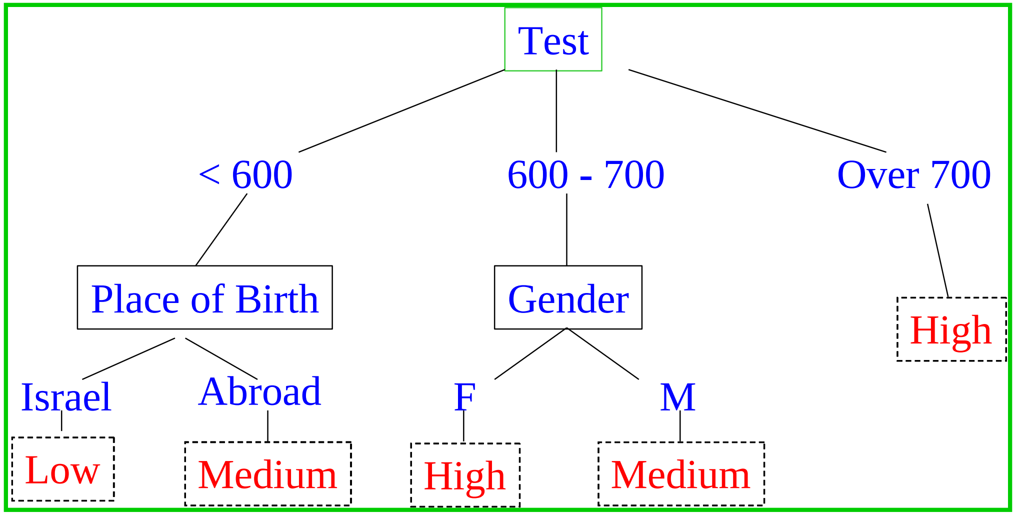
Lecture No. 5 – Decision Tree Learning II

- Rule Extraction
- Discretization of Continuous Attributes
- Alternative Splitting Rules
  - Information Gain Ratio
  - Gini Index
  - Twoing
- CART Overview
- Comparison of Decision Trees

Rule Extraction – Student Admission

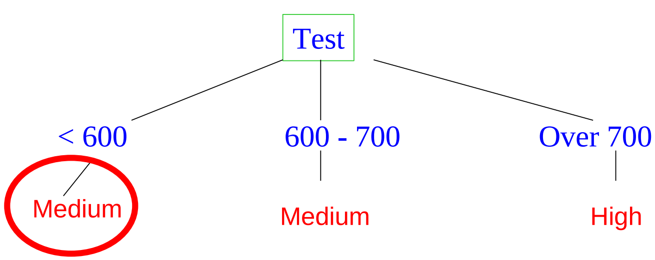


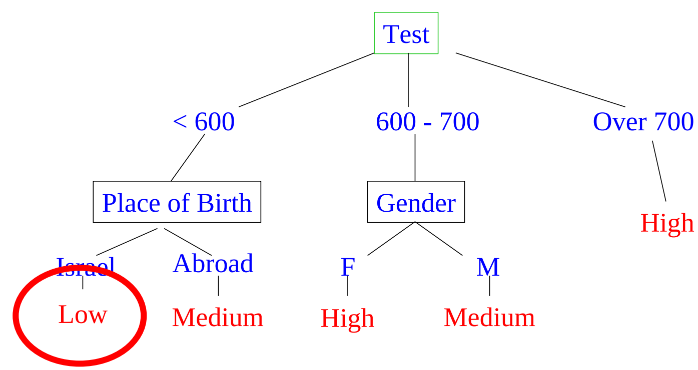
Complete set of extracted rules

- If (Test <600) and (Place of Birth = Israel) Then Grade = Low
- If (Test <600) and (Place of Birth = Abroad) Then Grade = Medium
- If (600<Test <700) and (Gender = F) Then Grade = High
- If (600<Test <700) and (Gender = M) Then Grade = Medium
- If (Test >700) Then Grade = High

Rule Coverage and Accuracy

- Coverage of a rule:
  - Fraction of records that satisfy the antecedent of a rule
- Accuracy of a rule:
  - Fraction of records that satisfy both the antecedent and consequent of a rule
- What are the coverage and the accuracy of each rule extracted from the following trees?





