Introduction to Machine Learning Applications

Spring 2023

Decision trees

Minor Gordon

gordom6@rpi.edu

Rensselaer

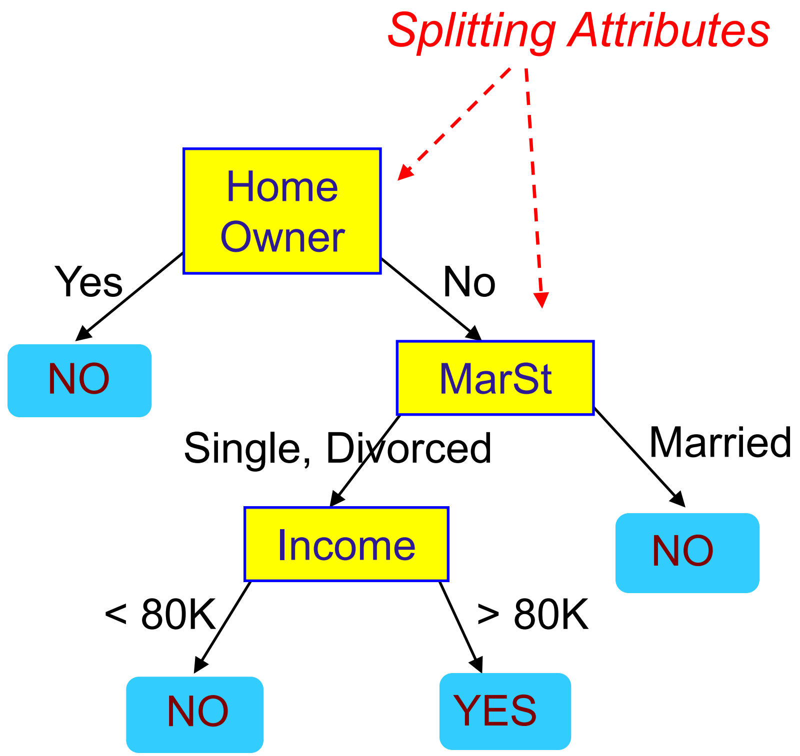
“A decision tree is a flowchart-like structure in which each internal node represents a "test" on an attribute (e.g. whether a coin flip comes up heads or tails), each branch represents the outcome of the test and each leaf node represents a class label (decision taken after computing all attributes).” –Wikipedia

https://en.wikipedia.org/wiki/Decision_tree

Example of a Decision Tree

ID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
	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

