Introduction to Artificial Intelligence

Acknowledgement

- The slides of this lecture have been taken from the lecture slides of CS307 – “Introduction to Artificial Intelligence” and CSE652 – “Knowledge Discovery and Data mining” by Dr. Sajjad Haider.

Course Outline

- Overview of Artificial Intelligence ✔
- State Space Representation ✔
- Search Techniques ✔
- AI in Adversarial Games ✔
- **Machine Learning**
- Propositional and Predicate Logic
- Probabilistic Reasoning
- Introduction to Robots
- Computer Vision
- Natural Language Processing
- Reinforcement Learning

Machine Learning

- As a broad subfield of artificial intelligence, machine learning is concerned with the **design and development of algorithms and techniques that allow computers to "learn".**
- A major focus of machine learning research is to automatically learn to recognize complex patterns and make intelligent decisions based on data.

Types of Learning

- Supervised Learning
  - Classification
  - Regression
- Unsupervised Learning
- Reinforcement Learning

* The focus in this course in on Classification techniques

Popular Machine Learning Techniques

- Classification
  - Classification Trees
  - Naïve Bayes
  - Artificial Neural Networks
- Clustering
  - K-Means
- Reinforcement Learning*

Classification Example

Venue	Type of Wicket	Type of match	Batted first	Winning Team
Pakistan	Slow	ODI	Pakistan	Pakistan
India	Fast	Test	Pakistan	Pakistan
India	Slow	ODI	India	India
Pakistan	Slow	ODI	Pakistan	India
Third country	Fast	ODI	India	Pakistan
India	Fast	ODI	India	India
Pakistan	Fast	Test	India	Pakistan
Third country	Fast	Test	Pakistan	India
Third country	Slow	Test	India	Pakistan
Third country	Slow	ODI	Pakistan	Pakistan
Pakistan	Fast	ODI	Pakistan	India
Third country	Slow	Test	Pakistan	Pakistan
Pakistan	Fast	ODI	India	Pakistan
Third country	Fast	Test	Pakistan	India
India	**Slow**	**ODI**	**Pakistan**	**???**

Classification

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

Venue	Type of Wicket	Type of match	Batted first	Winning Team
Pakistan	Slow	ODI	Pakistan	Pakistan
India	Fast	Test	Pakistan	Pakistan
India	Slow	ODI	India	India
Pakistan	Slow	ODI	Pakistan	India
Third country	Fast	ODI	India	Pakistan
India	Fast	ODI	India	India
Pakistan	Fast	Test	India	Pakistan
Third country	Fast	Test	Pakistan	India
Third country	Slow	Test	India	Pakistan
Third country	Slow	ODI	Pakistan	Pakistan
Pakistan	Fast	ODI	Pakistan	India
Third country	Slow	Test	Pakistan	Pakistan
Pakistan	Fast	ODI	India	Pakistan
Third country	Fast	Test	Pakistan	India
India	**Slow**	**ODI**	**Pakistan**	**???**

Classification

- Goal: <u>previously unseen</u> 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.

Application of Classification

- E-mail Classification (Spam vs. Inbox)
- Intrusion Detection
- Credit Scoring
  - Loan Defaulter
  - Fraud Detection
- Biometric Identification
  - Fingerprinting
  - Handwriting
  - Speech Recognition
- Search Engines

An Example application

- A credit card company typically receives thousands of applications for new cards. The application contains information regarding several different attributes, such as annual salary, any outstanding debts, age etc. The problem is to categorize applications into those who have good credit, bad credit, or fall into a gray area (thus requiring further human analysis).

