Data Mining
Classification: Decision Trees
(chapter 3.3)

Instructor: Sharma Chakravarthy
sharmac@cse.uta.edu
The University of Texas at Arlington

Classification: preamble

- It can be used as a predictive model
  - For previously unlabeled instances
- It can be used as a descriptive model (rules)
  - Explanation of prediction in terms of attribute values
  - Especially critical for some applications (e.g., medical diagnosis)
  - Remember, one of the complaints about the Neural Networks (ANN, CNN, …) is that it is non-descriptive!
- The class attribute needs to be nominal (not necessarily binary) or converted to finite ranges!
- Other attributes can be of any types (discrete, continuous, qualitative, quantitative)
- Not all attributes may be relevant; finding optimal combination of attributes that best discriminates instances is challenging!

2

Classification: preamble

- Classifier and model are often used synonymously
- A classification technique may build a single model, multiple models, or an ensemble of models
  - Decision tree builds a single model
  - k-nearest neighbors does not build an explicit model
  - Ensemble classifiers combine the output of a collection of models
- The induction (building the model) and deduction (predicting on unseen instances) steps are done separately
- Confusion matrix is used for assessing the performance (accuracy and error rate) of a classifier

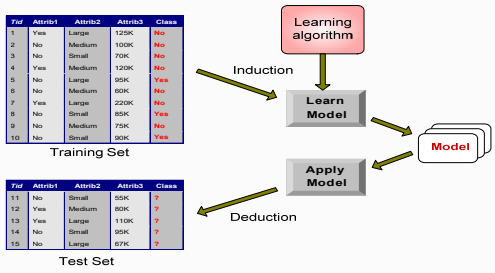
3

Classification: Definition

- Given a collection of records (*training set*)
  - Each record contains a set of *attributes*, one of the attributes being the *class*.
- Find a *model* for class attribute as a function of the values of other attributes.
- Goal: previously unseen records should be assigned a class as accurately as possible.
  - A *test set* is used to determine the accuracy of the model. Usually, the given data set is divided into training and test sets, with training set used to build the model and test set used to validate it.

4

Illustrating Classification Task



5

Examples of Classification Task

- Predicting tumor cells as benign or malignant
- Classifying credit card transactions as legitimate or fraudulent
- Classifying secondary structures of protein as alpha-helix, beta-sheet, or random coil
- Categorizing news stories as finance, weather, entertainment, sports, etc.

6

Classification vs. Prediction

Classification

- Predicts categorical class labels
- Most suited for nominal attributes
- Less effective for ordinal attributes (why?)

Prediction

- models continuous-valued functions or ordinal attributes, i.e., predicts unknown or missing values
- E.g., Linear regression

7

Supervised vs. Unsupervised Learning

Supervised learning (e.g., classification)

- Supervision: The training data (observations, measurements, etc.) are accompanied by labels indicating the class of the observations
- New data is classified based on the training set

Unsupervised learning (e.g., clustering)

- The class labels of training data is unknown
- Given a set of measurements, observations, etc. with the aim of establishing the existence of classes or clusters in the data

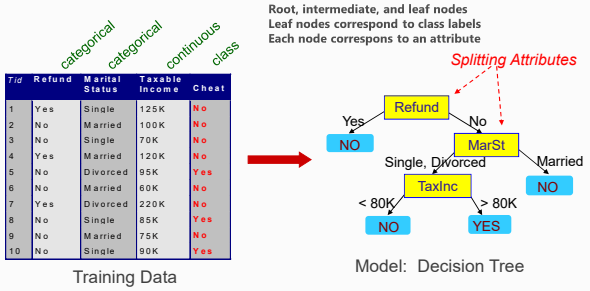
8

Classification Techniques

- Decision Tree based Methods
- Rule-based Methods
- Memory based reasoning
- Neural Networks
- Naïve Bayes classification
  - Bayesian Belief Networks
- Tf-idf or Support Vector Machines (SVM)

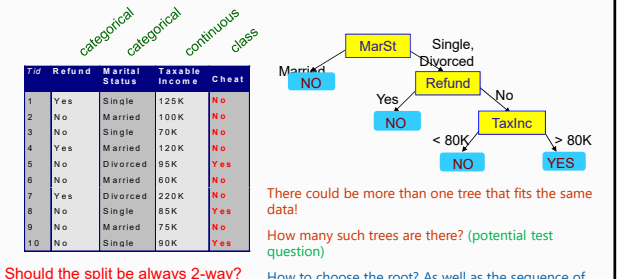
9

Example of a Decision Tree



10

Another Example of Decision Tree for the same data



Should the split be always 2-way?

