Classification

Decision Trees

Huiping Cao

Examples of a 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

Refund: categorical

Marital Status: categorical

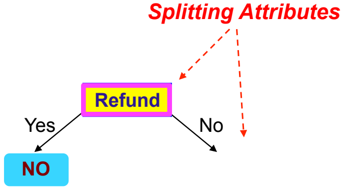
Taxable Income: continuous

Cheat: class

Examples of a Decision Tree (cont.)

	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

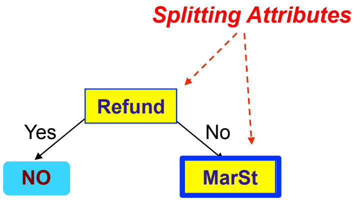


Model: Decision Tree

Examples of a Decision Tree (cont.)

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



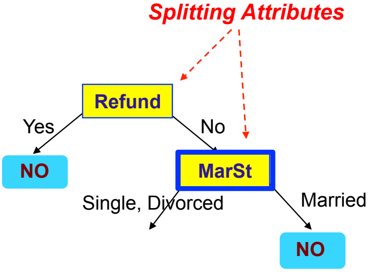
Model: Decision Tree

Examples of a Decision Tree (cont.)

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



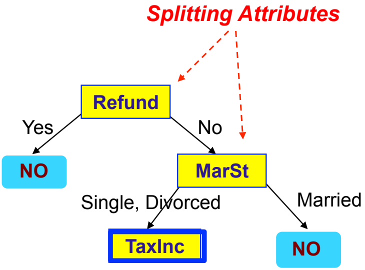
Model: Decision Tree

Examples of a Decision Tree (cont.)

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



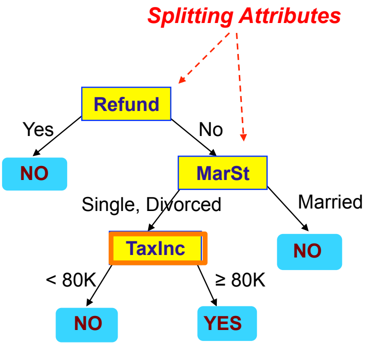
Model: Decision Tree

Examples of a Decision Tree (cont.)

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



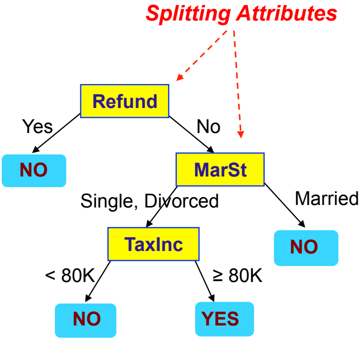
Model: Decision Tree

Examples of a Decision Tree (cont.)

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

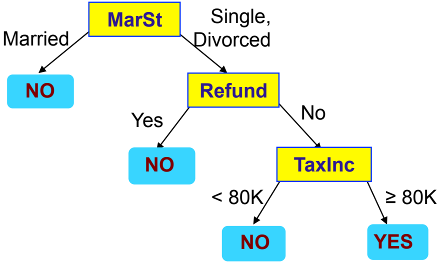
Training Data



Model: Decision Tree

Examples of a Decision Tree (cont.)

	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**

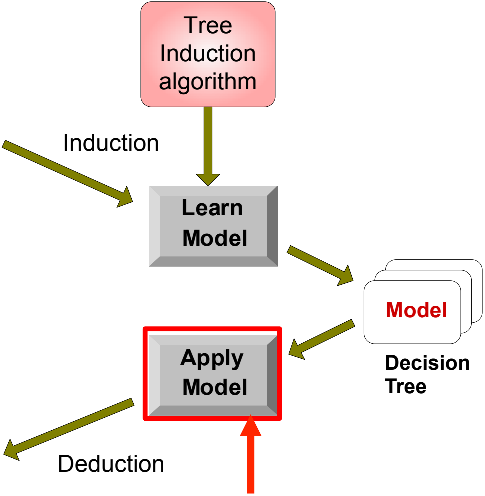


There could be more than one tree that fits the same data!

