

Note to other teachers and users of these slides.
Andrew would be delighted if you found this source material useful in giving your own lectures. Feel free to use these slides verbatim, or to modify them to fit your own needs. PowerPoint originals are available. If you make use of a significant portion of these slides in your own lecture, please include this message, or the following link to the source repository of Andrew's tutorials: <http://www.cs.cmu.edu/~awm/tutorials>. Comments and corrections gratefully received.

Decision Trees

Andrew W. Moore
Professor
School of Computer Science
Carnegie Mellon University

www.cs.cmu.edu/~awm
awm@cs.cmu.edu
412-268-7599

Copyright © Andrew W. Moore

Slide 1

Outline

- ▶ Machine Learning Datasets
- What is Classification?
- Contingency Tables
- OLAP (Online Analytical Processing)
- What is Data Mining?
- Searching for High Information Gain
- Learning an unpruned decision tree recursively
- Training Set Error
- Test Set Error
- Overfitting
- Avoiding Overfitting
- Information Gain of a real valued input
- Building Decision Trees with real Valued Inputs
- Andrew's homebrewed hack: Binary Categorical Splits
- Example Decision Trees

Copyright © Andrew W. Moore

Slide 2

Here is a dataset

age	employment	education	education_num	marital	...	job	relation	race	gender	hours	country	wealth
39	State_gov	Bachelors	13	Never_married	...	Adm_clerk	Not_in_family	White	Male	40	United_States	poor
51	Self_employed	Bachelors	13	Married	...	Exec_manager	Husband	White	Male	13	United_States	poor
39	Private	HS_grad	9	Divorced	...	Handlers_cleaners	Not_in_family	White	Male	40	United_States	poor
54	Private	11th	7	Married	...	Handlers_cleaners	Husband	Black	Male	40	United_States	poor
28	Private	Bachelors	13	Married	...	Prof_spec	Wife	Black	Female	40	Cuba	poor
38	Private	Masters	14	Married	...	Exec_manager	Wife	White	Female	40	United_States	poor
50	Private	9th	5	Married_spl	...	Other_serv	Not_in_family	Black	Female	16	Jamaica	poor
52	Self_employed	HS_grad	9	Married	...	Exec_manager	Husband	White	Male	45	United_States	rich
31	Private	Masters	14	Never_married	...	Prof_spec	Not_in_family	White	Female	50	United_States	rich
42	Private	Bachelors	13	Married	...	Exec_manager	Husband	White	Male	40	United_States	rich
37	Private	Some_coll	10	Married	...	Exec_manager	Husband	Black	Male	80	United_States	rich
30	State_gov	Bachelors	13	Married	...	Prof_spec	Husband	Asian	Male	40	India	rich
24	Private	Bachelors	13	Never_married	...	Adm_clerk	Own_child	White	Female	30	United_States	poor
33	Private	Assoc_acad	12	Never_married	...	Sales	Not_in_family	Black	Male	50	United_States	poor
41	Private	Assoc_voc	11	Married	...	Craft_repair	Husband	Asian	Male	40	MissingV	rich
34	Private	7th_8th	4	Married	...	Transportation	Husband	Amer_Indian	Male	45	Mexico	poor
26	Self_employed	HS_grad	9	Never_married	...	Farming_fishing	Own_child	White	Male	35	United_States	poor
33	Private	HS_grad	9	Never_married	...	Machine_c	Unmarried	White	Male	40	United_States	poor
38	Private	11th	7	Married	...	Sales	Husband	White	Male	50	United_States	poor
44	Self_employed	Masters	14	Divorced	...	Exec_manager	Unmarried	White	Female	45	United_States	rich
41	Private	Doctorate	16	Married	...	Prof_spec	Husband	White	Male	60	United_States	rich
:	:	:	:	:	:	:	:	:	:	:	:	:

48,000 records, 16 attributes [Kohavi 1995]

Copyright © Andrew W. Moore

Slide 3

Outline

Machine Learning Datasets



What is Classification?

Contingency Tables

OLAP (Online Analytical Processing)

What is Data Mining?

Searching for High Information Gain

Learning an unpruned decision tree recursively

Training Set Error

Test Set Error

Overfitting

Avoiding Overfitting

Information Gain of a real valued input

Building Decision Trees with real Valued Inputs

Andrew's homebrewed hack: Binary Categorical Splits

Example Decision Trees

Copyright © Andrew W. Moore

Slide 4

Classification

- A Major Data Mining Operation
- Give one attribute (e.g wealth), try to predict the value of new people's wealths by means of some of the other available attributes.
- Applies to categorical outputs
 - Categorical attribute: an attribute which takes on two or more discrete values. Also known as a symbolic attribute.
 - Real attribute: a column of real numbers

Copyright © Andrew W. Moore

Slide 5

Today's lecture

- Information Gain for measuring association between inputs and outputs
- Learning a decision tree classifier from data

Copyright © Andrew W. Moore

Slide 6

About this dataset

- It is a tiny subset of the 1990 US Census.
- It is publicly available online from the UCI Machine Learning Datasets repository

Used Attributes

age	edunum	race	hours_worked
employment	marital	gender	country
taxweighting	job	capitalgain	wealth
education	relation	capitalloss	agegroup

This color = Real-valued This color = Symbol-valued

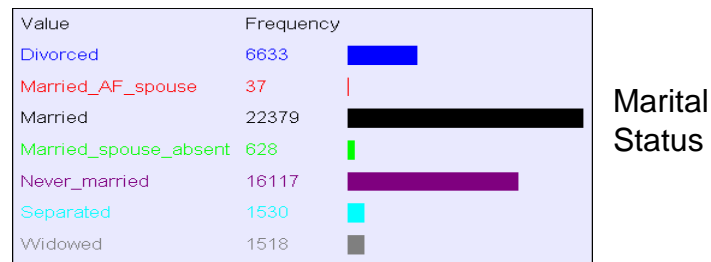
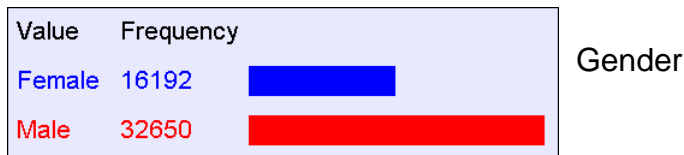
Successfully loaded a new dataset from the file \adult.fds. It has 16 attributes and 48842 records.

Copyright © Andrew W. Moore

Slide 7

What can you do with a dataset?

- Well, you can look at histograms...



Copyright © Andrew W. Moore

Slide 8

Outline

Machine Learning Datasets
What is Classification?
▶ Contingency Tables
OLAP (Online Analytical Processing)
What is Data Mining?
Searching for High Information Gain
Learning an unpruned decision tree recursively
Training Set Error
Test Set Error
Overfitting
Avoiding Overfitting
Information Gain of a real valued input
Building Decision Trees with real Valued Inputs
Andrew's homebrewed hack: Binary Categorical Splits
Example Decision Trees

Copyright © Andrew W. Moore

Slide 9

Contingency Tables

- A better name for a histogram:
A One-dimensional Contingency Table
- Recipe for making a k-dimensional contingency table:
 1. Pick k attributes from your dataset. Call them a_1, a_2, \dots, a_k .
 2. For every possible combination of values, $a_1 = x_1, a_2 = x_2, \dots, a_k = x_k$, record how frequently that combination occurs

Fun fact: A database person would call this a "k-dimensional datacube"

Copyright © Andrew W. Moore

Slide 10

A 2-d Contingency Table










wealth values:		poor	rich
agegroup	10s	2507	3
	20s	11262	743
	30s	9468	3461
	40s	6738	3986
	50s	4110	2509
	60s	2245	809
	70s	668	147
	80s	115	16
	90s	42	13

- For each pair of values for attributes (agegroup, wealth) we can see how many records match.

Copyright © Andrew W. Moore

Slide 11

A 2-d Contingency Table

wealth values:		poor	rich	
agegroup	10s	2507	3	
	20s	11262	743	
	30s	9468	3461	
	40s	6738	3986	
	50s	4110	2509	
	60s	2245	809	
	70s	668	147	
	80s	115	16	
	90s	42	13	

- Easier to appreciate graphically

Copyright © Andrew W. Moore

Slide 12

A 2-d Contingency Table

wealth values: poor rich	
agegroup	10s 2507 3
	20s 11262 743
	30s 9468 3461
	40s 6738 3986
	50s 4110 2509
	60s 2245 809
	70s 668 147
	80s 115 16
	90s 42 13

- Easier to see “interesting” things if we stretch out the histogram bars

Copyright © Andrew W. Moore

Slide 13

A bigger 2-d contingency table

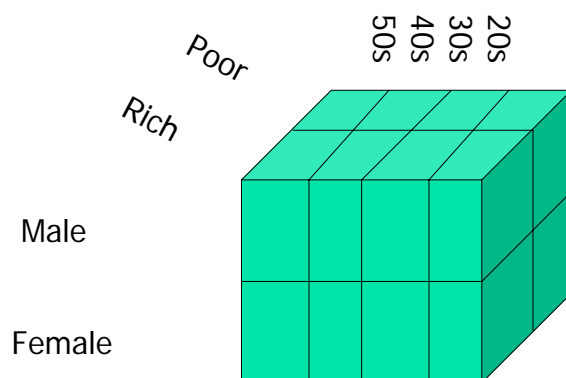
job values: Adm_clerical Craft_repair Farming_fishing Machine_op_inspct Priv_house_serv Protective_serv Tech_support	
MissingValue	Armed_Forces Exec_managerial Handlers_cleaners Other_service Prof_specialty Sales Transport_moving
marital Divorced	270 1192 0 679 890 90 197 434 762 46 795 121 664 239 254
Married_AF_spouse	5 6 0 4 3 1 1 1 5 0 4 1 5 0 1
Married	928 1495 7 3818 3600 869 724 1469 1088 27 3182 583 2491 609 1489
Married_spouse_absent	45 84 0 77 52 35 32 37 92 9 64 7 55 9 30
Never_married	1242 2360 8 1301 1260 434 1029 872 2442 99 1849 237 1992 506 486
Separated	97 224 0 160 126 23 63 123 275 21 145 23 146 48 56
Widowed	222 250 0 73 155 38 26 86 259 40 133 11 151 35 39

Copyright © Andrew W. Moore

Slide 14

3-d contingency tables

- These are harder to look at!



