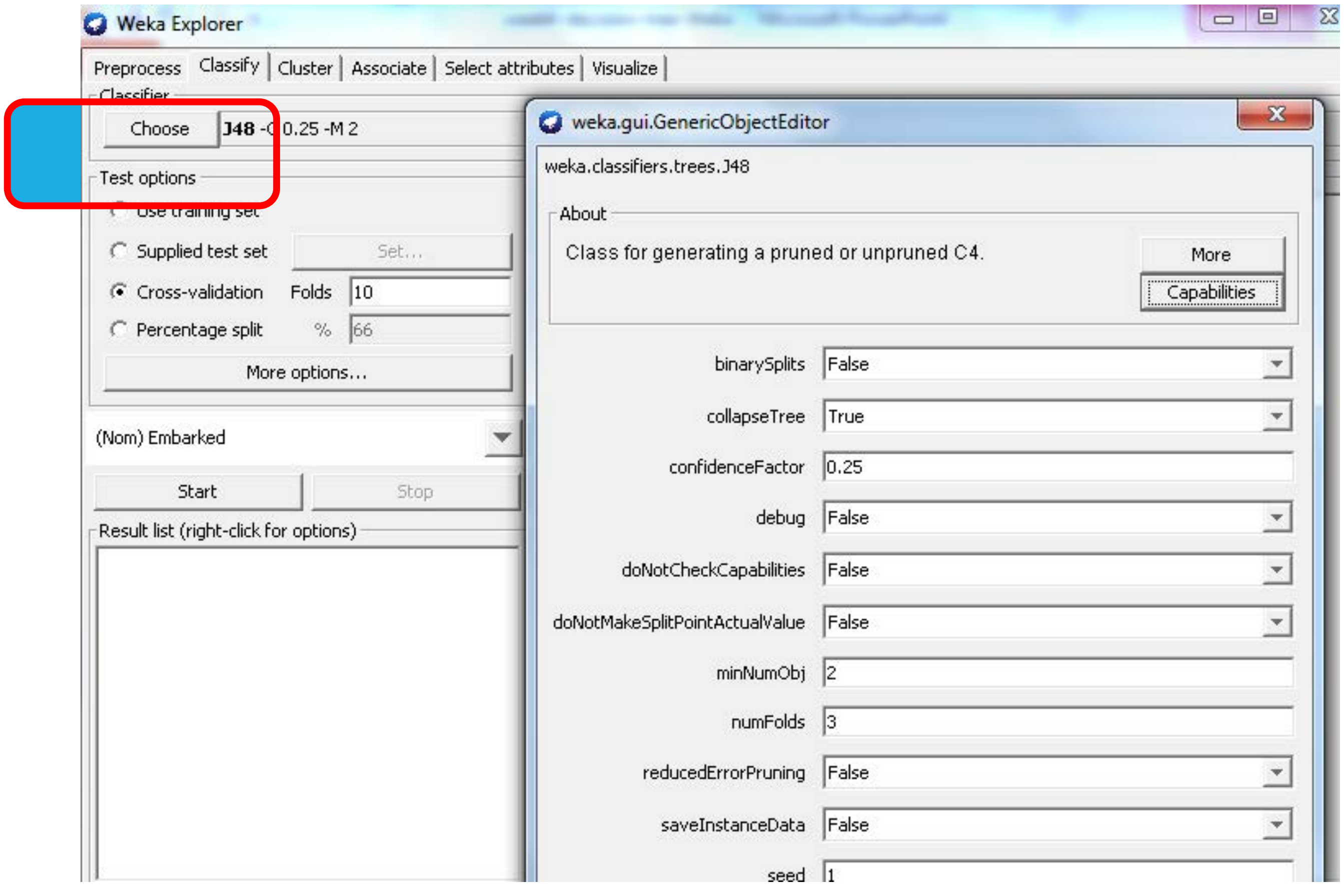
How to evaluate which model is the best?

HOW TO TUNE PARAMETERS TO GET BETTER DECISION TREE MODELS



Step 1: Find J48 algorithm, located in the “Classify” tab, under “Trees.”