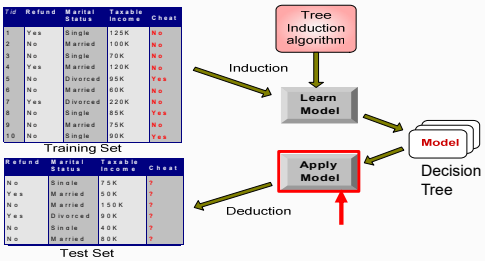
There could be more than one tree that fits the same data!

How many such trees are there? (potential test question)

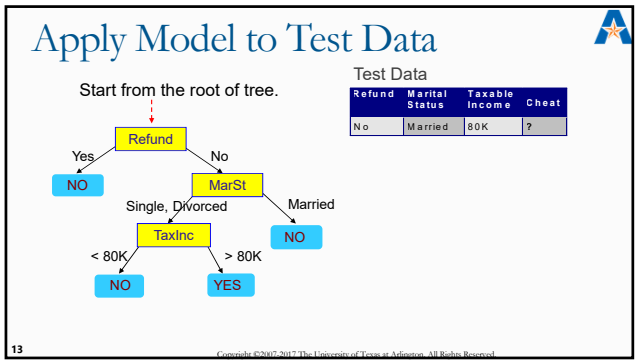
How to choose the root? As well as the sequence of nodes for splitting?

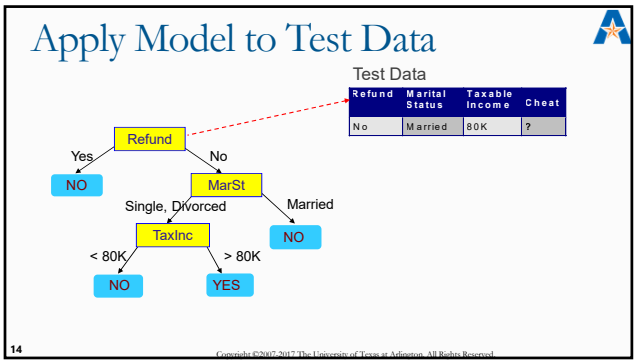
11

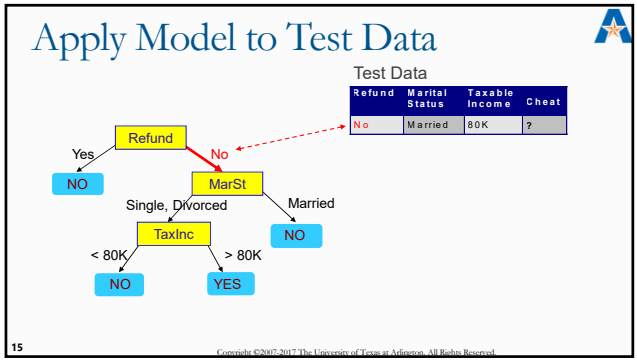
Decision Tree Classification Task

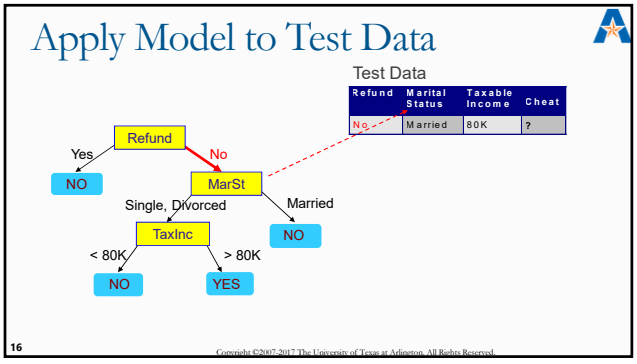


12





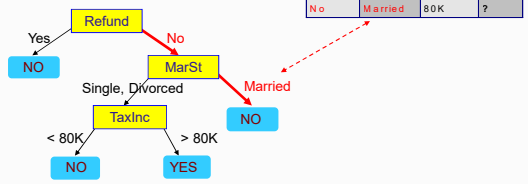




Apply Model to Test Data

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

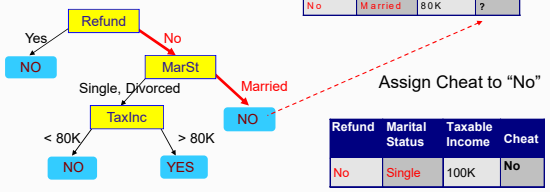


17

Apply Model to Test Data

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

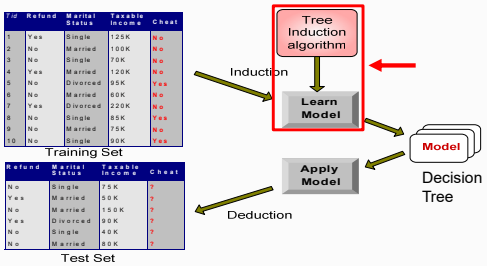


Assign Cheat to "No"