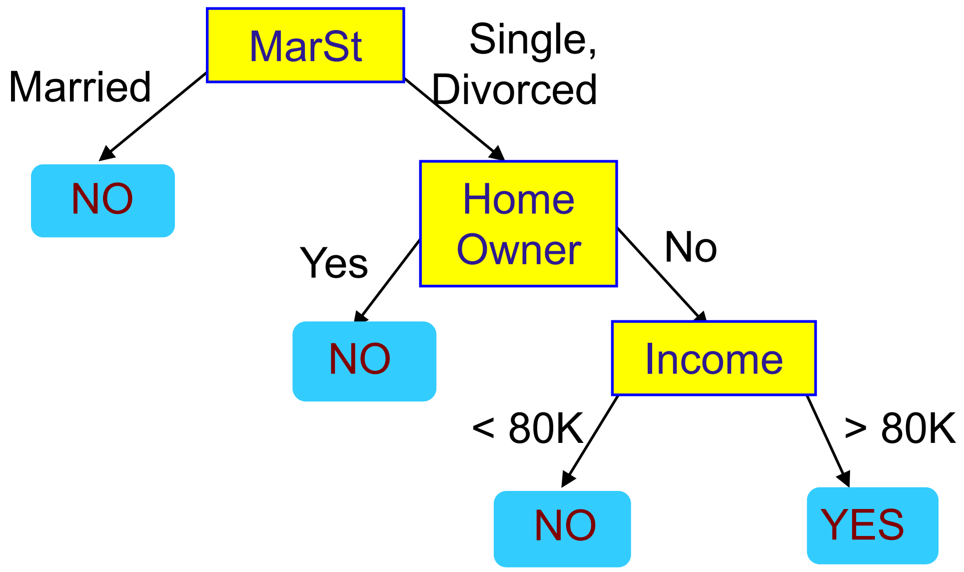
Training Data



Model: Decision Tree

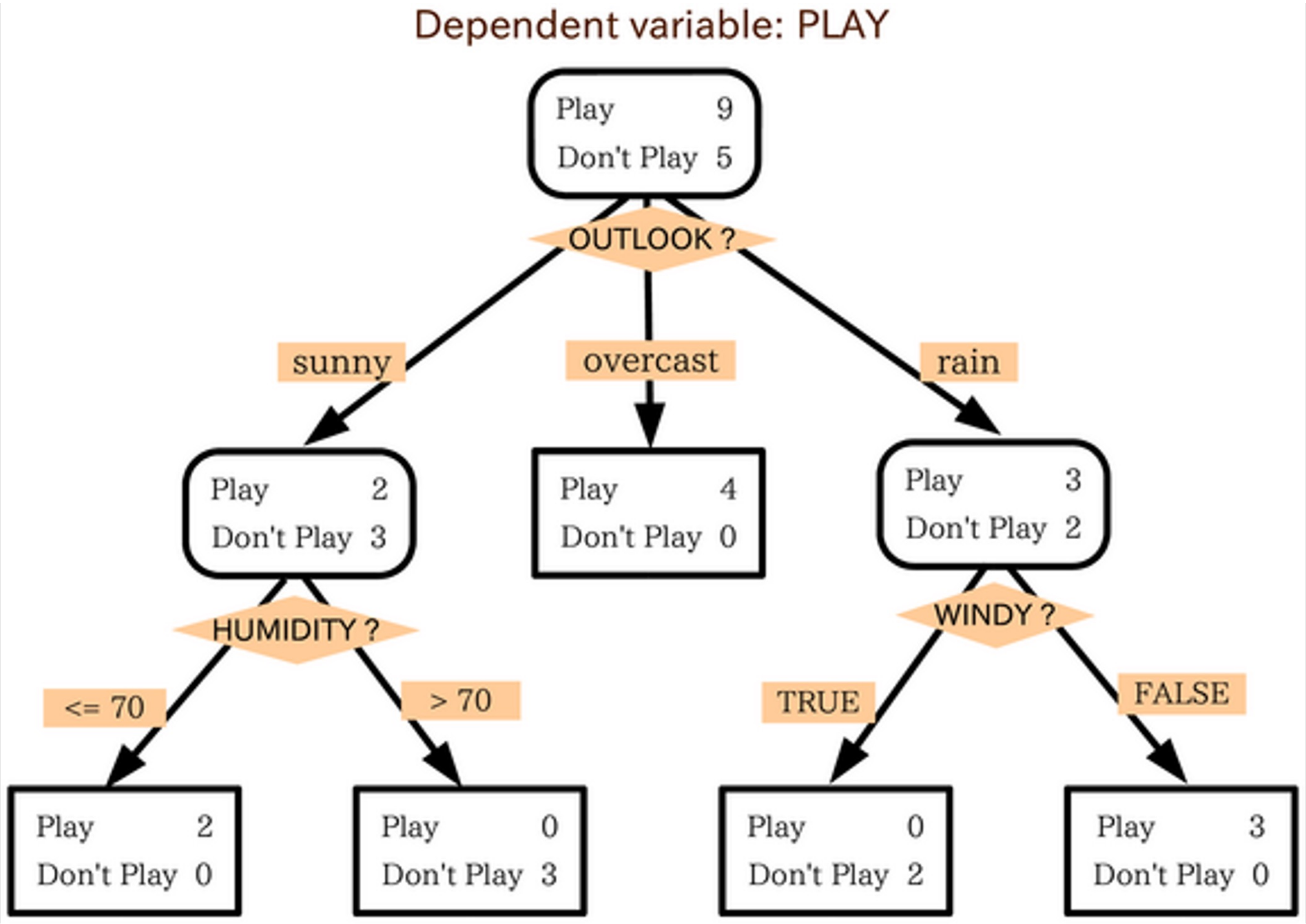
Another Example of Decision Tree

ID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
	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

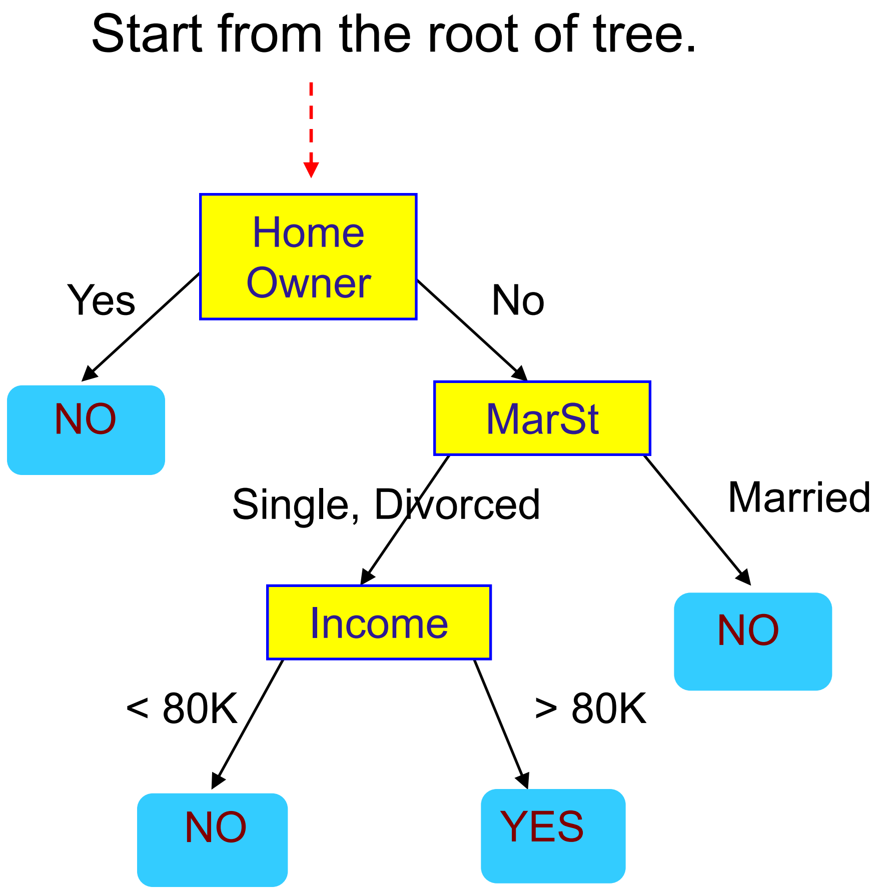


There could be more than one tree that fits the same data!

Decision Tree - Golf



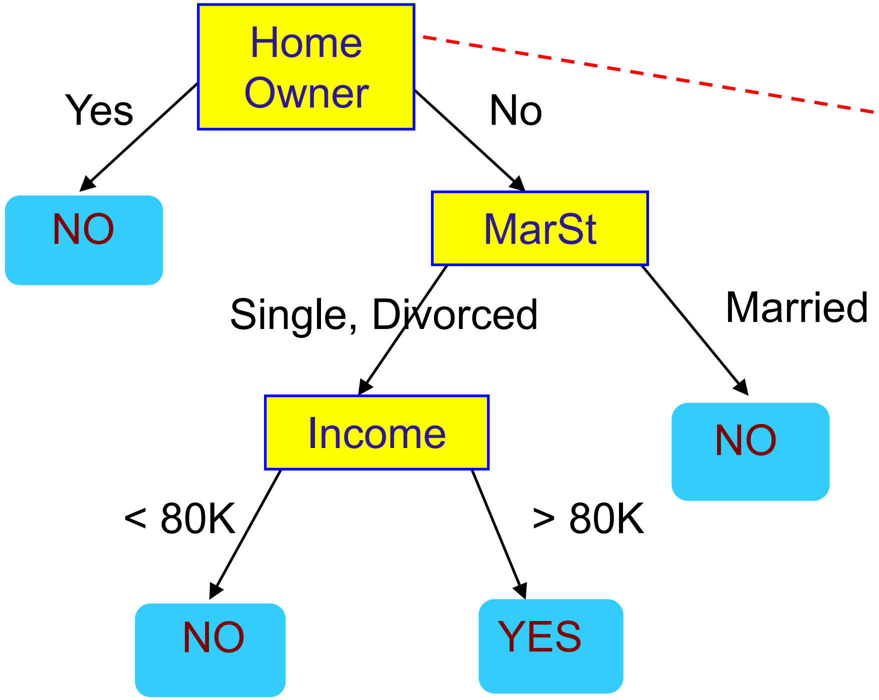
Apply Model to Test Data



Test Data

Home Owner	Marital Status	Annual Income	Defaulted Borrower
No	Married	80K	?

