THE UNIVERSITY OF MELBOURNE

COMP20008 Elements of Data Processing

Classification Methodologies

Announcements

- Project marking
  - We expect to release marks + feedback for Phase 1 by Thursday 13th April

- Phase 2A (Concept formulation and preliminary investigation): Due 25th April
  - In workshops next week (10-13th April) – Half of the time will be devoted to discussion regarding Phase 2A of the project

- Phase 2B (peer feedback): Due 28th April
  - In lecture on Monday 24th April, we will discuss strategies for giving peer feedback

Plan

- Introduction to classification
  - Decision tree classification
  - k nearest neighbor classification (on Monday)

- Predicting disease from microarray data

	Gene 1	Gene 2	Gene 3	…	Gene n	Cancer
Person 1	2.3	1.1	0.3	…	2.1	1
Person 2	3.2	0.2	1.2	…	1.1	1
Person 3	1.9	3.8	2.7	…	0.2	0
…	…	…	…	…	…	…
Person m	2.8	3.1	2.5	…	3.4	0

Test data

	Gene 1	Gene 2	Gene 3	…	Gene n	Cancer
Person m+1	2.1	0.9	0.6	…	1.9	?

Classification example

- Animal classification

Name	Body Temperature	Skin Cover	Gives Birth	Aquatic Creature	Aerial Creature	Has Legs	Hiber-nates	Class Label
human	warm-blooded	hair	yes	no	no	yes	no	mammal
python	cold-blooded	scales	no	no	no	no	yes	reptile
salmon	cold-blooded	scales	no	yes	no	no	no	fish
whale	warm-blooded	hair	yes	yes	no	no	no	mammal
frog	cold-blooded	none	no	semi	no	yes	yes	amphibian
komodo dragon	cold-blooded	scales	no	no	no	yes	no	reptile
bat	warm-blooded	hair	yes	no	yes	yes	yes	mammal
pigeon	warm-blooded	feathers	no	no	yes	yes	no	bird
cat	warm-blooded	fur	yes	no	no	yes	no	mammal
leopard shark	cold-blooded	scales	yes	yes	no	no	no	fish
turtle	cold-blooded	scales	no	semi	no	yes	no	reptile
penguin	warm-blooded	feathers	no	semi	no	yes	no	bird
porcupine	warm-blooded	quills	yes	no	no	yes	yes	mammal
eel	cold-blooded	scales	no	yes	no	no	no	fish
salamander	cold-blooded	none	no	semi	no	yes	yes	amphibian

Test data

Name	Body Temperature	Skin Cover	Gives Birth	Aquatic Creature	Aerial Creature	Has Legs	Hiber-nates	Class Label
gila monster	cold-blooded	scales	no	no	no	yes	yes	?

https://www-users.cs.umn.edu/~kumar/dmbook/ch4.pdf

Classification example

- Banking: classifying borrower

binary categorical continuous class

Tid	Home Owner	Marital Status	Annual Income	Defaulted Borrower
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

Training set for predicting borrowers who will default on loan payments.

Test data

Tid	Home Owner	Marital status	Annual Income	Defaulted Borrower
11	No	Single	55K	?

Classification example

- Detecting tax fraud

Tid	Refund	Marital Status	Taxable Income	Cheat
	categorical	categorical	continuous	class
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

Test data

Tid	Refund	Marital Status	Taxable Income	Cheat
11	Yes	Married	125K	?

Classification: Definition

- Given a collection of records (*training set*)
  – Each record contains a set of *attributes*, one of the attributes is the *class*.
- Find a *model* for class attribute as a function of the values of other attributes.

$$y = f(x_1, x_2, \ldots, x_n)$$

  – y: discrete value, target variable
  – $x_{1,} \ldots x_n$: attributes, predictors

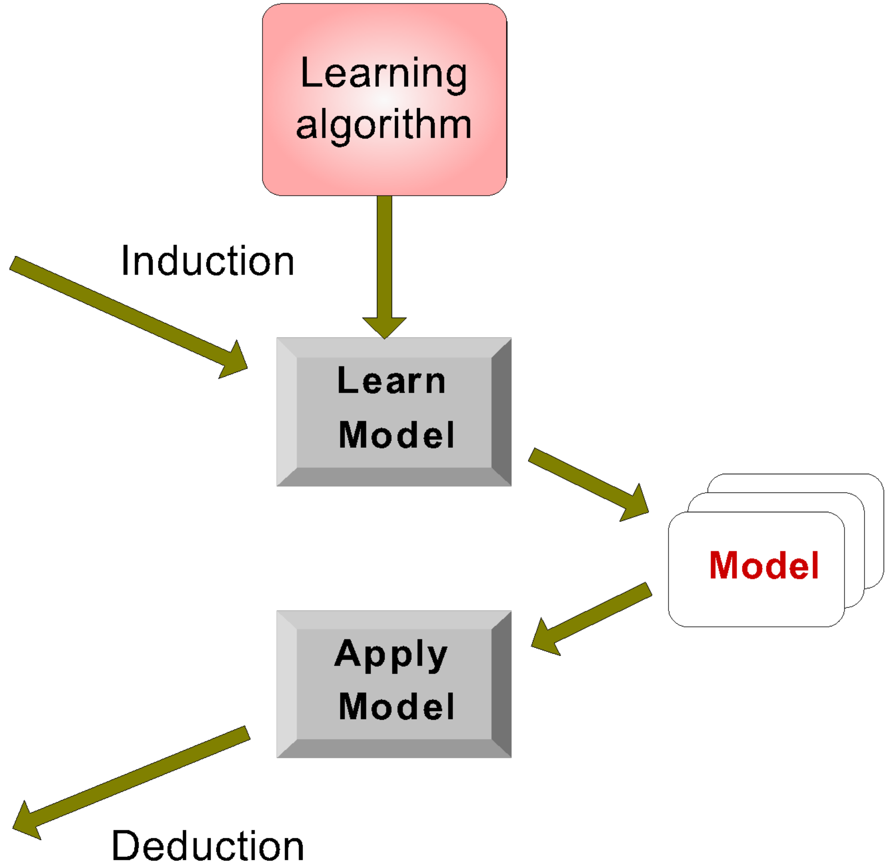
- Goal: previously unseen records should be assigned a class as accurately as possible.

  – 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.

Classification framework

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set



Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

Regression

- Given a collection of records (*training set*)
  - Each record contains a set of *attributes*, one of the attributes is the *target variable*.

