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...

Summary of Basic Decision Tree Building

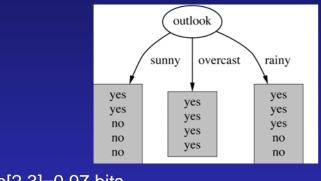
From www.cs.cmu.edu/~awm/tutorials

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, Output)

Where DS_i built consists of all those records in DataSet for which X = th distinct value of X.

Information gain (IG) of an attribute (outlook)



H(playtennis/outlook=sunny)

=Info[2,3]

=entropy(2/5, 3/5)

= -2/5 log 2/5 - 3/5 log 3/5

= 0.97 bits

H(playtennis/outlook)=

Average info of subtree(weighted)

 $= 0.97 \times 5/14 + 0 \times 4/14 + 0.97 \times 5/14$

= 0.693 bits

Info[2,3]=0.97 bits
Info[4,0]=0 bits

H(playtennis) = Info of all training samples= Info[9,5] = 0.94

IG(playtennis/outlook) = 0.94 - 0.693 = 0.247 bits

Decision Trees and Data mining

- Divide and conquer approach to decision tree induction
- Also called "top-down induction of decision trees"
- Information gain algorithm known as ID3
- Led to widely used system known as C4.5

US census Data Set

(from Kohavi/Moore) 48,000 records, 16 fields

age	emplovme	education	edun	marital		job	relation	race	gender	hour	country	wealth
-30			-			,			3		,	11 00
39	State_gov	Bachelors	13	Never_mar		Adm_cleric	Not_in_fan	White	Male	40	United_S	ta poor
	Self_emp_					Exec_man		White	Male	13	United_S	ta poor
39	Private	HS_grad	9	Divorced		Handlers_c	Not_in_fan	White	Male		United_S	
54	Private	11th	7	Married		Handlers_c	Husband	Black	Male	40	United_S	ta poor
28	Private	Bachelors	13	Married		Prof_speci	Wife	Black	Female	40	Cuba	poor
38	Private	Masters	14	Married		Exec_man	Wife	White	Female	40	United_S	ta 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_S	tarich
31	Private	Masters	14	Never_mar		Prof_speci	Not_in_fan	White	Female	50	United_S	tarich
42	Private	Bachelors	13	Married		Exec_man	Husband	White	Male	40	United_S	tarich
37	Private	Some_coll	10	Married		Exec_man	Husband	Black	Male	80	United_S	ta 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_S	ta poor
33	Private	Assoc_acc	12	Never_mar		Sales	Not_in_fan	Black	Male	50	United_S	ta poor
41	Private	Assoc_voc	11	Married		Craft_repai	Husband	Asian	Male	40	*Missing\	/i 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_S	ta poor
33	Private	HS_grad	9	Never_mar		Machine_c	Unmarried	White	Male		United_S	
38	Private	11th		Married		Sales	Husband	White	Male		United_S	-
44	Self_emp_	Masters	14	Divorced		Exec_man	Unmarried	White	Female		United_S	
41	Private	Doctorate	16	Married		Prof_speci	Husband	White	Male	60	United_S	ta rich
:	:	:	:	:	:	:	:	:	:	:	:	:

About this Data set

- Tiny subset of the 1990 US census
- Publicly available from the UCI Machine Learning Repository

```
Used Attributes

age edunum race hours_worked

employment marital gender country

taxweighting job capitalgain wealth

education relation capitalloss agegroup
```

