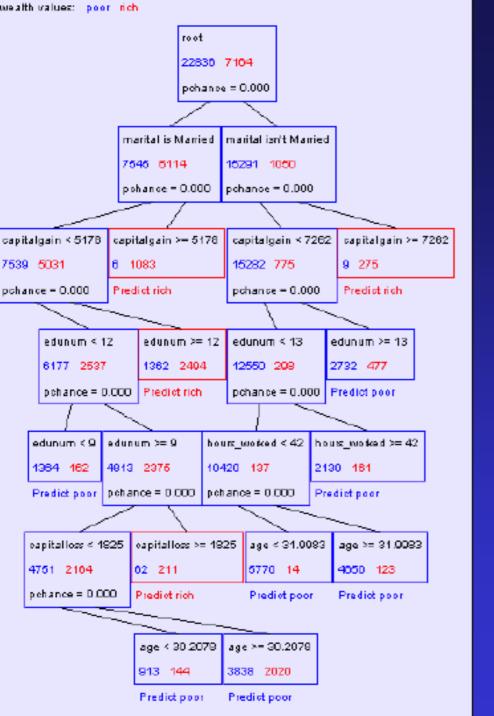
Census Data

| age | emplovme | education | edun | marital | | job | relation | race | aender | hour | country | wealth |
|-----|-----------|-----------|------|------------|---|-------------|------------|------------|--------|------|-----------|----------|
| -3- | | | | | | 1 | | | 3 | | , , | 11001111 |
| 39 | State_gov | Bachelors | 13 | Never_mar | | Adm cleric | Not_in_fan | White | Male | 40 | United_St | poor |
| | Self_emp_ | | | Married | | Exec_man | | White | Male | | United_St | |
| | Private | HS_grad | 9 | Divorced | | | Not_in_fan | White | Male | 40 | United_St | poor |
| 54 | Private | 11th | 7 | Married | | Handlers_c | Husband | Black | Male | | United_St | - |
| 28 | Private | Bachelors | 13 | Married | | Prof_speci | Wife | Black | Female | | Cuba | poor |
| 38 | Private | Masters | 14 | Married | | Exec_man | Wife | White | Female | 40 | United_St | poor |
| 50 | Private | 9th | 5 | Married_sr | | Other_serv | Not_in_fan | Black | Female | 16 | Jamaica | poor |
| 52 | Self_emp_ | HS_grad | 9 | Married | | Exec_man | Husband | White | Male | 45 | United_St | rich |
| 31 | Private | Masters | 14 | Never_mar | | Prof_speci | Not_in_fan | White | Female | 50 | United_St | rich |
| 42 | Private | Bachelors | 13 | Married | | Exec_man | Husband | White | Male | 40 | United_St | rich |
| 37 | Private | Some_coll | 10 | Married | | Exec_man | Husband | Black | Male | 80 | United_St | rich |
| 30 | State_gov | Bachelors | 13 | Married | | Prof_speci | Husband | Asian | Male | 40 | India | rich |
| 24 | Private | Bachelors | 13 | Never_mar | | Adm_cleric | Own_child | White | Female | 30 | United_St | poor |
| 33 | Private | Assoc_acc | 12 | Never_mar | | Sales | Not_in_fan | Black | Male | 50 | United_St | poor |
| 41 | Private | Assoc_voc | 11 | Married | | Craft_repai | Husband | Asian | Male | 40 | *MissingV | rich |
| 34 | Private | 7th_8th | 4 | Married | | Transport_ | Husband | Amer_India | Male | 45 | Mexico | poor |
| 26 | Self_emp_ | HS_grad | 9 | Never_mar | | Farming_fi | Own_child | White | Male | 35 | United_St | poor |
| 33 | Private | HS_grad | 9 | Never_mar | | Machine_c | Unmarried | White | Male | 40 | United_St | poor |
| | Private | 11th | | Married | | Sales | Husband | White | Male | | United_St | |
| 44 | Self_emp_ | Masters | 14 | Divorced | | Exec_man | Unmarried | White | Female | | United_St | |
| 41 | Private | Doctorate | 16 | Married | | Prof_speci | Husband | White | Male | 60 | United_St | rich |
| : | : | : | : | : | : | : | : | : | : | : | : | : |