Refund	Marital Status	Taxable Income	Cheat
No	Single	100K	No

18

Decision Tree Classification Task



19

Decision Tree problem

- Large search space (why?) what does it depend on?
  - # of tuples in training set? # of attributes?
  - Exponential in size, with respect to the set of attributes
- Finding the optimal decision tree is computationally infeasible (why?)
  - Hence, need an Efficient algorithm for suboptimal decision tree
  - Greedy strategy (top-down, recursive, divide-and-conquer)
    - Grow the tree by making locally optimal decisions in selecting the attributes
- Generating an optimal tree is NP-complete
- What roles does the attribute values play?

20

Decision Tree size

- Can we say anything about depth, breadth of the tree?
  - Worst case?
- What is the effect of the number of attributes
  - Determines depth
- What is the effect of attribute values
  - Breadth of the tree
- In addition, what else plays a role in complexity?
  - Permutation order of attributes to choose and split

21

Algorithm for Decision Tree Induction

- Basic algorithm (a greedy algorithm)
  - Tree is constructed in a top-down, recursive, divide-and-conquer manner
  - At start, all the training examples are at the root
  - Attributes are categorical (if continuous-valued, they are discretized in advance)
  - Examples are partitioned recursively based on selected attributes
  - Test attributes are selected on the basis of a heuristic or statistical measure (we will discuss a couple of them)
- Conditions for stopping partitioning
  - All samples for a given node belong to the same class
  - There are no remaining attributes for further partitioning – majority voting is employed for classifying the leaf
  - There are no samples left

22

Decision Tree Induction

- Many Algorithms:
  - Hunt's Algorithm (one of the earliest) (CH 3) -- 1966
  - CART (uses only binary splits) (1984) uses GINI
  - ID3 (1986), C4.5 (1993, successor) uses entropy
  - SLIQ, SPRINT

23

General Structure of Hunt's Algorithm

- Let D_t be the set of training records that are associated with node t (at any point) and $y = \{y1, y2, \ldots, yn\}$ be the class labels.
- Steps:

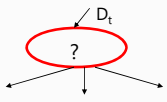
1. If D_t contains records that belong the same class y_t, then t is a leaf node labeled as y_t.

   No need to discriminate any further!

2. If D_t is an empty set, then t is a leaf node labeled by the majority class among the records of Dt's parent node.

   No need to continue any further!

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



24

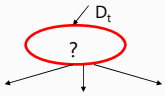
General Structure of Hunt's Algorithm

- Let D_t be the set of training records that are associated with node t (at any point) and y = {y1, y2, …,yn} be the class labels.
- Steps:

3. If D_t contains records that have identical values on all attributes but the class attribute, then t is a leaf node labeled by the majority class among D_t's records.
   Cannot expand/split this node any further!
4. If none of the above conditions is satisfied, use an attribute test to split the data into smaller subsets. Recursively apply the procedure to each subset.

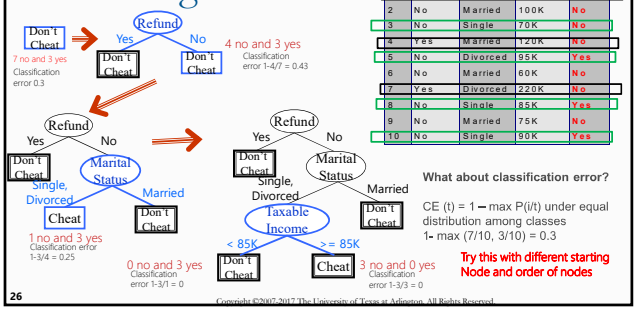
This is the non-stopping condition that continues to split the node and develop the decision tree.

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



25

Hunt's Algorithm



Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

What about classification error?

CE (t) = 1 – max P(i/t) under equal distribution among classes
1- max (7/10, 3/10) = 0.3

Try this with different starting Node and order of nodes

26

Tree Induction

- Greedy strategy.
  - Split the records based on an attribute test that optimizes certain criterion.
- Issues
  - Determine how to split the records
    - How to specify the attribute test condition?
    - How to determine the best split?
  - Determine when to stop splitting

27

How to Specify Test Condition?

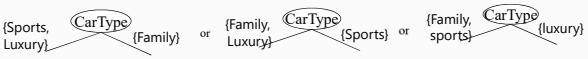
- Depends on attribute types
  - Categorical vs. Numeric
    - Categorical attributes: Nominal, Ordinal
    - Numeric attributes: Interval, Ratio
  - Discrete vs. Continuous
- Depends on number of ways to split
  - 2-way split
  - Multi-way split

28

Splitting Based on Nominal Attributes: Example

- Multi-way split: #Partitions equal to distinct values.