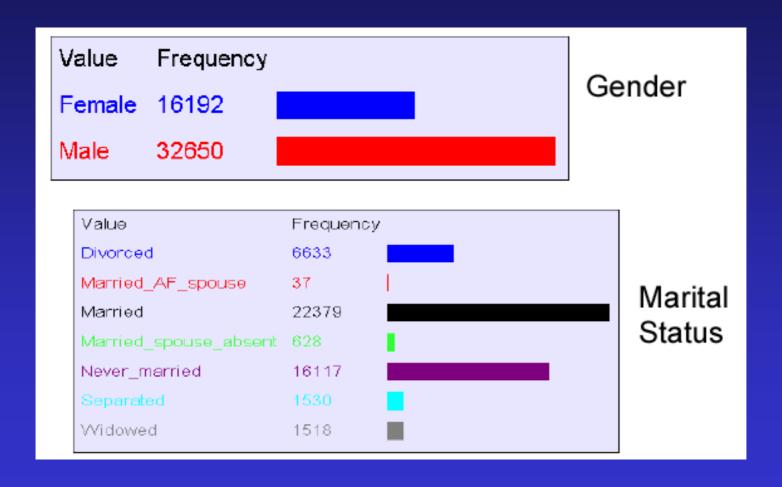
This color = Real-valued This color = Symbol-valued

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

Classification

- A major data mining operation
- Try to predict an attribute, e.g., wealth by means of other 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

What can you do with a data set? Histograms



2-D Contingency Table

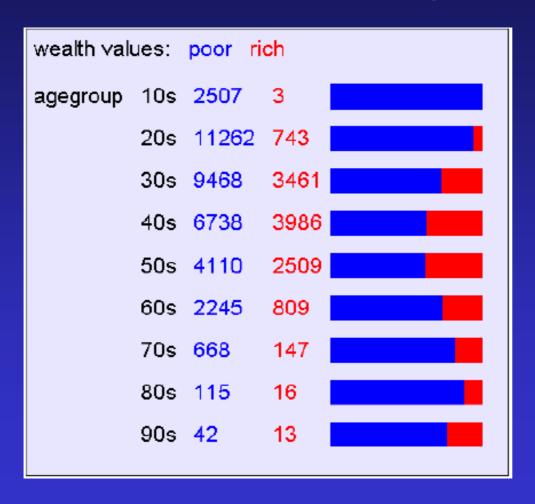
wealth valu	ues:	poor ri	ch
agegroup	10s	2507	3
	20s	11262	743
	30s	9468	3461
	40s	6738	3986
	50s	4110	2509
	60s	2245	809
	70s	668	147
	80s	115	16
	90s	42	13

- For each pair of values for attributes (agegroup, wealth)
- Count how many records match

Graphical Representation of 2-D contingency Table



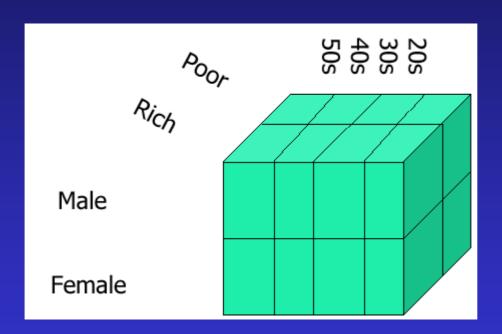
Easier to see more interesting things if we stretch out histogram bars



2-D Contingency Table showing all 16 attributes

job valu	ies:	Adm_derical	Cratt_	repair		Farm	ning_fis	hing	Ma	chine_c	op_insp	oct	Priv_ho	ouse_	serv	Prote	ctive_s	erv	Tech_support	
Missin	gValue	Armed_Forces	Exec_	manage	eria	Hand	ders_c	leane	rs Oth	ner_ser	vice		Prof_s	pecial	ty	Sales			Transport_mo	ving
marital	Divorce	d	270	1192	0	679	890	90	197	434	762		795	121	664	239	254			
	Married	_AF_spouse	5	6	0	4	3	1	1	1	5		4	1	5	0	1			
	Married		928	1495	7	3818	3600	869	724	1469	1088		3182	583	2491	609	1489			
	Married	_spouse_absent	45	84	0	77	52	35	32	37	92		64	7	55	9	30			
	Never_r	narried	1242	2360	8	1301	1260	434	1029	872	2442		1849	237	1992	506	486			
	Separat	ed	97	224	0	160	126	23	63	123	275		145	23	146	48	56			
	Widowe	ed	222	250	0	73	155	38	26	86	259		133	11	151	35	39			