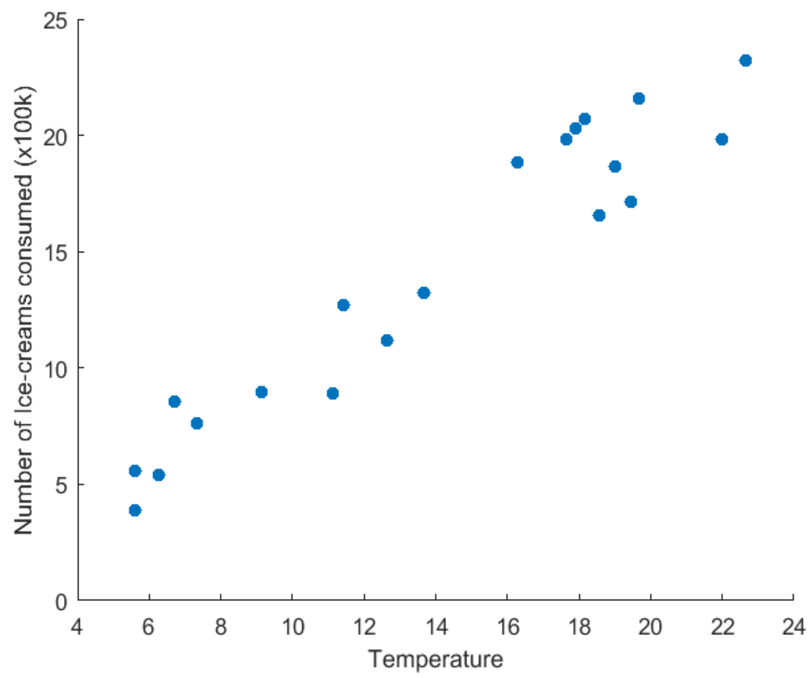
- Learn predictive model from data

$$y = f(x_1, x_2, \ldots, x_n)$$

- y: continuous real value, target variable
- $x_1, \ldots x_n$: attributes, predictors

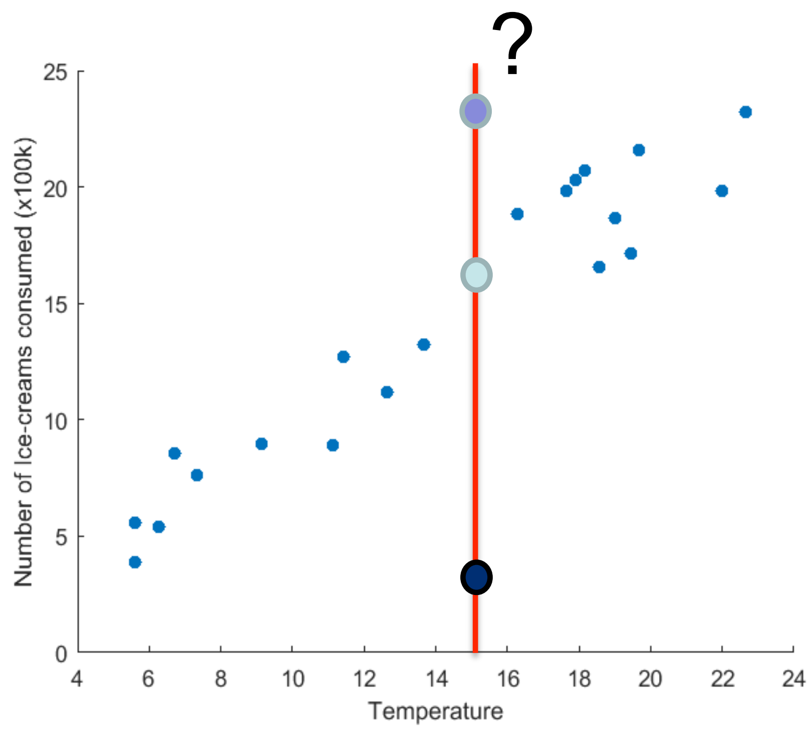
Regression

- Predicting ice-creams consumption from temperature: $y = f(x)$



Regression

- Predicting ice-creams consumption from temperature: $y = f(x)$



Regression

- Predicting activity level of a target gene

	Gene 1	Gene 2	Gene 3	…	Gene n
Person 1	2.3	1.1	0.3	…	2.1
Person 2	3.2	0.2	1.2	…	1.1
Person 3	1.9	3.8	2.7	…	0.2
…	…	…	…	…	…

Gene n+1
3.2
1.1
0.2
…
0.9

	Gene 1	Gene 2	Gene 3	…	Gene n
Person m+1	2.1	0.9	0.6	…	1.9

Gene n+1
?

Classification and Regression

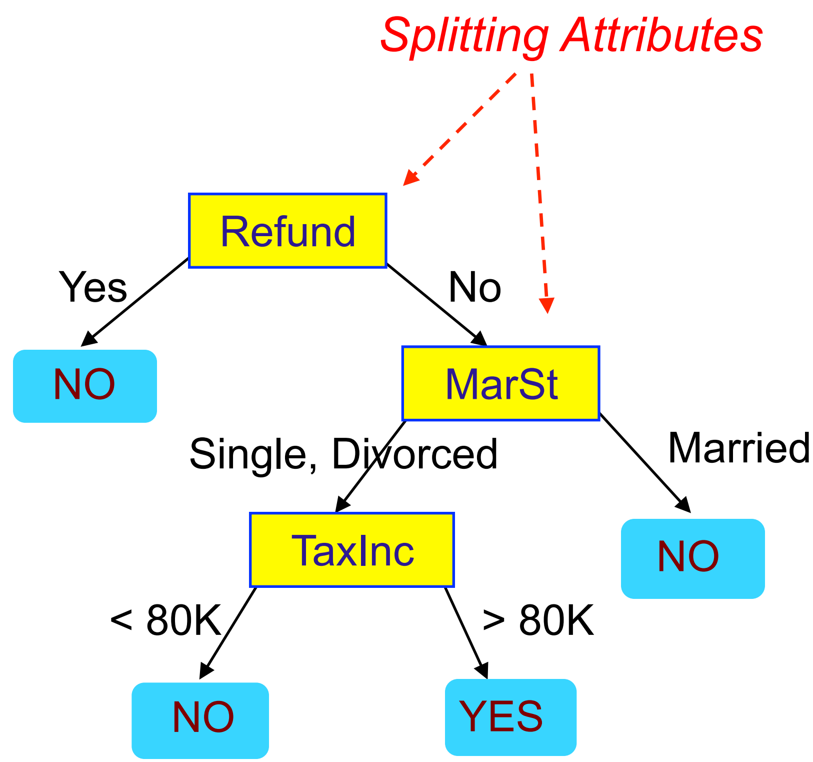
- What is Classification and Regression?
- Classification algorithms:
  - Decision tree (today)
  - K-Nearest Neighbor Classifier (K-NN) (tomorrow)