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

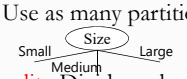


HW: How many **binary splits** are possible for k (4) attribute values? 10 **(how? List them out)**

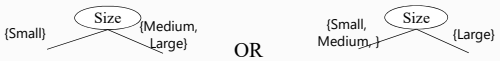
29

Splitting Based on Ordinal Attributes: another example

- Multi-way split: Use as many partitions as distinct values.



- Binary (2-way) split: Divides values into two subsets.
- Need to find optimal partitioning.



- What about this split?

HW: How many **binary splits** are possible for 4 (k) attribute values? **For non-binary splits?**

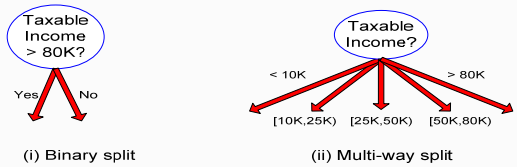
30

Splitting Based on Continuous Attributes

- Different ways of handling
  - Discretization to form an ordinal categorical attribute
    - Static – discretize once at the beginning
    - Dynamic – ranges can be found by equal interval bucketing, equal frequency bucketing (remember histograms) (percentiles), or clustering
  - Binary split: $(A < v)$ or $(A \geq v)$
    - consider all possible splits and find the best cut
    - can be more compute intensive

31

Splitting Based on Continuous Attributes



Remember open and closed intervals "(" and "]"

32

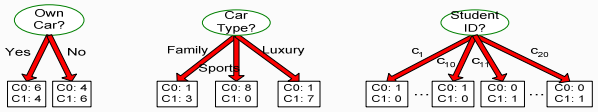
Split Strategies for Tree Induction

- Greedy strategy.
  - Split the records based on an attribute test that optimizes certain criterion. We have not discussed this yet
- Issues
  - Determine how to split the records
    - How to specify the attribute test condition?
    - How to determine the best split?
  - Determine when to stop splitting
    - When no more instances are left
    - When all remaining instances have the same values in all attributes
    - Early stopping is also possible!

33

How to determine the Best Split

Before Splitting: 10 records of class c0, 10 records of class c1



Error rate?

What do these numbers in each class mean?

Which test condition is the best? And why?

But is it?

Based on impurity, 3rd one seems best?

Large split with small sets; semantics is a big problem!

34

How to determine the Best Split

- Impurity of a node measures how dissimilar the class labels are for the data instances belonging to a common node,
  - Hence, nodes with homogeneous class distribution are preferred!
- Need a measure of node impurity:

C0: 5
C1: 5

Non-homogeneous,

High degree of impurity

C0: 9
C1: 1

Homogeneous,

Low degree of impurity

How can we quantify this?

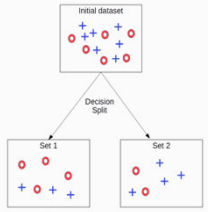
35

Entropy in Decision Trees?

- Entropy is a measure of disorder
- Can also be explained in terms of information theory
- *Entropy is an indicator of how messy your data is*
- If everything is in its place, there is no disorder (high entropy); if they are scattered, then there is disorder (low entropy)
- Fortunately, can be measured mathematically for data.
  - In decision trees, the goal is to tidy the data. You try to separate your data and group the samples together into the classes they belong to. You know their label since you construct the trees from the training set. You **maximize the purity** of the groups as much as possible each time you create a new node of the tree (meaning you cut your set in two or more). Of course, at the end of the tree, you want to have a clear answer.
  - So, what kind of function are we looking for?

36

Tidying up



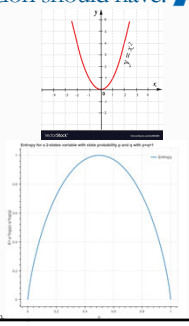
Children more tidier
Than the parent?

From: https://bricaud.github.io/personal-blog/entropy-in-decision-trees/

37

What property this function/computation should have?

1. Having a single class (all labels being same) is very good/pure (low value)
2. Having equal distribution is very bad/impure (high value)
3. All other combinations should fall in between!
4. What is a good function for this?



38

Measures of Node Impurity

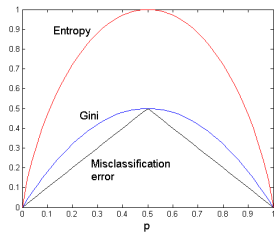
1. Gini Index
2. Entropy (information gain)
3. Misclassification error

- Both GINI and Entropy use impurity.
- Both give a zero impurity value if the node contains instances from a single class and maximum impurity value if the node contains equal proportion of instances from multiple classes

39

Comparison among Splitting Criteria

For a 2-class problem:



X and y axis are relative Frequency ($p_i(t)$) of training instances That belong to class i at node t

Measure of Impurity: GINI

- Gini Index for a given node t :

$$GINI(t) = 1 - \sum_j [p(j \mid t)]^2$$

(NOTE: $p(j \mid t)$ is the relative frequency of class j at node t).