3-D Contingency Table: much harder to look at



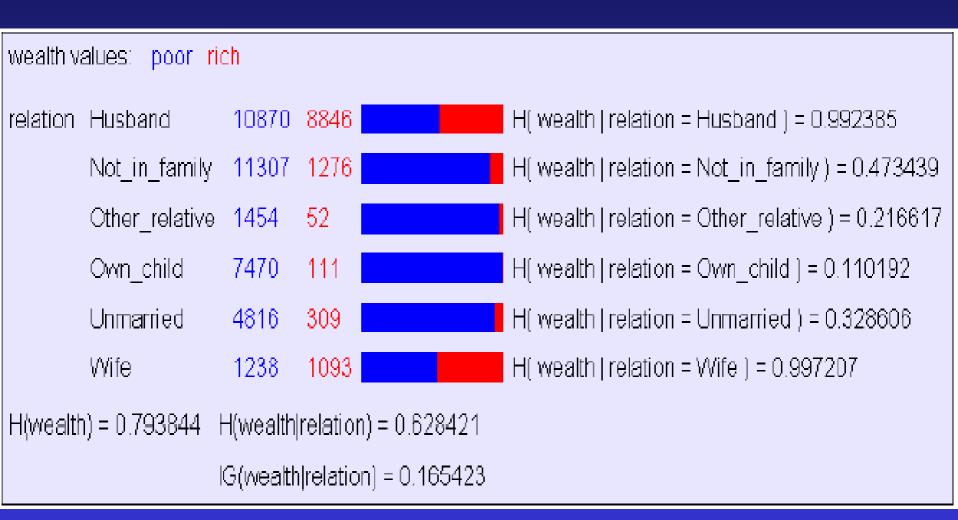
On Line Analytical Processing (OLAP)

- 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
- Too many tables though
 - With 16 attributes there are:
 - 16 1-D contingency tables
 - ${}^{16}C_2 = 120 2-D$ contingency tables
 - 560 3-D contingency tables

Information Gain in Census Data

age	employme	education	edun	marital		job	relation	race	gender	hour	country	wealth
39	State_gov	Bachelors	13	Never_mar		Adm_cleric	Not_in_fan	White	Male	40	United_St	ta poor
51	Self_emp_	Bachelors	13	Married		Exec_man	Husband	White	Male	13	United_St	ta poor
39	Private	HS_grad	9	Divorced		Handlers_c	Not_in_fan	White	Male	40	United_St	te poor
54	Private	11th	7	Married		Handlers_c	Husband	Black	Male	40	United_St	ta poor
28	Private	Bachelors	13	Married		Prof_speci	Wife	Black	Female	40	Cuba	poor
38	Private	Masters	14	Married		Exec_man	Wife	White	Female	40	United_St	ta 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	*Missing\	/i 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	ta poor
33	Private	HS_grad	9	Never_mar		Machine_c	Unmarried	White	Male		United_St	
38	Private	11th	7	Married		Sales	Husband	White	Male	50	United_St	poor
44	Self_emp_	Masters	14	Divorced		Exec_man	Unmarried	White	Female	45	United_St	rich
41	Private	Doctorate	16	Married		Prof_speci	Husband	White	Male	60	United_St	rich
:	:	:	:	:	:	:	:	:	:	:	:	:

Information Gain

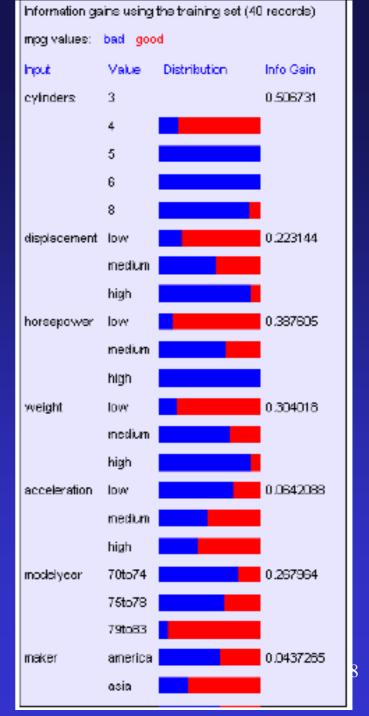


MPG Data Set (40 records from UCI repository)

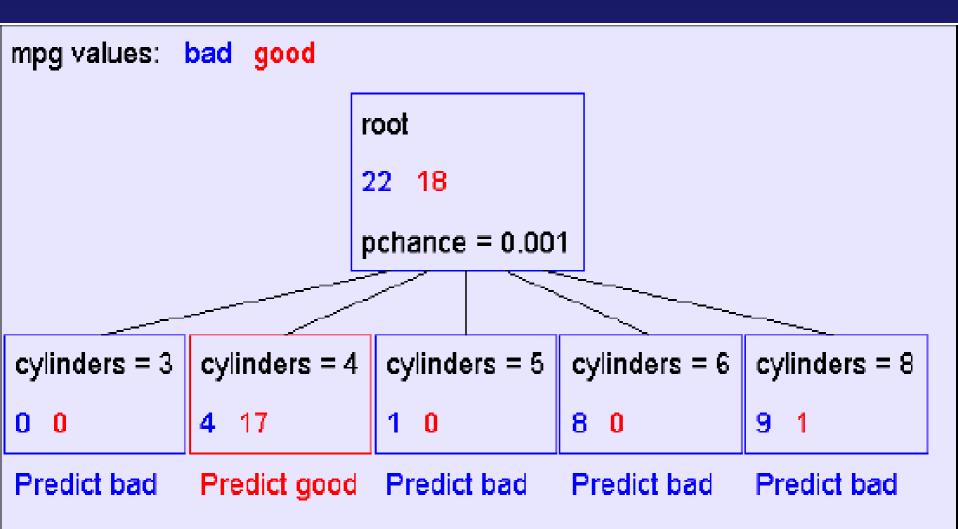
mpg	cylinders	displacement	horsepower	weight	acceleration	modelyear	maker
good	4	low	low	low	high	75to78	asia
bad	6	medium	medium	medium	medium	70to74	america
bad	4	medium	medium	medium	low	75to78	europe
bad	8	high	high	high	low	70to74	america
bad	6	medium	medium	medium	medium	70to74	america
bad	4	low	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	europe

Predicting MPG

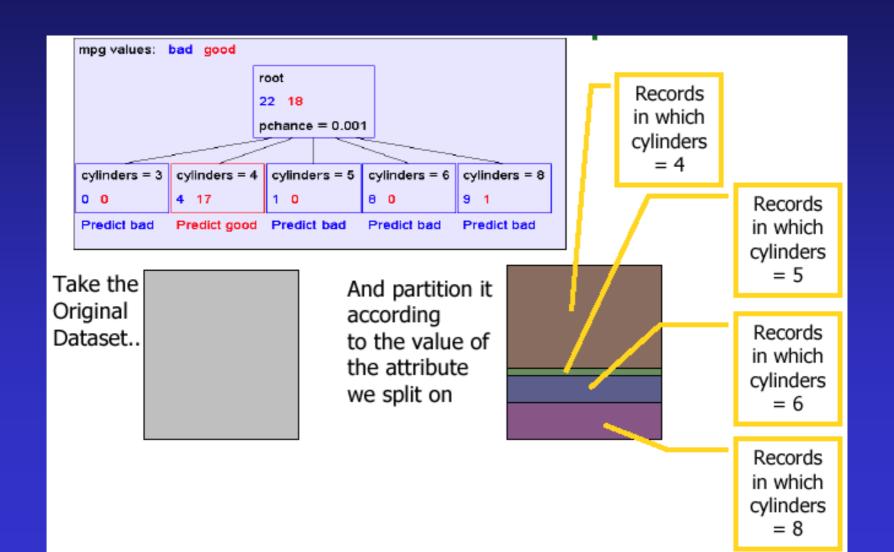
- Has good/bad values
- Look at all the Information Gains
- Cylinders has the highest IG