Characteristics of Rule-Based Classifier

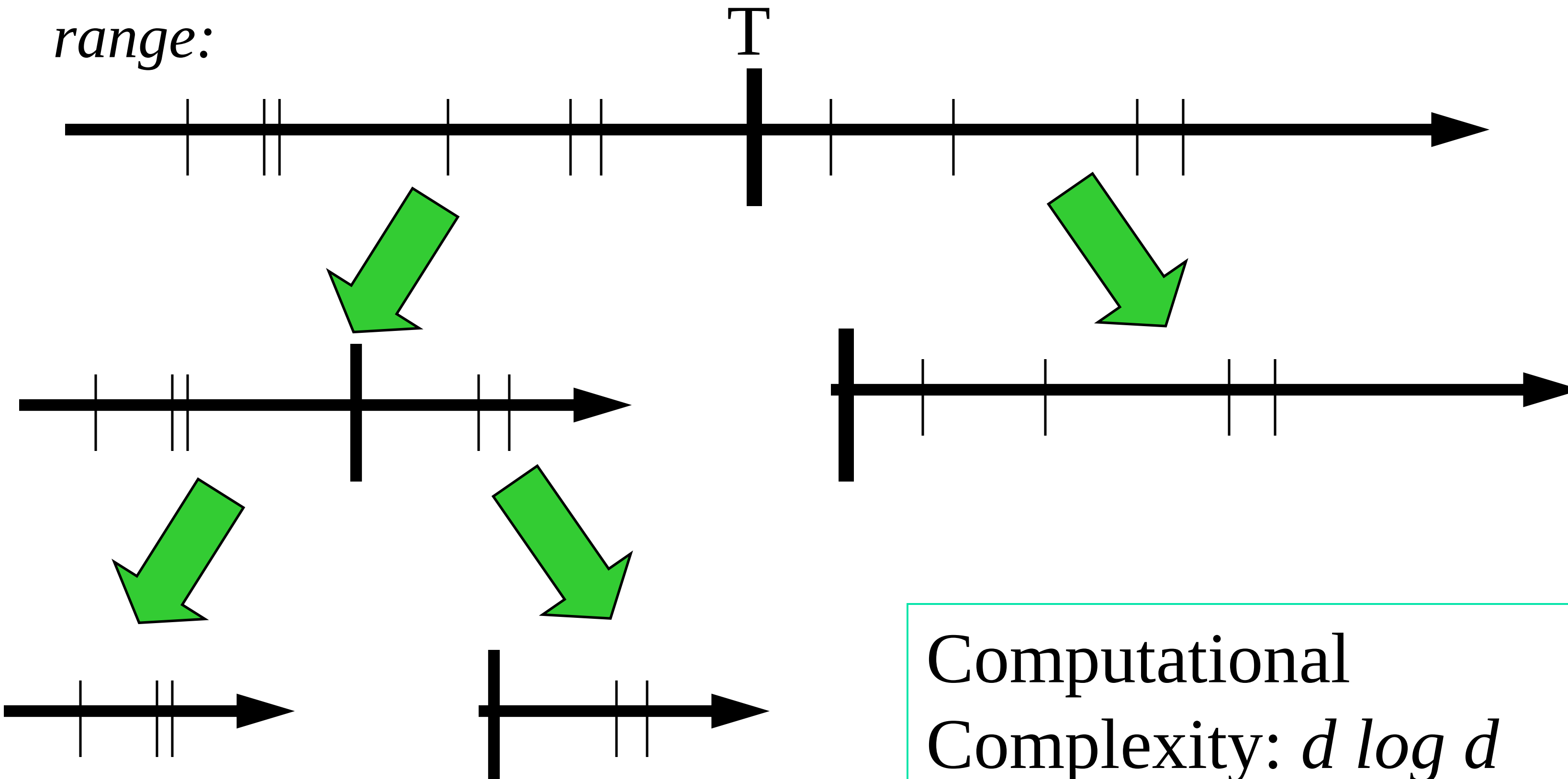
- Mutually exclusive rules (חוקים זרים)
  - Classifier contains mutually exclusive rules if the rules are independent of each other
  - Every record is covered by <u>at most</u> one rule
- Exhaustive rules (חוקים בעלי כיסוי מלא)
  - Classifier has exhaustive coverage if it accounts for every possible combination of attribute values
  - Each record is covered by <u>at least</u> one rule
- Example: mutually exclusive and exhaustive?
  - If (Test < 600) Then Grade = Low
  - If (600 ≤ Test < 700) Then Grade = Medium
  - If (Test ≥ 700) Then Grade = High

Lecture No. 5 – Decision Tree Learning II

- Rule Extraction
- Discretization of Continuous Attributes
- Alternative Splitting Rules
  - Information Gain Ratio
  - Gini Index
  - Twoing
- CART Overview
- Comparison of Decision Trees

Discretization Algorithm

Attribute range:



Computational Complexity: *d log d*

d – number of distinct values

Discretization Algorithm (continued)

- Notation
  - S - entire set of instances
  - A - attribute (feature)
  - T - threshold (partition boundary)
  - S_1 - set of instances below the threshold ($v \leq T$)
  - S_2 - set of instances above the threshold ($v > T$)

Discretization Algorithm (continued)

- Entropy induced by *T*:

$$E(A,T;S)=\frac{|S_1|}{|S|}Ent(S_1)+\frac{|S_2|}{|S|}Ent(S_2)$$

- Information Gain:
  - *Gain (A,T;S) = Ent (S) - E(A,T;S)*
- Example:

Record	1	2	3	4	5
Value	1	1.5	1.5	1.7	2.1
Class	0	1	0	1	1

Discretization Example

Record	1	2	3	4	5
Value	1	1.5	1.5	1.7	2.1
Class	0	1	0	1	1

$$E(A,T;S)=\frac{|S_1|}{|S|}Ent(S_1)+\frac{|S_2|}{|S|}Ent(S_2)$$

Value <=	Pos (0)	Neg (1)	Total		Prob (0)	Prob (1)
1	1	0	1		1.00	0.00
1.5	2	1	3		0.67	0.33
1.7	2	2	4		0.50	0.50
2.1	2	3	5		0.40	0.60
Value >	**Pos (0)**	**Neg (1)**	**Total**		**Prob (0)**	**Prob (1)**
1	1	3	4		0.25	0.75
1.5	0	2	2		0.00	1.00
1.7	0	1	1		0.00	1.00
2.1	0	0	0			

The best threshold

Value <=	plogp	plogp	Total		Entropy	Info Gain
1	0.000		0.000		0.649	0.322
1.5	0.390	0.528	0.918		0.551	**0.420**
1.7	0.500	0.500	1.000		0.800	0.171
2.1	0.529	0.442	0.971		0.971	
Value >	**plogp**	**plogp**				
1	0.500	0.311	0.811			
1.5		0.000	0.000			
1.7		0.000	0.000			
2.1						

Lecture No. 5 – Decision Tree Learning II

- Rule Extraction
- Discretization of Continuous Attributes
- Alternative Splitting Rules
  - Information Gain Ratio
  - Gini Index
  - Twoing
- CART Overview
- Comparison of Decision Trees

Splitting Rules

- General Requirements
  - Function of class probabilities $f(p_1, p_2, \ldots)$
  - Symmetric around 1/2 $f(p) = f(1-p)$
  - Convex function
- Possible splitting functions (rules)
  - Entropy (Information Gain and Gain Ratio)
  - Twoing
  - Gini Index

Information Gain Ratio

- ID3 selects the attribute which maximizes the mutual information (information gain):
  - $gain(A) = I(p, n) - E(A)$
    - $I(p, n)$ – unconditional entropy (does not depend on the choice of A)
    - $E(A)$ - conditional entropy after splitting the root node by the test attribute A
- The information gain is maximal when $E(A)$ is equal to zero
- $E(A) = 0$ if for each value of A
  - Either all examples are positive
  - Or all examples are negative

Gain Ratio (cont.)