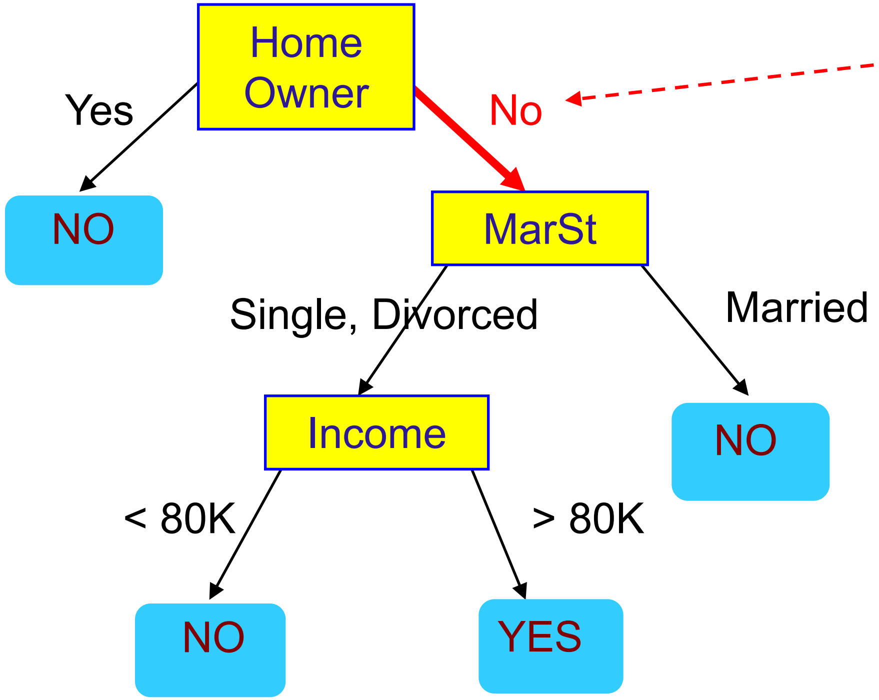
Apply Model to Test Data



Test Data

Home Owner	Marital Status	Annual Income	Defaulted Borrower
No	Married	80K	?

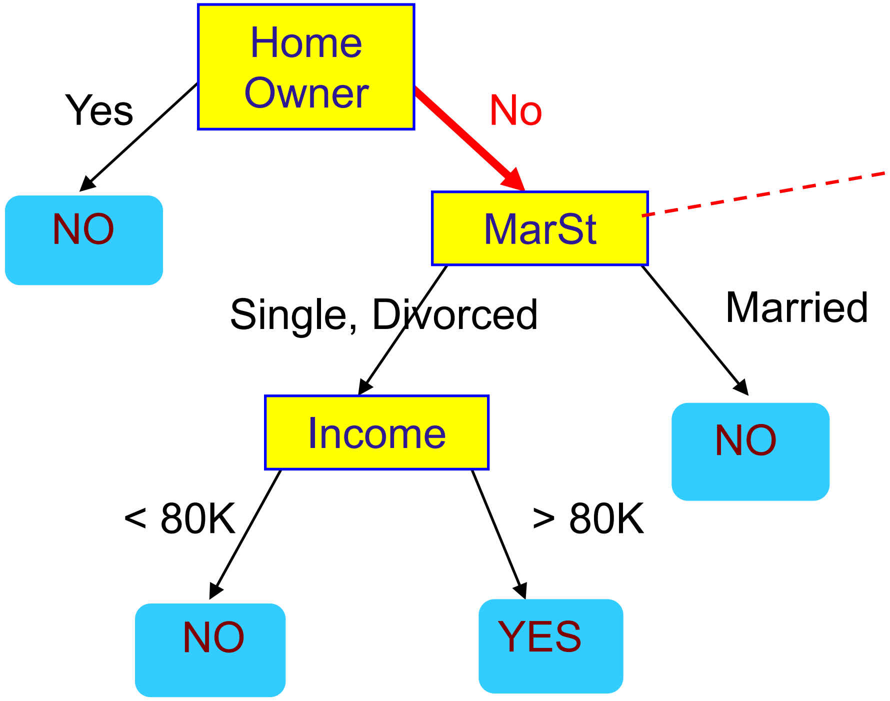
Apply Model to Test Data



Test Data

Home Owner	Marital Status	Annual Income	Defaulted Borrower
No	Married	80K	?

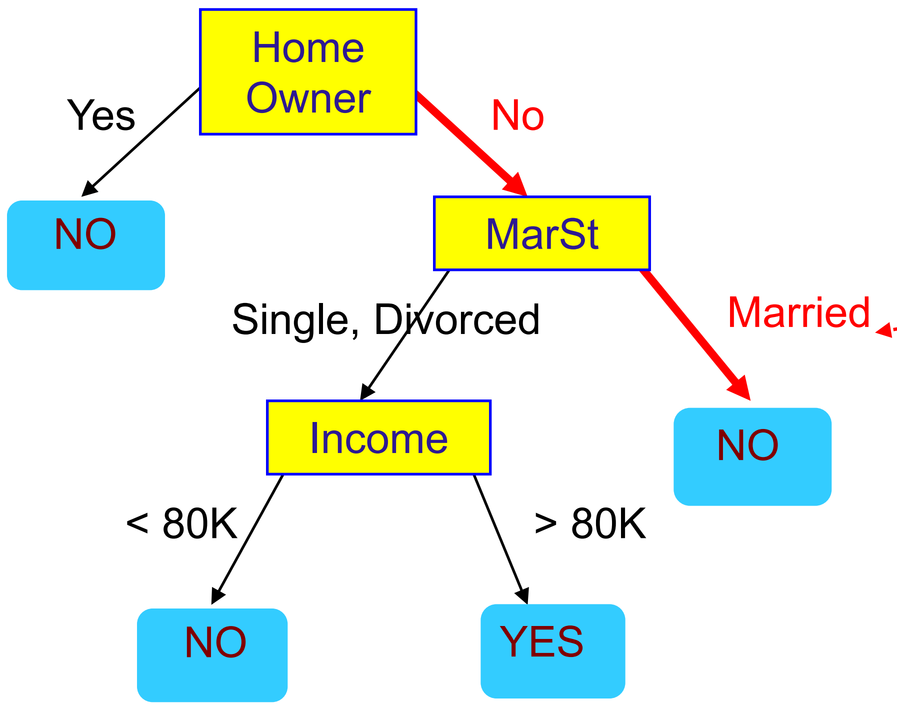
Apply Model to Test Data



Test Data

Home Owner	Marital Status	Annual Income	Defaulted Borrower
No	Married	80K	?