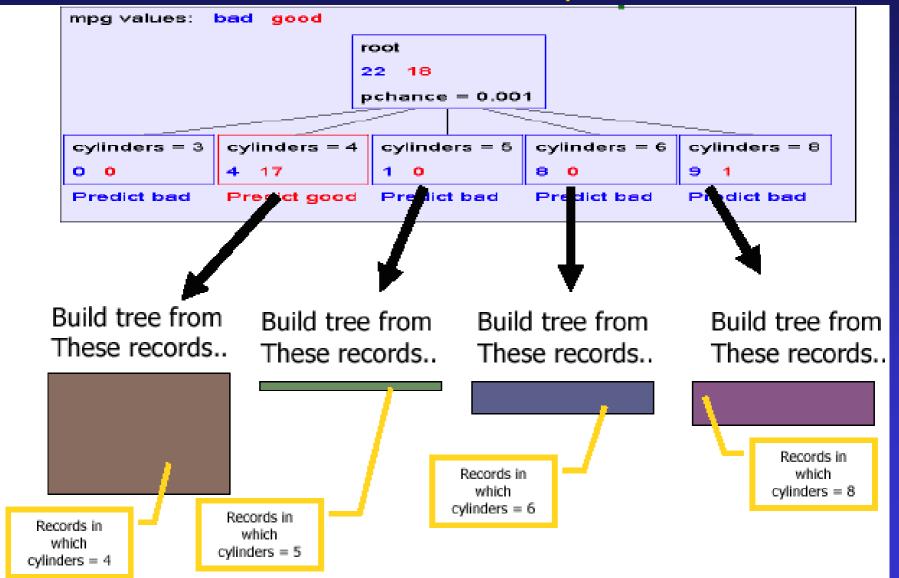
A Decision Stump



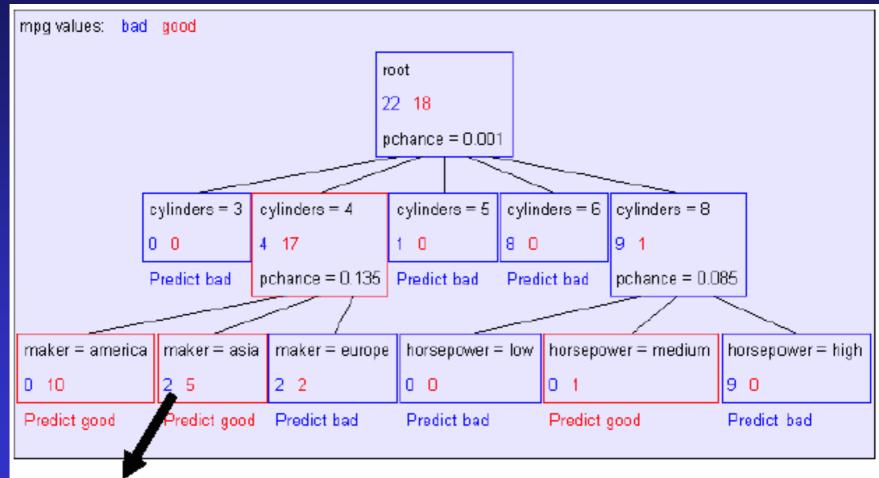
Recursion Step



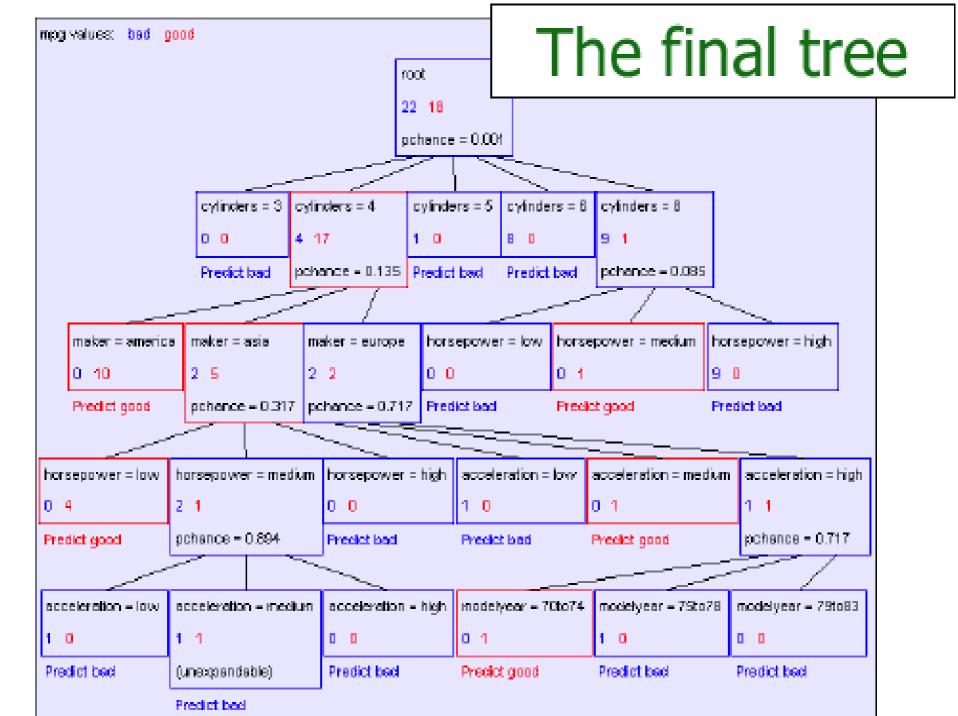
Recursion Step



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)



Incorporating Continuous-Valued Attributes

- Initial definition of ID3 is restricted to attributes that take on discrete values
 - target attribute must be discrete valued
 - attributes tested at decision nodes must be discrete valued
- Example:
 - Convert a continuous-valued test attribute A into a Boolean valued attribute A_c
 - create a Boolean attribute A_c that is true if A < c, false otherwise

Example of Continuous-valued Features

	Feature	Minimum	Maximum
1	Ave Entropy	0.275	1.423216
2	Ave Grey-Threshold	170	226
3	NoBlackPixels	81105	873557
4	Ave NoExtContours	4	44
5	Ave NoIntContours	5	35
6	Ave SlopeHorizontal	0.094	0.5707
7	Ave SlopePositive	0.1	0.542
8	Ave SlopeVertical	0.066	0.546
9	Ave SlopeNegative	0.048	0.36
10	Ave StrokeWidth	4	11
11	Ave Slant	23.04	36.876
12	Ave Height	13	66

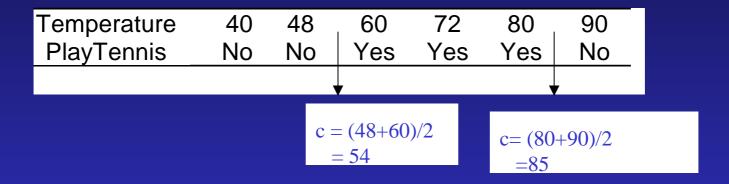
Thresholding a continuous value to obtain a Boolean Attribute

Algorithm

- Sort examples according to continuous attribute A
- Identify adjacent examples that differ in target classification
- Generate thresholds midway between corresponding values of A