Example of a Decision Tree

categorical categorical continuous class

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**

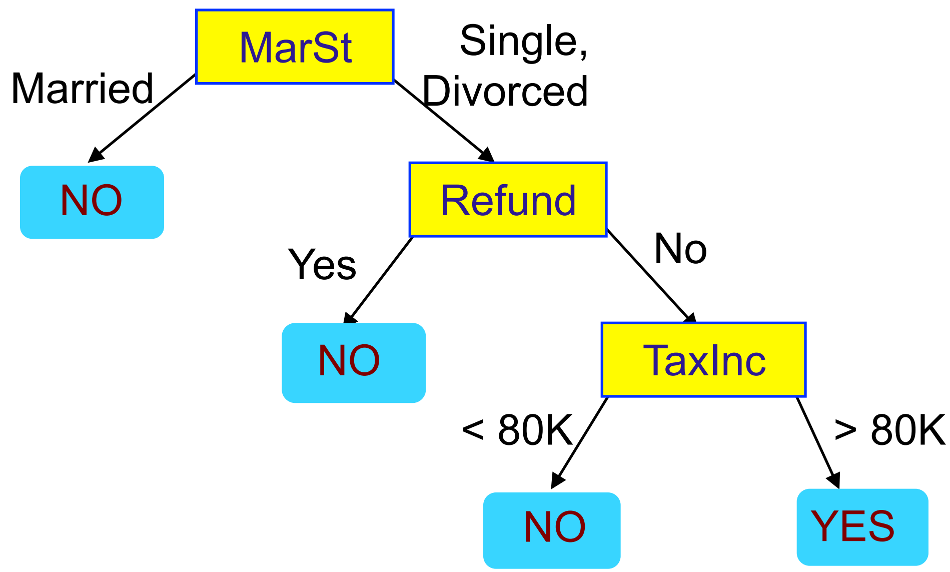
Training Data



Model: Decision Tree

Another Example of Decision Tree

Tid	Refund (categorical)	Marital Status (categorical)	Taxable Income (continuous)	Cheat (class)
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**



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

THE UNIVERSITY OF MELBOURNE

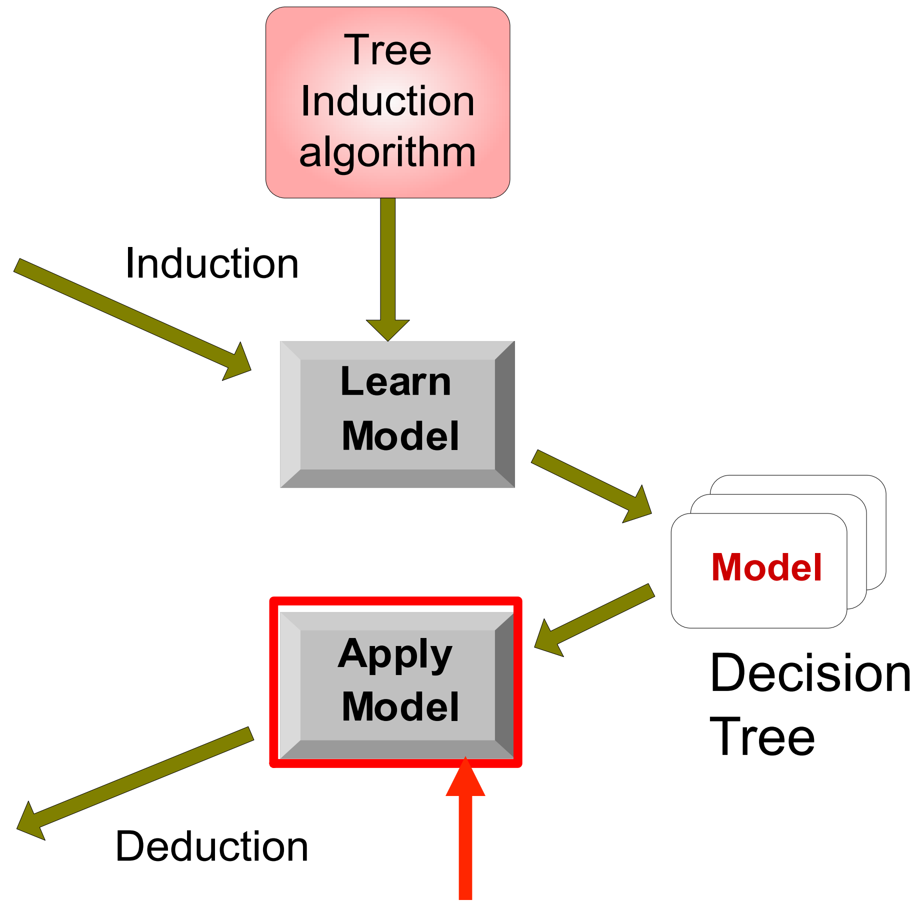
Decision Tree Classification Task

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

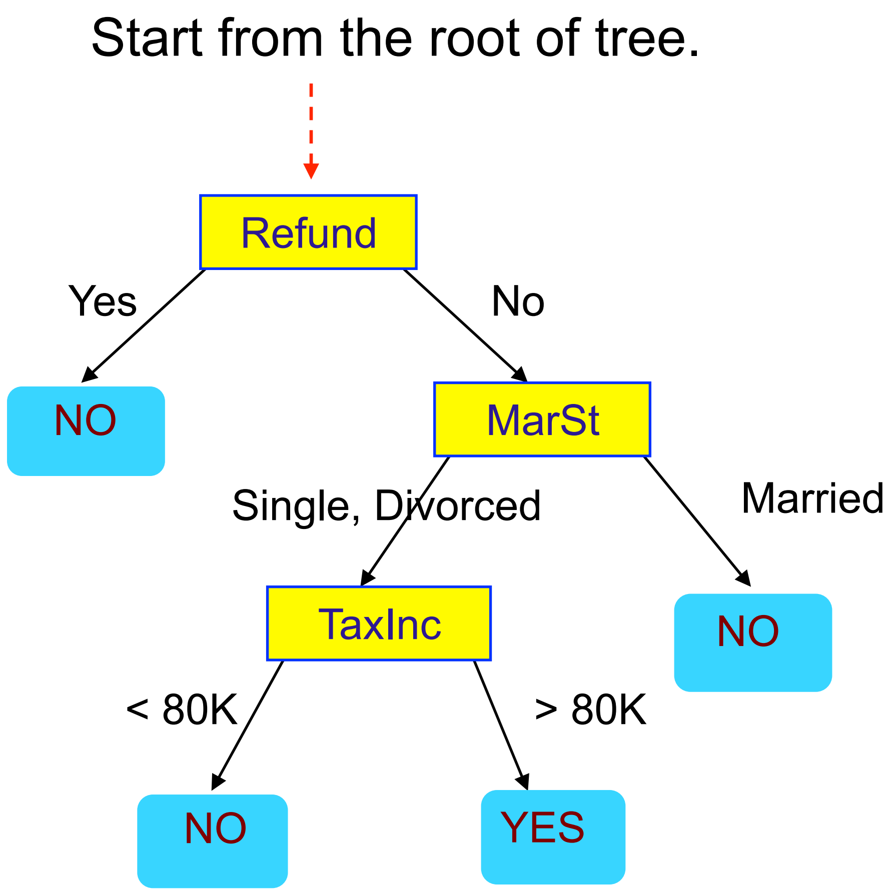
Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set