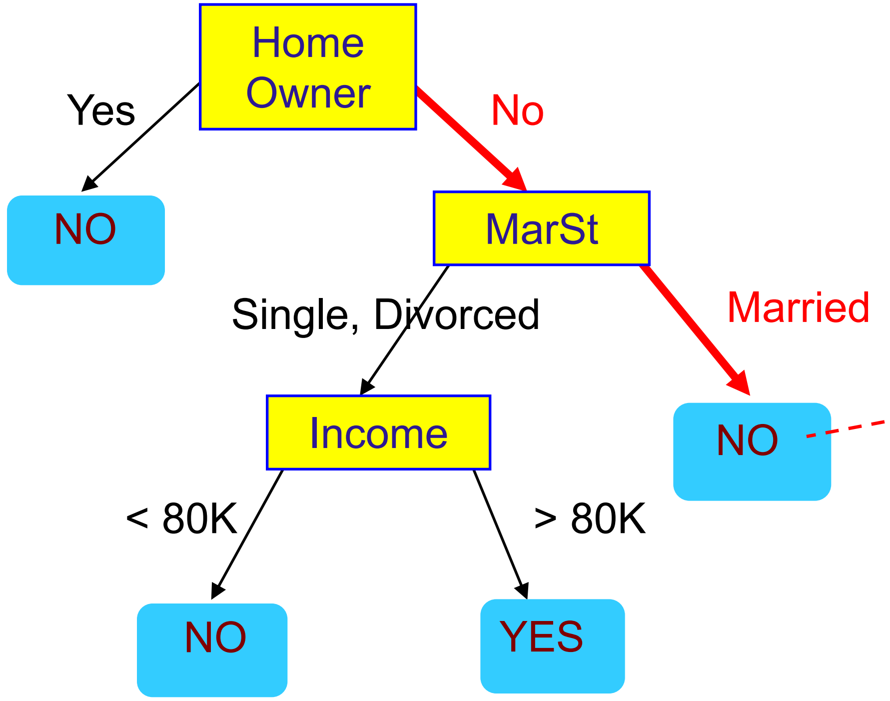
Apply Model to Test Data



Test Data

Home Owner	Marital Status	Annual Income	Defaulted Borrower
No	Married	80K	?

Apply Model to Test Data



Test Data

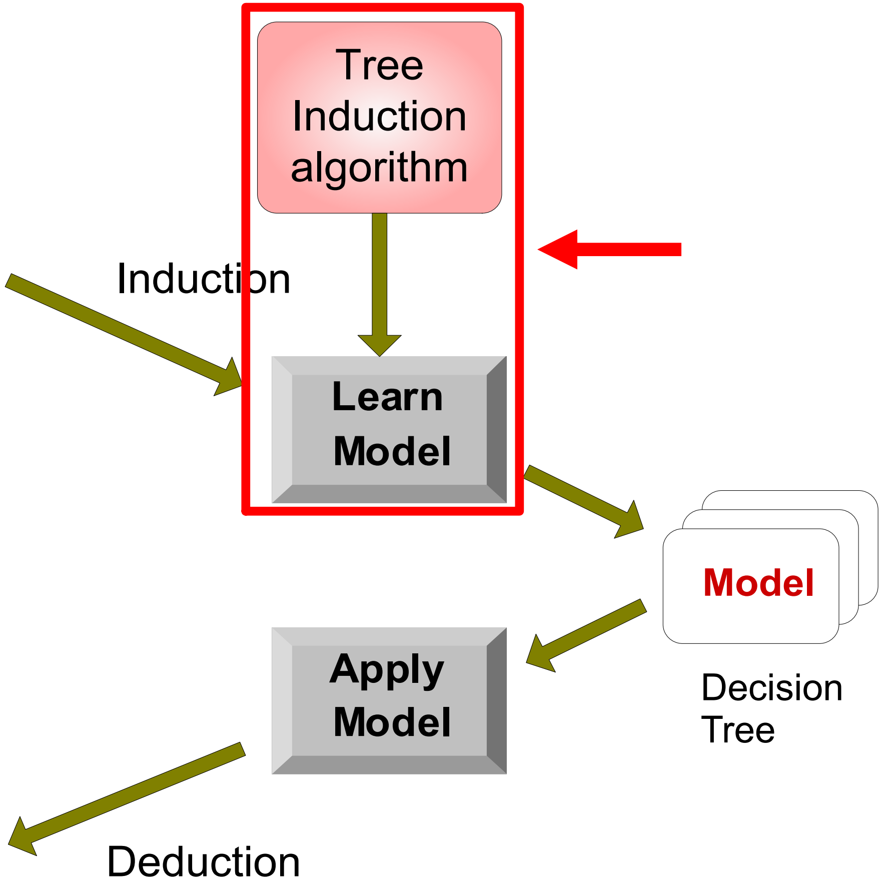
Home Owner	Marital Status	Annual Income	Defaulted Borrower
No	Married	80K	?