- The problem with multi-valued and continuous attributes in noisy databases
  - The probability of a subset of examples to have the same class increases monotonically with a decrease in the subset size
    - The extreme case is a subset of one example
  - The average size of a subset decreases with an increase in the total number of attribute values (e.g., attribute Date)
- Conclusion
  - Information gain is biased towards multi-valued and continuous attributes

Gain Ratio (cont.)

- The Gain Ratio Approach
  - "Punish" the multi-valued attributes via dividing (normalizing) their information gain by the *Split Information:*

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

  - The *Split Information* represents the *entropy* of the tested attribute (in contrast to the entropy of the target attribute)
  - The Gain Ratio: *Gain(A)/SplitInfo(A)*

Gain Ratio – Student Example

שם פרטי	שם משפחה	מגדר	מקום לידה	ציון פסיכומטרי	ממוצע ציונים
First Name	**Last Name**	**Gender**	**Place of Birth**	**Test Grade**	**GPA**
David	Cohen	M	USA	Over 700	High
Ophir	Levy	M	Israel	600-700	Medium
Sharon	Grosman	F	Israel	600-700	High
Diana	Liberman	F	Russia	0-600	Medium
Anat	Klein	F	Israel	0-600	Low

Let us assume that the attribute "Place of Birth" has three possible values: USA, Israel, and Russia

Information Gain – *Place of Birth*

	Place of Birth			**Total**
	Israel	USA	Russia	
Low	1	0	0	1
p	0.333	0.000	0.000	
-logp	1.585	0.000	0.000	
Medium	1	0	1	2
p	0.333	0.000	1.000	
-logp	1.585	0.000	0.000	
High	1	1	0	2
p	0.333	1.000	0.000	
-logp	1.585	0.000	0.000	
Total	3	1	1	5
p	0.60	0.20	0.20	0.80
Entropy	1.585	0.000	0.000	**0.951**
Gain				**0.571**

$$\boxed{Info(D) = 1.522}$$

Split Information and Gain Ratio

Student Example

$$\boxed{Info(D) = 1.522}$$

	Place of Birth			**Total**
	Israel	USA	Russia	
Total	3	1	1	5
p	0.60	0.20	0.20	1.00
-logp	0.737	2.322	2.322	1.371
Gain				0.571
Gain Ratio				**0.416**

- Split Information

  Information Gain

  Information Gain Ratio

Gain Ratio (Test Grade) = **0.474**

Gain Ratio (Gender) = 0.176

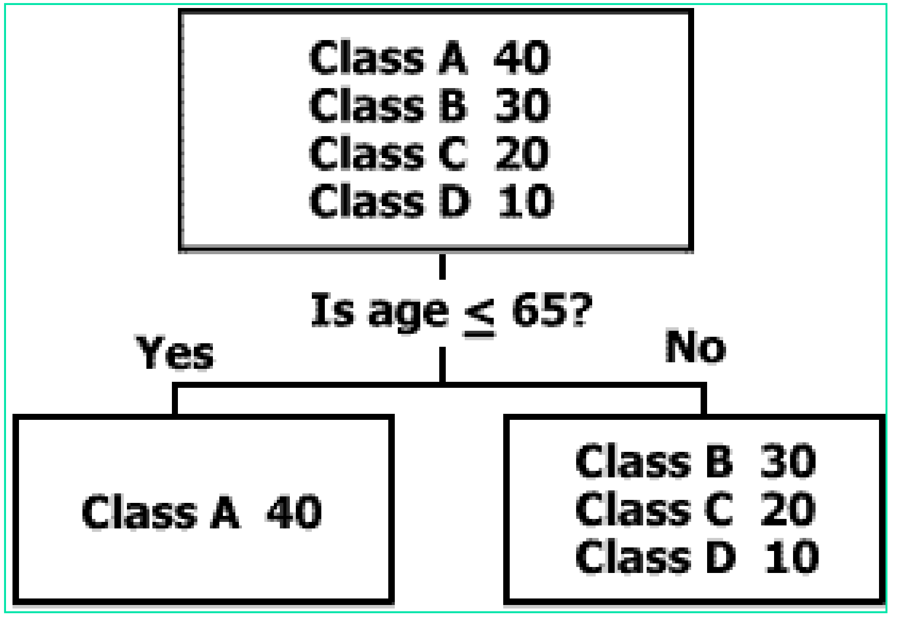
Gain Ratio (Place of Birth) = 0.416

Gini index

- All attributes are assumed continuous-valued
- Assume there exist several possible split values for each attribute
- May need other tools, such as clustering, to get the possible split values
- Can be modified for categorical attributes

Gini Splitting Rule

- Looks for the largest class in the data set and strives to isolate it from all other classes



Gini Index

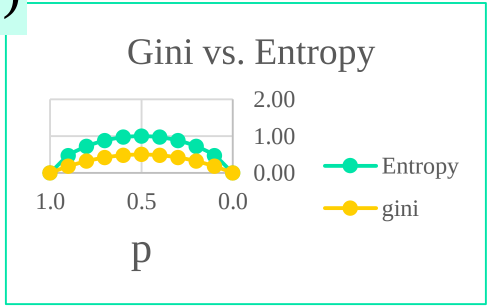
- If a data set T contains examples from n classes, gini index, $gini(T)$ is defined as

$$gini(T) = 1 - \sum_{j=1}^{n} p_j^2$$

  - where p_j is the relative frequency of class j in T.
- If a data set T is split into two subsets T_1 and T_2 with sizes N_1 and N_2 respectively, the gini index of the split data contains examples from n classes, the gini index $gini(T)$ is defined as

$$gini_{split}(T) = \frac{N_1}{N} gini(T_1) + \frac{N_2}{N} gini(T_2)$$

- Reduction in Impurity: $\Delta gini(A) = gini(D) - gini_A(D)$
- The attribute provides the smallest $gini_{split}(T)$ (or the largest reduction in impurity) is chosen to split the node (need to enumerate all possible splitting points for each attribute).