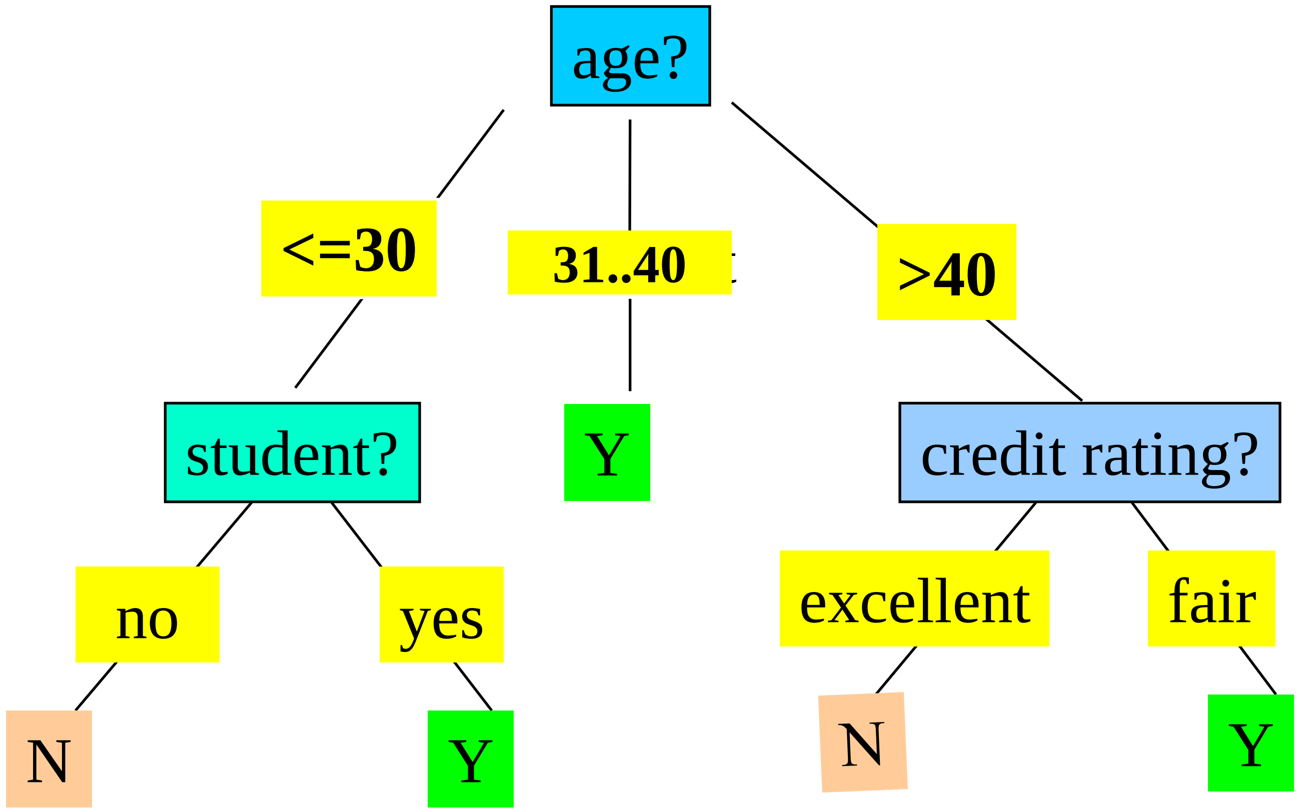
Another Application

- An emergency room in a hospital measures 17 variables (e.g., blood pressure, age, etc) of newly admitted patients. A decision has to be taken whether to put the patient in an intensive-care unit. Due to the high cost of ICU, those patients who may survive less than a month are given higher priority. The problem is to predict high-risk patients and discriminate them from low-risk patients.