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

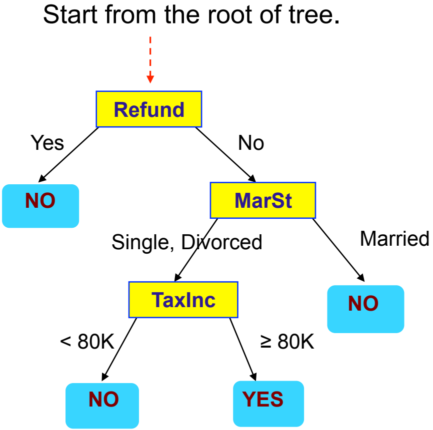
Training Set



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

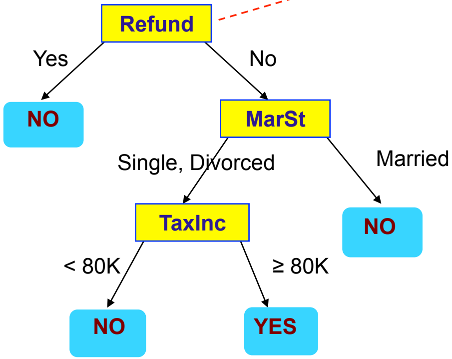
Example: Apply Model to Test Data



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Example: Apply Model to Test Data (cont.)



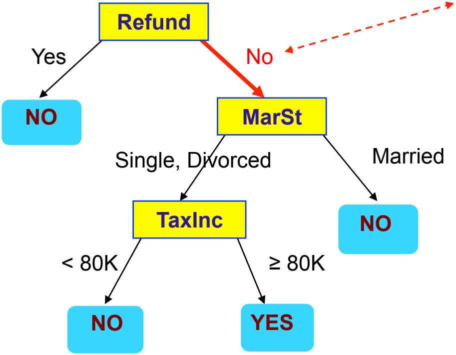
Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

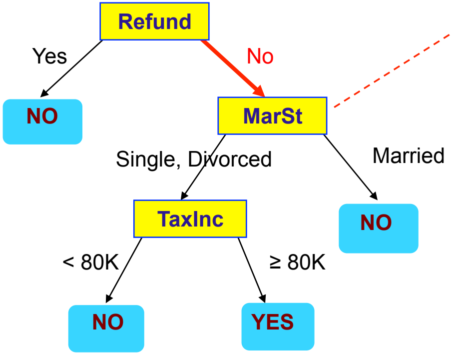
Example: Apply Model to Test Data (cont.)

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



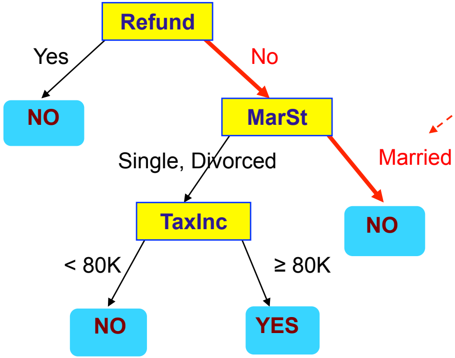
Example: Apply Model to Test Data (cont.)



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

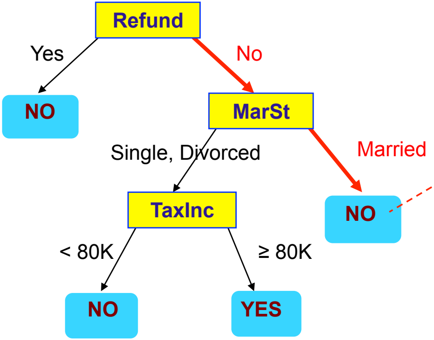
Example: Apply Model to Test Data (cont.)



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Example: Apply Model to Test Data (cont.)



