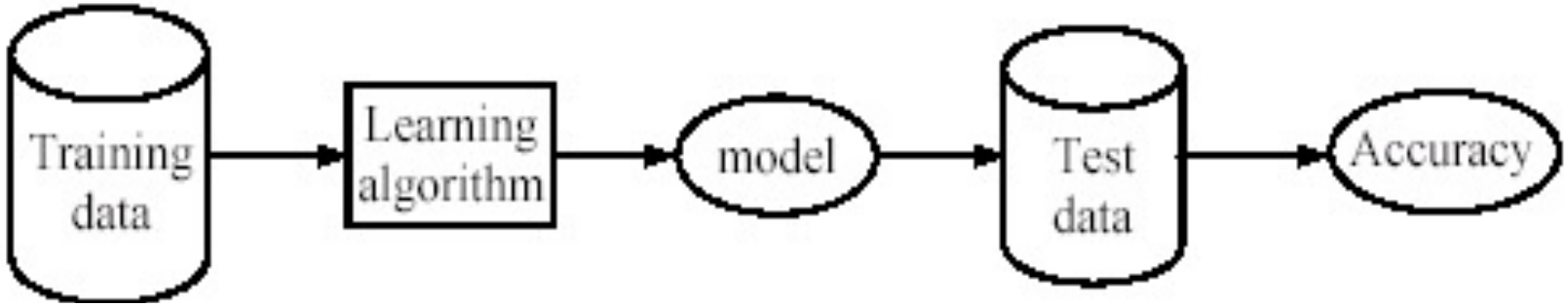
CLASSIFICATION

SUPERVISED LEARNING PROCESS: TWO STEPS

- Learning (training): learn a model via the training data
- Testing: test the model via test data and evaluate the model accuracy

$$Accuracy = \frac{\text{Number of correct classifications}}{\text{Total number of test cases}},$$



A DECISION TREE FROM THE LOAN DATA

features ⇕ attributes

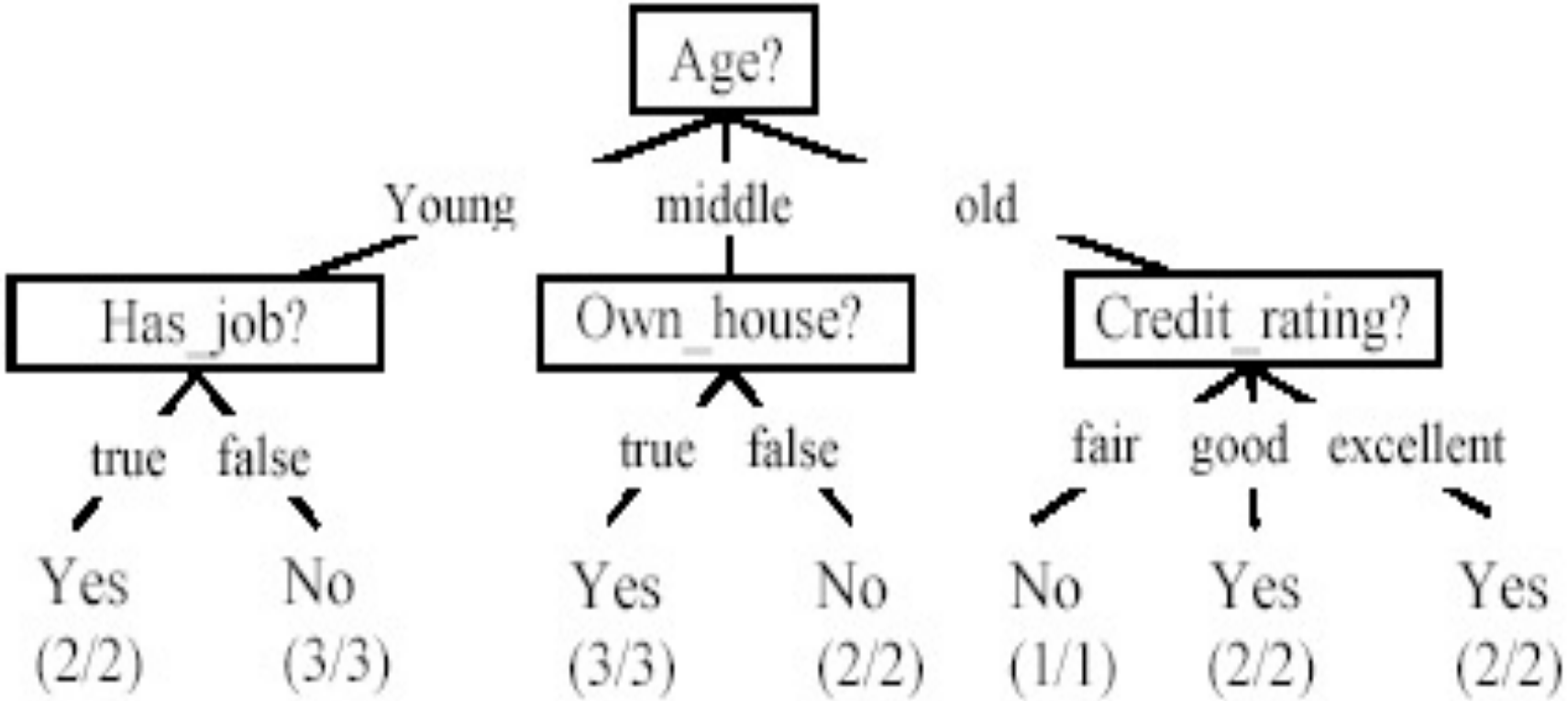
ID	Age	Has_Job	Own_House	Credit_Rating	Class
1	young	false	false	fair	No
2	young	false	false	good	No
3	young	true	false	good	Yes
4	young	true	true	fair	Yes
5	young	false	false	fair	No
6	middle	false	false	fair	No
7	middle	false	false	good	No
8	middle	true	true	good	Yes
9	middle	false	true	excellent	Yes
10	middle	false	true	excellent	Yes
11	old	false	true	excellent	Yes
12	old	false	true	good	Yes
13	old	true	false	good	Yes
14	old	true	false	excellent	Yes
15	old	false	false	fair	No