Gini Splitting Rule - Example

$$gini(T) = 1 - \sum_{j=1}^{n} p_j^2$$

Split 1

Class:	A	B	C	D	Total	Pl/Pr
Age <= 65	40	0	0	10	50	0.5
Age > 65	0	30	20	0	50	0.5
Total	40	30	20	10	**100**	

Split 2

Class:	A	B	C	D	Total	Pl/Pr
Age <= 65	40	0	0	0	40	0.4
Age > 65	0	30	20	10	60	0.6
Total	40	30	20	10	**100**	

Better split

Split 1

Prob						
A	**B**	**C**	**D**	**Gini**	**Gini Split**	Gini Drop
0.80	0.00	0.00	0.20	**0.32**	0.400	0.300
0.00	0.60	0.40	0.00	**0.48**		
0.40	0.30	0.20	0.10	**0.70**		

Split 2

Prob						
A	**B**	**C**	**D**	**Gini**	**Gini Split**	**Gini Drop**
1.00	0.00	0.00	0.00	**0.00**	0.367	**0.333**
0.00	0.50	0.33	0.17	**0.61**		
0.40	0.30	0.20	0.10	**0.70**		

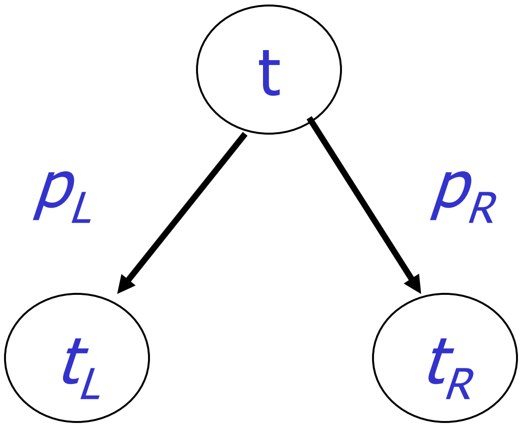
Twoing Splitting Rule (CART™)

Source: L. Breiman, J. Friedman, R. Olshen, and C. Stone (1984), *Classification and Regression Trees*, Pacific Grove: Wadsworth

- Maximize

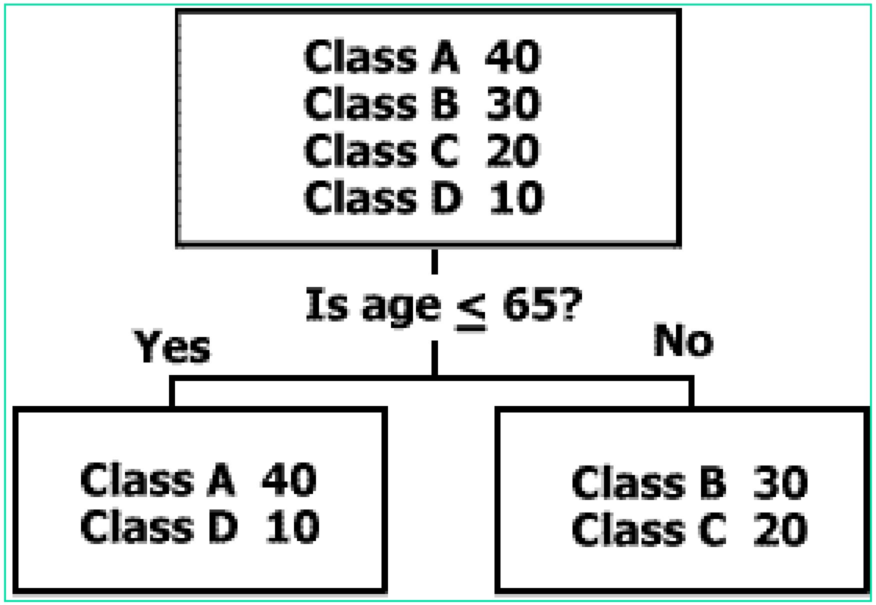
$$\frac{p_L p_R}{4}\left[\sum_j \left| p(j/t_L) - p(j/t_R) \right| \right]^2$$

- Notation
  - p_L – proportion of cases going to the left node
  - p_R – proportion of cases going to the right node
  - j – class index
  - $p\ (j/t_L)$ – probability of class j at the left node
  - $p\ (j/t_R)$ – probability of class j at the right node
- Attempts to find groups of up to 50% of the data each
- If impossible - power-modified twoing

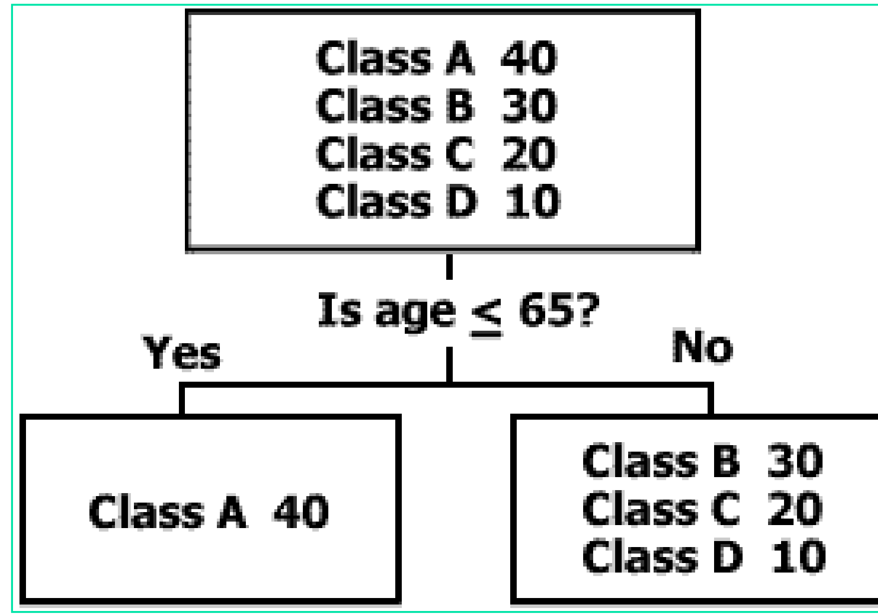


Twoing Splitting Rule - Example

Split 1



Split 2



Which split is better?