Assign Defaulted 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

Design Issues of Decision Tree Induction

- How should training **records be split**?
  - Method for specifying test condition
    - depending on attribute types
  - Measure for evaluating the goodness of a test condition

- How should the **splitting procedure stop**?
  - Stop splitting if all the records belong to the same class or have identical attribute values
  - Early termination

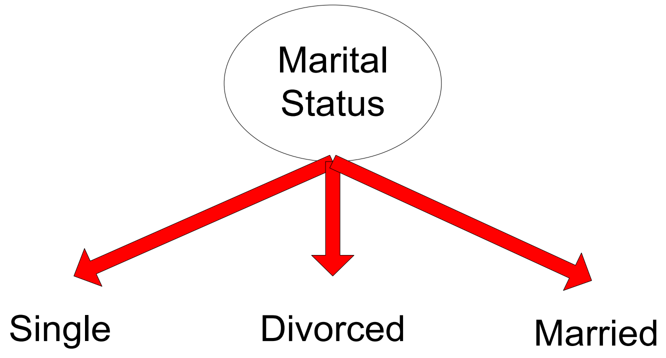
Methods for Expressing Test Conditions

- Depends on attribute types
  - Binary
  - Nominal
  - Ordinal
  - Continuous

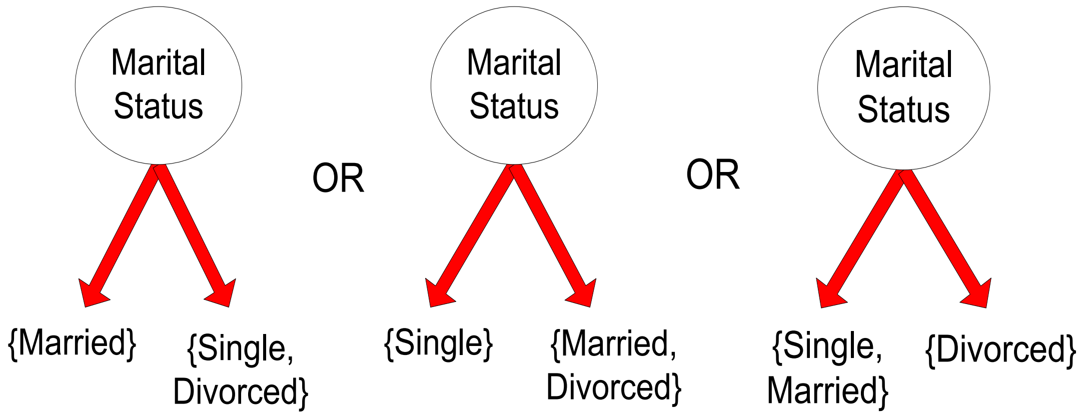
- Depends on number of ways to split
  - 2-way split
  - Multi-way split