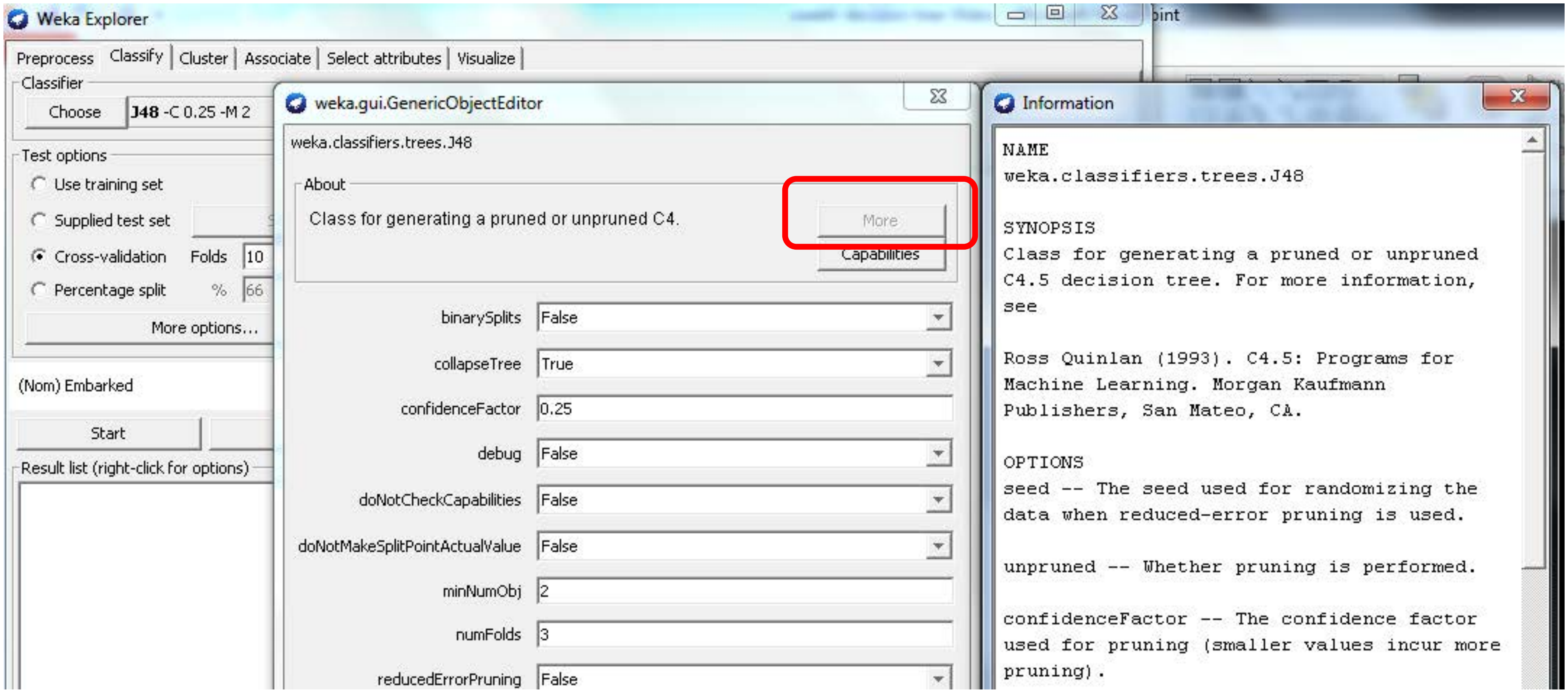
J48 PARAMETER PANEL



Click the algorithm (in red box) to pop up the parameter panel.

SYRACUSE UNIVERSITY
School of Information Studies

J48 PARAMETER MEANING



Click the “More” button (in red box) to pop up the explanation for the parameters.

J48 PARAMETERS

Tuning these parameters requires the theoretical knowledge of their purpose and empirical knowledge of their performance on different kinds of data.

Several important parameters to tune

- “BinarySplit”: True or False
  - True: A deep tree with two branches at each level
  - False (default): A wide tree with many branches at each level
  - Which one works better? Depends on data

J48 PARAMETERS

"**unpruned**": True or False

- True: Grow a tree completely without pruning.
- False (default): Prune the tree.