Twoing Splitting Rule - Example

$$\frac{p_L p_R}{4}\left[\sum_j \left| p(j/t_L) - p(j/t_R) \right|\right]^2$$

Split 1

Class:	A	B	C	D	Total	Pl/Pr
Age <= 65	40	0	0	10	50	0.5
Age > 65	0	30	20	0	50	0.5
Total					100	

Better split

Split 2

Class:	A	B	C	D	Total	Pl/Pr
Age <= 65	40	0	0	0	40	0.4
Age > 65	0	30	20	10	60	0.6
Total					100	

Prob			
A	B	C	D
0.80	0.00	0.00	0.20
0.00	0.60	0.40	0.00

Abs						
A	B	C	D	Total		Twoing
0.800	0.600	0.400	0.200	2.000		0.250

Prob			
A	B	C	D
1.00	0.00	0.00	0.00
0.00	0.50	0.33	0.17

Abs						
A	B	C	D	Total		Twoing
1.000	0.500	0.333	0.167	2.000		0.240

Using Splitting Rules (Gini, Twoing, Entropy)

- Gini -- is usually best for yes/no outcomes
- Twoing - similar to entropy but more flexible because it has a tuning parameter
  - excellent for multi-class outcomes
  - twoing excellent for hard to classify problems
    - problems where accuracy for all methods will be low
    - inherently difficult problems or low signal/noise ratio
- Entropy- popular in Machine Learning literature

Lecture No. 5 – Decision Tree Learning II

- Rule Extraction
- Discretization of Continuous Attributes
- Alternative Splitting Rules
  - Information Gain Ratio
  - Gini Index
  - Twoing
- CART Overview
- Comparison of Decision Trees

CART™ Algorithm
Main Steps

- Grow the maximal tree based on the entire data set
  - A binary splitting procedure
  - Splitting rules
  - Stopping criteria
- Derive a set of pruned sub-trees
  - Create “efficiency frontier”
- Select the best tree by using validation set or cross-validation

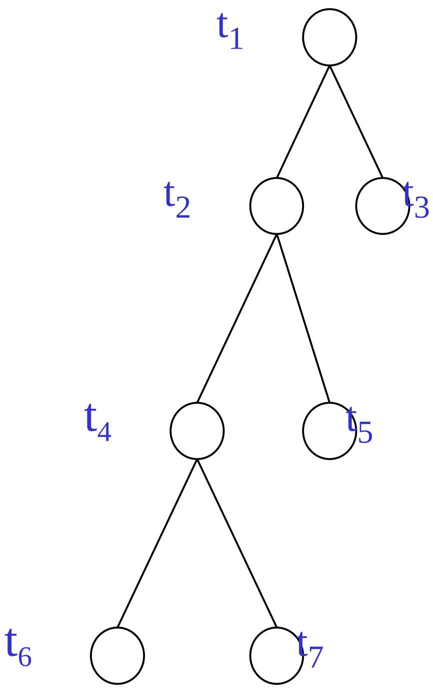
CART™ : Binary Splitting Procedure

- Continuous (Ordinal) Attributes
  - Each distinct value is considered for threshold
  - Branching rule: $x \leq C$
  - M possible splits (M - number of distinct values)
- Nominal (Categorical) Attributes
  - The branching rule is determined separately for each possible value
  - $2^{M-1} - 1$ possible splits (M - number of values)

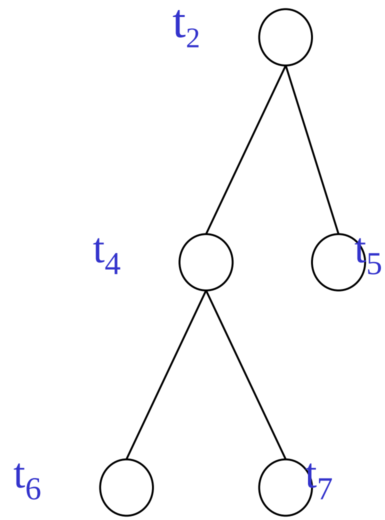
CART™ : Stopping Criteria

- Splitting is impossible
  - One case left in a node
  - All the cases in the node have the same target value
- Other reasons
  - Too few cases in the node (default = 10 cases)

Pruning Trees

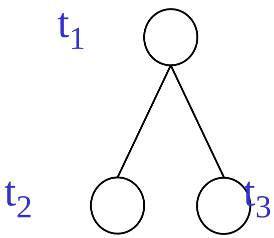


Tree T



Branch

T_{t2}



Sub-tree

$T - T_{t2}$

Deriving a set of pruned sub-trees

- Objective: minimizing the cost-complexity function

$$R_\alpha(T) = R(T) + \alpha \cdot \left|\tilde{T}\right|$$

  - T - a tree
  - $R(T)$ - the training error rate of a tree
  - $R_\alpha(T)$ - the cost-complexity of a tree
  - $\left|\tilde{T}\right|$ - number of terminal nodes in a tree
  - α - complexity parameter (real number, greater than zero)