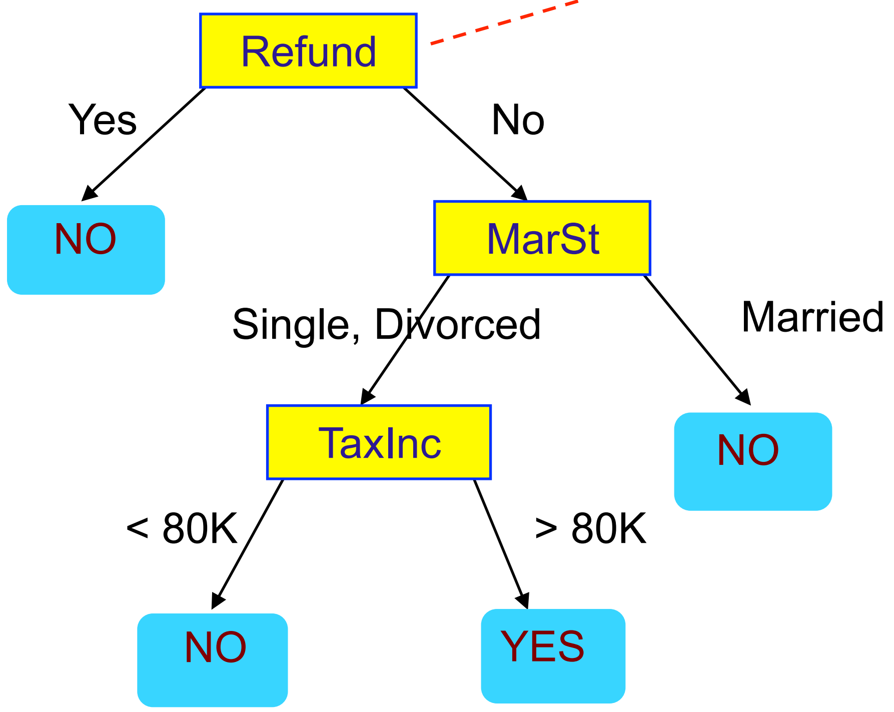
Apply Model to Test Data



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

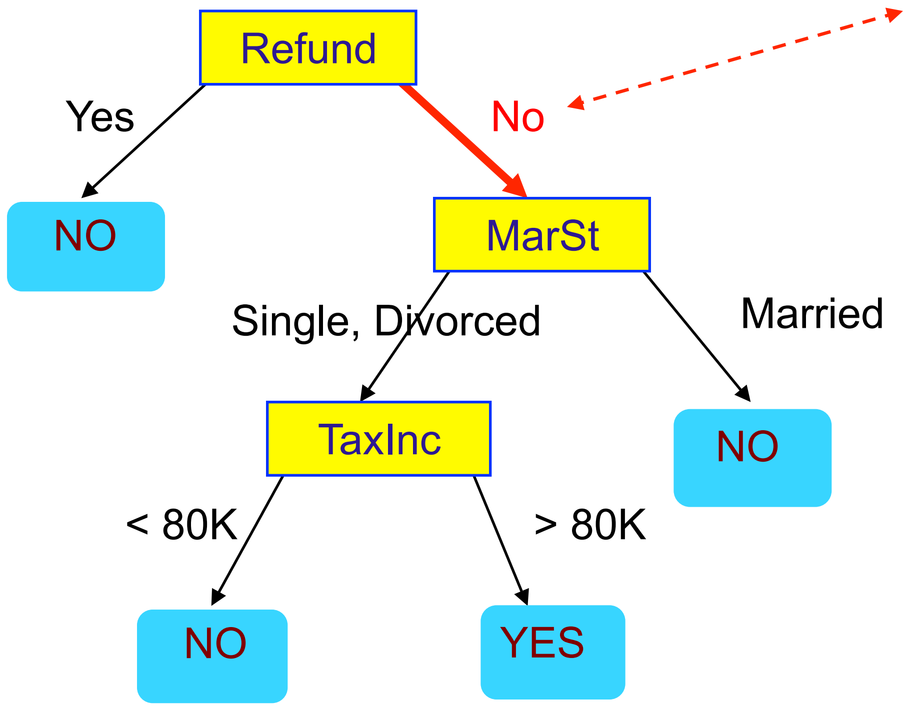
Apply Model to Test Data



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

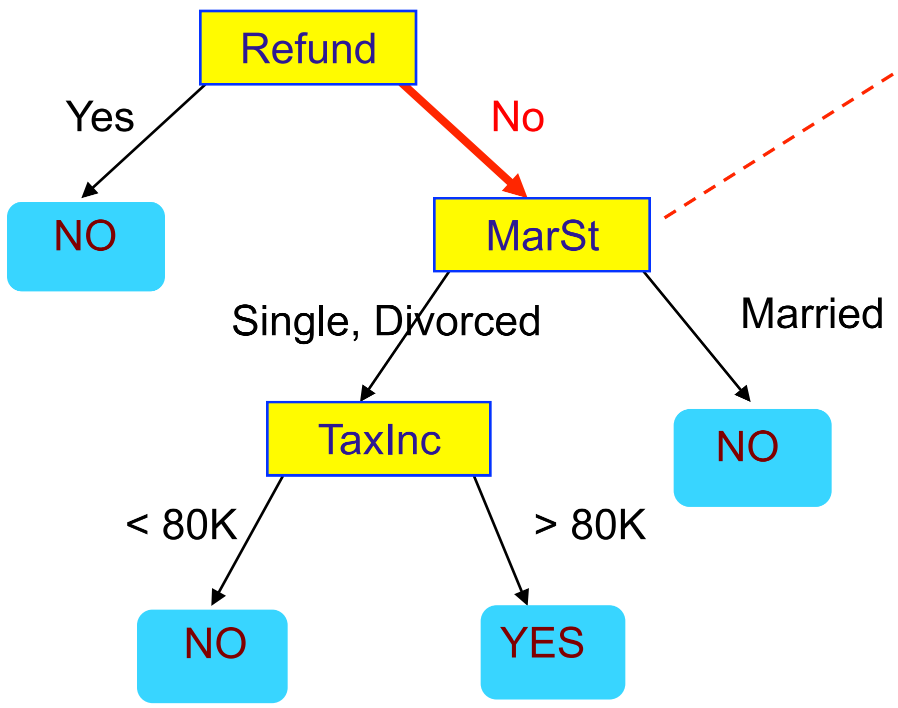
Apply Model to Test Data



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

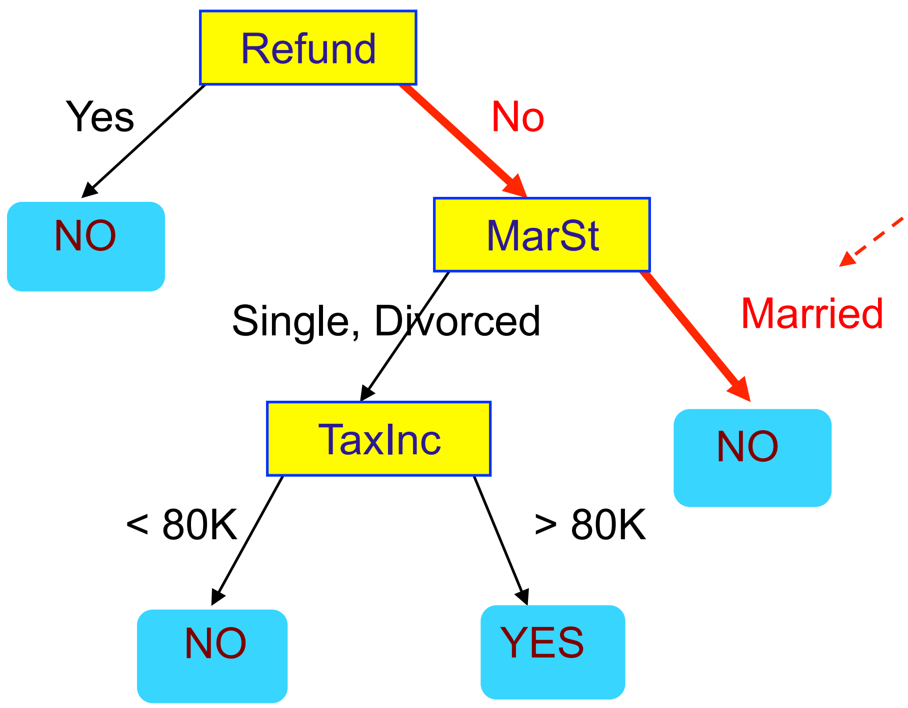
Apply Model to Test Data