- Maximum $(1 - 1/n_c)$ when records are equally distributed among all c classes, implying least interesting information
- Minimum (0.0) when all records belong to one class, implying most interesting information

C1	0
C2	6
Gini=0.000	

$1-(0/6)^2 - (6/6)^2$

C1	1
C2	5
Gini=0.278	

$1-(1/6)^2 - (5/6)^2$

C1	2
C2	4
Gini=0.444	

$1-(2/6)^2 - (4/6)^2$

C1	3
C2	3
Gini=0.500	

$1-(3/6)^2 - (3/6)^2$

- Try applying this formula for 3 classes with different distributions

Examples for computing GINI

$$GINI(t) = 1 - \sum_j [p(j \mid t)]^2$$

C1	0
C2	6

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1

Gini = $1 - P(C1)^2 - P(C2)^2 = 1 - 0 - 1 = 0$

C1	1
C2	5

P(C1) = 1/6 P(C2) = 5/6

Gini = $1 - (1/6)^2 - (5/6)^2 = 0.278$

C1	2
C2	4

P(C1) = 2/6 P(C2) = 4/6

Gini = $1 - (2/6)^2 - (4/6)^2 = 0.444$

Splitting Based on GINI

- Used in CART, SLIQ, SPRINT.
- When a parent node p is split into k partitions (children), the quality of split is computed as,

$$GINI_{split} = \sum_{i=1}^{k} \frac{n_i}{n} GINI(i)$$

where, n_i = number of records at child i,

n = number of records at node p.

Choose the split that has maximum value (why?) for

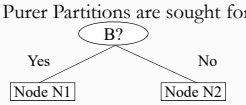
$$GINI_{parent} - GINI_{split}$$

Binary Attributes: Computing GINI Index

- Splits into two partitions
- Effect of Weighing partitions:
  - Larger and Purer Partitions are sought for.

$GINI_{parent} - GINI_{children} = 0.5 - 0.371 = 0.13$

	Parent
C1	6
C2	6
Gini = 0.500	



Gini(N1)
= $1 - (5/7)^2 - (2/7)^2$
= 0.408

Gini(N2)
= $1 - (1/5)^2 - (4/5)^2$
= 0.32

	N1	N2
C1	5	1
C2	2	4
Gini=0.371		

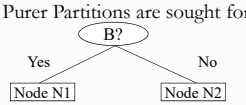
Gini(Children)
= 7/12 * 0.408 +
5/12 * 0.32
= 0.371

Binary Attributes: Computing GINI Index (2)

- Splits into two partitions
- Effect of Weighing partitions:
  - Larger and Purer Partitions are sought for.

$GINI_{parent} - GINI_{children} = 0.5 - 0.0 = 0.5$

	Parent
C1	6
C2	6
Gini = 0.500	



Gini(N1)
= 1 – (6/6)² – (0/6)²
= 1- 1-0 = 0

Gini(N2)
= 1 – (0/6)² – (6/6)²
= 1 – 0 -1 = 0

	N1	N2
C1	6	0
C2	0	6
Gini= ?		

Gini(Children)
= 6/12 * 0.0 +
6/12 * 0.0
= 0.0

45

Categorical Attributes: Computing Gini Index

- For each distinct value, gather counts for each class in the dataset
- Use the count matrix to make decisions

Multi-way split

CarType	Family	Sports	Luxury
C1	1	2	1
C2	4	1	1
Gini	0.393		

Two-way split
(find best partition of values)

CarType	{Sports, Luxury}	{Family}
C1	3	1
C2	2	4
Gini	0.400	

CarType	{Sports}	{Family, Luxury}
C1	2	2
C2	1	5
Gini	0.419	

- HW: what does the parent look like? Compute the GINI of the parent
- HW: Using GINI gain, which of the above split is the best?
  - Do you necessarily need the parent GINI?

46

Continuous Attributes: Computing Gini Index

- Use Binary Decisions based on one value
- Several Choices for the splitting value
  - Number of possible splitting values
    = Number of distinct values
- Each splitting value has a count matrix associated with it
  - Class counts in each of the partitions, A < v and A ≥ v
- Simple method to choose best v
  - For each v, scan the database to gather count matrix and compute its Gini index
  - Computationally Inefficient! Repetition of work.

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	100K	Yes

47

Continuous Attributes: Computing Gini Index...