Test Condition for Nominal Attributes

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

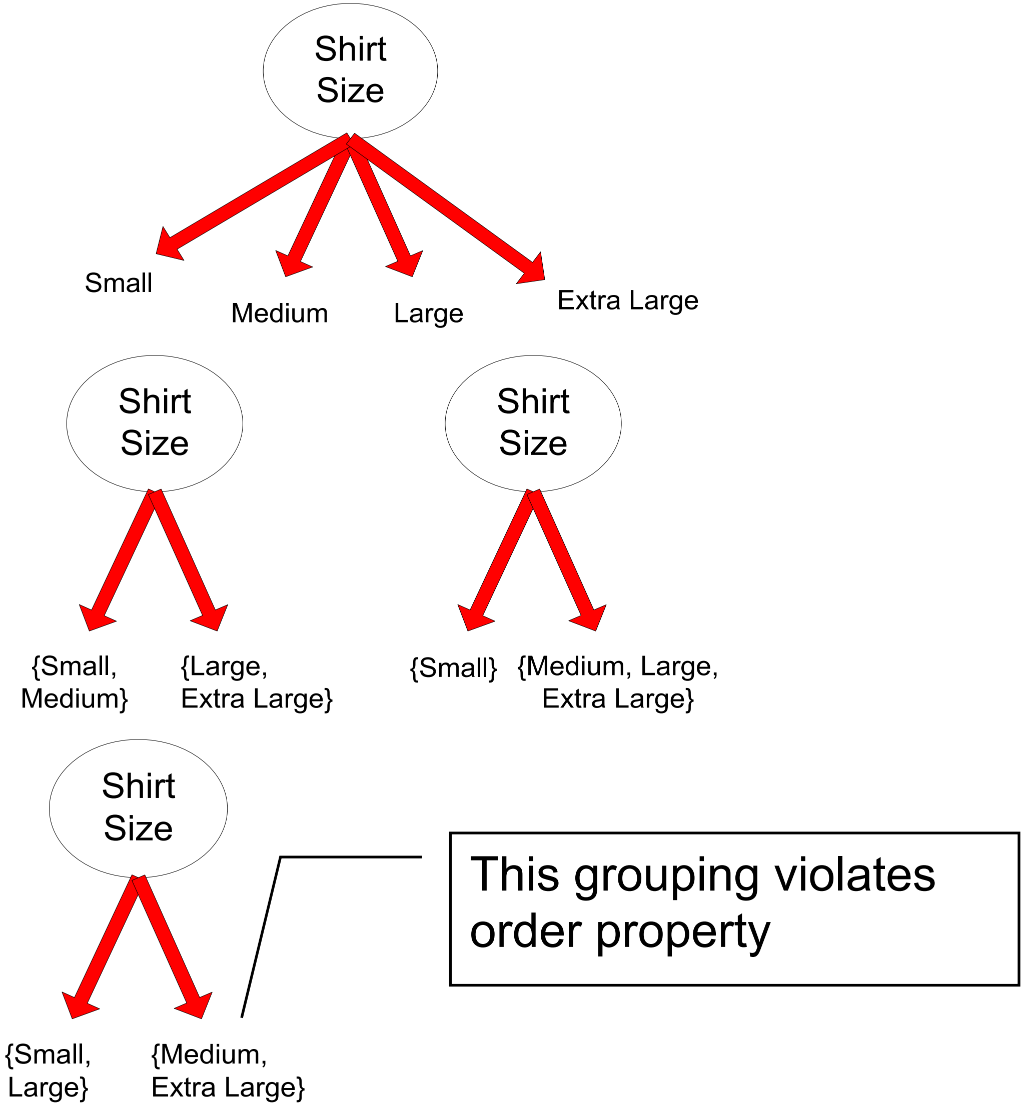


- Binary split:
  - Divides values into two subsets

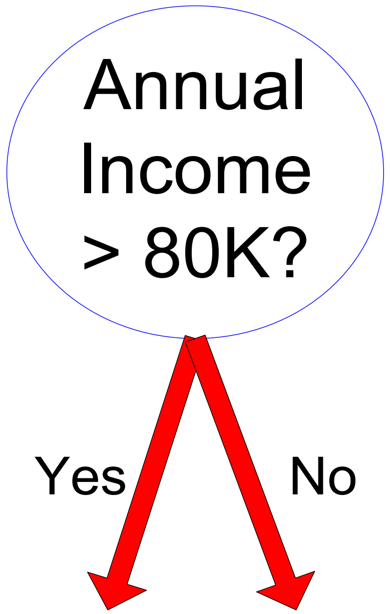


Test Condition for Ordinal Attributes

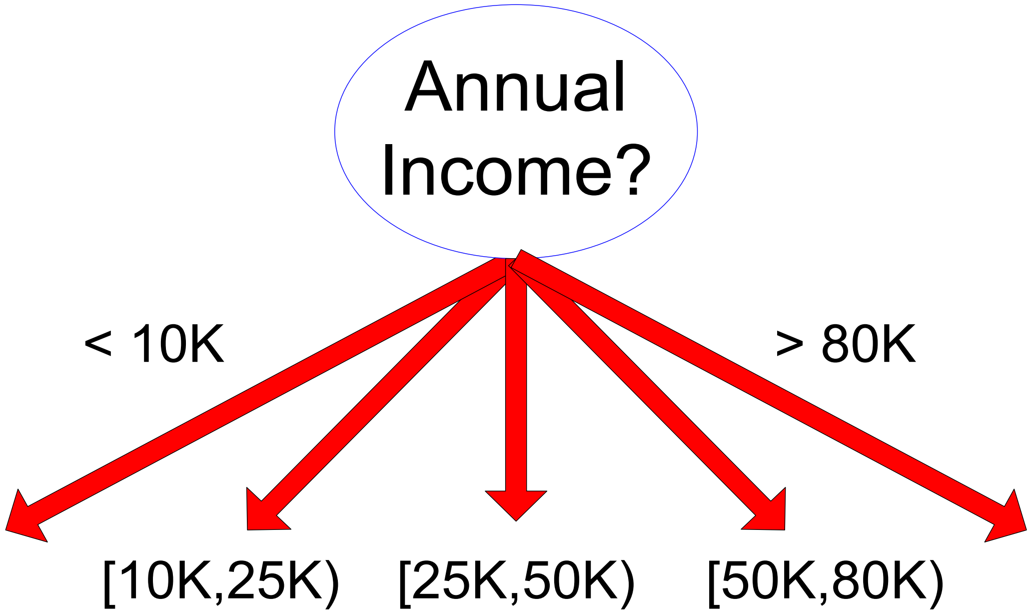
- **Multi-way split:**
  - Use as many partitions as distinct values.
- **Binary split:**
  - Divides values into two subsets
  - Preserve order property among attribute values



Test Condition for Continuous Attributes



(i) Binary split



(ii) Multi-way split

Splitting Based on Continuous Attributes

- Different ways of handling
  - Discretization to form an ordinal categorical attribute

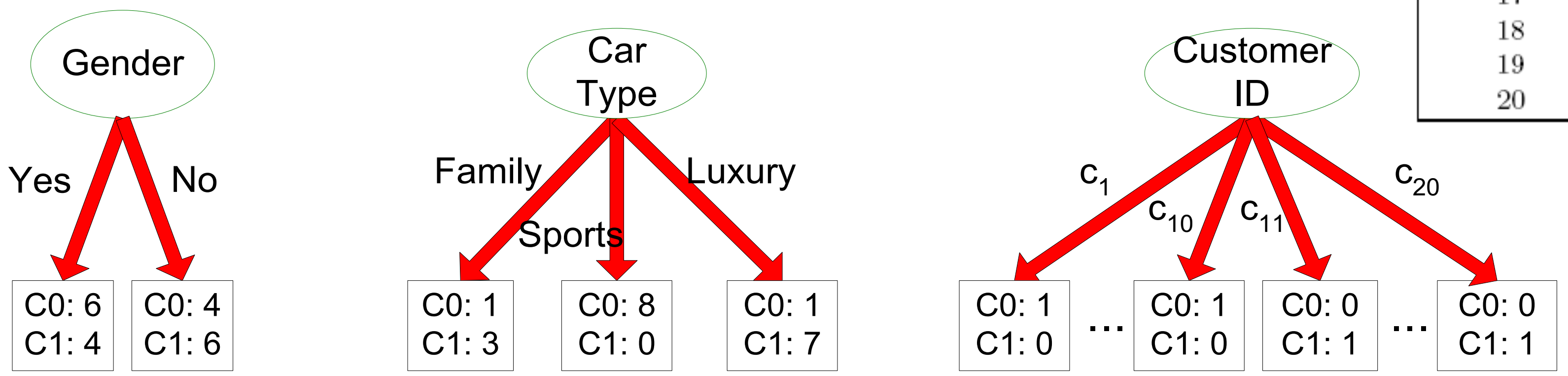
    Ranges can be found by equal interval bucketing, equal frequency bucketing (percentiles), or clustering.
    - Static – discretize once at the beginning
    - Dynamic – repeat at each node

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

How to determine the best split

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

Customer Id	Gender	Car Type	Shirt Size	Class
1	M	Family	Small	C0
2	M	Sports	Medium	C0
3	M	Sports	Medium	C0
4	M	Sports	Large	C0
5	M	Sports	Extra Large	C0
6	M	Sports	Extra Large	C0
7	F	Sports	Small	C0
8	F	Sports	Small	C0
9	F	Sports	Medium	C0
10	F	Luxury	Large	C0
11	M	Family	Large	C1
12	M	Family	Extra Large	C1
13	M	Family	Medium	C1
14	M	Luxury	Extra Large	C1
15	F	Luxury	Small	C1
16	F	Luxury	Small	C1
17	F	Luxury	Medium	C1
18	F	Luxury	Medium	C1
19	F	Luxury	Medium	C1
20	F	Luxury	Large	C1



Which test condition is the best?