Copyright © Andrew W. Moore

Slide 15

Outline

Machine Learning Datasets
What is Classification?
Contingency Tables
▶ OLAP (Online Analytical Processing)
What is Data Mining?
Searching for High Information Gain
Learning an unpruned decision tree recursively
Training Set Error
Test Set Error
Overfitting
Avoiding Overfitting
Information Gain of a real valued input
Building Decision Trees with real Valued Inputs
Andrew's homebrewed hack: Binary Categorical Splits
Example Decision Trees

Copyright © Andrew W. Moore

Slide 16

On-Line Analytical Processing (OLAP)

- Software packages and database add-ons to do this are known as OLAP tools
- They usually include point and click navigation to view slices and aggregates of contingency tables
- They usually include nice histogram visualization

Copyright © Andrew W. Moore

Slide 17

Time to stop and think

- Why would people want to look at contingency tables?

Copyright © Andrew W. Moore

Slide 18

Let's continue to think

- With 16 attributes, how many 1-d contingency tables are there?
- How many 2-d contingency tables?
- How many 3-d tables?
- With 100 attributes how many 3-d tables are there?

Copyright © Andrew W. Moore

Slide 19

Let's continue to think

- With 16 attributes, how many 1-d contingency tables are there? **16**
- How many 2-d contingency tables? **16-choose-2 = $16 * 15 / 2 = 120$**
- How many 3-d tables? **560**
- With 100 attributes how many 3-d tables are there? **161,700**

Copyright © Andrew W. Moore

Slide 20

Manually looking at contingency tables

- Looking at one contingency table: *can be as much fun as reading an interesting book*
- Looking at ten tables: *as much fun as watching CNN*
- Looking at 100 tables: *as much fun as watching an infomercial*
- Looking at 100,000 tables: *as much fun as a three-week November vacation in Duluth with a dying weasel.*

Copyright © Andrew W. Moore

Slide 21

Outline

Machine Learning Datasets
What is Classification?
Contingency Tables
OLAP (Online Analytical Processing)
▶ What is Data Mining?
Searching for High Information Gain
Learning an unpruned decision tree recursively
Training Set Error
Test Set Error
Overfitting
Avoiding Overfitting
Information Gain of a real valued input
Building Decision Trees with real Valued Inputs
Andrew's homebrewed hack: Binary Categorical Splits
Example Decision Trees

Copyright © Andrew W. Moore

Slide 22

Data Mining

- Data Mining is all about automating the process of searching for patterns in the data.

Which patterns are interesting?

Which might be mere illusions?

And how can they be exploited?

Copyright © Andrew W. Moore

Slide 23

Data Mining

- Data Mining is all about automating the process of searching for patterns in the data.

Which patterns are interesting?

Which might be mere illusions?

And how can they be exploited?

That's what we'll look at right now.

And the answer will turn out to be the engine that drives decision tree learning.

Copyright © Andrew W. Moore

Slide 24

Deciding whether a pattern is interesting

- We will use **information theory**
- A very large topic, originally used for compressing signals
- But more recently used for data mining...

Copyright © Andrew W. Moore

Slide 25

Deciding whether a pattern is interesting

- We will use **information theory**
- A very large topic, originally used for compressing signals
- But more recently used for data mining...

(The topic of Information Gain will now be discussed, but you will find it in a separate Andrew Handout)

Copyright © Andrew W. Moore

Slide 26

Outline

Machine Learning Datasets
 What is Classification?
 Contingency Tables
 OLAP (Online Analytical Processing)
 What is Data Mining?
 ► Searching for High Information Gain
 Learning an unpruned decision tree recursively
 Training Set Error
 Test Set Error
 Overfitting
 Avoiding Overfitting
 Information Gain of a real valued input
 Building Decision Trees with real Valued Inputs
 Andrew's homebrewed hack: Binary Categorical Splits
 Example Decision Trees

Copyright © Andrew W. Moore

Slide 27

Searching for High Info Gains

- Given something (e.g. wealth) you are trying to predict, it is easy to ask the computer to find which attribute has highest information gain for it.

wealth values: poor rich			
relation	Husband	10870 8846	$H(\text{wealth} \mid \text{relation} = \text{Husband}) = 0.992385$
	Not_in_family	11307 1276	$H(\text{wealth} \mid \text{relation} = \text{Not_in_family}) = 0.473439$
	Other_relative	1454 52	$H(\text{wealth} \mid \text{relation} = \text{Other_relative}) = 0.216617$
	Own_child	7470 111	$H(\text{wealth} \mid \text{relation} = \text{Own_child}) = 0.110192$
	Unmarried	4816 309	$H(\text{wealth} \mid \text{relation} = \text{Unmarried}) = 0.328606$
	Wife	1238 1093	$H(\text{wealth} \mid \text{relation} = \text{Wife}) = 0.997207$
$H(\text{wealth}) = 0.793844$ $H(\text{wealth} \mid \text{relation}) = 0.628421$			
$IG(\text{wealth} \mid \text{relation}) = 0.165423$			

Copyright © Andrew W. Moore

Slide 28

Outline

Machine Learning Datasets
What is Classification?
Contingency Tables
OLAP (Online Analytical Processing)
What is Data Mining?
Searching for High Information Gain
▶ Learning an unpruned decision tree recursively
Training Set Error
Test Set Error
Overfitting
Avoiding Overfitting
Information Gain of a real valued input
Building Decision Trees with real Valued Inputs
Andrew's homebrewed hack: Binary Categorical Splits
Example Decision Trees

Copyright © Andrew W. Moore

Slide 29

Learning Decision Trees

- A Decision Tree is a tree-structured plan of a set of attributes to test in order to predict the output.
- To decide which attribute should be tested first, simply find the one with the highest information gain.
- Then recurse...

Copyright © Andrew W. Moore

Slide 30

A small dataset: Miles Per Gallon

40
Records

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	europa
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	europa
bad	5	medium	medium	medium	medium	75to78	europa

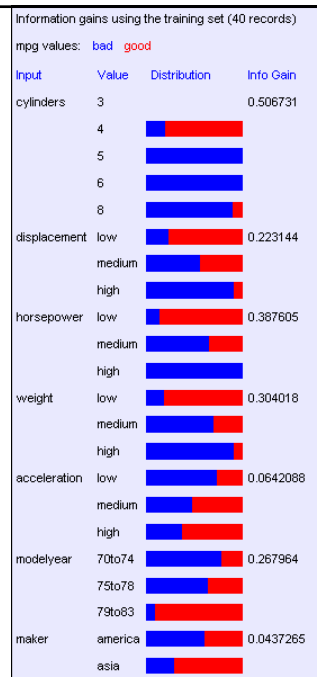
From the UCI repository (thanks to Ross Quinlan)

Copyright © Andrew W. Moore

Slide 31

Suppose we want to
predict MPG.

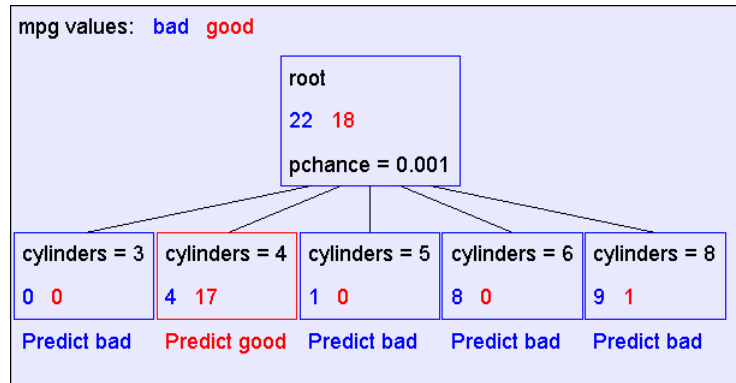
Look at all
the
information
gains...



Copyright © Andrew W. Moore

Slide 32

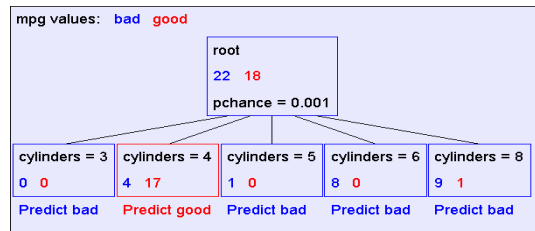
A Decision Stump



Copyright © Andrew W. Moore

Slide 33

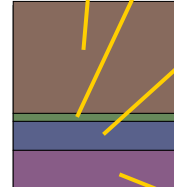
Recursion Step



Take the
Original
Dataset..