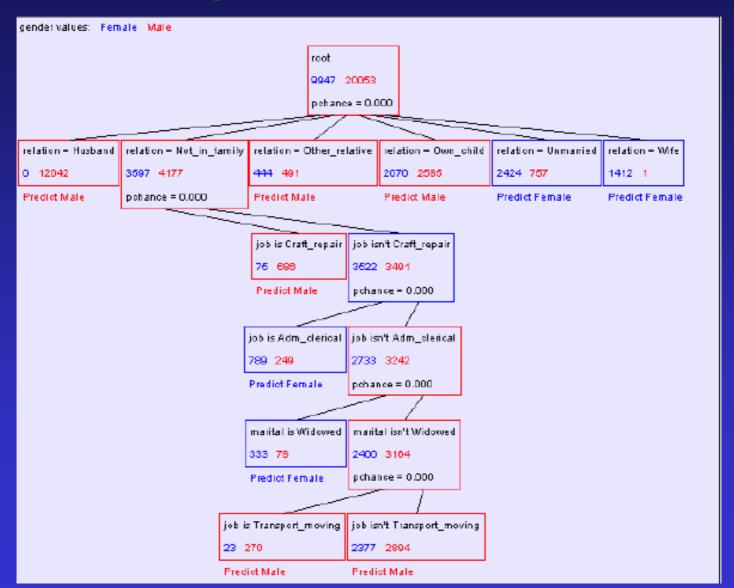


Predicting Wealth from Census



Predicting Age from Census

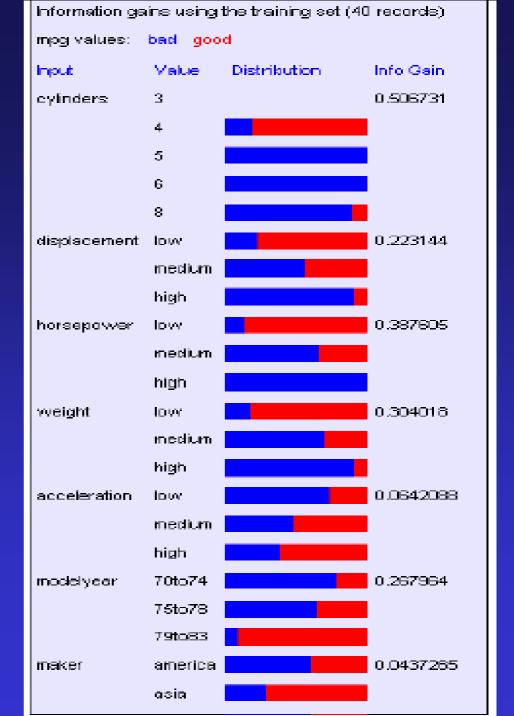
Predicting Gender from Census



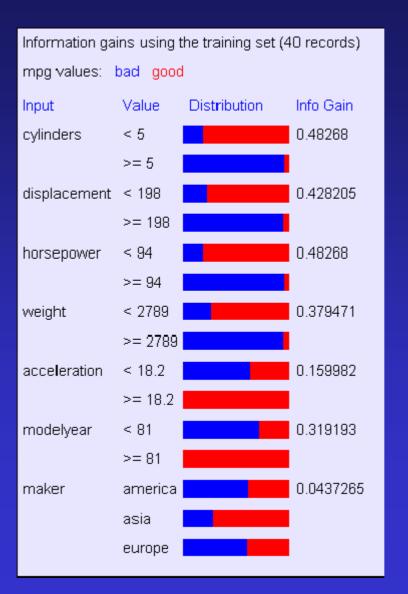
MPG Data Set (40 records from UCI repository)

| mpg | cylinders | displacement | horsepower | weight | acceleration | modelyear | maker |
|------|-----------|--------------|------------|--------|--------------|-----------|---------|
| | | | | | | | |
| good | 4 | low | low | low | high | 75to78 | asia |
| bad | 6 | medium | medium | medium | medium | 70to74 | america |
| bad | 4 | medium | medium | medium | low | 75to78 | europe |
| bad | 8 | high | high | high | low | 70to74 | america |
| bad | 6 | medium | medium | medium | medium | 70to74 | america |
| bad | 4 | low | medium | low | medium | 70to74 | asia |
| bad | 4 | low | medium | low | low | 70to74 | asia |
| bad | 8 | high | high | high | low | 75to78 | america |
| : | : | : | : | : | : | : | : |
| : | : | : | : | - | : | | : |
| : | | | : | - | : | | : |
| bad | 8 | high | high | high | low | 70to74 | america |
| good | 8 | high | medium | high | high | 79to83 | america |
| bad | 8 | high | high | high | low | 75to78 | america |
| good | 4 | low | low | low | low | 79to83 | america |
| bad | 6 | medium | medium | medium | high | 75to78 | america |
| good | 4 | medium | low | low | low | 79to83 | america |
| good | 4 | low | low | medium | high | 79to83 | america |
| bad | 8 | high | high | high | low | 70to74 | america |
| good | 4 | low | medium | low | medium | 75to78 | europe |
| bad | 5 | medium | medium | medium | medium | 75to78 | ешгоре |

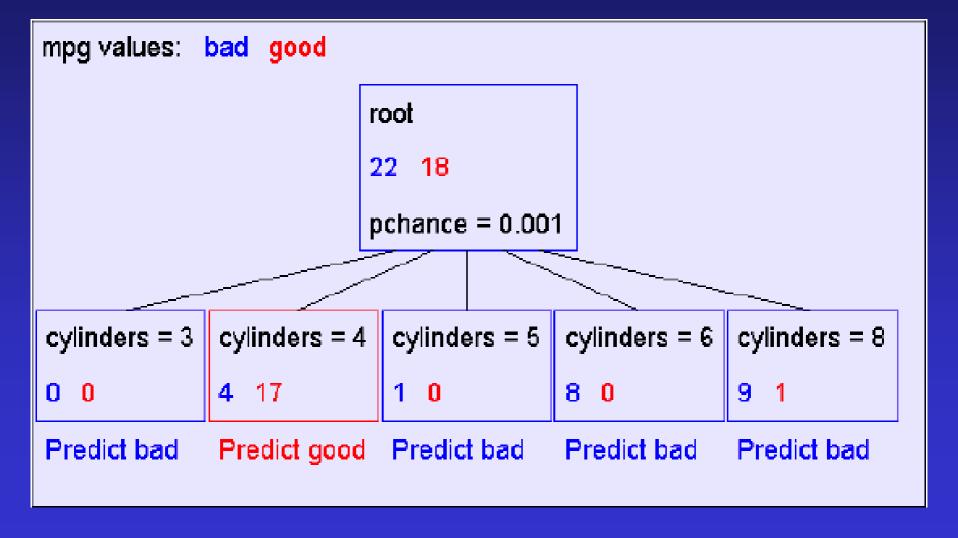
- Predicting MPG
 - Has good/bad values
- Look at all the Information Gains
- Cylinders has the highest IG