- For efficient computation: for each attribute,
  - Sort the attribute on values
  - Linearly scan these values, each time updating the count matrix and computing gini index (one scan vs. 10 scans for this example!)
  - Choose the split position that has the least Gini index (why?)

Sorted Values →
Split Positions →

Cheat	No	No	No	Yes	Yes	Yes	No	No	No	No
Taxable Income										
	60	70	75	85	90	95	100	120	125	220
	55	65	72	80	87	92	97	110	122	172	230
	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >
Yes	0 3	0 3	0 3	0 3	1 2	2 1	3 0	3 0	3 0	3 0	3 0	3 0
No	0 7	1 6	2 5	3 4	3 4	3 4	3 4	4 3	5 2	6 1	7 0
Gini	0.420	0.400	0.375	0.343	0.417	0.400	0.300	0.343	0.375	0.400	0.420

1-sq(3/10) – sq(7/10)
1-0.09-0.49 = 0.42
Gini(children) =
0/10 * 0 + 10/10*0.42
= 0.42

1-sq(3/6) – sq(3/6)
1-0.25-0.25 = 0.5
1-sq(0/4)-sq(4/4) =0
Gini(children) =
6/10*0.5 + 4/10*0=0.3

Hw: Figure out the Parent class instances!
Is it the same for each split?

48

How to determine the Best Split (another way)

- Impurity of a node measures how dissimilar the class labels are for the data instances belonging to a common node,
  - Hence, nodes with homogeneous class distribution are preferred!
- Need a measure of node impurity:

C0: 5 C1: 5	C0: 9 C1: 1
Non-homogeneous, High degree of impurity	Homogeneous, Low degree of impurity

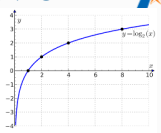
How can we quantify this in Another way?

49

2. Alternative Splitting Criteria based on INFO

- Entropy at a given node t:

$$Entropy\quad (t) = -\sum_j p(j \mid t) \log_2 p(j \mid t)$$



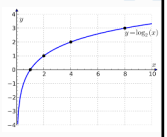
- (NOTE: $p(j \mid t)$ is the relative frequency of class j at node t).
  - Consider two labels and N items with m and N-m distribution
  - Let p = n/N and q = m/N or 1-p
  - The entropy of the above is
  - E = -p $\log_2$ (p) – q $\log_2$ (q)
  - What is the max value and min value of the above formula?
    - It is minimum when either n is 0 or m is 0
      - - [0/N $\log_2$ (0) + N/N $\log_2$ (1)] or - [N/N $\log_2$ (1) + 0/N $\log_2$ (0)]
      - 0 + 0 or 0 + 0
    - It is maximum when n = m = N/2
      - - [1/2 $\log_2$ (1/2) + ½ $\log_2$ (1/2)]
      - -[½ * -1 + ½ * -1] = -[-1/2 -1/2]
      - 1

50-

2. Alternative Splitting Criteria based on INFO

- Entropy at a given node t:

$$Entropy\quad (t) = -\sum_j p(j \mid t) \log_2 p(j \mid t)$$



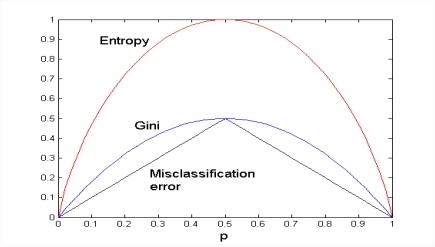
(NOTE: $p(j \mid t)$ is the relative frequency of class j at node

- Measures homogeneity of a node.
  - Maximum (log n_c (contrast with GINI!)) when records are equally distributed among all classes implying least information ($\log_2$ 2 is 1, $\log_2$ 1 is 0, $\log_2$ 0.5 is -1)
  - Minimum (0.0) when all records belong to one class, implying most information (0 $\log_2$ 0 is taken as 0)
  - Higher entropy ➔ higher uncertainty **Hw: Why do we need the negative sign?**
  - Lower entropy ➔ lower uncertainly
- Entropy based computations are similar to the GINI index computations

51

Comparison among Splitting Criteria

For a 2-class problem:



52

Examples for computing Entropy

$$Entropy \quad (t) = -\sum_j p(j \mid t) \log_2 p(j \mid t)$$

C1	0
C2	6

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1

Entropy = – 0 log 0 – 1 log 1 = – 0 – 0 = 0

C1	1
C2	5

P(C1) = 1/6 P(C2) = 5/6

0 log 0 is taken as 0

Entropy = – (1/6) $\log_2$ (1/6) – (5/6) $\log_2$ (5/6)