And partition it
according
to the value of
the attribute
we split on



Records
in which
cylinders
= 4

Records
in which
cylinders
= 5

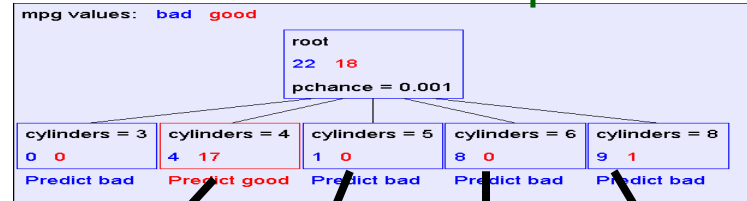
Records
in which
cylinders
= 6

Records
in which
cylinders
= 8

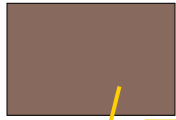
Copyright © Andrew W. Moore

Slide 34

Recursion Step



Build tree from
These records..



Records in
which
cylinders = 4

Build tree from
These records..



Records in
which
cylinders = 5

Build tree from
These records..



Records in
which
cylinders = 6

Build tree from
These records..

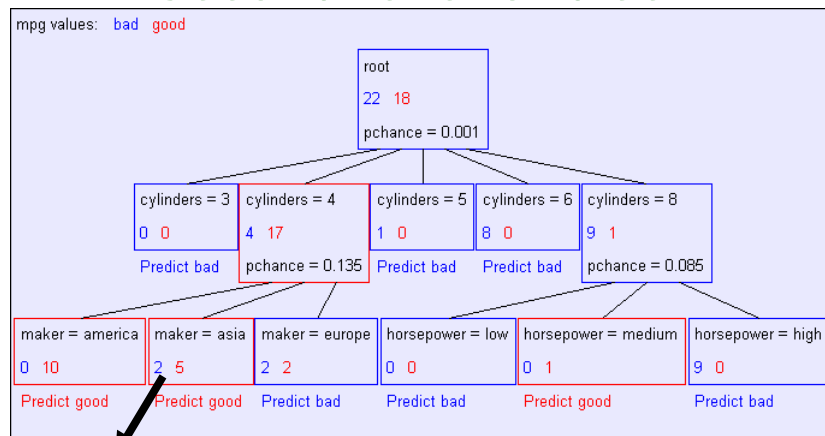


Records in
which
cylinders = 8

Copyright © Andrew W. Moore

Slide 35

Second level of tree

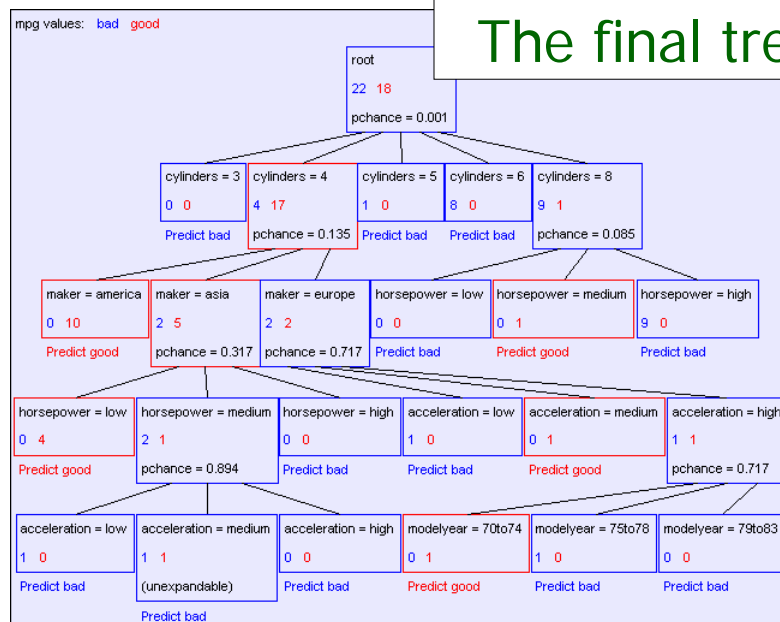


Recursively build a tree from the seven
records in which there are four cylinders and
the maker was based in Asia

(Similar recursion in the
other cases)

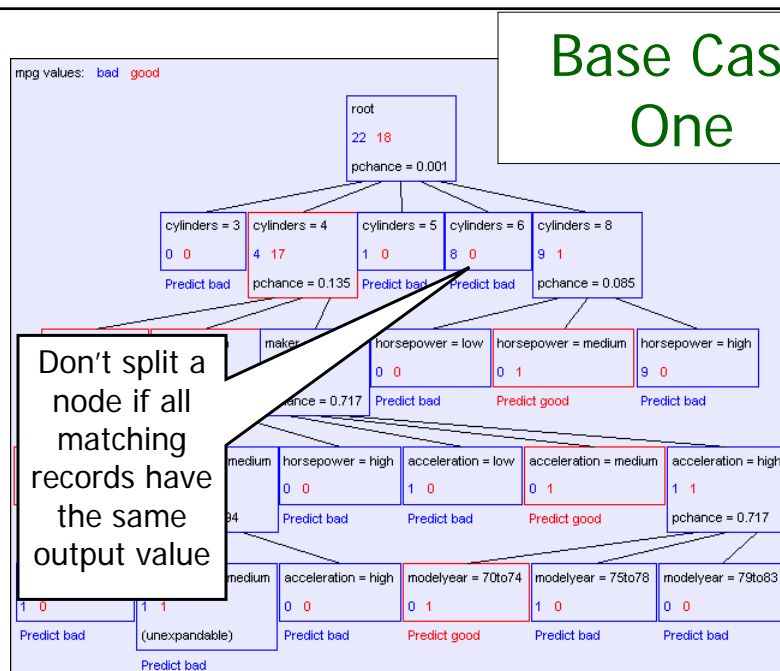
Copyright © Andrew W. Moore

Slide 36



Copyright © Andrew W. Moore

Slide 37

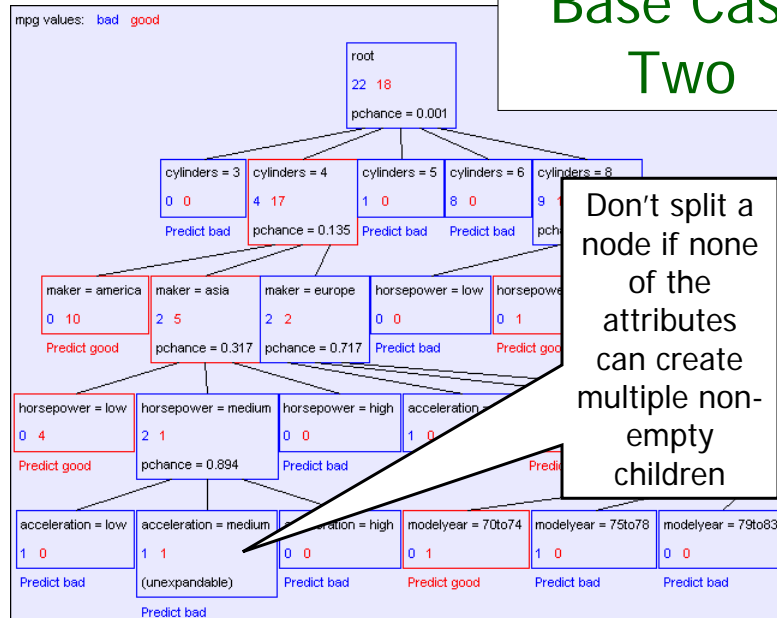


Don't split a node if all matching records have the same output value

Copyright © Andrew W. Moore

Slide 38

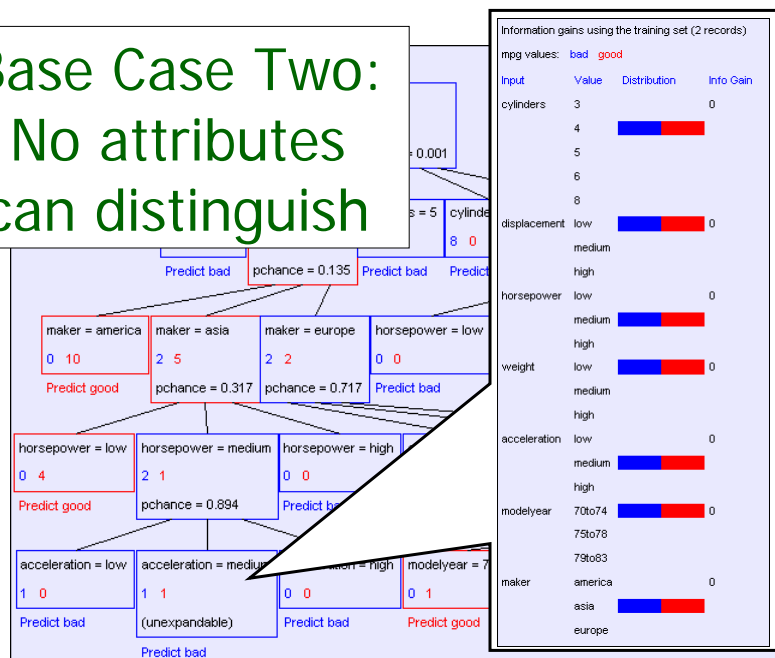
Base Case Two



Copyright © Andrew W. Moore

Slide 39

Base Case Two: No attributes can distinguish



Copyright © Andrew W. Moore

Slide 40

Base Cases

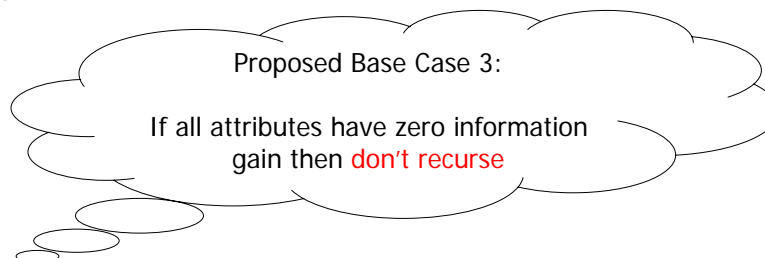
- Base Case One: If all records in current data subset have the same output then **don't recurse**
- Base Case Two: If all records have exactly the same set of input attributes then **don't recurse**