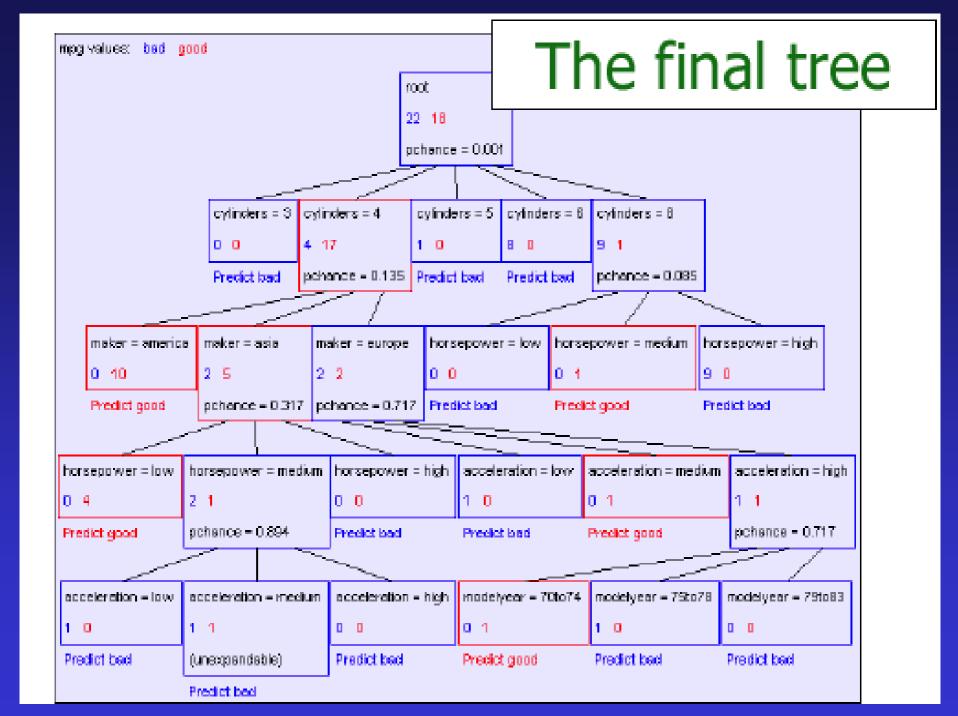


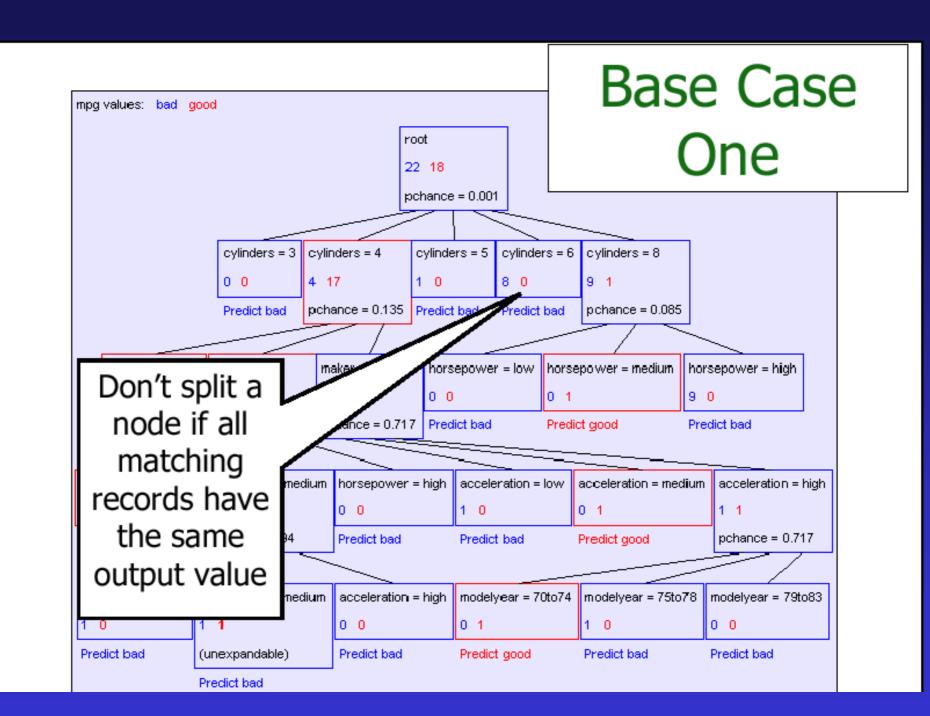
Example with MPG



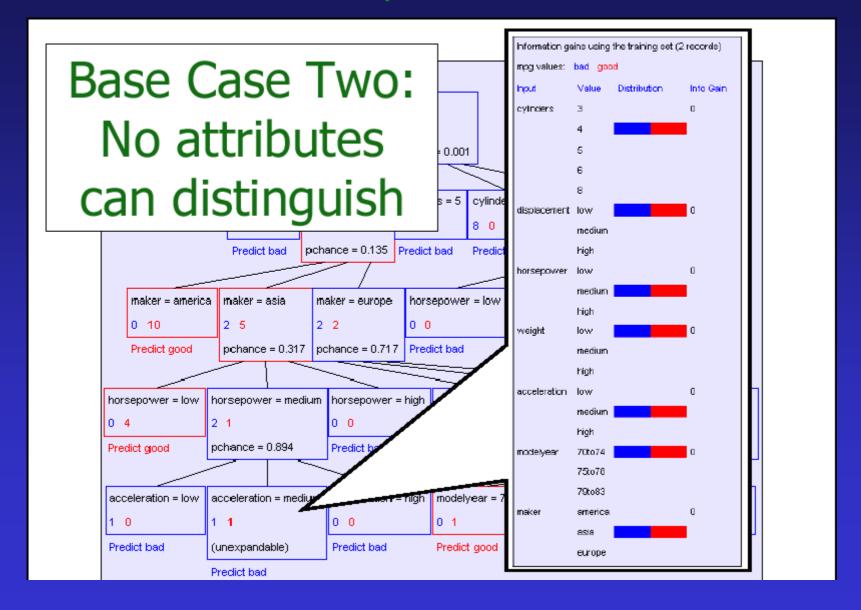
A Decision Stump

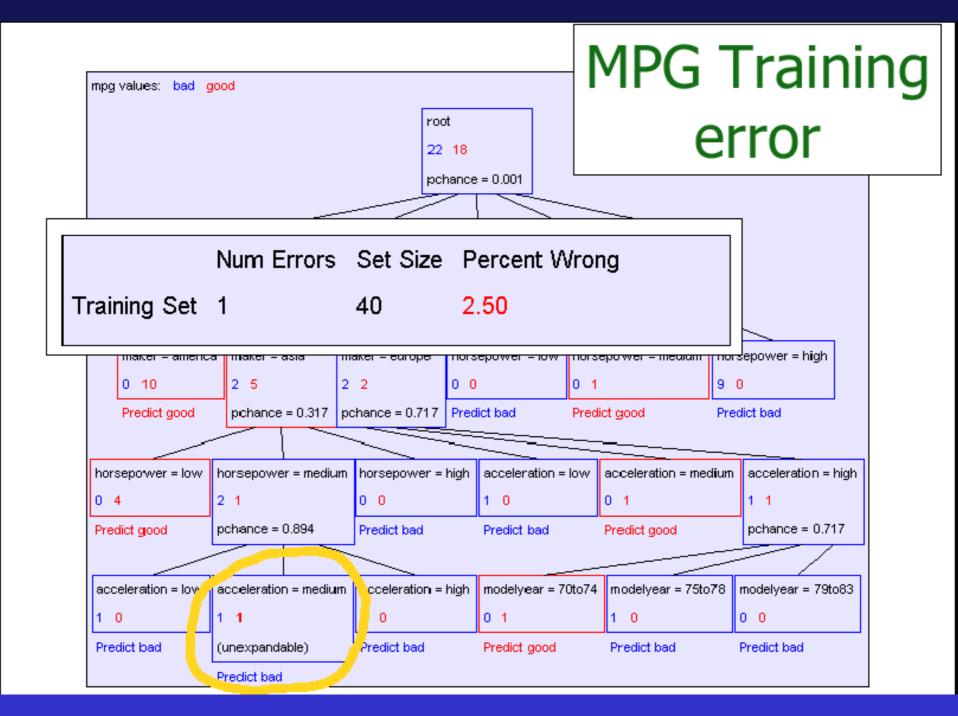


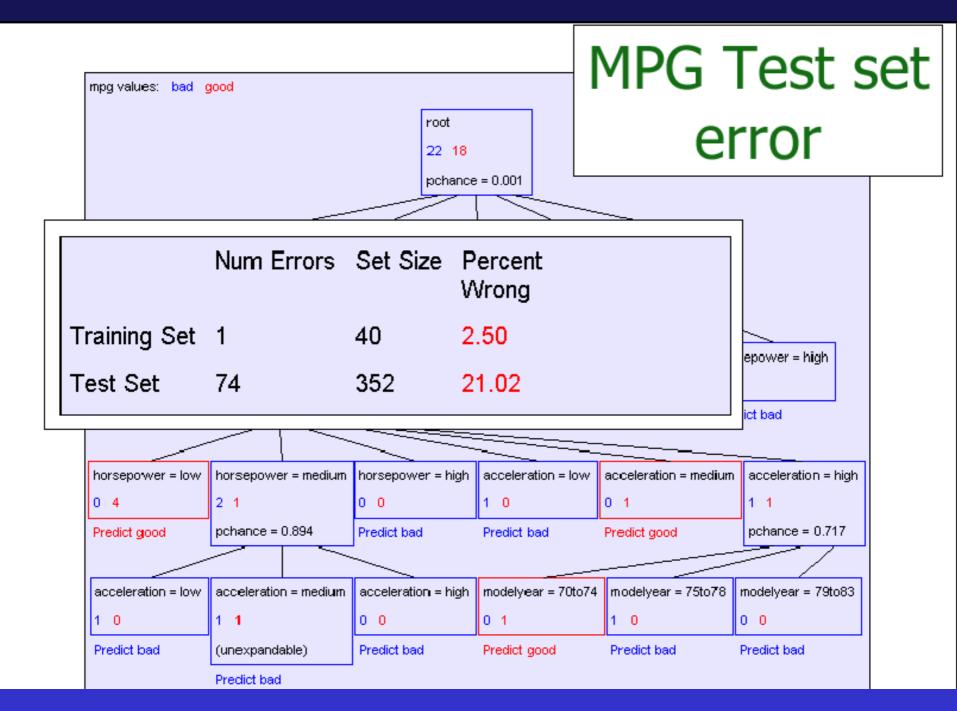


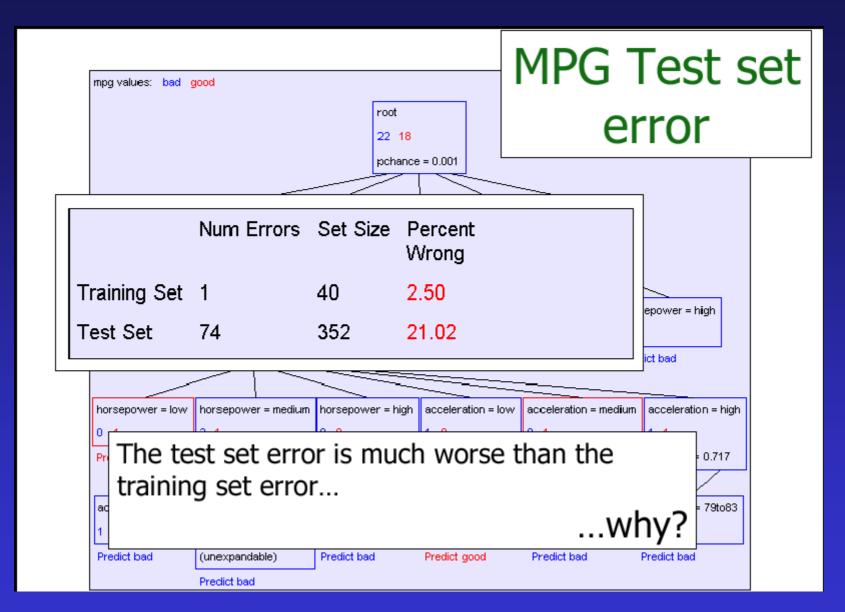


Base Case 2: Don't split if no attribute is useful



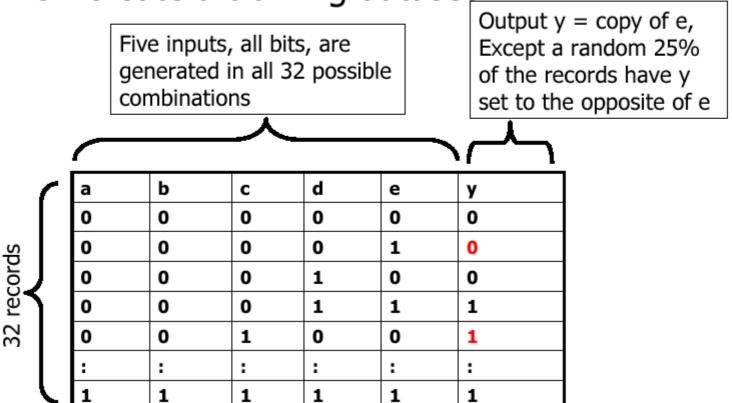






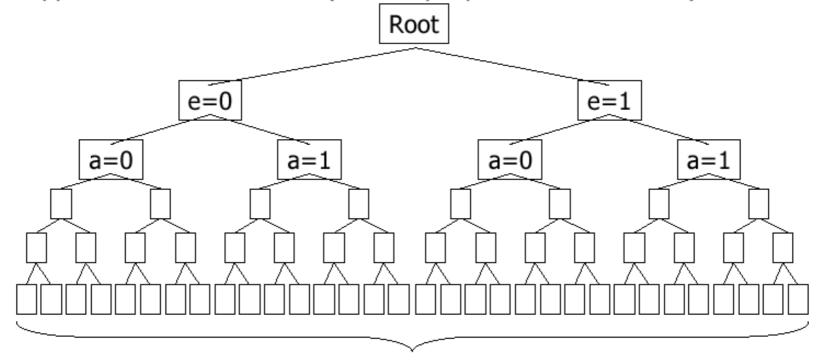
An artificial example

We'll create a training dataset