= -.166*.-2.59 - .833 * -0.26 = 0.646

C1	2
C2	4

P(C1) = 2/6 P(C2) = 4/6

Entropy = – (2/6) $\log_2$ (2/6) – (4/6) $\log_2$ (4/6) = 0.92

C1	3
C2	3

P(C1) = 3/6 P(C2) = 3/6

Entropy = – (3/6) $\log_2$ (3/6) * (3/6) $\log_2$ (3/6) = (-0.5*-1) + (– 0.5*-1) = 1.0

53

Why is that 0 log 0 = 0 ?

$$\lim_{x \to 0} x \log_2(x) = \lim_{x \to 0} \frac{ln(x)}{ln(2)} \frac{}{x^{-1}} = \lim_{x \to 0} \frac{x^{-1}}{ln(2)} \frac{}{-x^{-2}} = \lim_{x \to 0} \frac{-x}{ln(2)} = 0$$

L'Hospital's Rule (Wikipedia)

If

$\lim_{x \to c} f(x) = \lim_{x \to c} g(x) = 0$ or $\pm\infty$, and

$\lim_{x \to c} \frac{f'(x)}{g'(x)}$ exists, and

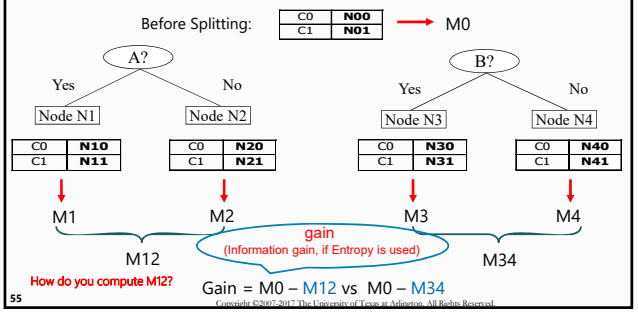
$g'(x) \neq 0$ for all x in I with $x \neq c$,

then

$$\lim_{x \to c} \frac{f(x)}{g(x)} = \lim_{x \to c} \frac{f'(x)}{g'(x)}.$$

54

How to Find the Best Split



55

Splitting Based on INFO...

Information Gain:

$$GAIN_{split} = Entropy \quad (p) - \left(\sum_{i=1}^{k} \frac{n_i}{n} Entropy \quad (i) \right)$$

Parent Node, p is split into k partitions;

n_i is number of records in partition i

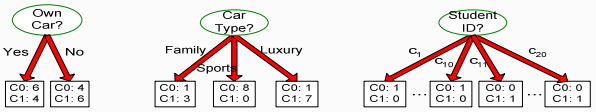
- Measures Reduction in Entropy achieved because of the split. Choose the split that achieves most reduction (maximizes GAIN)
- Used in ID3 and C4.5
- Disadvantage: Tends to prefer splits that result in large number of partitions, each being small but pure.

56

Limitations of entropy and GINI

Consider the following splits:

Before Splitting: 10 records of class 0,
10 records of class 1



Both tend to favor qualitative attributes with large number of distinct values each being small and pure

Student ID has more homogeneous partitions! But car type is a better split!

How can we overcome this limitation? Having a low impurity value alone is not sufficient!

57

Classification: preamble

- Having a large number of child nodes can make a decision tree more complex and consequently more susceptible to overfitting
- Hence the number of child nodes produced by a splitting need to be taken into account as well
- This can be done in two ways:
  - Generate only binary trees (e.g., CART)
  - Modify splitting criterion to take into account the number of partitions produced by the attributes
  - A measure known as gain ratio was introduced by C4.5 decision tree algorithm

58

Splitting Based on INFO...

Gain Ratio:

$$GainRATIO_{split} = \frac{GAIN_{Split}}{SplitINFO} \qquad SplitINFO = -\sum_{i=1}^{k} \frac{n_i}{n} \log \frac{n_i}{n}$$

Parent Node, p is split into k partitions
n_i is the number of records in partition i

- Instead of using GAIN split, use GainRATIO split for choosing a split!
- Adjusts Information Gain by the entropy of the partitioning (SplitINFO). Higher entropy partitioning (large number of small partitions) is penalized!
- Used in C4.5
- Designed to overcome the disadvantage of Information Gain
- Larger the number of child nodes means larger the denominator; will reduce the GainRATIO

Hw: modify the formula to apply this to GINI

59

3. Splitting Criteria based on Classification Error

- Classification error at a node t :

$$Error(t) = 1 - \max_i P(i \mid t)$$

- Measures misclassification error made by a node.
  - Maximum $(1 - 1/n_c)$ when records are equally distributed among all classes, implying least interesting information
  - Minimum (0.0) when all records belong to one class, implying most interesting information