Classification Example

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

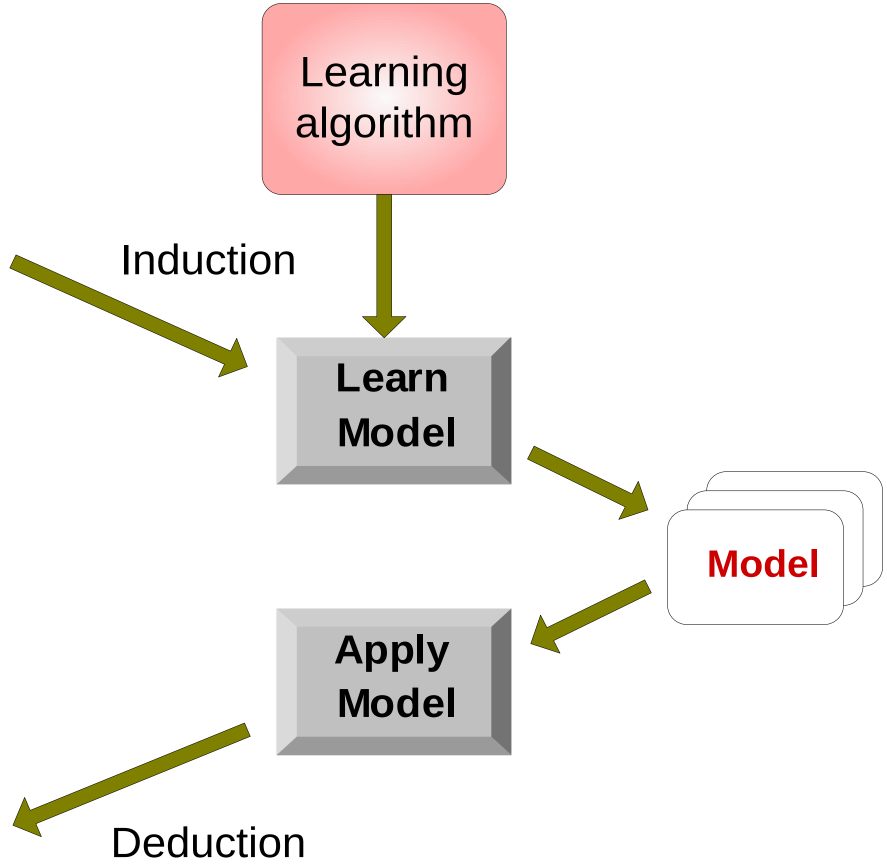
Classification Tree



Illustrating 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



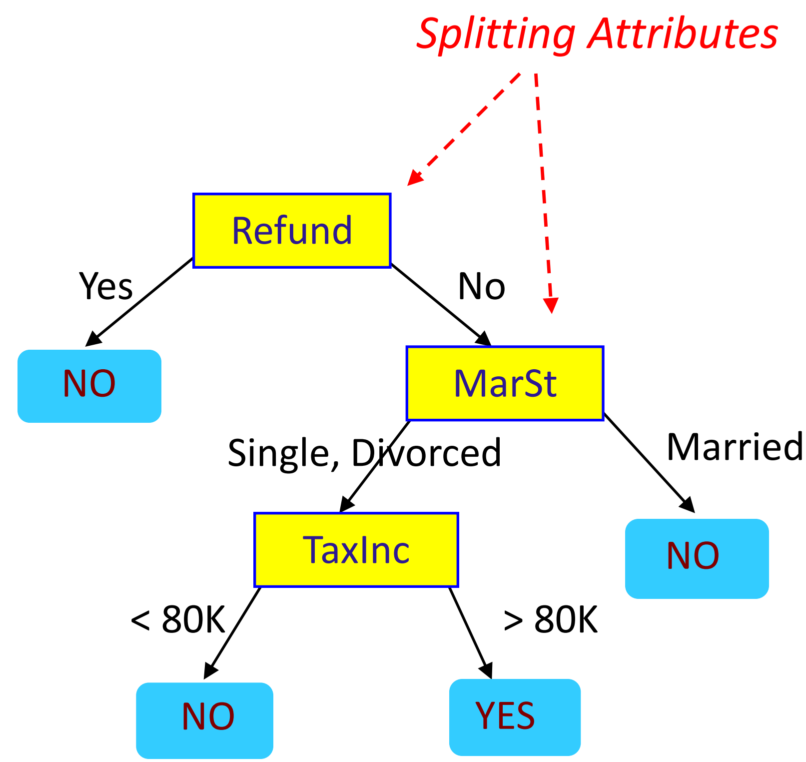
Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