Building a tree with the artificial training set

Suppose we build a full tree (we always split until base case 2)



25% of these leaf node labels will be corrupted

Test Set generated by the same method

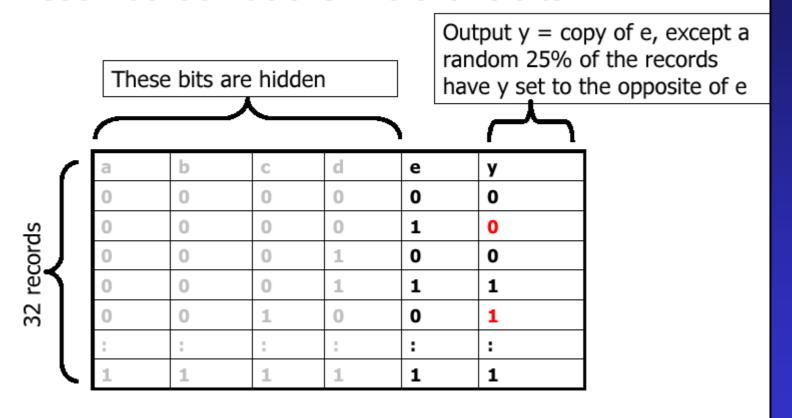
Testing the tree with the test set

| | 1/4 of the tree nodes are corrupted | 3/4 are fine |
|-------------------------------------------|----------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|
| 1/4 of the test set records are corrupted | 1/16 of the test set will be correctly predicted for the wrong reasons | 3/16 of the test set will be wrongly predicted because the test record is corrupted |
| 3/4 are fine | 3/16 of the test predictions will be wrong because the tree node is corrupted | 9/16 of the test predictions will be fine |

In total, we expect to be wrong on 3/8 of the test set predictions

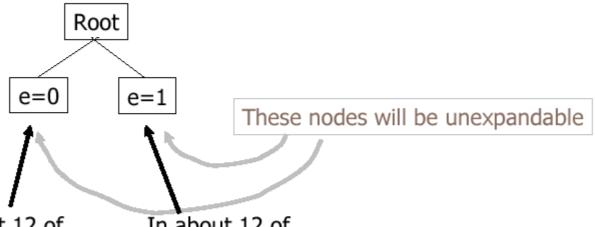
Suppose we had less data

Let's not look at the irrelevant bits



What decision tree would we learn now?

Without access to the irrelevant bits...