How to determine the best split

- Greedy approach:
  - Nodes with purer class distribution are preferred

- Need a measure of node impurity:

C0: 5	C0: 9
C1: 5	C1: 1
High degree of impurity	Low degree of impurity

Measures of Node Impurity

- Gini Index

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

- Entropy

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

- Misclassification error

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

Finding the best split

1. Compute impurity measure (P) before splitting
2. Compute impurity measure (M) after splitting
    1. Compute impurity measure of each child node
    2. M is the weighted impurity of children
3. Choose the attribute test condition that produces the highest gain

$$\text{Gain} = P - M$$

or equivalently, lowest impurity measure after splitting (M)

Revisiting Entropy

Information and Probability

- Information relates to possible outcomes of an event
  - transmission of a message, flip of a coin, or measurement of a piece of data

- The more certain an outcome, the less information that it contains and vice-versa
  - For example, if a coin has two heads, then an outcome of heads provides no information
  - More quantitatively, the information is related the probability of an outcome
    - The smaller the probability of an outcome, the more information it provides and vice-versa
  - Blog post: [“Entropy is a measure of uncertainty”]()

Entropy

- For
  - a variable (event), X,
  - with n possible values (outcomes), $x_1, x_2 \ldots, x_n$
  - each outcome having probability, $p_1, p_2 \ldots, p_n$
  - the entropy of X , $H(X)$, is given by

$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$$

- Entropy is between 0 and $\log_2 n$ and is measured in bits
  - Thus, entropy is a measure of how many bits it takes to represent an observation of X on average

Entropy Examples

- For a coin with probability p of heads and probability $q = 1 - p$ of tails

$$H = -p \log_2 p - q \log_2 q$$

  - For $p = 0.5$, $q = 0.5$ (fair coin) $H = 1$
  - For $p = 1$ or $q = 1$, $H = 0$

- What is the entropy of a fair four-sided die?

Entropy for Sample Data: Example

Hair Color	Count	p	$-p\log_2 p$
Black	75	0.75	0.3113
Brown	15	0.15	0.4105
Blond	5	0.05	0.2161
Red	0	0.00	0
Other	5	0.05	0.2161
Total	100	1.0	1.1540

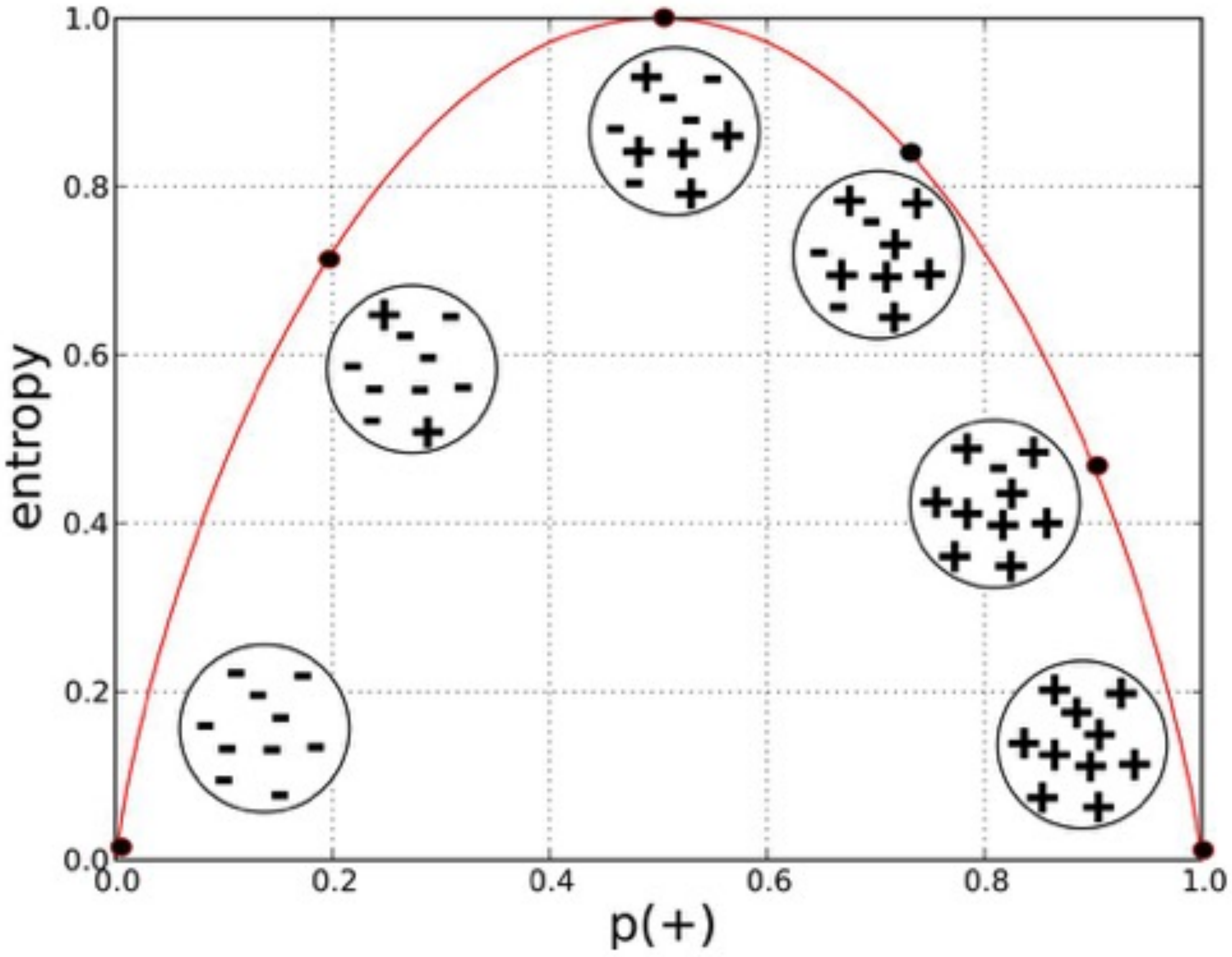
Measure of Impurity: Entropy

- Entropy at a given node t:

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

  - (NOTE: $p(j \mid t)$ is the relative frequency of class j at node t).
  - 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 quite similar to Gini index computations



Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

Computing Entropy of a Single Node

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

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

P(c1)=0.5; p(c2)=0.5

Entropy = -(1/2)log(1/2)- (1/2)log(1/2)

Computing Information Gain after Splitting

- Information Gain

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

Parent Node p is split into k partitions; n is the total number of records being split; n_i is number of records in partition i

- Choose the split that achieves most (entropy) reduction (maximizes GAIN) on the target variable (C1/C2)
  - i.e., how much entropy we removed with this split

InformationGain =entropy(parent) – [average entropy(children)]