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



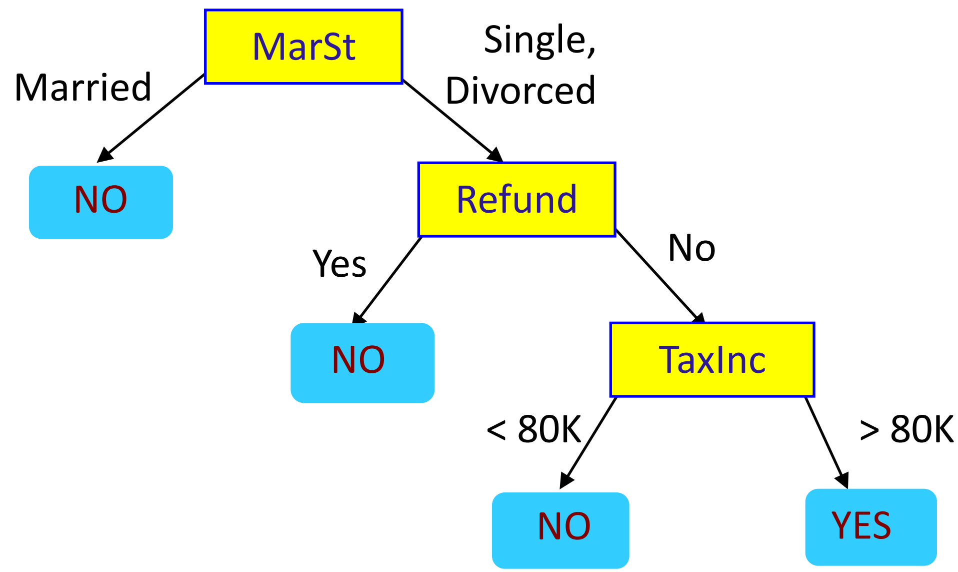
Training Data

Model: Decision Tree

Another Example of Decision Tree

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



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

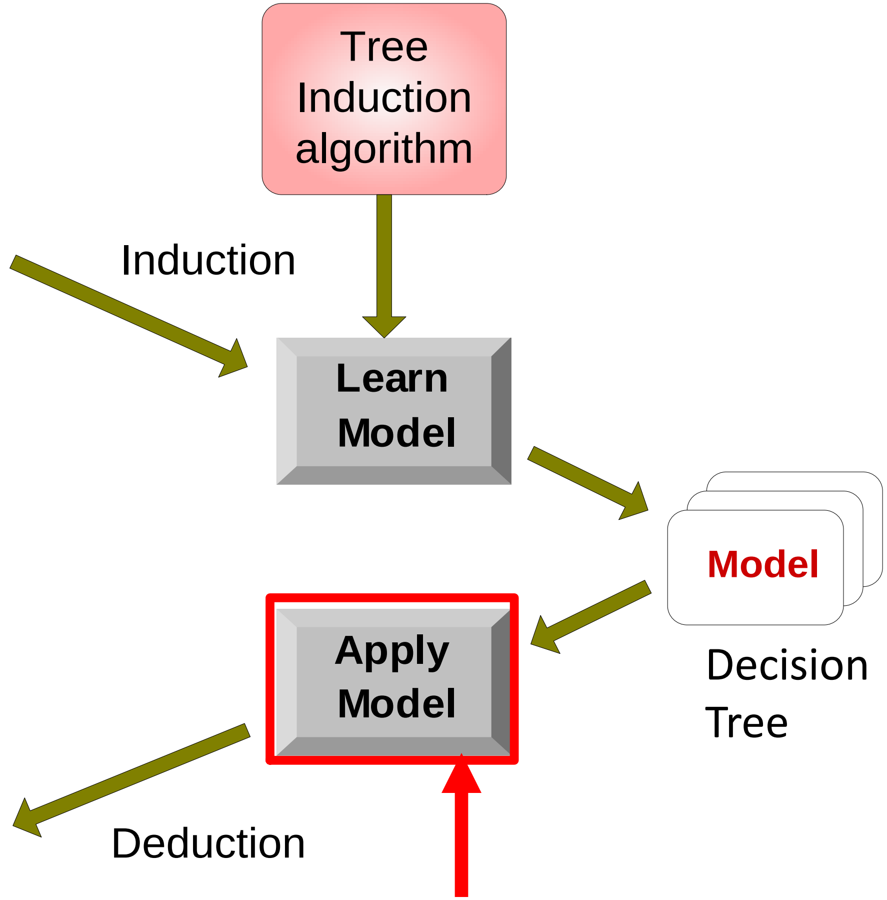
Decision Tree Classification Task

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

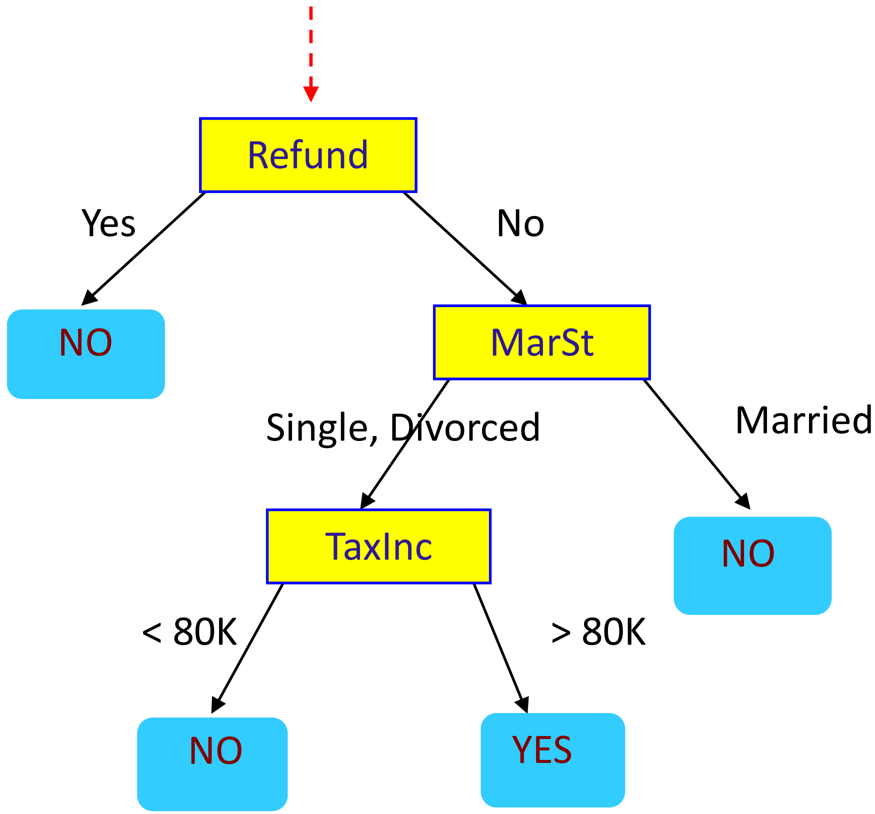
Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set