CART™ Pruning Algorithm

$$R_\alpha(T) = R(T) + \alpha \cdot \left|\tilde{T}\right|$$

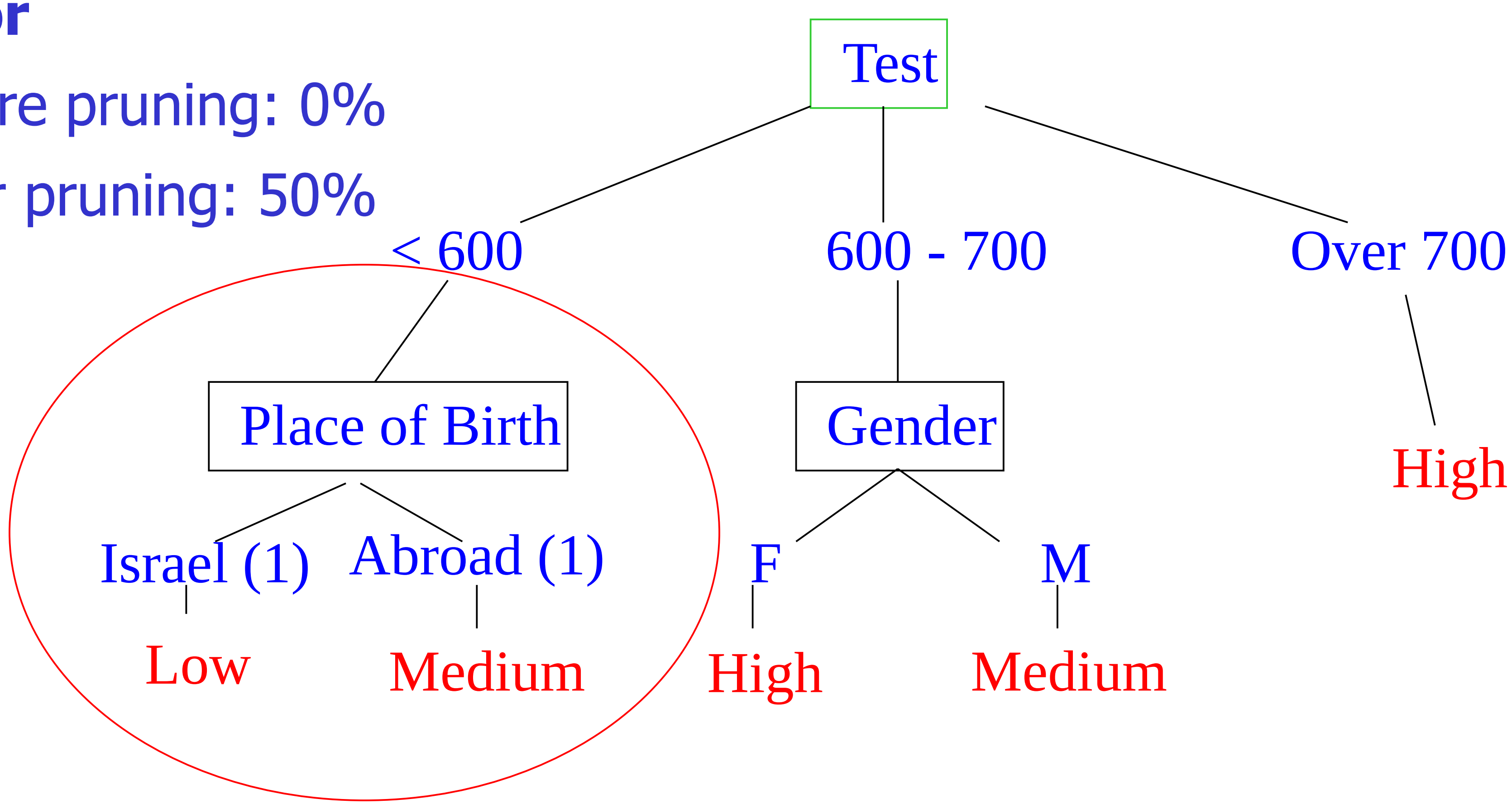
- Step 1 - Initialize the list of optimal trees with the maximal tree
- Step 2 - Initialize $\alpha = 0$
- Step 3 - Increase α until the tree ceases to be optimal
- Step 4 - Find a new sub-tree, which is optimal with the new value of α
- Step 5 - Add the new sub-tree to the list of optimal trees.
- Step 6 - If the new sub-tree has more than one terminal node, go to Step 3. Otherwise, stop.

CART™ Student Example Maximal Tree ($\alpha = 0$)