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

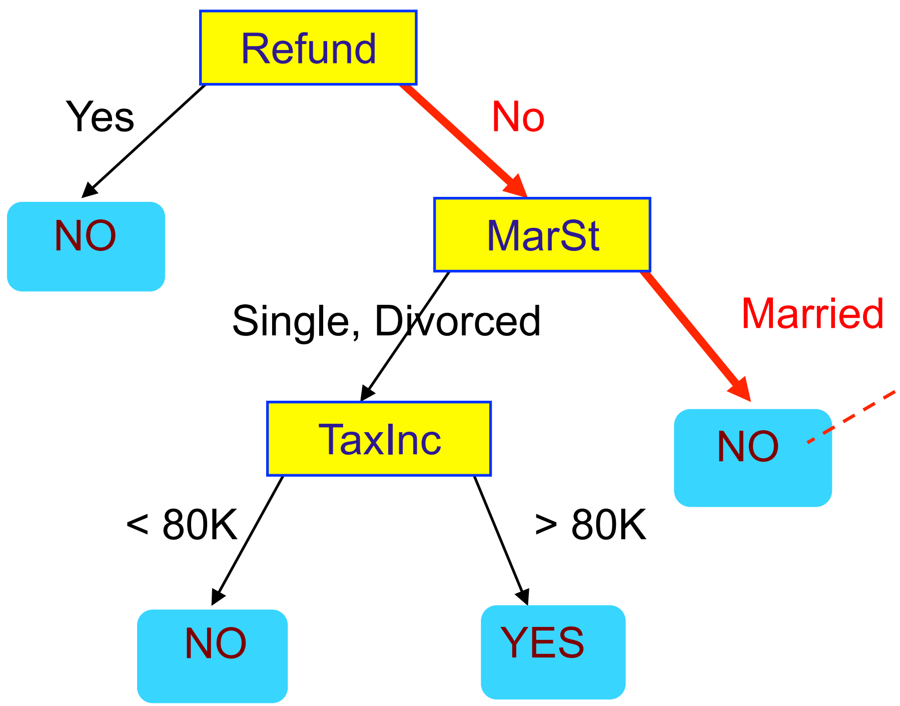
Apply Model to Test Data



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Apply Model to Test Data



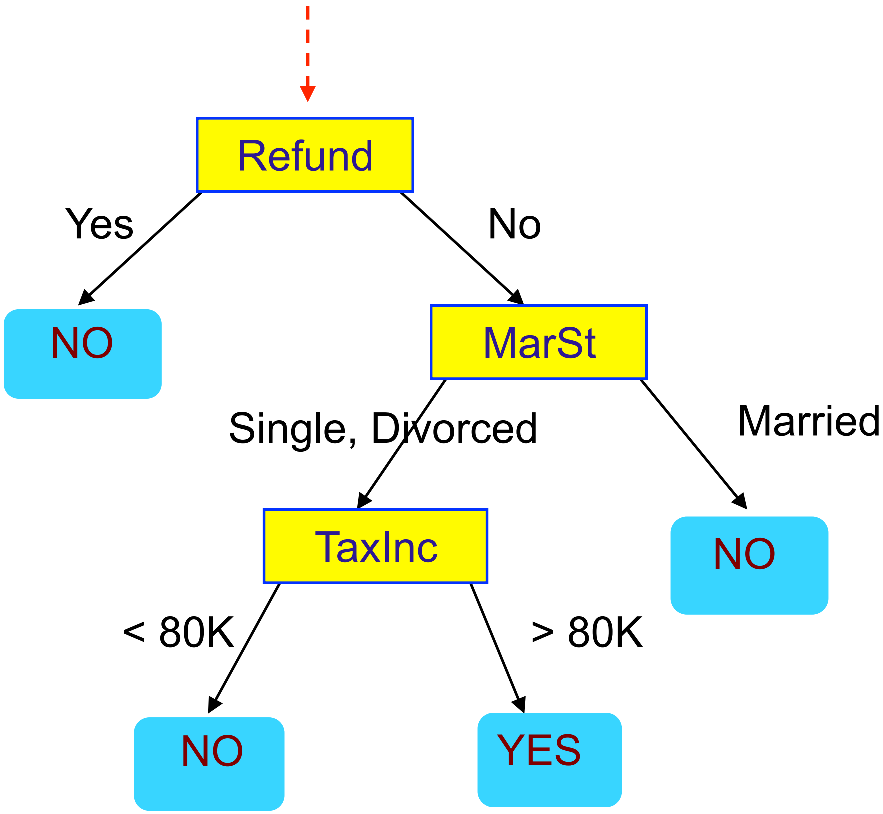
Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Assign Cheat to “No”

Apply Model to Test Data

Start from the root of tree.



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Single	100K	?