Apply Model to Test Data

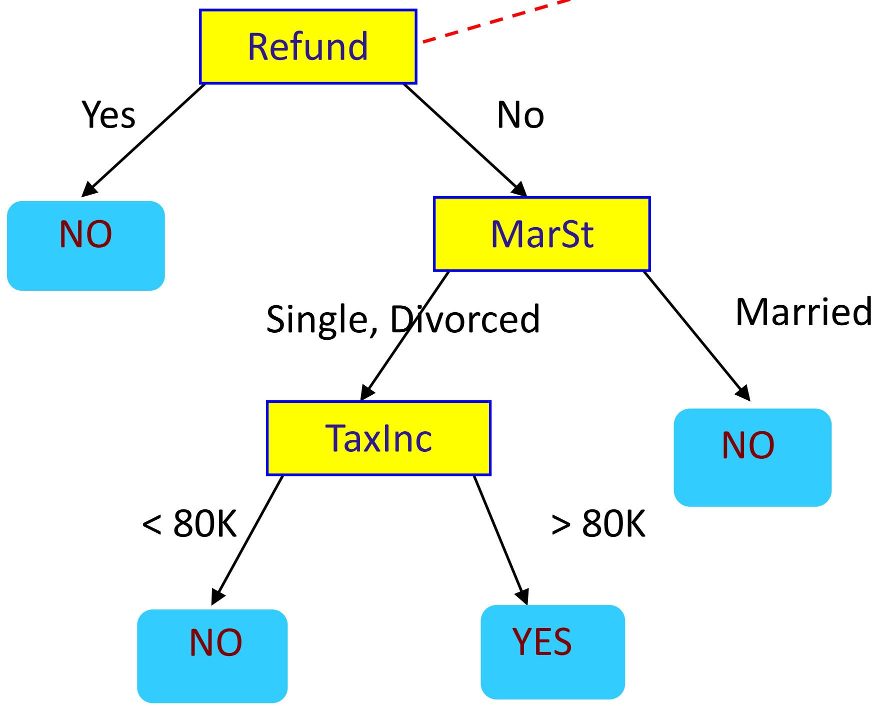
Start from the root of tree.



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Apply Model to Test Data



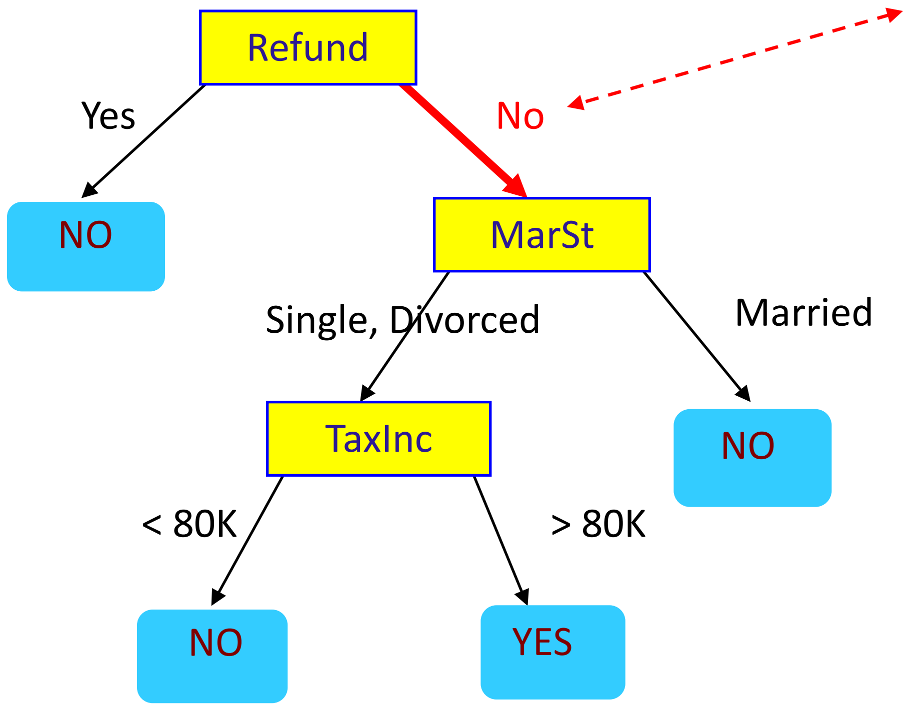
Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