Error

Before pruning: 0%

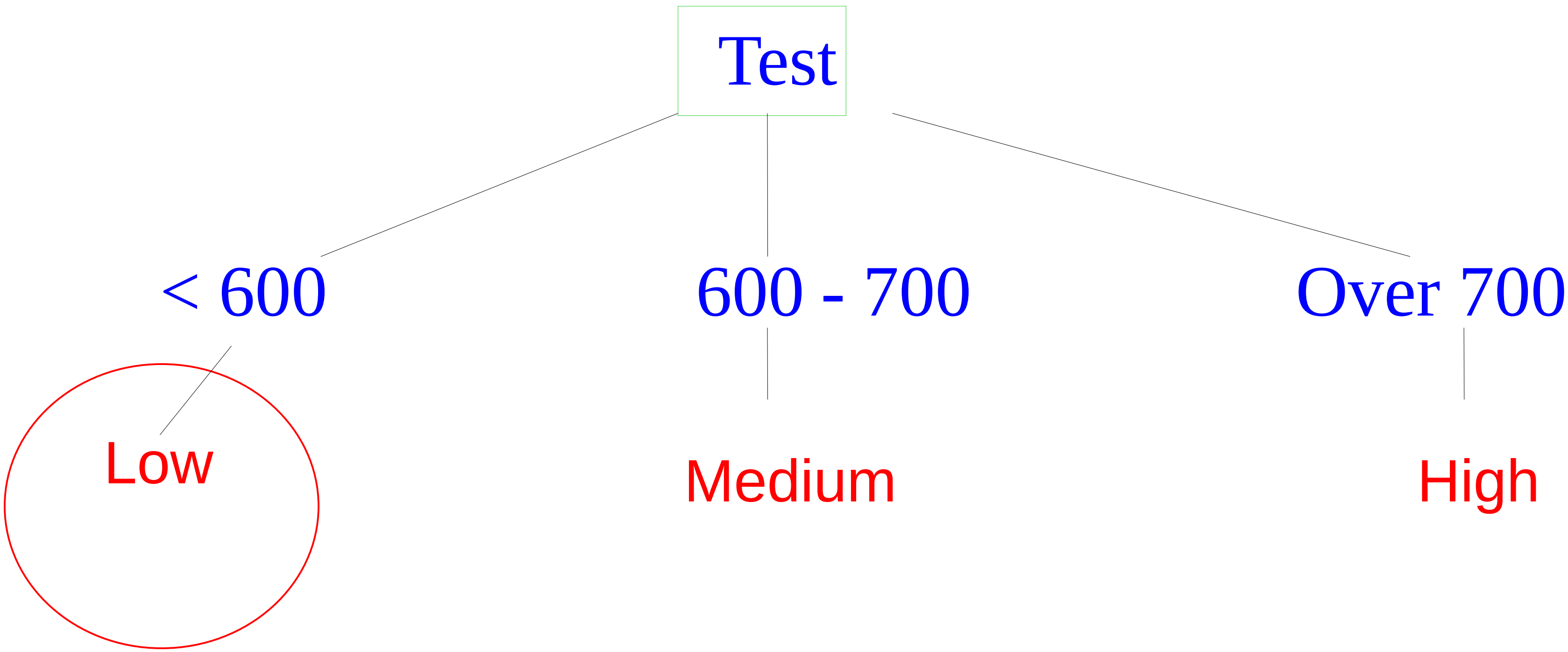
After pruning: 50%



CART™ Student Example (cont'd) Removing *Place of Birth*

- Cost-complexity of the single node t
  - $R_\alpha(\{t\}) = R(t) + \alpha*1 = 0.50 + \alpha$
- Cost-complexity of the branch T_t
  - $R_\alpha(T_t) = R(T_t) + \alpha*|\check{T}_t| = 0 + \alpha*2$
- The critical value of α
  - $R_\alpha(\{t\}) = R_\alpha(T_t)$
  - $0.50 + \alpha = 2\alpha$
  - $\alpha = 0.50$

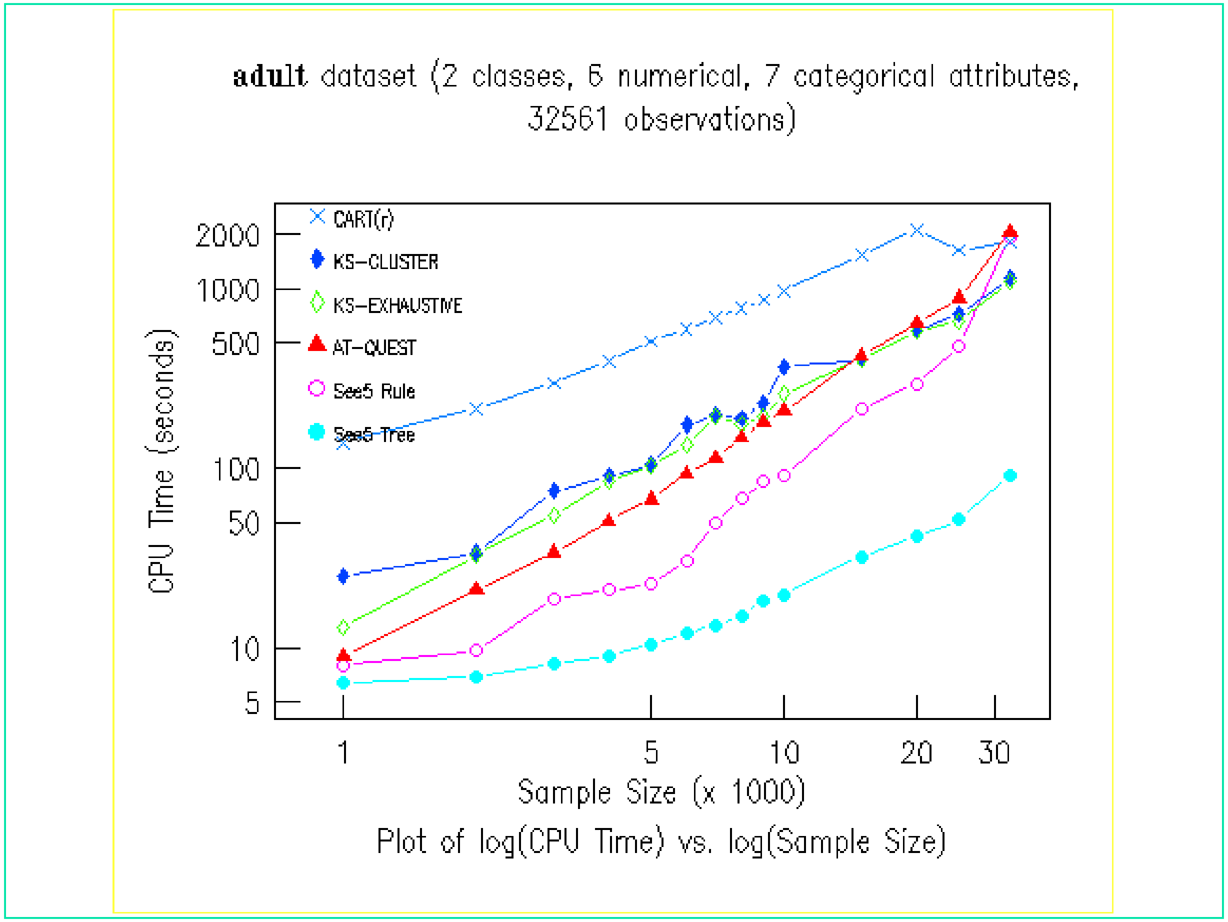
CART™ Student Example (cont'd) New Sub-Tree (α = 0.50)