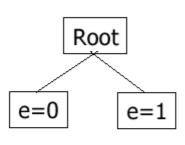
In about 12 of the 16 records in this node the output will be 0

So this will almost certainly predict 0

In about 12 of the 16 records in this node the output will be 1

So this will almost certainly predict 1

Without access to the irrelevant bits...



| | almost certainly none of the tree nodes are corrupted | almost certainly all are fine |
|-------------------------------------------------|----------------------------------------------------------------|------------------------------------------------------------------------------------------------|
| 1/4 of the test set records are corrupted | n/a | 1/4 of the test set will be wrongly predicted because the test record is corrupted |
| 3/4 are fine | n/a | 3/4 of the test predictions will be fine |

In total, we expect to be wrong on only 1/4 of the test set predictions

Over-fitting the Data

- ID3 grows each branch of tree just deeply enough to perfectly classify training samples
- Overfitting the data
 - when there is noise in the data or
 - when the number of training examples is too small to produce a representative sample of the true target function.

Over-fitting the Data

Definition:

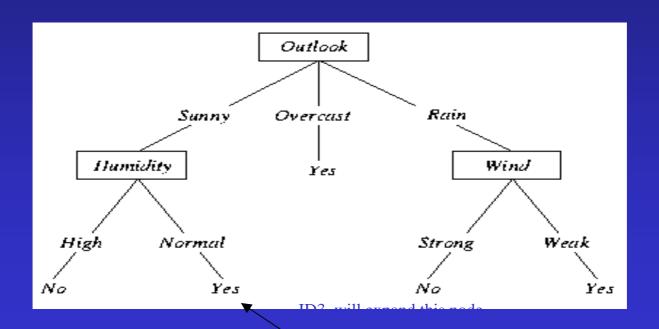
Given a hypothesis space H,

a hypothesis $h \in H$ is said to *overfit* the training data if there exists some alternative hypothesis $h' \in H$, such that h has smaller error than h' over the training examples,

but h' has a smaller error than h over the entire distribution of instances.

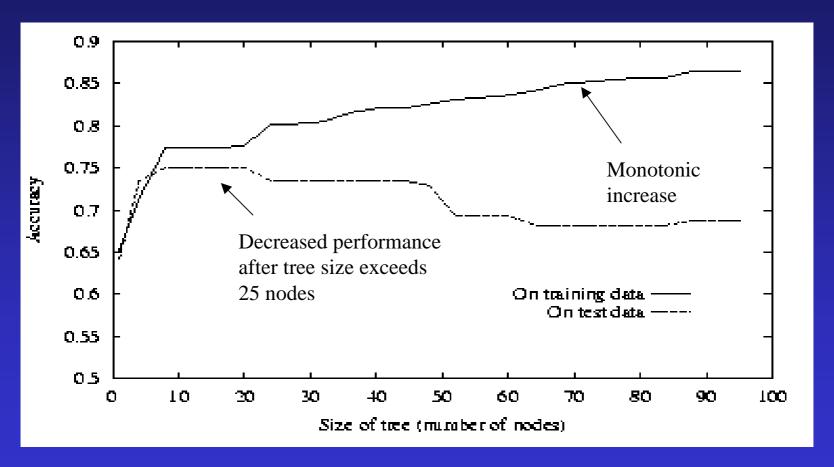
Effect of Noise in training data: cause over-fit

Consider effect of adding *incorrectly* labeled sample: (Outlook=Sunny,Temperature=Hot,Humidity=Normal,Wind=Strong, PlayTennis=No)



Over-fitting with ID3

Learning which medical patients have a form of diabetes



Noise in training data can cause over-fit

Two Approaches To Prevent Over-fitting

- 1. Stop growing tree earlier, before it reaching point where it perfectly classifies training data.
- 2. Allow tree to over-fit the data and then post-prune the tree.
- Although first approach is more direct, second approach found more successful in practice: because difficult to estimate when to stop
- Both need a criterion to determine final tree size

Criterion to Determine Correct Tree Size

- 1. Training and Validation Set Approach:
 - Use a separate set of examples, distinct from the training examples, to evaluate the utility of post-pruning nodes from the tree.
- 2. Use all available data for training,
 - but apply a statistical test (Chi-square test) to estimate whether expanding (or pruning) a particular node is likely to produce an improvement.
- 3. Use an explicit measure of the complexity
 - for encoding the training examples and the decision tree,
 - halting growth when this encoding size is minimized.

Training and Validation Set Approach

- Training Set: used to form learned hypotheses
- Validation Set:
 - used to evaluate the accuracy of this hypothesis over subsequent data
 - also, evaluate impact of pruning hypothesis
- Philosophy:
 - Validation set is unlikely to exhibit same random fluctuations as Training set
 - check against over-fitting

Validation Set

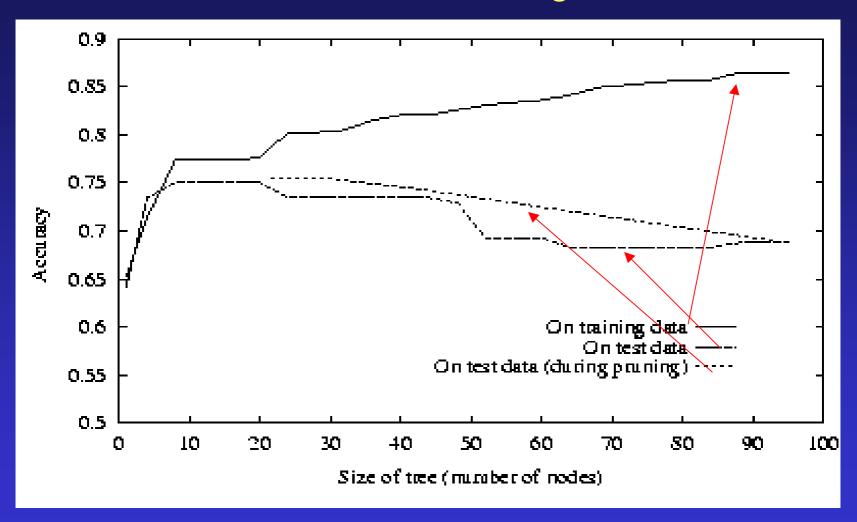
- Provides a safety check against overfitting spurious characteristics of data
- Needs to be large enough to provide a statistically significant sample of instances
- Typically validation set is one half size of training set