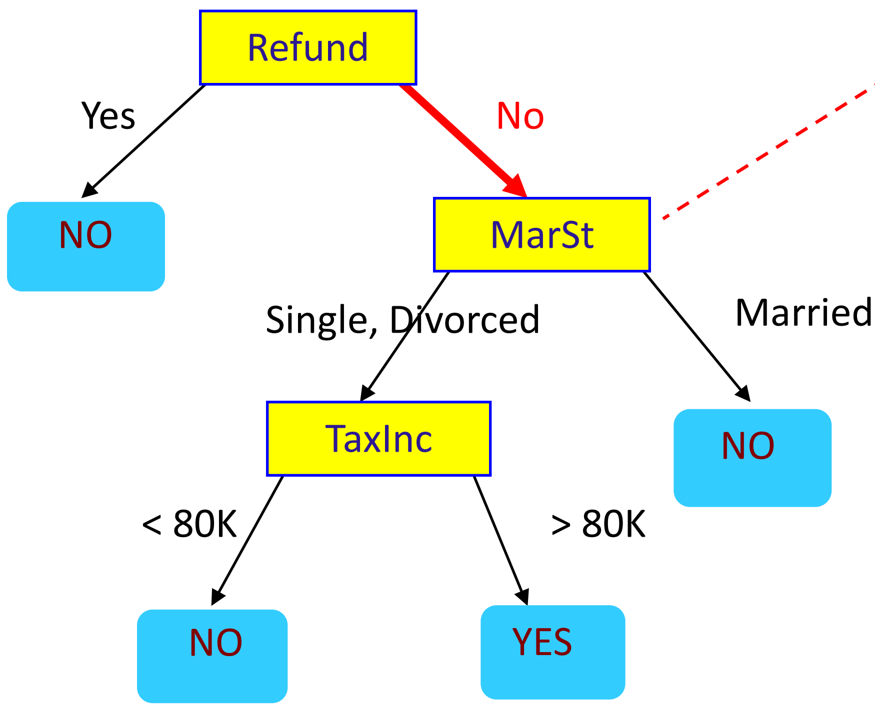
Apply Model to Test Data

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



Apply Model to Test Data



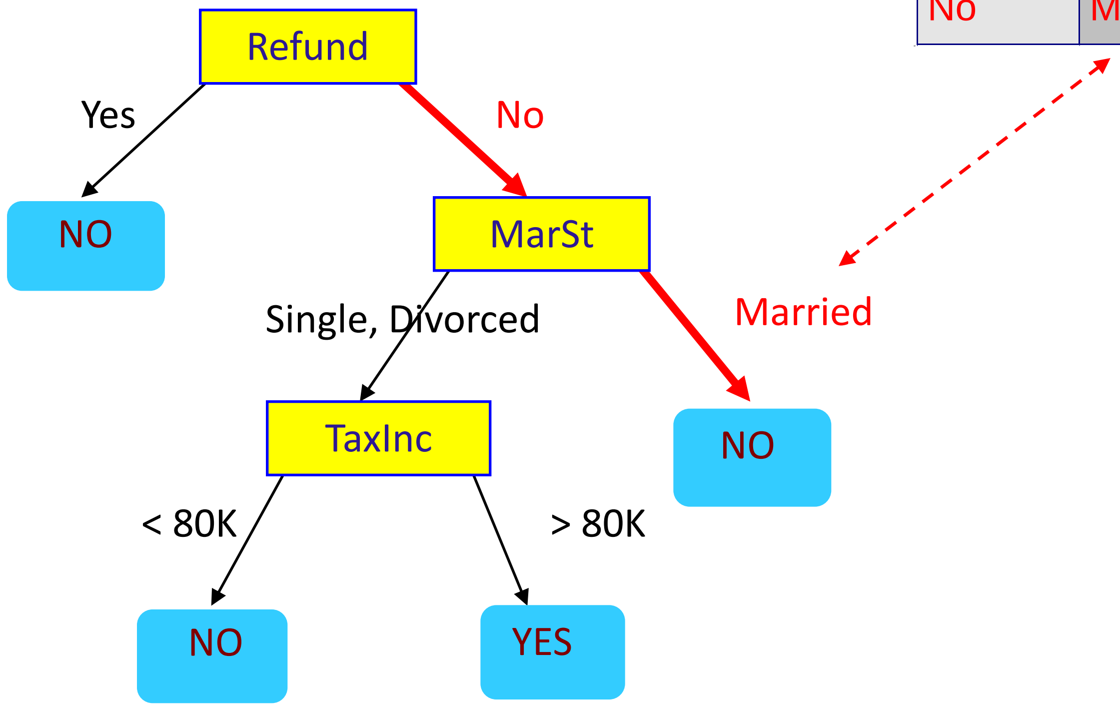
Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Apply Model to Test Data