Copyright © Andrew W. Moore

Slide 41

Base Cases: An idea

- Base Case One: If all records in current data subset have the same output then **don't recurse**
- Base Case Two: If all records have exactly the same set of input attributes then **don't recurse**



• *Is this a good idea?*

Copyright © Andrew W. Moore

Slide 42





The problem with Base Case 3

a	b	y
0	0	0
0	1	1
1	0	1
1	1	0

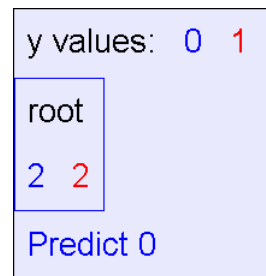
$$y = a \text{ XOR } b$$

The information gains:

Information gains using the training set (4 records)
y values: 0 1

Input	Value	Distribution	Info Gain
a	0		0
	1		
b	0		0
	1		

The resulting decision tree:



Copyright © Andrew W. Moore

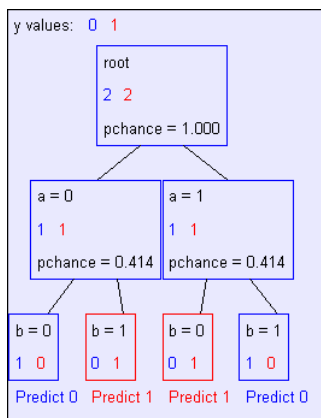
Slide 43

If we omit Base Case 3:

a	b	y
0	0	0
0	1	1
1	0	1
1	1	0

$$y = a \text{ XOR } b$$

The resulting decision tree:



Copyright © Andrew W. Moore

Slide 44

Basic Decision Tree Building Summarized

BuildTree(DataSet, Output)

- If all output values are the same in *DataSet*, return a leaf node that says “predict this unique output”
- If all input values are the same, return a leaf node that says “predict the majority output”
- Else find attribute X with highest Info Gain
- Suppose X has n_X distinct values (i.e. X has arity n_X).
 - Create and return a non-leaf node with n_X children.
 - The i th child should be built by calling

BuildTree(DS _{i} , Output)

Where $DS_{i,built}$ consists of all those records in *DataSet* for which $X = i$ th distinct value of X .

Copyright © Andrew W. Moore

Slide 45

Outline

Machine Learning Datasets

What is Classification?

Contingency Tables

OLAP (Online Analytical Processing)

What is Data Mining?

Searching for High Information Gain

Learning an unpruned decision tree recursively



Training Set Error

Test Set Error

Overfitting

Avoiding Overfitting

Information Gain of a real valued input

Building Decision Trees with real Valued Inputs

Andrew's homebrewed hack: Binary Categorical Splits

Example Decision Trees

Copyright © Andrew W. Moore

Slide 46

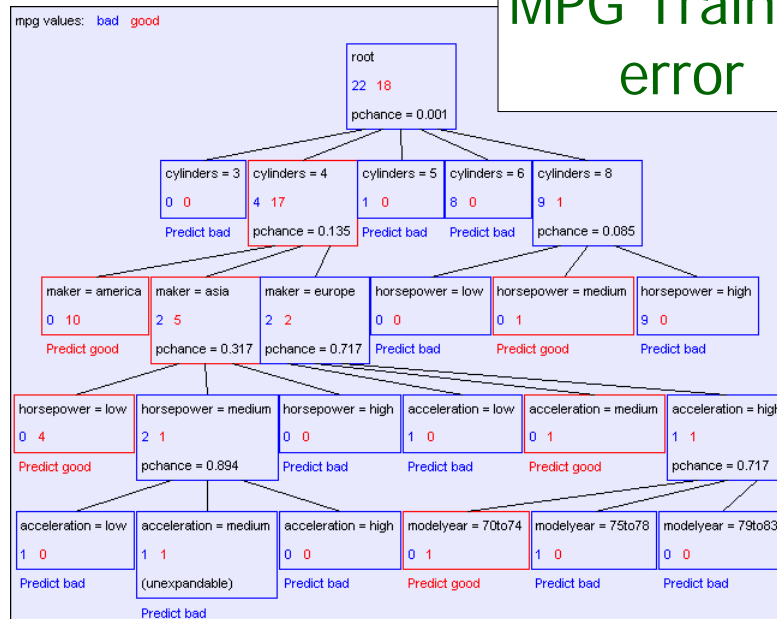
Training Set Error

- For each record, follow the decision tree to see what it would predict
For what number of records does the decision tree's prediction disagree with the true value in the database?
- This quantity is called the *training set error*. The smaller the better.

Copyright © Andrew W. Moore

Slide 47

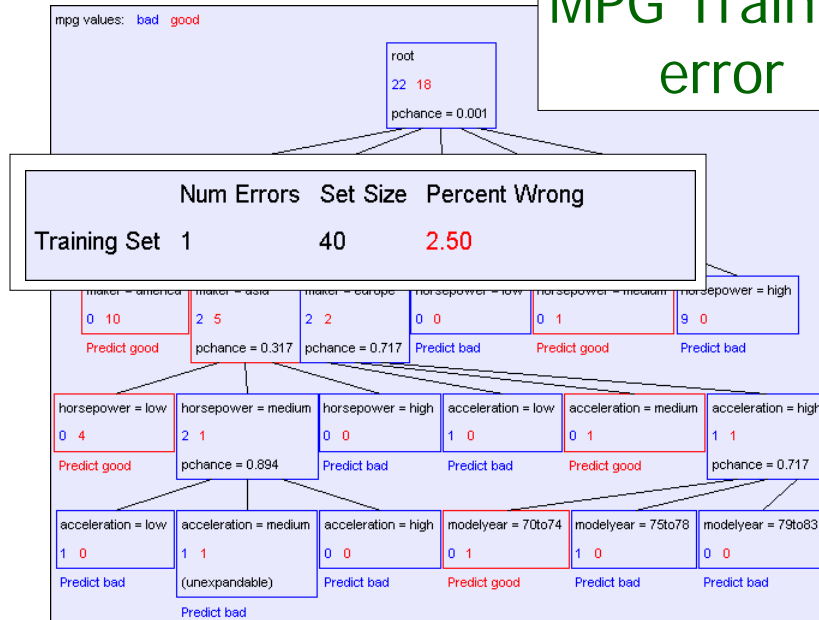
MPG Training error



Copyright © Andrew W. Moore

Slide 48

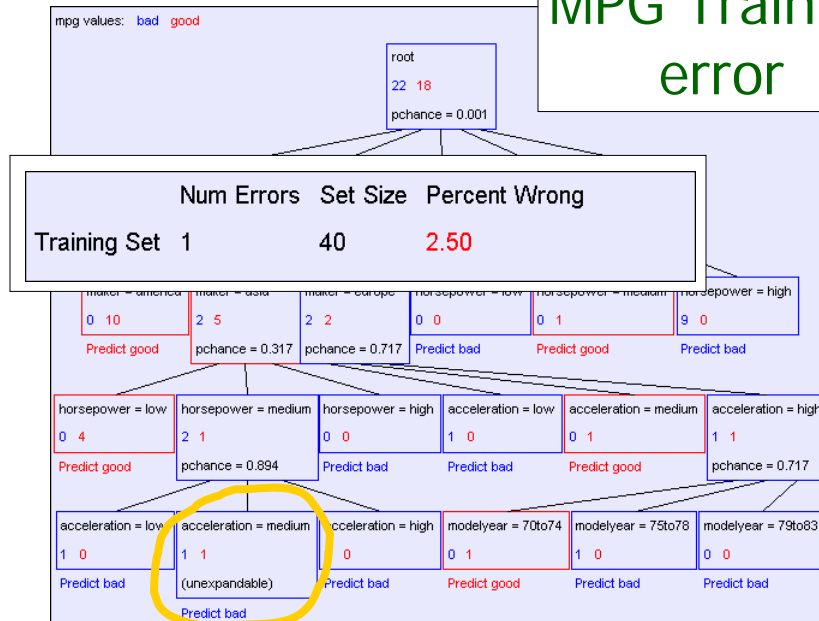
MPG Training error



Copyright © Andrew W. Moore

Slide 49

MPG Training error



Copyright © Andrew W. Moore

Slide 50

Stop and reflect: Why are we doing this learning anyway?

- It is not usually in order to predict the training data's output on data we have already seen.

Copyright © Andrew W. Moore

Slide 51

Stop and reflect: Why are we doing this learning anyway?

- It is not usually in order to predict the training data's output on data we have already seen.
- It is more commonly in order to predict the output value for **future data** we have not yet seen.

Copyright © Andrew W. Moore

Slide 52

Stop and reflect: Why are we doing this learning anyway?

- It is not usually in order to predict the training data's output on data we have already seen.
- It is more commonly in order to predict the output value for **future data** we have not yet seen.

Warning: A common data mining misperception is that the above two bullets are the only possible reasons for learning. There are at least a dozen others.