Decision Trees

- Decision tree
  - A flow-chart-like tree structure
  - Internal node denotes a test on an attribute
  - Branch represents an outcome of the test
  - Leaf nodes represent class labels or class distribution

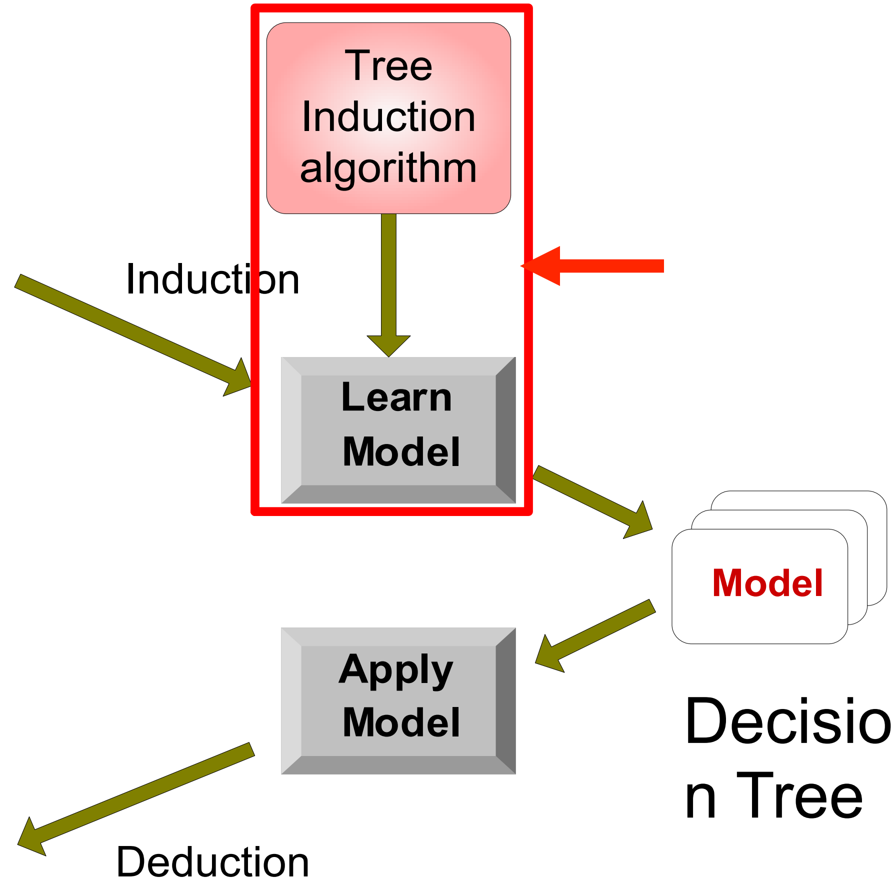
Decision Tree Classification Task

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set



Decision Tree

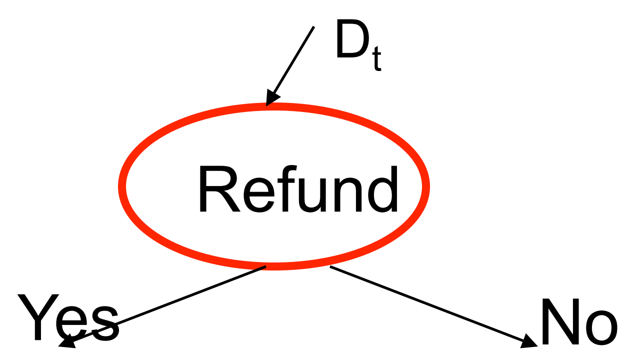
Decision Tree Creation

- Many Algorithms:
  - We will look at a representative one (Hunt's algorithm)

General Structure of Hunt's Algorithm

- Let D_t be the set of training records that reach a node t
- General Procedure:
  - If D_t contains records that belong the same class y_t, then t is a leaf node labeled as y_t
  - If D_t is an empty set, then t is a leaf node labeled by the default class, y_d
  - If D_t contains records that belong to more than one class, use an attribute test to split the data into smaller subsets. Recursively apply the procedure to each subset.

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



Example

If D_t contains records that belong to more than one class, use an attribute test to split the data into smaller subsets. Recursively apply the procedure to each subset.

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

D_t

Refund

Yes

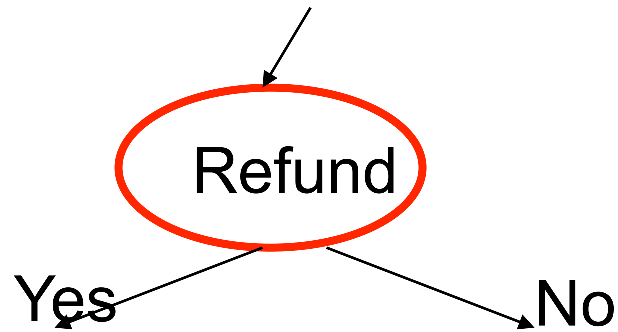
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
4	Yes	Married	120K	No
7	Yes	Divorced	220K	No

No

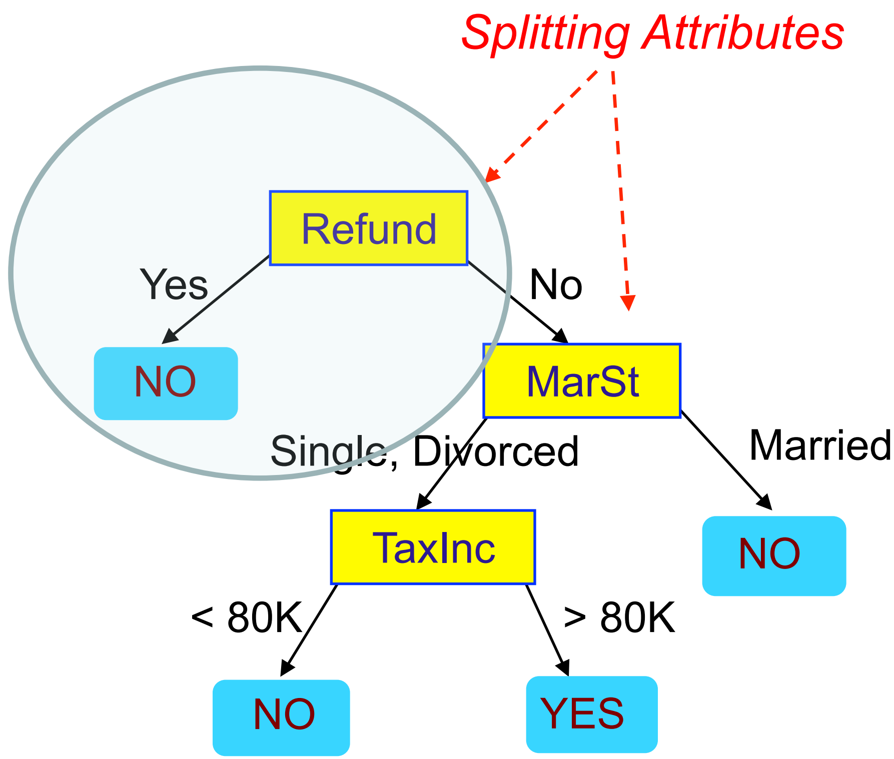
Tid	Refund	Marital Status	Taxable Income	Cheat
2	No	Married	100K	No
3	No	Single	70K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Stopping condition: leaf node