60

Examples for Computing Error

$$Error\ (t) = 1 - \max_i P(i \mid t)$$

C1	**0**
C2	**6**

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1

Error = 1 – max (0/6, 6/6) = max (0, 1) = 1 – 1 = 0

C1	**1**
C2	**5**

P(C1) = 1/6 P(C2) = 5/6

Error = 1 – max (1/6, 5/6) = 1 – 5/6 = 1/6 = 0.167

C1	**2**
C2	**4**

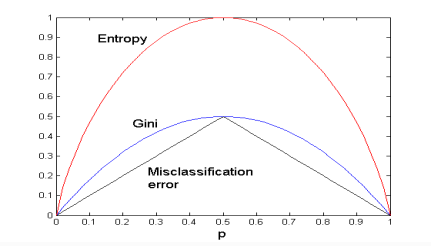
P(C1) = 2/6 P(C2) = 4/6

Error = 1 – max (2/6, 4/6) = 1 – 4/6 = 1/3 = 0.33

61

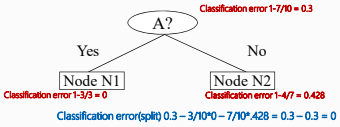
Comparison among Splitting Criteria

For a 2-class problem:



62

Misclassification Error vs. Gini



Classification error(split) 0.3 – 3/10*0 – 7/10*.428 = 0.3 – 0.3 = 0

The split would stop! Why does GINI not stop!

	Parent
C1	7
C2	3
Gini = 0.42	

Gini(N1)
= 1 – (3/3)² – (0/3)²
= 0

Gini(N2)
= 1 – (4/7)² – (3/7)²
= 0.489

	N1	N2
C1	3	4
C2	0	3
Gini=0.342		

Gini(Children)
= 3/10 * 0
+ 7/10 * 0.489
= 0.342

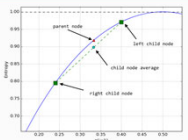
Gini improves !!

Q: try with entropy

63

Misclassification error vs. gini

- In contrast to the average classification error, the average child node GINI is **not** equal to the Gini of the parent node. Thus, the splitting rule would continue until the child nodes are pure.
- So, why is this happening?
- The green square-shapes are the gini values for p(3/0) and (4/7) of the first two child nodes in the decision tree model above, connected by a green (dashed) line. To recapitulate: the decision tree algorithm aims to find the feature and splitting value that leads to a maximum decrease of the average child node impurities over the parent node. So, if we have 2 gini values (left and right child node), the average will fall onto the straight, connecting line. However – **and this is the important part –** we can see that the Gini is always larger than the averaged Gini due to its "bell shape," which is why we keep continuing to split the nodes in contrast to the misclassification error!



64

Tree Induction

- Greedy strategy.
  - Split the records based on an attribute test that optimizes certain criterion.
- Issues
  - Determine how to split the records
    - How to specify the attribute test condition?
    - How to determine the best split?
  - Determine when to stop splitting

65

Stopping Criteria for Tree Induction

- Stop expanding a node when all the records belong to the same class
- Stop expanding a node when all the records have similar attribute values
  - What to do? majority voting
- Early termination, e.g., when the information gain is below a threshold.

66

Decision Tree Based Classification

- Advantages:
  - Inexpensive to construct
  - Extremely fast at classifying unknown records
  - Easy to interpret for small-sized trees
  - Accuracy is comparable to other classification techniques for many data sets

67

Example: C4.5

- Simple depth-first construction.
- Uses Information Gain
- Sorts Continuous Attributes at each node.
- Needs entire data to fit in memory.
- Unsuitable for Large Datasets.
  - Needs out-of-core sorting.
- You can download the software from: http://www.cse.unsw.edu.au/~quinlan/c4.5r8.tar.gz

68

Decision trees: final thoughts

- One of the very early classification techniques
- Overfitting: intuitively, this occurs when the model fits the data too well
  - Tree size is larger than needed!
  - This results in lower accuracy for the unseen data set!
  - Splits, cross validation can be used to choose a model
  - Pruning the tree is another approach. Pruning reduces specialization!
    - Again, need some quantitative way to do this!
- Underfitting: intuitively, this occurs when the model does not fit the data well enough
  - Tree size is smaller than needed!
  - Does not generalize to new data
  - Can be detected using a good performance metric
- Ideally, want a model at the sweet spot between under- and over-fitting!

69

Decision trees: greedy vs. optimal

- Complexity of generating a single decision tree is exponential (in the worst case) in the number of attributes!
  - That is the number of decision making nodes!
- For global optimization, you have to generate or explore an exponential number of such trees!
  - Even then it will only be optimal for the training data
  - Does not guarantee the same for unseen data!
- Generating a globally optimal decision tree Complexity is proportional to $2^N * 2^N$ where N is the number of attributes

70

Thank You !!!

For more information visit:
http://itlab.uta.edu