Test Data

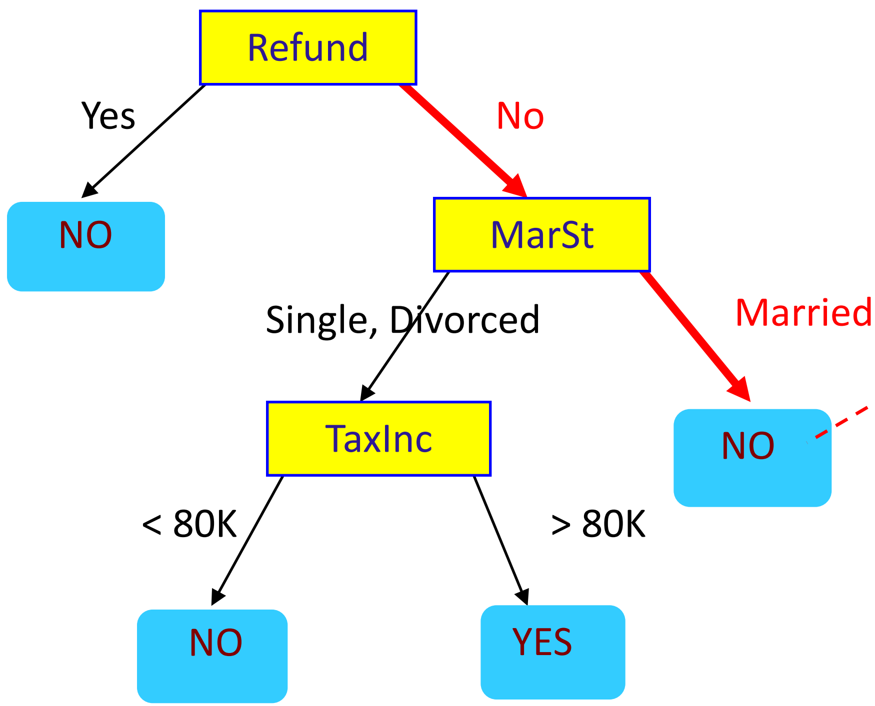
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



Apply Model to Test Data

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

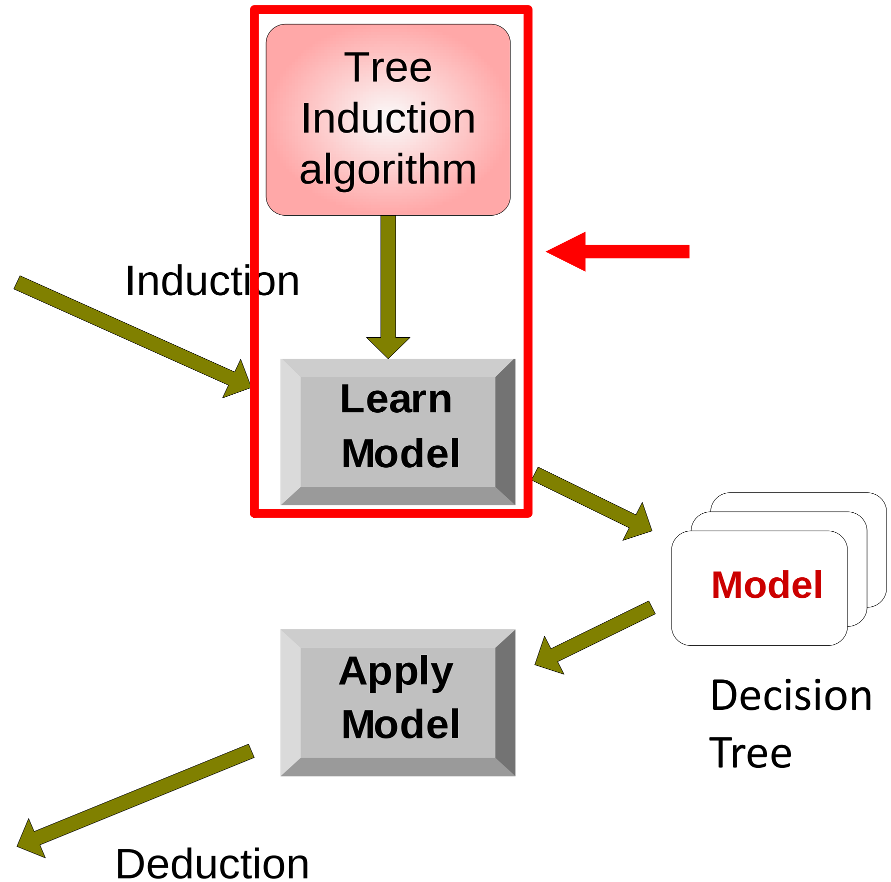


Assign Cheat to “No”

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

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

How to Specify Test Condition?