- If D_t contains records that belong the same class y_t, then t is a leaf node labeled as y_t
- If D_t is an empty set, then t is a leaf node labeled by the default class, y_d



Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
4	Yes	Married	120K	No
7	Yes	Divorced	220K	No



Model: Decision Tree

Tree Induction (creation)

- 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 node has only a single class of instances

Tree Induction

- 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

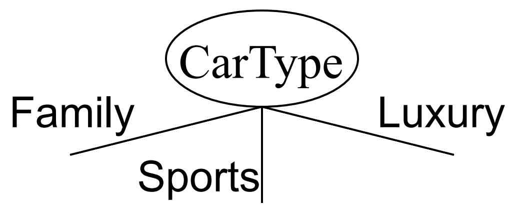
How to Specify Test Condition?

- Depends on attribute types
  - Nominal
  - Ordinal
  - Continuous

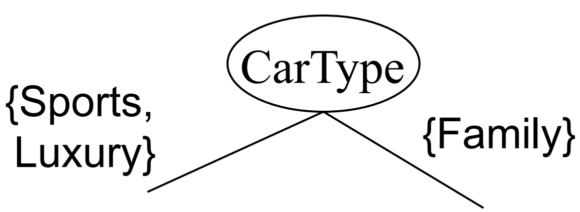
- Depends on number of ways to split
  - 2-way split
  - Multi-way split

Splitting Based on Nominal Attributes

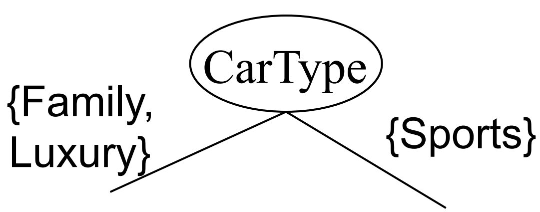
- Multi-way split: Use as many partitions as distinct values.



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

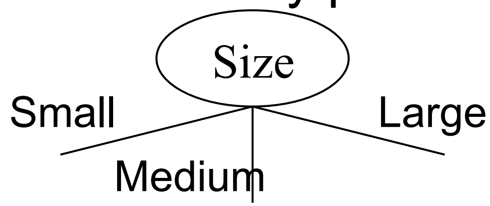


OR

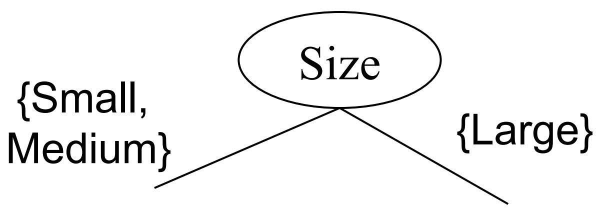


Splitting Based on Ordinal Attributes

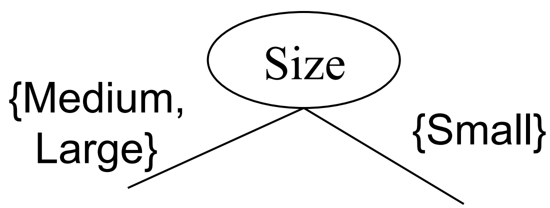
- Multi-way split: Use as many partitions as distinct values.



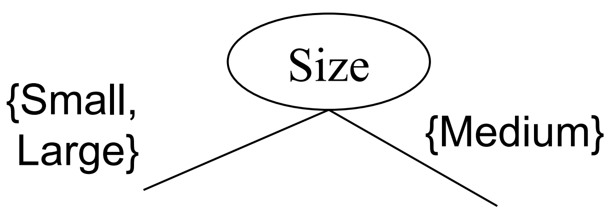
- Binary split: Divides values into two subsets.
  Need to find optimal partitioning.



OR



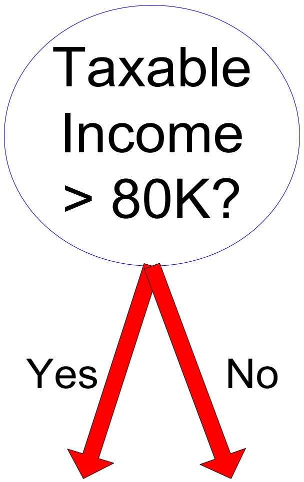
- What about this split?



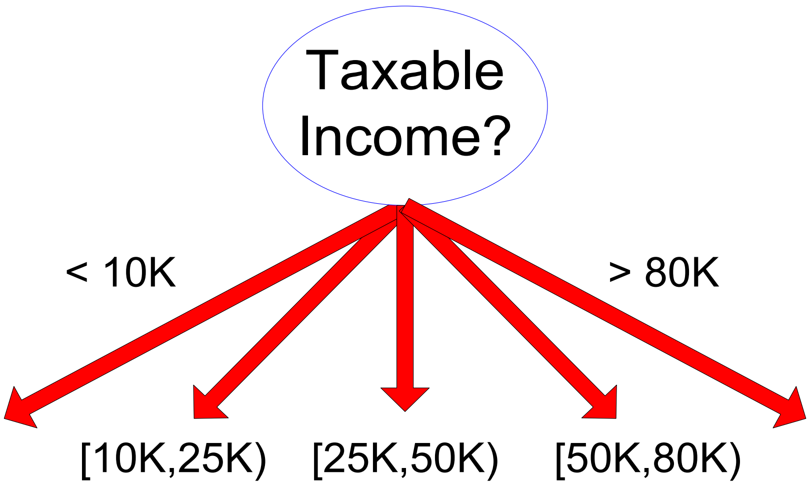
Splitting Based on Continuous Attributes

- 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 (percentiles), or clustering.
  - Binary Decision: $(A < v)$ or $(A \geq v)$
    - consider all possible splits and finds the best cut
    - can be more compute intensive