How to use validation set to Prune



- Consider each node of the decision nodes in the tree to be candidates for pruning
- Pruning a decision tree consists of
 - removing a sub-tree rooted at the node
 - making it a leaf node
 - assigning it the most common classification of the training examples affiliated with that node.
- Nodes are removed only if the resulting pruned tree performs no worse than the original over the validation set.

Effect of Pruning



Reduced Error Pruning Properties

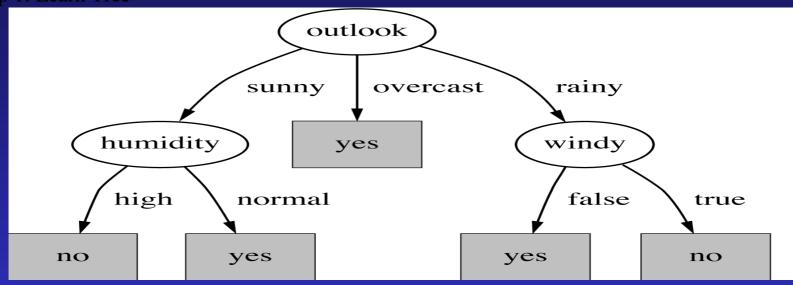
- When pruning begins tree is at maximum size and lowest accuracy over test set
- As pruning proceeds no of nodes is reduced and accuracy over test set increases
- Disadvantage: when data is limited, no of samples available for training is further reduced
 - Rule post-pruning is one approach (discussed next)
 - Alternatively, partition available data several times in multiple ways and then average the results

Rule Post-Pruning

- Useful when data is limited
- practical method for finding high accuracy hypotheses
- variant of rule post-pruning is used by C4.5
- C4.5 system is an outgrowth of ID3 algorithm
- C4.5 also allows dealing with numerical attributes, missing values, noisy data

Example for Rule Post-pruning

Step 1: Learn Tree



Step 2: Convert tree to equivalent rules: generate one rule for each leaf node

Leftmost Path: IF (Outlook= Sunny) ^ (Humidity = High) THEN Play-Tennis = no

Why Convert to rules before pruning?

- Converting to rules allows distinguishing among the different contexts in which a decision node is used.
- Converting to rules removes the distinction between attribute tests that occur near the root of the tree and those that occur near the leaves.
- Converting to rules improves readability.

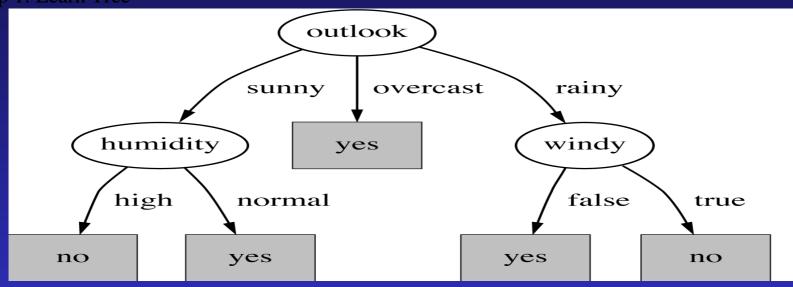
Rule Post-Pruning: Four steps

1 Infer the decision tree from the training set, growing the tree until the training data fits as well as possible and allowing the overfitting to occur.

2 Convert the learned tree into an equivalent set of rules by creating one rule for each path from the root node to the leaf node.

Example for Rule Post-pruning

Step 1: Learn Tree



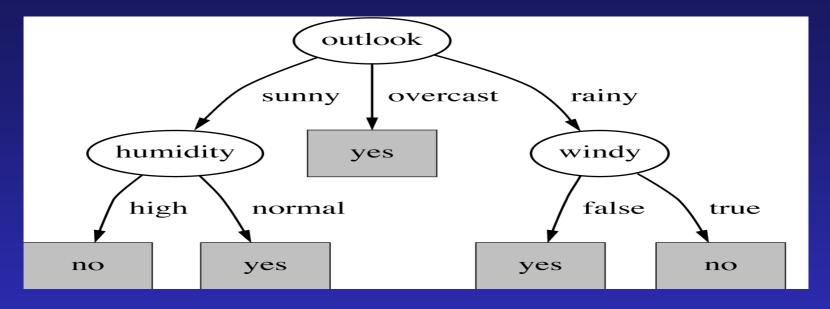
Step 2: Convert tree to equivalent rules: generate one rule for each leaf node

Leftmost Path: IF (Outlook= Sunny) ^ (Humidity = High)
THEN Play-Tennis = no

Rule Post-Pruning: Four steps

- 3 Prune (generalize) each rule by removing any preconditions that result in improving its estimated accuracy.
- 4 Sort the pruned rules by their estimated accuracy and consider them in this sequence when classifying subsequent instances.