Copyright © Andrew W. Moore

Slide 53

Outline

Machine Learning Datasets
What is Classification?
Contingency Tables
OLAP (Online Analytical Processing)
What is Data Mining?
Searching for High Information Gain
Learning an unpruned decision tree recursively
Training Set Error
▶ Test Set Error
Overfitting
Avoiding Overfitting
Information Gain of a real valued input
Building Decision Trees with real Valued Inputs
Andrew's homebrewed hack: Binary Categorical Splits
Example Decision Trees

Copyright © Andrew W. Moore

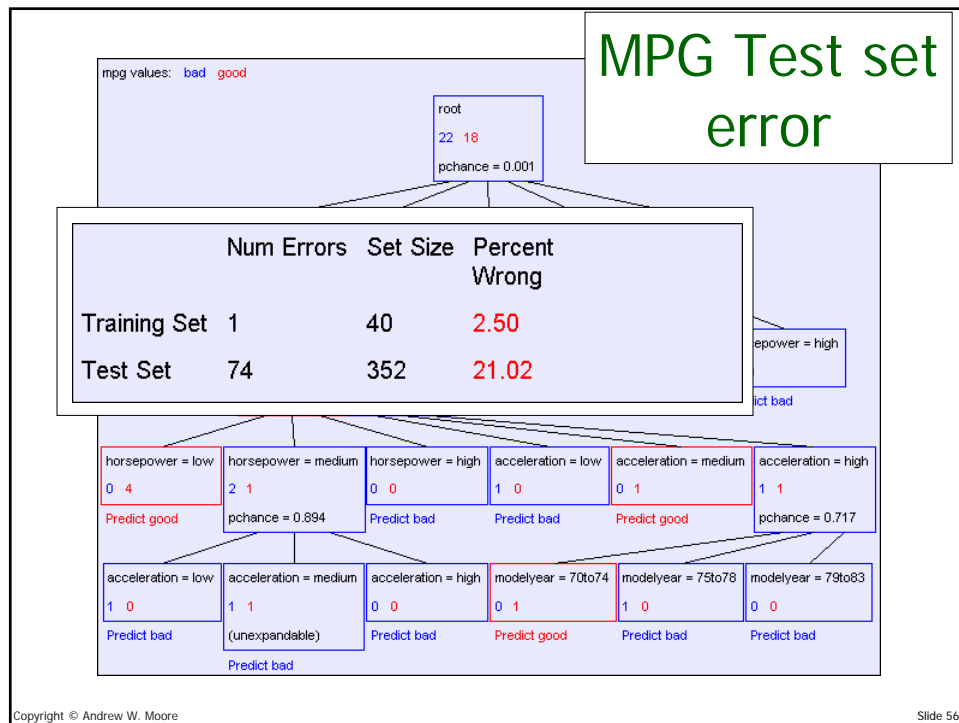
Slide 54

Test Set Error

- Suppose we are forward thinking.
- We hide some data away when we learn the decision tree.
- But once learned, we see how well the tree predicts that data.
- This is a good simulation of what happens when we try to predict future data.
- And it is called **Test Set Error**.

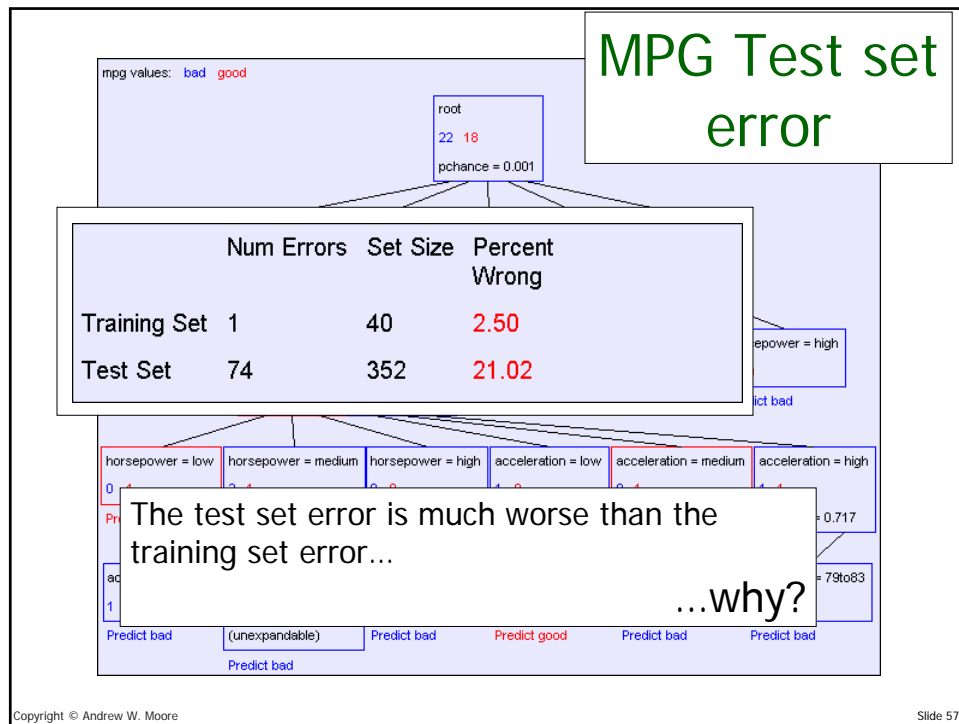
Copyright © Andrew W. Moore

Slide 55



Copyright © Andrew W. Moore

Slide 56



Outline

- Machine Learning Datasets
- What is Classification?
- Contingency Tables
- OLAP (Online Analytical Processing)
- What is Data Mining?
- Searching for High Information Gain
- Learning an unpruned decision tree recursively
- Training Set Error
- Test Set Error
- ▶ Overfitting
- Avoiding Overfitting
- Information Gain of a real valued input
- Building Decision Trees with real Valued Inputs
- Andrew's homebrewed hack: Binary Categorical Splits
- Example Decision Trees

Copyright © Andrew W. Moore

Slide 58

An artificial example

- We'll create a training dataset

Five inputs, all bits, are generated in all 32 possible combinations

Output y = copy of e ,
Except a random 25% of the records have y set to the opposite of e

32 records

a	b	c	d	e	y
0	0	0	0	0	0
0	0	0	0	1	0
0	0	0	1	0	0
0	0	0	1	1	1
0	0	1	0	0	1
:	:	:	:	:	:
1	1	1	1	1	1

Copyright © Andrew W. Moore

Slide 59

In our artificial example

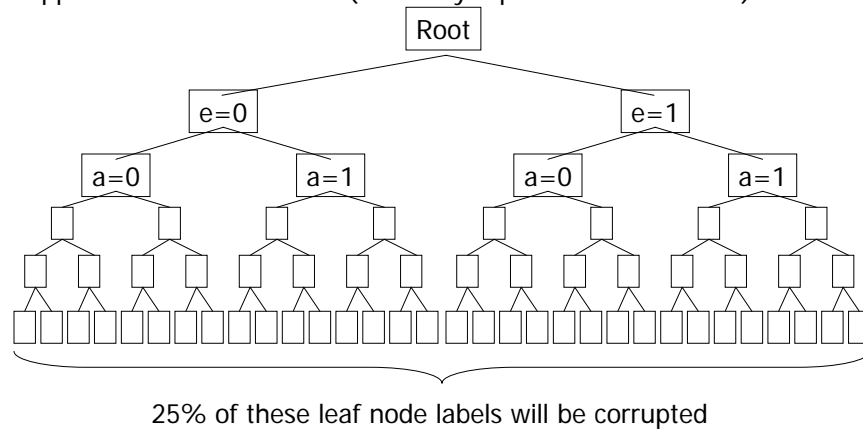
- Suppose someone generates a test set according to the same method.
- The test set is identical, except that some of the y 's will be different.
- Some y 's that were corrupted in the training set will be uncorrupted in the testing set.
- Some y 's that were uncorrupted in the training set will be corrupted in the test set.

Copyright © Andrew W. Moore

Slide 60

Building a tree with the artificial training set

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



Copyright © Andrew W. Moore

Slide 61

Training set error for our artificial tree

All the leaf nodes contain exactly one record and so...

- We would have a training set error of zero

Copyright © Andrew W. Moore

Slide 62

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

Copyright © Andrew W. Moore

Slide 63

What's this example shown us?

- This explains the discrepancy between training and test set error
- But more importantly... ..it indicates there's something we should do about it if we want to predict well on future data.

Copyright © Andrew W. Moore

Slide 64

Suppose we had less data

- Let's not look at the irrelevant bits

These bits are hidden

Output y = copy of e , except a random 25% of the records have y set to the opposite of e

32 records

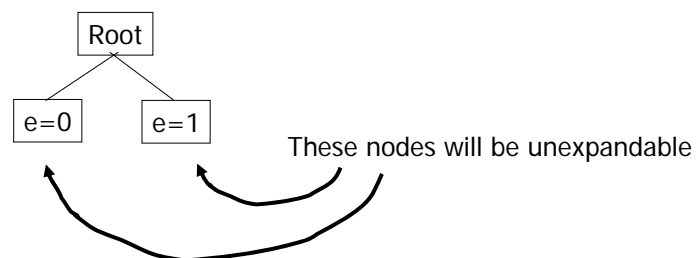
a	b	c	d	e	y
0	0	0	0	0	0
0	0	0	0	1	0
0	0	0	1	0	0
0	0	0	1	1	1
0	0	1	0	0	1
:	:	:	:	:	:
1	1	1	1	1	1

What decision tree would we learn now?