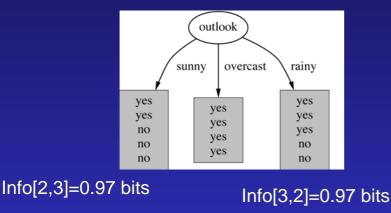
Temperature	40	48	60	72	80	90
PlayTennis	No	No	Yes	Yes	Yes	No
		•			•	_

Thresholding Algorithm



- There are two Boolean valued candidate attributes
 - Temperature_{>54} and Temperature_{>85}
- Which one is better?
 - One that produces greatest information gain

Information gain of an attribute



Info[4,0]=0 bits

```
Info[2,3]=entropy(2/5, 3/5)
= -2/5 log 2/5 - 3/5 log 3/5
= 0.97 bits
```

```
Average info of subtree(weighted)= 0.97x5/14 + 0 \times 4/14 + 0.97 \times 5/14 = 0.693 bits
```

Info of all training samples, info[9,5] = 0.94

$$gain(outlook) = 0.94 - 0.693 = 0.247 bits$$

Algorithm for thresholding a continuous-valued feature

- At a given node
 - For a given continuous-valued feature
 - Obtain candidate thresholds for continuous-valued feature:
 - points where target attribute changes value
 - Determine information gain for each threshold
 - Choose threshold with highest information gain

Data Reduction

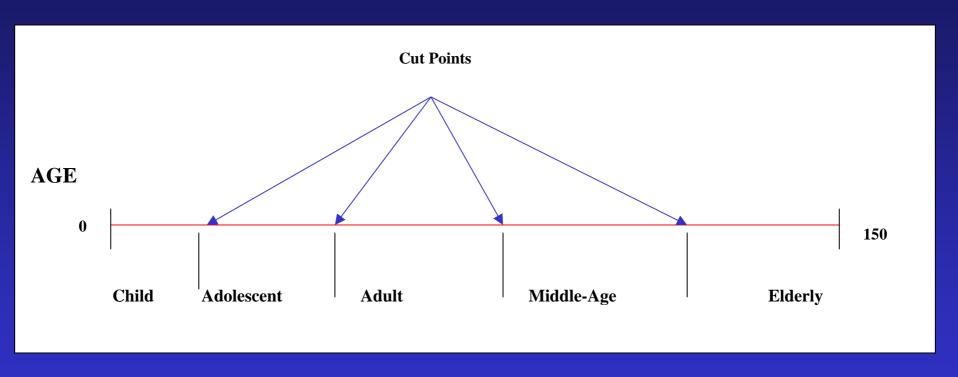
Feature Discretization

- Discretize values of continuous features into a small number of intervals
- Each interval is mapped into a discrete symbol
- Each feature handled independently of other features

Benefits

- Simplified data description
- Easy to understand data
- Easy to understand data mining results
- Applicability of several machine learning algorithms
- Quality of the analysis improved

Discretization of the age feature



Automated Discretization Techniques: not based on target attribute value

- Values are placed in bins
- Partitioning into groups with close values
- All values in a bin will be merged into a single concept represented by a single value – usually the mean or median of the bin's values

Clustering for binning

- Set of values for a given feature= {3,2,1,5,4,3,1,7,5,3}
- Sorted set = {1,1,2,3,3,3,4,5,5,7}
- Split into three bins of approx same size
 - **{1,1,2,** 3,3,3 4,5,5,7**}**
 - Bin1 Bin2 Bin3
- Bin representative selection
 - Modes:
 - **{1,1,1** 3,3,3 5,5,5,5}
 - Mean Values:
 - **{1.33, 1.33, 1.33** 3,3,3 5.25, 5.25, 5.25}
 - Closest Boundary Values:
 - **{1,1,2** 3,3,3 4,4,4,7}

K-means clustering

- Sort all values for a given feature
- Assign approximately equal numbers of sorted adjacent values to each of k bins
- Move a border element from one bin to next when that reduces the sum of all distances from each number to the mean or mode of the assigned bin
- {1,1,2,2,2,5,6,8,8,9}
- Gets assigned as {1,2,2,2,2, 5,6, 8,8,9}