Splitting Based on Continuous Attributes



(i) Binary split



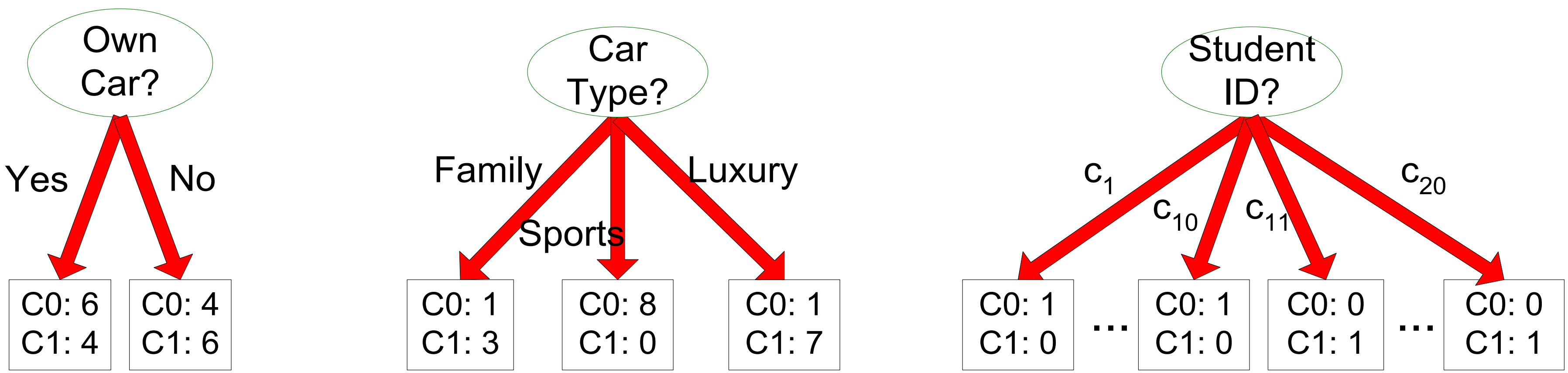
(ii) Multi-way split

Tree Induction

- 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

How to determine the Best Split

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



Which test condition is the best?

How to determine the Best Split

- Greedy approach:
  - 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

Measures of Node Impurity

- Entropy
  - We have seen entropy in the feature correlation section, where it was used to measure the amount of uncertainty in an outcome
  - *Entropy can also be viewed as an impurity measure*
    - The set {A,B,C,A,A,A,A,A} has low entropy: low uncertainty and **high purity**
    - The set {A,B,C,D,B,E,A,F} has high entropy: high uncertainty and **low purity**

Node Impurity Criteria based on Entropy

- Entropy (H) at a given node t:

$$H(t) = -\sum_{j} p(j \mid t) \log p(j \mid t)$$

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

- Measures homogeneity of a node.
  - Maximum ($\log n_c$) when records are equally distributed among all classes
  - Minimum (0.0) when all records belong to one class

Examples for computing Entropy

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

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

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

Entropy = – 0 $\log_2$ 0 – 1 $\log_2$ 1 = – 0 – 0 = 0

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

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

Entropy = – (1/6) $\log_2$ (1/6) – (5/6) $\log_2$ (1/6) = 0.65

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

?

Examples for computing Entropy

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

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

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

Entropy = – 0) $\log_2$ 0 – 1) $\log_2$ 1 = – 0 – 0 = 0

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

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

Entropy = – (1/6) $\log_2$ (1/6) – (5/6) $\log_2$ (1/6) = 0.65

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

How good is a Split?

- Compare the impurity (entropy) of parent node (before splitting)
- With the impurity (entropy) of the children nodes (after splitting)

$$
\begin{aligned}
Gain &= H(Parent) - H(Parent|Child) \\
&= H(Parent) - \sum_{j=1}^{k} \frac{N(v_j)}{N} H(v_j)
\end{aligned}
$$

- $H(v_j)$: impurity measure of node v_j
- j: children node index
- $N(v_j)$: number of data points in child node v_j
- N: number of data points in parent node
- The larger the gain, the better

THE UNIVERSITY OF MELBOURNE

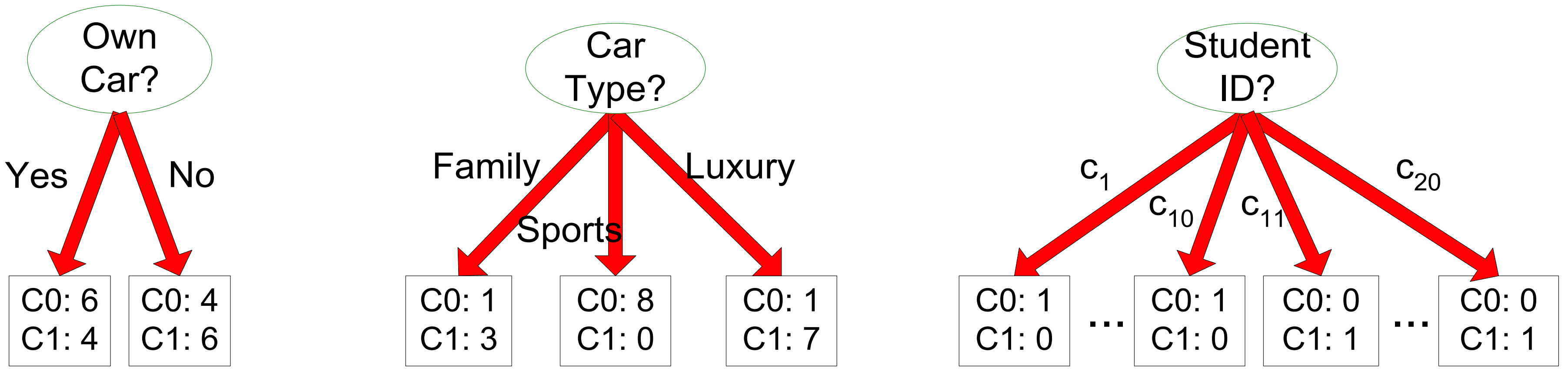
How good is a Split?

$$\begin{aligned} Gain = & \quad H(Parent) - H(Parent|Child) \\ = & \quad H(Parent) - \sum_{j=1}^{k} \frac{N(v_j)}{N} H(v_j) \end{aligned}$$

- Note: the information gain is equivalent to the mutual information between the class feature and the feature being split on
- Thus splitting using the information gain is to choose the feature with highest information shared with the class variable

How to determine the Best Split?

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



Which test condition is the best?

- Compute the gain of all splits
- Choose the one with largest gain

Question 2c) from 2016 exam

Given a dataset with two classes, A and B, suppose the root node of a decision tree has 50 instances of class A and 150 instances of class B. Consider a candidate split of this root node into two children, the first with (25 class A and 25 class B), the second with (25 class A and 125 class B). Write a formula to measure the utility of this split using the entropy criterion. Explain how this formula helps measure split utility

Points to remember from this lecture

- Understand what is meant by the terms classification and regression and why it is useful to build models for these tasks
- Understand how a decision tree may be used to make predictions about the class of a test instance
- Understand the key steps in building a decision tree
  - How to split the instances, how to specify the attribute test condition, how to determine the best split and how to decide when to stop splitting
- Understand the use of entropy as a node impurity measure for decision tree node splitting. Understand the benefits of entropy for this task and why it is effective for assessing the goodness of a split

References and Acknowledgement

This lecture was prepared using some material adapted from:

- https://www-users.cs.umn.edu/~kumar/dmbook/ch4.pdf
- CS059 - Data Mining -- Slides
- http://www-users.cs.umn.edu/~kumar/dmbook/dmslides/chap4_basic_classification.ppt