Test Data

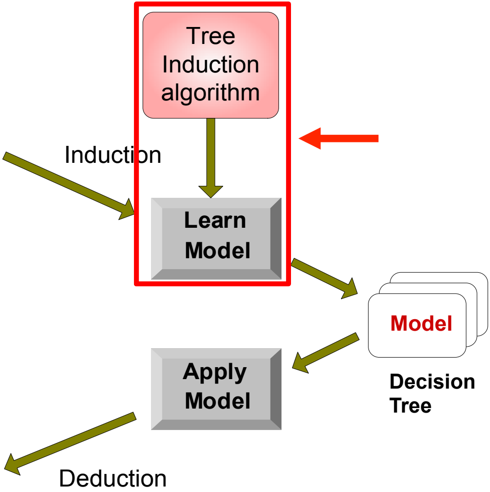
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

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

Decision Tree Induction

- How many trees? Exponential in the number of attributes
- Many Algorithms: reasonably accurate, suboptimal, reasonable amount of time
  - Hunt's Algorithm (basis of many others)
  - CART (Classification and Regression Trees), a book by Breiman et al.
  - ID3, C4.5 by Quinlan

Hunt's algorithm

- Let D_t be the set of training records that reach a node t
- $y = \{y_1, y_2, \cdots, y_c\}$ are class labels
- General procedure
  - If D_t contains records that belong to the same class y_t, then t is a leaf node labeled as y_t
  - 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

Decision Tree Induction Algorithms – Design Issues

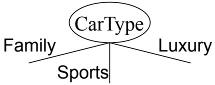
- How should the training records be split?
  - How to specify the attribute test condition?
  - How to determine the best split?
  - Greedy strategy: split the records based on an attribute test that optimizes certain criterion
- When to stop splitting
  - Naive: (1) all the records have identical attribute values; or (2) all the records belong to the same class
  - Is there any better way? Early stop?

Specify the Attribute Test Condition?

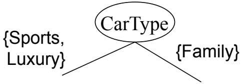
- Depends on attribute types
  - Nominal
  - Ordinal
  - Continuous
- Depends on number of ways to split
  - 2-way split
  - Multi-way split

Splitting Based on Nominal Attributes

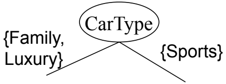
- Multi-way split: Use as many partitions as distinct values



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

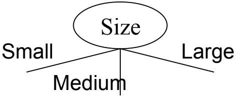


OR

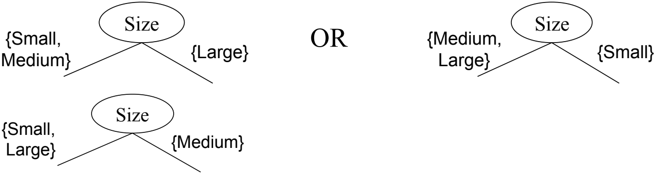


Splitting Based on Ordinal Attributes

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



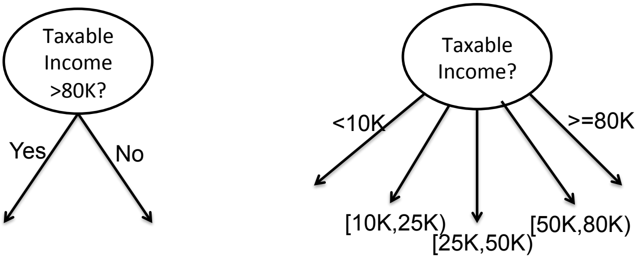
- Binary split: Divides values into two subsets. Need to find optimal partitioning and preserve the order among attribute values.



Splitting Based on Continuous Attributes

- 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 find the best cut
  - Can be more compute intensive

Splitting Based on Continuous Attributes – Example



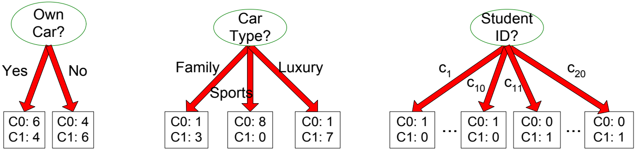
(i) Binary split

(ii) Multi-way split

Determine the best split

Before Splitting:

- 10 records of class 0
- 10 records of class 1



Which test condition is the best?

Determine the Best Split – Node Impurity

- Splitting criterion
  - Splitting attribute
  - Splitting point or splitting subset
  - Ideally, the resulting partitions at each branch are as "pure" as possible.
- Need a measure of node impurity
  - The smaller the degree of impurity, the more skewed the class distribution. The BETTER.
  - Node with class distribution (0,1) has zero impurity.
  - Node with class distribution (0.5,0.5) has highest impurity.

References

- Chapter 3: Introduction to Data Mining (2nd Edition) by Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, and Vipin Kumar

- DecisionTreeClassifier: `https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html`