Chimerge: an automatic discretization algorithm

 Analyzes the quality of multiple intervals for a given feature by using chi-squared statistics

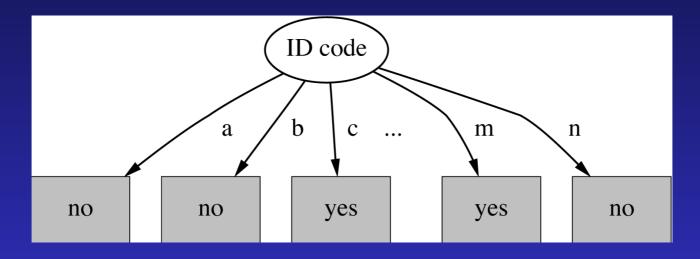
Highly Branching Attributes

- When attribute has a large number of possible values, information gain is large
 - example attributes are Date (March 4, 1979), or ID code
- Consider case when each instance has an Identification Code attribute

Play Tennis Data with Identification Codes

ID Code	Outlook	Temperature	Humidity	Windy	Play
а	sunny	hot	high	false	no
b	sunny	hot	high	true	no
С	overcast	hot	high	false	yes
d	rainy	mild	high	false	yes
е	rainy	cool	normal	false	yes
f	rainy	cool	normal	true	no
g	overcast	cool	normal	true	yes
h	sunny	mild	high	false	no
i	sunny	cool	normal	false	yes
j	rainy	mild	normal	false	yes
k	sunny	mild	normal	true	yes
I	overcast	mild	high	true	yes
m	overcast	hot	normal	false	yes
n	rainy	mild	high	true	no

Tree Stump for the ID Code Attribute



Information gain of each node is just the information at the root (0.94)

Therefore chosen as the splitting attribute!

Alternative Measures for Selecting Attributes

 Split information is sensitive to how broadly and uniformly the attribute splits the data

SplitInformation(S, A) =
$$-\sum_{i=1}^{c} \frac{|s_i|}{|S|} \log_2 \frac{|s_i|}{|S|}$$

- S1 through Sc are c subsets of examples resulting from partitioning the c-values of attribute A
- It is the entropy of S with respect to the values of attribute A
- (not wrt attribute, as considered before)

Alternative Measures for Selecting Attributes, continued

$$GainRatio(S, A) \equiv \frac{Gain(S, A)}{SplitInformation(S, A)}$$

S is the collection of data and A is the attribute

The Splitinformation term discourages the selection of attributes with many uniformly distributed values

Gain Ratio

- Take into account number and size of child nodes
- For ID code attribute:
 - Split Information[1,1,...,1]=-1/14 x log 1/14 x 14
 - = log 14 = 3.807 bits
 - Gain ratio = 0.940/3.807
 - = 0.246
 - Gain Ratio reduces the effect of high branching

Gain Ratio Calculations for the Tree Stumps

Outlook		Temperature	
info:	0.693	info:	0.911
gain:0.940-0.693	0.247	gain:0.940-0.911	0.029
split info: info([5,4,5])	1.577	split info: info([4,6,4])	1.362
gain ratio:0.247/1.577	0.156	0.029/1.362	0.021
Humidity		Windy	
info:	0.788	info:	0.892
gain:0.940-0.788	0.152	gain:0.940-0.892	0.048
split info: info([7,7])	1.000	split info: info([8,6])	0.985
gain ratio:0.152/1	0.152	gain ratio:0.048/0.985	0.049

Handling Training Examples with Missing Attribute Values (3.7.4)

- Medical domain
 - Predict patient outcome based on lab tests
 - Lab test available only for subset of patients
- Estimate missing attribute using examples with known values

Handling Training Examples with Missing Attribute Values

- Gain (S,A) is to be calculated at node n to test whether atribute A is the best attribute to test at this decision node
- Suppose <x, c(x)> is a training example in S and the value of A(x) is unknown

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