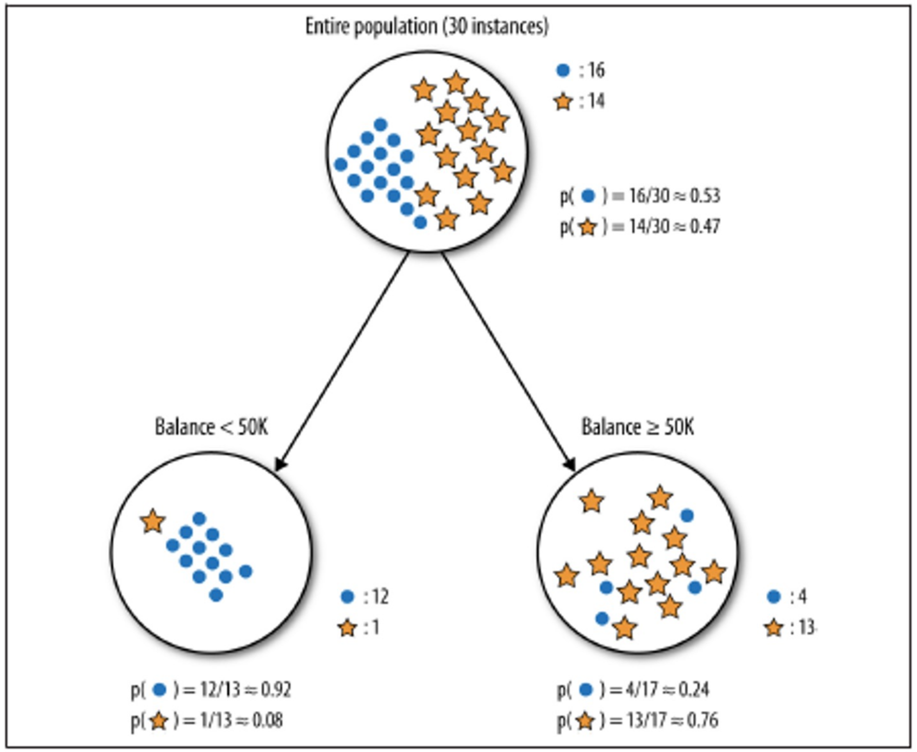
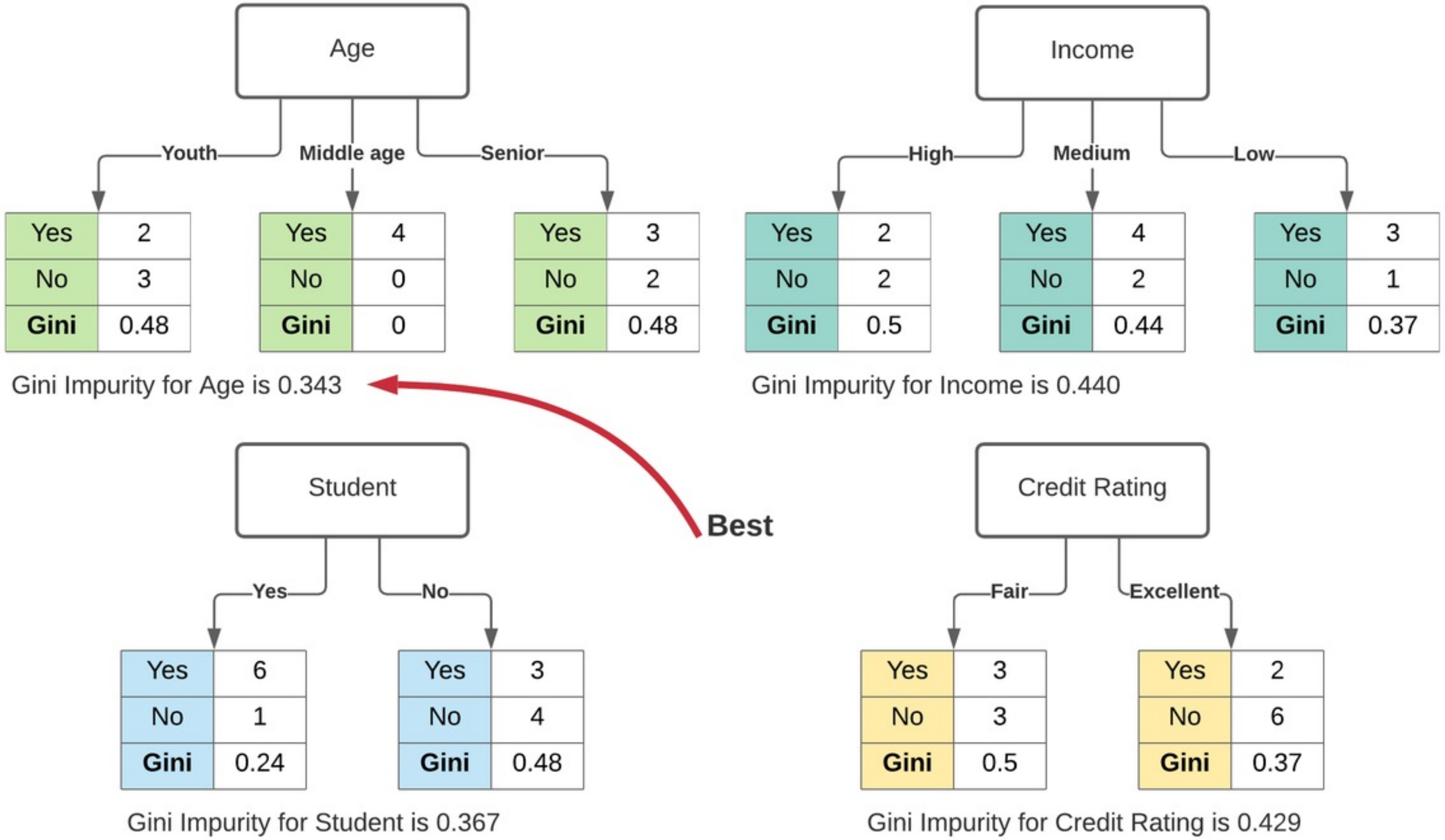


Figure 3-4. Splitting the "write-off" sample into two segments, based on splitting the Balance attribute (account balance) at 50K.

Gini Impurity/Index

$$Gini = 1 - \sum_{i=1}^{C} (p_i)^2$$



- Probability of classifying a randomly-chosen data point incorrectly, classifying according to the class distribution
- [0, 1]
- Less is better
- 0 = everything is same class
- 0.5 = items uniformly distributed over classes
- 1 = items randomly distributed over classes