Example for Rule Post-pruning



Leftmost Path: IF (Outlook= Sunny) ^ (Humidity = High) THEN Play-Tennis = no

Each leaf node is a consequent(post-condition),

Each path from root node is the antecedent (precondition)

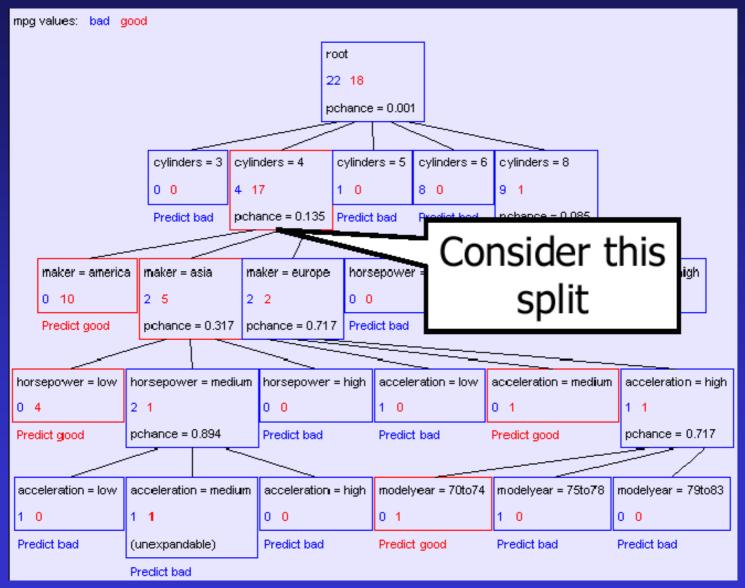
Pruned Rules

- IF (Outlook = sunny) ^ (Humidity = High)
- THEN PlayTennis = No
- Consider removing the Preconditions
 - (Outlook = sunny)
 - (Humidity = High)
- Select whichever pruning steps produces the greatest improvement in estimated accuracy
- Consider second precondition as the next pruning step

Estimation of Rule Accuracy

- C4.5 evaluates performance on the training set itself
- Use a pessimistic estimate to remove bias in favor of rules
 - Calculate rule accuracy over training examples to which it applies
 - Calculate standard deviation of accuracy assuming binomial distribution
 - For given estimate use lower bound of 95% confidence interval
 - observed accuracy 1.96 x standard deviation

Chi Squared Approach to Avoid Overfitting



A chi-squared test

- Suppose that mpg was completely uncorrelated with maker.
- What is the chance we'd have seen data of at least this apparent level of association anyway?

By using a particular kind of chi-squared test, the answer is 13.5%.

What is a chi squared test?

Chi Squared Test

Using Chi-squared to avoid overfitting

- Build the full decision tree as before.
- But when you can grow it no more, start to prune:
 - Beginning at the bottom of the tree, delete splits in which $p_{chance} > MaxPchance$.
 - Continue working you way up until there are no more prunable nodes.

MaxPchance is a magic parameter you must specify to the decision tree, indicating your willingness to risk fitting noise.

Handling Training Examples with Missing Attribute Values

Strategies

- assign missing attribute the value that is most common among training examples at node n.
- assign it the most common value among examples at node n that have classification c(x)
- More complex procedure:
 - assign a probability to each of the possible values of A rather than simply assigning the most common value to A(x)
 - e.g., P(A(x)=1) = 0.6 and P(A(x)=0)=0.4
 - use fractional samples for computing information Gain

Handling Attributes with Differing Costs (3.7.5)

- Strategies
 - replace information gain attribute selection measure by

$$\frac{Gain^2(S,A)}{Cost(A)}$$

use attribute selection measure

$$\frac{2^{Gain(S,A)}-1}{(Cost(A)+1)^w}$$

Inductive Bias in ID3

- What is the policy by which ID3 generalizes from observed training examples to classify unseen instances?
- Basis for choosing one consistent hypothesis over others

Inductive Bias in Decision Tree Learning

- Approximate inductive bias of ID3:
 - Shorter trees are preferred over larger trees.
- Breadth First Search ID3 (BFS-ID3)
 - Searches all trees of depth1, all trees of depth 2, etc and produces same tree as ID3
- ID3 is more efficient than BFS-ID3,
 - Performs greedy heuristic search
 - Does not conduct entire breadth-first search
- A closer approximation to the inductive bias of ID3:
 - Shorter trees are preferred over longer trees.
 - Trees that place high information gain attributes close to the root are preferred over those that do not.

Restriction Biases and Preference Biases

- ID3 has Preference Bias
 - searches a complete hypothesis space (i.e., one capable of expressing any finite discrete-valued function).
 - inductive bias is a preference for certain hypotheses
 - Referred to as preference bias or search bias
- Candidate-Elimination has Restriction Bias
 - The version space Candidate-Elimination algorithm searches an *incomplete* hypothesis space (*i.e.*, one that can express only a subset of the potentially teachable concepts).
 - inductive bias is a restriction on set of hypotheses considered
 - Referred to as restriction bias or language bias.

Why Prefer Short Hypotheses?

- ID3 has an inductive bias for favoring shorter decision trees
- Occam's Razor: Prefer the simplest hypothesis that fits the data.
- Philosophical issue, unresolved
 - William of Occam, ca 1320, while shaving
- There are fewer short hypotheses than long ones
 - less likely to find a short hypothesis that coincidentally fits the data
 - many long hypotheses fail to generalize subsequently
 - prefer 5-node tree to fit 20 examples than 500 node tree
 - polynomial versus linear fit of noisy data



Summary

- Decision tree learning provides a practical method for concept learning and for learning discrete-valued functions.
- ID3 searches a complete hypothesis space (i.e., the space of decision trees can represent any discretevalued function defined over discrete-valued instances).
- The inductive bias implicit in ID3 includes a preference for smaller trees.

Summary, continued

- Overfitting the training data is an important issue in decision tree learning.
- A large variety of extensions to the basic ID3 algorithm has been developed by different researchers.