Copyright © Andrew W. Moore

Slide 65

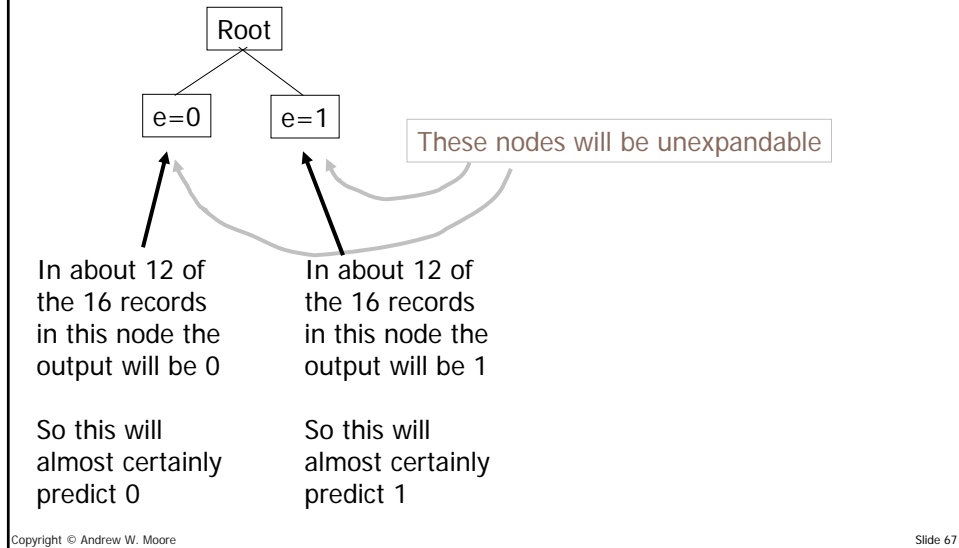
Without access to the irrelevant bits...



Copyright © Andrew W. Moore

Slide 66

Without access to the irrelevant bits...



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

Copyright © Andrew W. Moore

Slide 68

Overfitting

- Definition: If your machine learning algorithm fits noise (i.e. pays attention to parts of the data that are irrelevant) it is **overfitting**.
- Fact (theoretical and empirical): If your machine learning algorithm is overfitting then it may perform less well on test set data.

Copyright © Andrew W. Moore

Slide 69

Outline

Machine Learning Datasets
What is Classification?
Contingency Tables
OLAP (Online Analytical Processing)
What is Data Mining?
Searching for High Information Gain
Learning an unpruned decision tree recursively
Training Set Error
Test Set Error
Overfitting
▶ Avoiding Overfitting
Information Gain of a real valued input
Building Decision Trees with real Valued Inputs
Andrew's homebrewed hack: Binary Categorical Splits
Example Decision Trees

Copyright © Andrew W. Moore

Slide 70

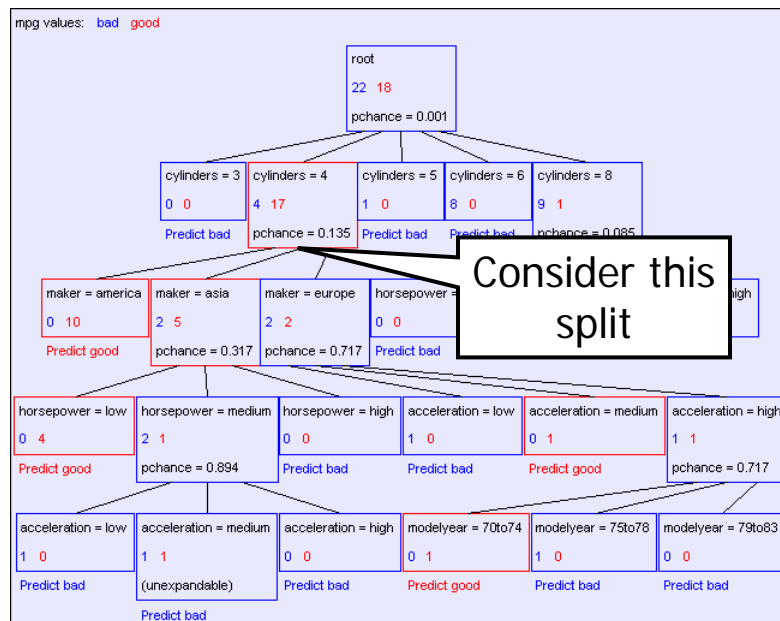
Avoiding overfitting

- Usually we do not know in advance which are the irrelevant variables
- ...and it may depend on the context
 - For example, if $y = a \text{ AND } b$ then b is an irrelevant variable only in the portion of the tree in which $a=0$

But we can use simple statistics to warn us that we might be overfitting.

Copyright © Andrew W. Moore




Slide 71



Copyright © Andrew W. Moore

Slide 72

A chi-squared test




mpg values: bad good			
maker	america	0 10	 $H(\text{mpg} \mid \text{maker} = \text{america}) = 0$
	asia	2 5	 $H(\text{mpg} \mid \text{maker} = \text{asia}) = 0.863121$
	europa	2 2	 $H(\text{mpg} \mid \text{maker} = \text{europa}) = 1$
$H(\text{mpg}) = 0.702467$ $H(\text{mpg} \mid \text{maker}) = 0.478183$			
$IG(\text{mpg} \mid \text{maker}) = 0.224284$			

- 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?

Copyright © Andrew W. Moore

Slide 73

A chi-squared test

mpg values: bad good			
maker	america	0 10	 $H(\text{mpg} \mid \text{maker} = \text{america}) = 0$
	asia	2 5	 $H(\text{mpg} \mid \text{maker} = \text{asia}) = 0.863121$
	europa	2 2	 $H(\text{mpg} \mid \text{maker} = \text{europa}) = 1$
$H(\text{mpg}) = 0.702467$ $H(\text{mpg} \mid \text{maker}) = 0.478183$			
$IG(\text{mpg} \mid \text{maker}) = 0.224284$			

- 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%.

Copyright © Andrew W. Moore

Slide 74

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 your way up until there are no more prunable nodes.

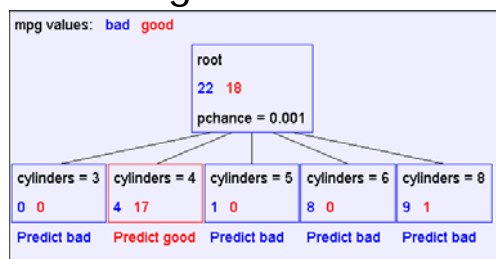
MaxPchance is a magic parameter you must specify to the decision tree, indicating your willingness to risk fitting noise.

Copyright © Andrew W. Moore

Slide 75

Pruning example

- With $MaxPchance = 0.1$, you will see the following MPG decision tree:



Note the improved test set accuracy compared with the unpruned tree

	Num Errors	Set Size	Percent Wrong
Training Set	5	40	12.50
Test Set	56	352	15.91

Copyright © Andrew W. Moore

Slide 76

MaxPchance

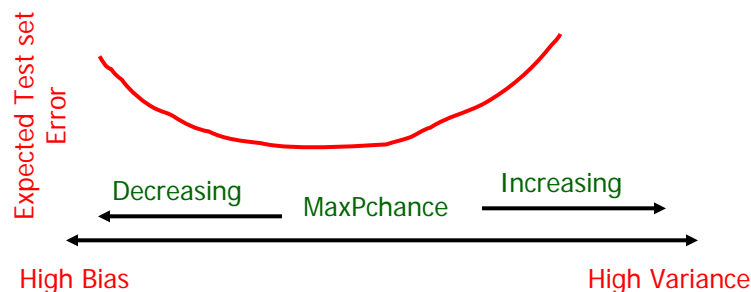
- **Good news:** The decision tree can automatically adjust its pruning decisions according to the amount of apparent noise and data.
- **Bad news:** The user must come up with a good value of MaxPchance. (Note, Andrew usually uses 0.05, which is his favorite value for any magic parameter).
- **Good news:** But with extra work, the best MaxPchance value can be estimated automatically by a technique called cross-validation.

Copyright © Andrew W. Moore

Slide 77

MaxPchance

- Technical note (dealt with in other lectures): MaxPchance is a regularization parameter.



Copyright © Andrew W. Moore

Slide 78

The simplest tree

- Note that this pruning is heuristically trying to find

The simplest tree structure for which all within-leaf-node disagreements can be explained by chance

- This is not the same as saying “the simplest classification scheme for which...”
- Decision trees are biased to prefer classifiers that can be expressed as trees.

Copyright © Andrew W. Moore

Slide 79

Expressiveness of Decision Trees

- Assume all inputs are Boolean and all outputs are Boolean.
- What is the class of Boolean functions that are possible to represent by decision trees?
- Answer: All Boolean functions.

Simple proof:

1. Take any Boolean function
2. Convert it into a truth table
3. Construct a decision tree in which each row of the truth table corresponds to one path through the decision tree.

Copyright © Andrew W. Moore

Slide 80

Outline

Machine Learning Datasets
 What is Classification?
 Contingency Tables
 OLAP (Online Analytical Processing)
 What is Data Mining?
 Searching for High Information Gain
 Learning an unpruned decision tree recursively
 Training Set Error
 Test Set Error
 Overfitting
 Avoiding Overfitting
 ► Information Gain of a real valued input
 Building Decision Trees with real Valued Inputs
 Andrew's homebrewed hack: Binary Categorical Splits
 Example Decision Trees

Copyright © Andrew W. Moore

Slide 81

Real-Valued inputs

- What should we do if some of the inputs are real-valued?