Lecture No. 5 – Decision Tree Learning II

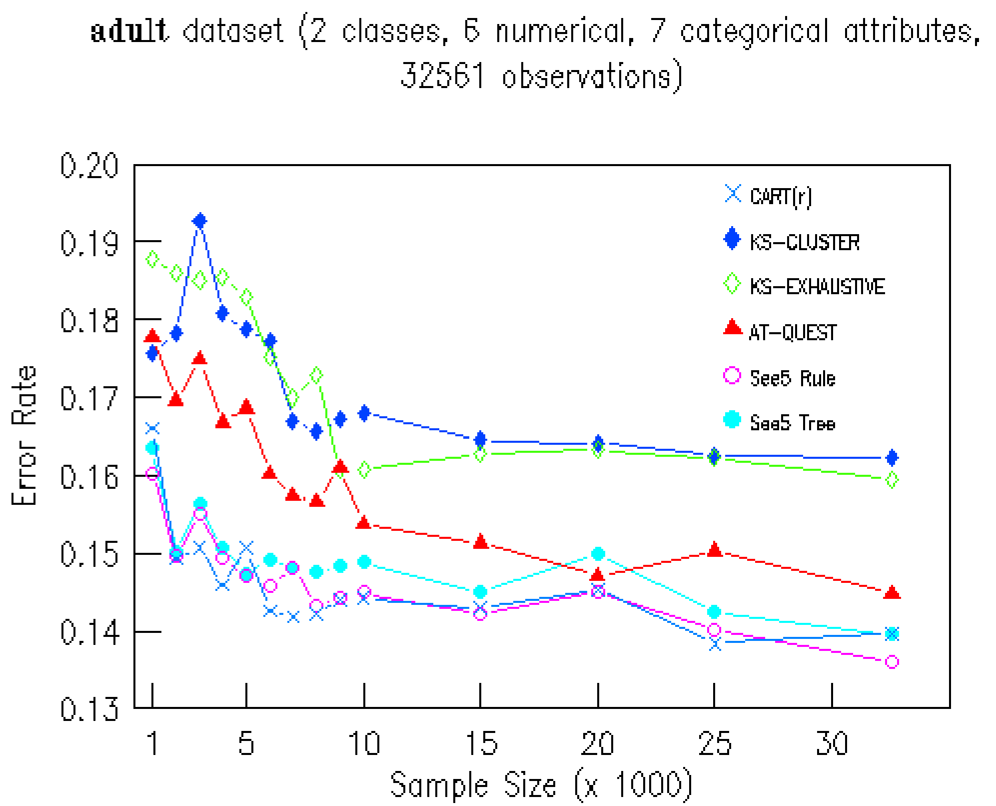
- Rule Extraction
- Discretization of Continuous Attributes
- Alternative Splitting Rules
  - Information Gain Ratio
  - Gini Index
  - Twoing
- CART Overview
- Comparison of Decision Trees

Comparison of Decision Trees
(based on Lim *et al*., Machine Learning, 40, 203–228, 2000)
Computational Complexity



Comparison of Decision Trees

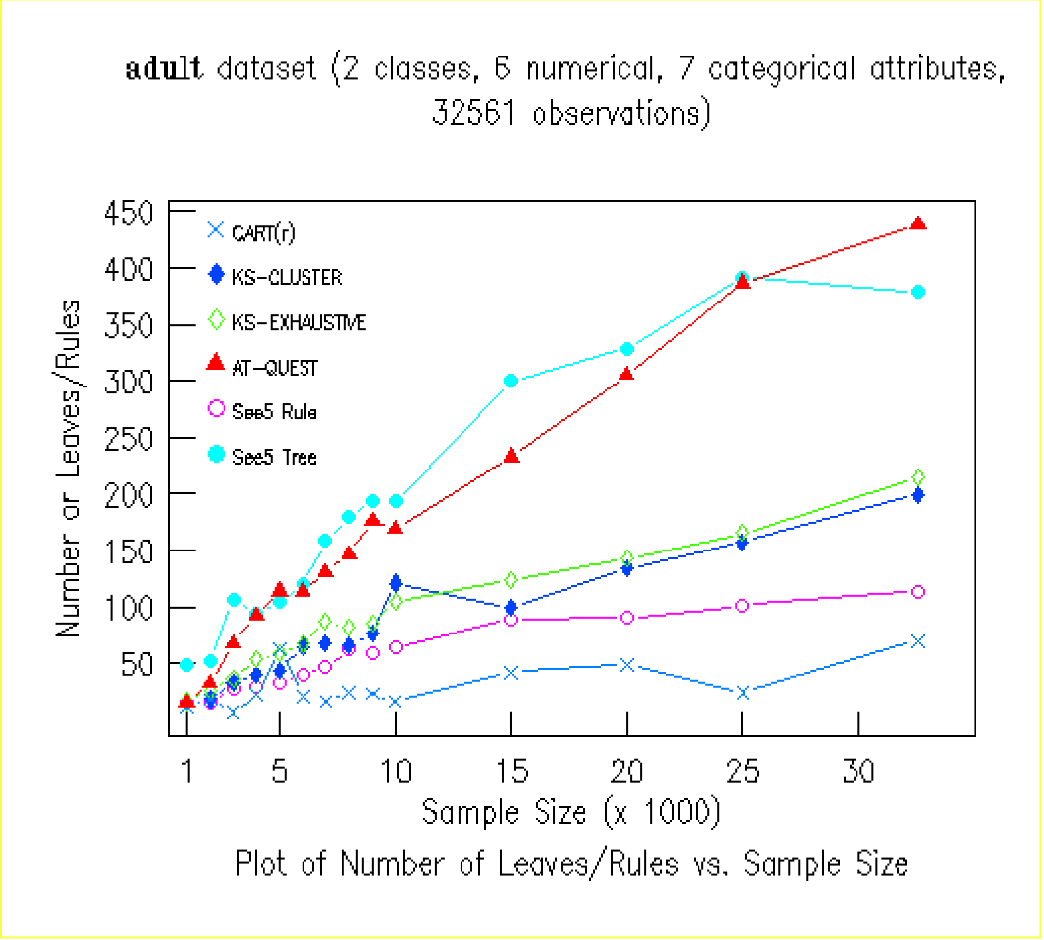
Error Rate



Plot of Error Rate vs. Sample Size

Comparison of Decision Trees

Tree Size



Lectures No. 4-5: Summary

- Classification tasks involve *model construction* and *model testing*
- Decision trees are one of the most popular classification models
- Decision trees are usually constructed in *a top-down recursive divide-and-conquer manner*
- Overfitting can be avoided with *pre-pruning* and *post-pruning* techniques
- Most popular splitting criteria include *Gini*, *Twoing*, and *Entropy*