"**ConfidenceFactor**": numeric (0 to 1)

- Decide how aggressively to prune the tree.
- Smaller values incur more pruning.
- Too aggressive pruning results in a too small tree that does not capture all patterns.
- Too conservative pruning results in a large tree that overfits the training data.
- How to find the balance point?
  - Use a good evaluation method, such as cross-validation (covered in later slides).

J48 PARAMETERS

“**minNumObj**”: Integer (1 to infinity)

The minimum number of examples in the leaf node

Default value: 2

It means all leaves with only one data example are pruned.

Increasing this value would result in more aggressive pruning.

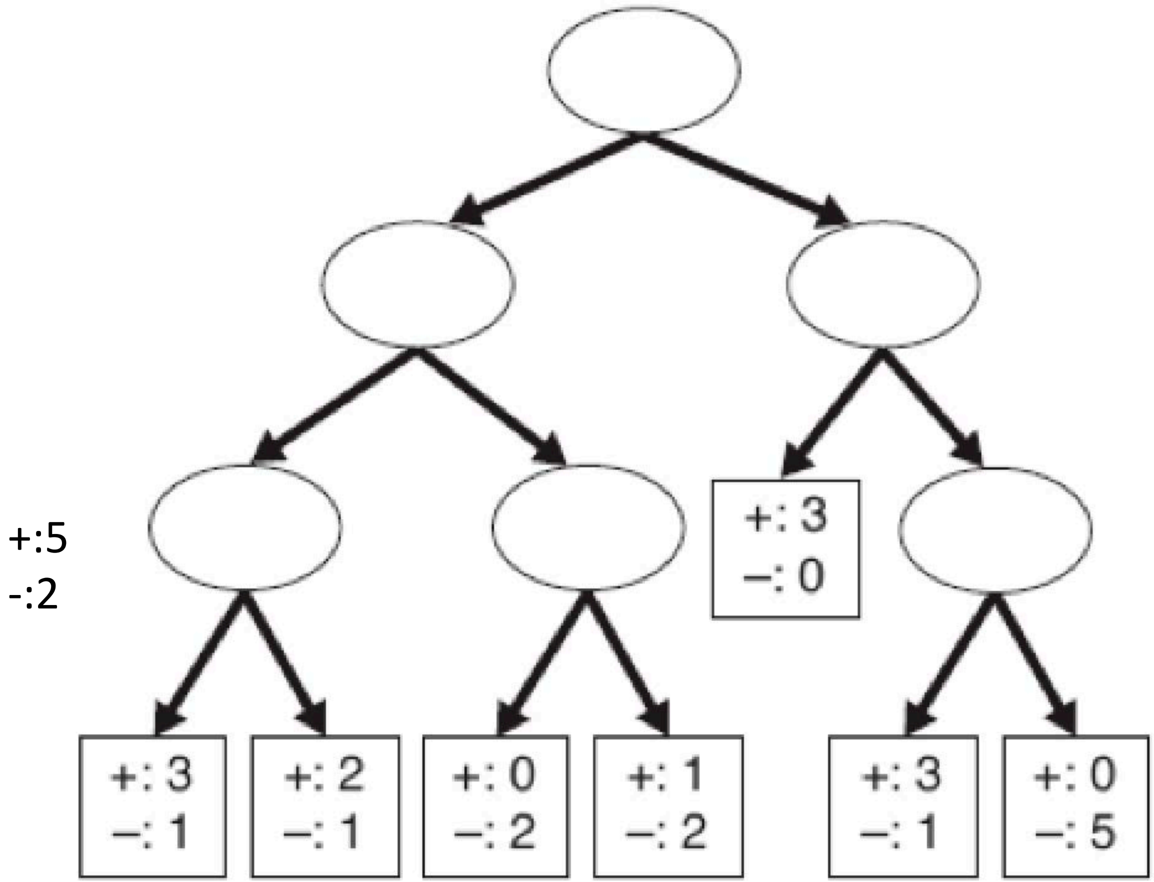
J48 PARAMETERS

Further techniques to prune a tree

"reducedErrorPruning": Replace a subtree with a leaf node with the most popular category label

"subtreeRaising": A subtree replaces its parent

REDUCED ERROR PRUNING



Decision Tree, T_L