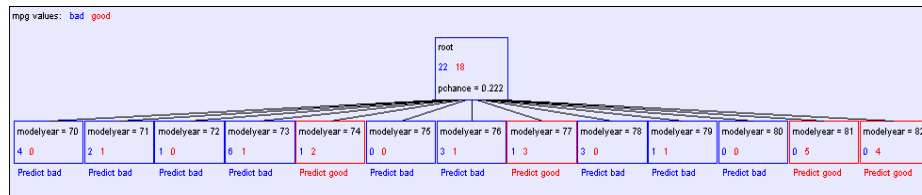
mpg	cylinders	displacemen	horsepower	weight	acceleration	modelyear	maker
good	4	97	75	2265	18.2	77	asia
bad	6	199	90	2648	15	70	america
bad	4	121	110	2600	12.8	77	europa
bad	8	350	175	4100	13	73	america
bad	6	198	95	3102	16.5	74	america
bad	4	108	94	2379	16.5	73	asia
bad	4	113	95	2228	14	71	asia
bad	8	302	139	3570	12.8	78	america
:	:	:	:	:	:	:	:
:	:	:	:	:	:	:	:
:	:	:	:	:	:	:	:
good	4	120	79	2625	18.6	82	america
bad	8	455	225	4425	10	70	america
good	4	107	86	2464	15.5	76	europa
bad	5	131	103	2830	15.9	78	europa

Idea One: Branch on each possible real value

Copyright © Andrew W. Moore

Slide 82

“One branch for each numeric value” idea:



Hopeless: with such high branching factor will shatter the dataset and over fit

Note pchance is 0.222 in the above...if MaxPchance was 0.05 that would end up pruning away to a single root node.

Copyright © Andrew W. Moore

Slide 83

A better idea: thresholded splits

- Suppose X is real valued.
- Define $IG(Y/X:t)$ as $H(Y) - H(Y/X:t)$
- Define $H(Y/X:t) = H(Y/X < t) P(X < t) + H(Y/X \geq t) P(X \geq t)$
 - $IG(Y/X:t)$ is the information gain for predicting Y if all you know is whether X is greater than or less than t
- Then define $IG^*(Y/X) = \max_t IG(Y/X:t)$
- For each real-valued attribute, use $IG^*(Y/X)$ for assessing its suitability as a split

Copyright © Andrew W. Moore

Slide 84

Computational Issues

- You can compute $IG^*(Y|X)$ in time

$$R \log R + 2 R n_y$$

- Where

R is the number of records in the node under consideration

n_y is the arity (number of distinct values of) Y

How?

Sort records according to increasing values of X . Then create a $2 \times n_y$ contingency table corresponding to computation of $IG(Y|X:x_{\min})$. Then iterate through the records, testing for each threshold between adjacent values of X , incrementally updating the contingency table as you go. For a minor additional speedup, only test between values of Y that differ.

Copyright © Andrew W. Moore

Slide 85

Information gains using the training set (40 records)

mpg values: bad good

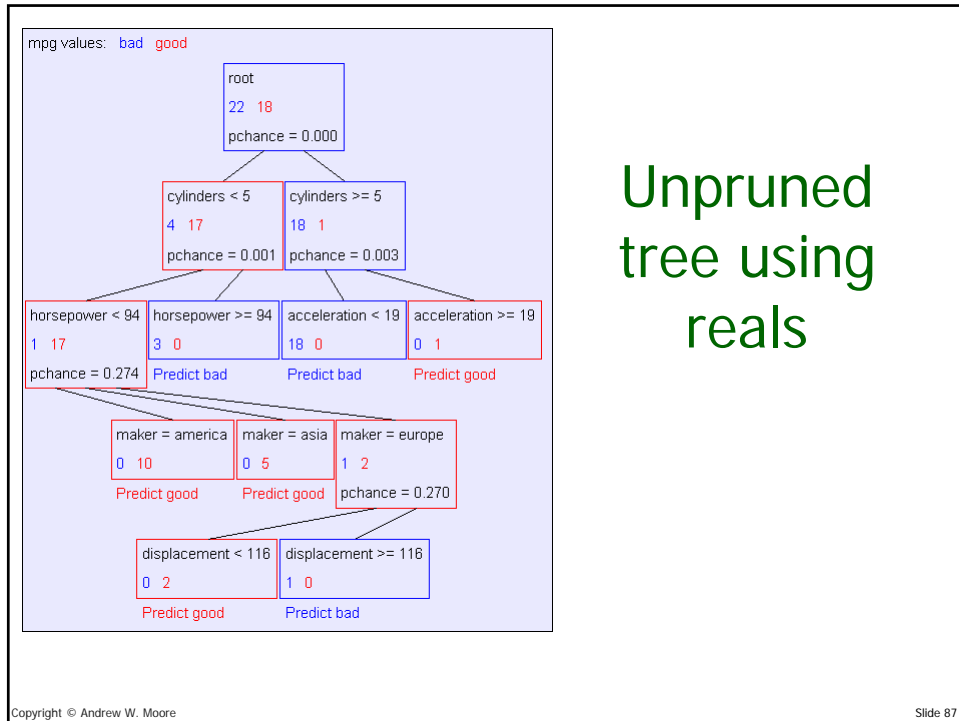
Input	Value	Distribution	Info Gain
cylinders	< 5		0.48268
	>= 5		
displacement	< 198		0.428205
	>= 198		
horsepower	< 94		0.48268
	>= 94		
weight	< 2789		0.379471
	>= 2789		
acceleration	< 18.2		0.159982
	>= 18.2		
modelyear	< 81		0.319193
	>= 81		
maker	america		0.0437265
	asia		
	europa		

Example with MPG

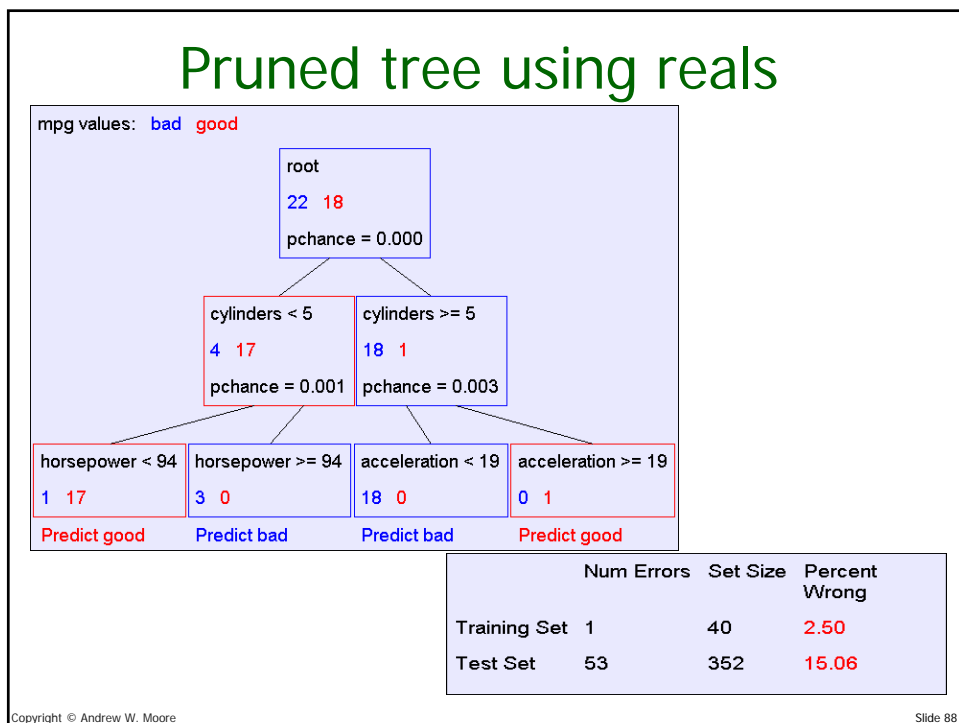
Copyright © Andrew W. Moore

Slide 86

Unpruned tree using reals



Pruned tree using reals



LearnUnprunedTree(X,Y)

Input: X a matrix of R rows and M columns where X_{ij} = the value of the j'th attribute in the i'th input datapoint. Each column consists of either all real values or all categorical values.

Input: Y a vector of R elements, where Y_i = the output class of the i'th datapoint. The Y_i values are categorical.

Output: An Unpruned decision tree

If all records in X have identical values in all their attributes (this includes the case where $R < 2$), return a Leaf Node predicting the majority output, breaking ties randomly. This case also includes

If all values in Y are the same, return a Leaf Node predicting this value as the output

Else

For j = 1 .. M

If j'th attribute is categorical

$IG_j = IG(Y|X_j)$

Else (j'th attribute is real-valued)

$IG_j = IG^*(Y|X_j)$ from about four slides back

Let $j^* = \text{argmax}_j IG_j$ (this is the splitting attribute we'll use)

If j^* is categorical then

For each value v of the j'th attribute

Let X^v = subset of rows of X in which $X_{ij} = v$. Let Y^v = corresponding subset of Y

Let $\text{Child}^v = \text{LearnUnprunedTree}(X^v, Y^v)$

Return a decision tree node, splitting on j'th attribute. The number of children equals the number of values of the j'th attribute, and the v'th child is Child^v

Else j^* is real-valued and let t be the best split threshold

Let X^{LO} = subset of rows of X in which $X_{ij} \leq t$. Let Y^{LO} = corresponding subset of Y

Let $\text{Child}^{LO} = \text{LearnUnprunedTree}(X^{LO}, Y^{LO})$

Let X^{HI} = subset of rows of X in which $X_{ij} > t$. Let Y^{HI} = corresponding subset of Y

Let $\text{Child}^{HI} = \text{LearnUnprunedTree}(X^{HI}, Y^{HI})$

Return a decision tree node, splitting on j'th attribute. It has two children corresponding to whether the j'th attribute is above or below the given threshold.

Copyright © Andrew W. Moore

Slide 89

LearnUnprunedTree(X,Y)

Input: X a matrix of R rows and M columns where X_{ij} = the value of the j'th attribute in the i'th input datapoint. Each column consists of either all real values or all categorical values.