- Depends on attribute types
  - Categorical
  - Numeric
    - Discrete
    - Continuous

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

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

Measures of Node Impurity

- Gini Index

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

- Entropy

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

Measure of Impurity: GINI

- Gini Index for a given node t :

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

(NOTE: *p(j | t)* is the relative frequency of class j at node t).

  - 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

C1	**0**
C2	**6**
Gini=0.000	

C1	**1**
C2	**5**
Gini=0.278	

C1	**2**
C2	**4**
Gini=0.444	

C1	**3**
C2	**3**
Gini=0.500	

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

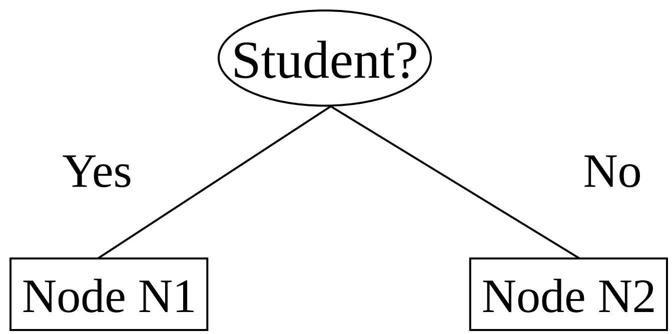
Gini = 1 – (2/6)^2 – (4/6)^2 = 0.444

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Binary Attributes: Computing GINI Index

- Splits into two partitions
- Effect of Weighing partitions:
  - Purer Partitions (having lower GINI index) are sought for.

	Buy Computer
Yes	**9**
No	**5**
Gini = 0.46	



Gini(N1)
$= 1 - (6/7)^2 - (1/7)^2$
$= 0.24$

Gini(N2)
$= 1 - (3/7)^2 - (4/7)^2$
$= 0.49$

	student yes	student no
Yes	**6**	**3**
No	**1**	**4**
Gini=0.365		

Gini(Student)
$= 7/14 * 0.24 +$
$7/14 * 0.49$
= ??

GINI Index for Buy Computer Example

- Gini (Income):
- Gini (Credit_Rating):
- Gini (Age):

Measure of Impurity: Entropy

- Entropy at a given node t:

$$\boxed{Entropy(t) = -\sum_j p(j\,|\,t)\log p(j\,|\,t)}$$

(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 implying least information
    - Minimum (0.0) when all records belong to one class, implying most information

  – Entropy based computations are similar to the GINI index computations

Entropy in a nut-shell

Low Entropy

High Entropy

Examples for computing Entropy

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

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

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

Inducing a decision tree

- There are many possible trees
- How to find the most compact one
  - that is consistent with the data?
- The *key* to building a decision tree - which attribute to choose in order to branch.
- The *heuristic* is to choose the attribute with the minimum GINI/Entropy.

Algorithm for Decision Tree Induction

- Basic algorithm (a greedy algorithm)
  - Tree is constructed in a top-down recursive manner
  - At start, all the training examples are at the root
  - Attributes are categorical
  - Examples are partitioned recursively based on selected attributes
  - Test attributes are selected on the basis of a heuristic or statistical measure (e.g., GINI/Entropy)
- Conditions for stopping partitioning
  - All examples 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 examples left

Extracting Classification Rules from Trees

- Represent the knowledge in the form of IF-THEN rules
- One rule is created for each path from the root to a leaf
- Each attribute-value pair along a path forms a conjunction. The leaf node holds the class prediction
- Rules are easier for humans to understand
- Example

IF *age* = “<=30” AND *student* = “*no*” THEN *buys_computer* = “*no*”

IF *age* = “<=30” AND *student* = “*yes*” THEN *buys_computer* = “*yes*”

IF *age* = “31…40” THEN *buys_computer* = “*yes*”

IF *age* = “>40” AND *credit_rating* = “*excellent*” THEN *buys_computer* = “*yes*”

IF *age* = “<=30” AND *credit_rating* = “*fair*” THEN *buys_computer* = “*no*”

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 simple data sets