Input: Y a vector of R elements, where Y_i = the output class of the i'th datapoint. The Y_i values are categorical.

Output: An Unpruned decision tree

If all records in X have identical values in all their attributes (this includes the case where $R < 2$), return a Leaf Node predicting the majority output, breaking ties randomly. This case also includes

If all values in Y are the same, return a Leaf Node predicting this value as the output

Else

For j = 1 .. M

If j'th attribute is categorical

$IG_j =$

Else (j'th attribute is real-valued)

$IG_j =$

Let $j^* = \text{argmax}_j IG_j$

If j^* is categorical then

For each value v of the j'th attribute

Let X^v = subset of rows of X in which $X_{ij} = v$. Let Y^v = corresponding subset of Y

Let $\text{Child}^v = \text{LearnUnprunedTree}(X^v, Y^v)$

Return a decision tree node, splitting on j'th attribute. The number of children equals the number of values of the j'th attribute, and the v'th child is Child^v

Else j^* is real-valued and let t be the best split threshold

Let $X^{LO} =$

Let $\text{Child}^{LO} =$

Let $X^{HI} =$

Let $\text{Child}^{HI} =$

Return a decision tree node, splitting on j'th attribute. It has two children corresponding to whether the j'th attribute is above or below the given threshold.

Things to note:

Below the root node, there is no point testing categorical attributes that have already been split upon further up the tree. This is because all the values of that attribute will be the same and IG must therefore be zero.

But it's worth retesting real-valued attributes, since they may have different values below the binary split, and may benefit from splitting further.

To achieve the above optimization, you should pass down through the recursion a current active set of attributes.

Pedantic detail: a third termination condition should occur if the best split attribute puts all its records in exactly one child (note that this means it and all other attributes have $IG=0$).

Copyright © Andrew W. Moore

Slide 90

Outline

Machine Learning Datasets
What is Classification?
Contingency Tables
OLAP (Online Analytical Processing)
What is Data Mining?
Searching for High Information Gain
Learning an unpruned decision tree recursively
Training Set Error
Test Set Error
Overfitting
Avoiding Overfitting
Information Gain of a real valued input
Building Decision Trees with real Valued Inputs
▶ Andrew's homebrewed hack: Binary Categorical Splits
Example Decision Trees

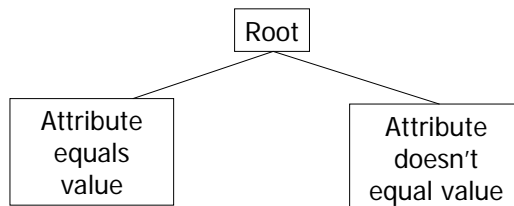
Copyright © Andrew W. Moore

Slide 91

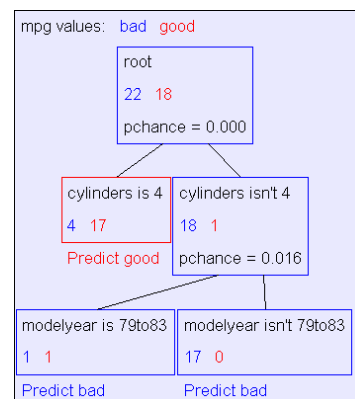
Warning: unlike what went before, this is an editorial "trick of the trade": not part of the official Decision Tree algorithm.

Binary categorical splits

- One of Andrew's favorite tricks
- Allow splits of the following form



Example:



Copyright © Andrew W. Moore

Slide 92

Outline

Machine Learning Datasets

What is Classification?

Contingency Tables

OLAP (Online Analytical Processing)

What is Data Mining?

Searching for High Information Gain

Learning an unpruned decision tree recursively

Training Set Error

Test Set Error

Overfitting

Avoiding Overfitting

Information Gain of a real valued input

Building Decision Trees with real valued Inputs

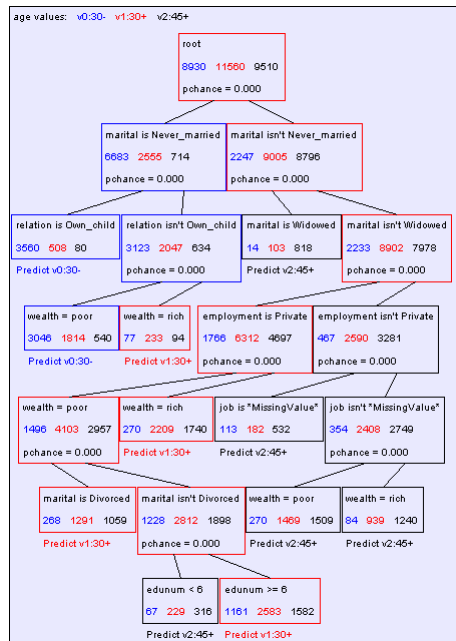
Andrew's homebrewed hack: Binary Categorical Splits



Example Decision Trees

Copyright © Andrew W. Moore

Slide 93

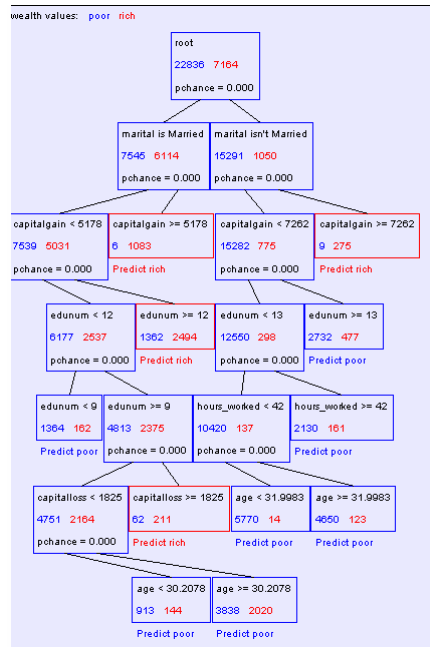


Copyright © Andrew W. Moore

Slide 94

Predicting age
from census

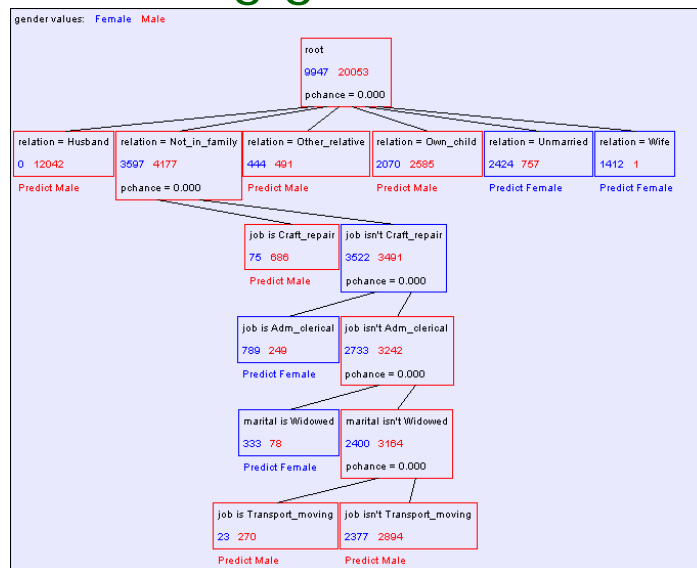
Predicting wealth from census



Copyright © Andrew W. Moore

Slide 95

Predicting gender from census



Copyright © Andrew W. Moore

Slide 96

Conclusions

- Decision trees are the single most popular data mining tool
 - Easy to understand
 - Easy to implement
 - Easy to use
 - Computationally cheap
- It's possible to get in trouble with overfitting
- They do classification: predict a categorical output from categorical and/or real inputs

Copyright © Andrew W. Moore

Slide 97

What you should know

- What's a contingency table?
- What's information gain, and why we use it
- The recursive algorithm for building an unpruned decision tree
- What are training and test set errors
- Why test set errors can be bigger than training set
- Why pruning can reduce test set error
- How to exploit real-valued inputs

Copyright © Andrew W. Moore

Slide 98

What we haven't discussed

- It's easy to have real-valued outputs too---these are called Regression Trees*
- Bayesian Decision Trees can take a different approach to preventing overfitting
- Computational complexity (straightforward and cheap) *
- Alternatives to Information Gain for splitting nodes
- How to choose MaxPchance automatically *
- The details of Chi-Squared testing *
- Boosting---a simple way to improve accuracy *

* = discussed in other Andrew lectures

Copyright © Andrew W. Moore

Slide 99

For more information

- Two nice books
 - L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. Classification and Regression Trees. Wadsworth, Belmont, CA, 1984.
 - C4.5 : Programs for Machine Learning (Morgan Kaufmann Series in Machine Learning) by J. Ross Quinlan
- Dozens of nice papers, including
 - Learning Classification Trees, Wray Buntine, Statistics and Computation (1992), Vol 2, pages 63-73
 - Kearns and Mansour, On the Boosting Ability of Top-Down Decision Tree Learning Algorithms, STOC: ACM Symposium on Theory of Computing, 1996"
- Dozens of software implementations available on the web for free and commercially for prices ranging between \$50 - \$300,000

Copyright © Andrew W. Moore

Slide 100

Discussion

- Instead of using information gain, why not choose the splitting attribute to be the one with the highest prediction accuracy?
- Instead of greedily, heuristically, building the tree, why not do a combinatorial search for the optimal tree?
- If you build a decision tree to predict wealth, and marital status, age and gender are chosen as attributes near the top of the tree, is it reasonable to conclude that those three inputs are the major causes of wealth?
- ..would it be reasonable to assume that attributes not mentioned in the tree are not causes of wealth?
- ..would it be reasonable to assume that attributes not mentioned in the tree are not correlated with wealth?