Label (Yes, No)

Training Data

A DECISION TREE FROM THE LOAN DATA

- Decision nodes and leaf nodes (classes)

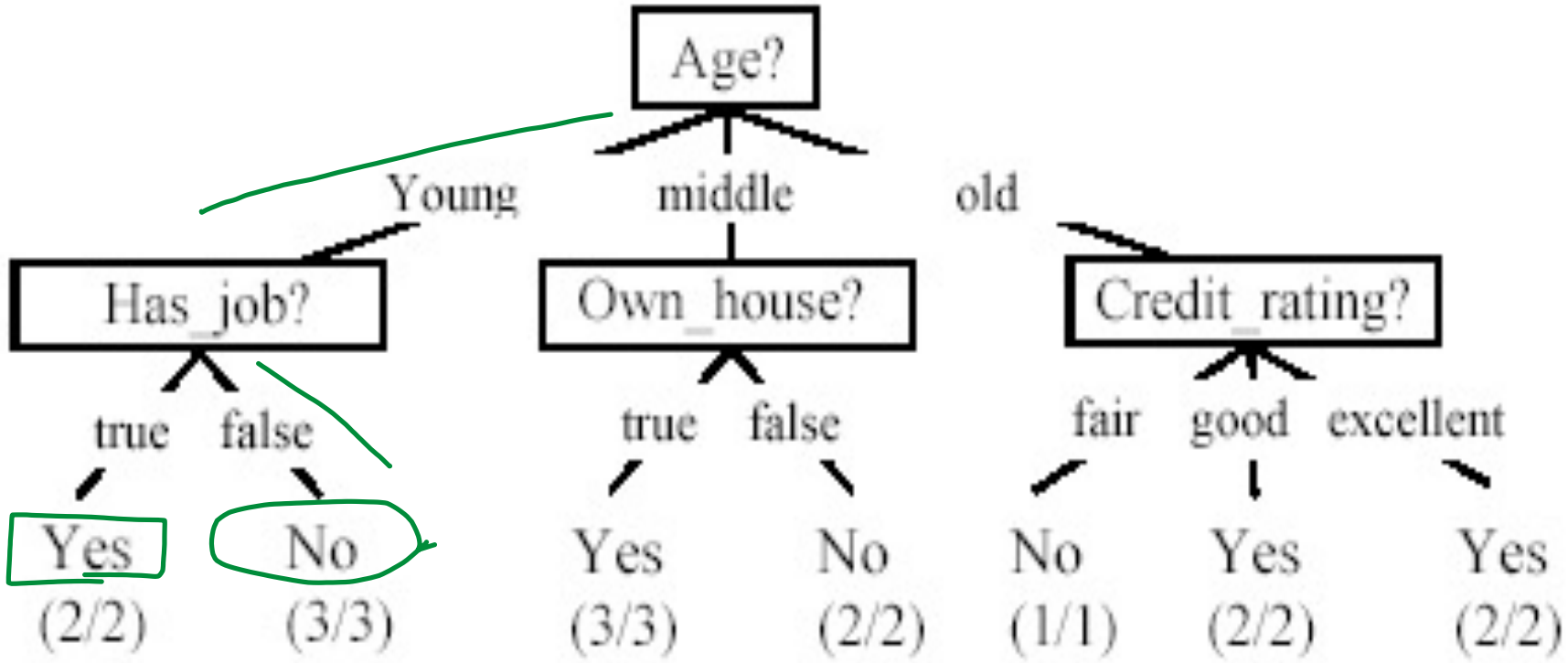


USE THE DECISION TREE

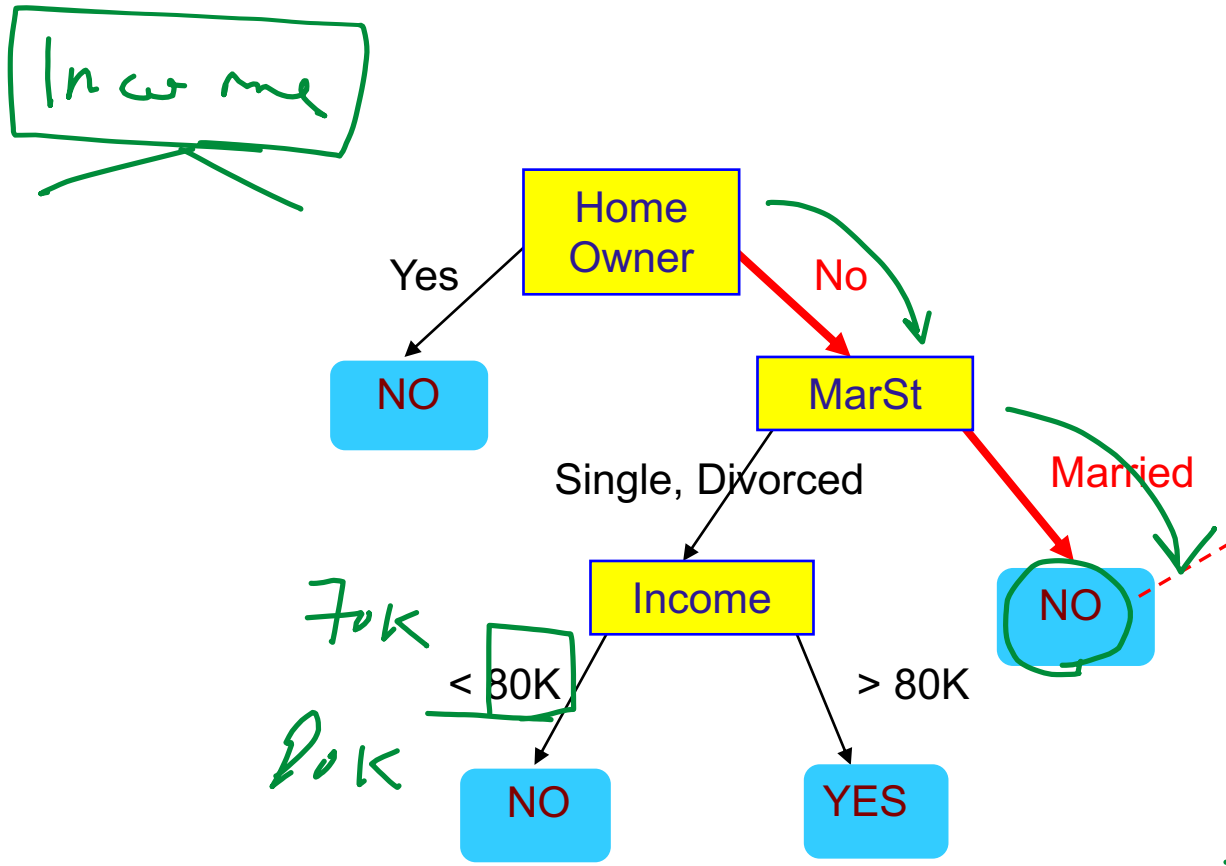
testing Data

Age	Has_Job	Own_house	Credit-Rating	Class
young	false	false	good	? No

label



ANOTHER EXAMPLE



Test Data

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

More than 1 trees that can be learned from TD

DECISION TREE INDUCTION

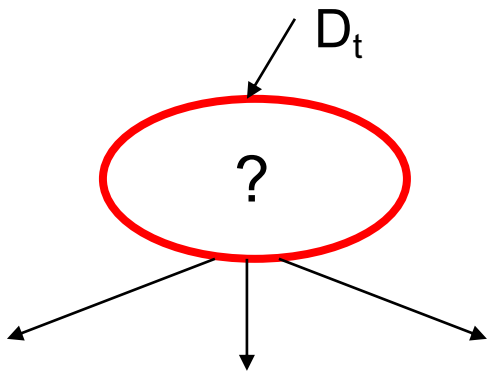
- Many Algorithms:
  - Hunt's Algorithm (one of the earliest)
  - CART
  - ID3, C4.5
  - SLIQ,SPRINT

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 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.

ID	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



DESIGN ISSUES OF DECISION TREE INDUCTION

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

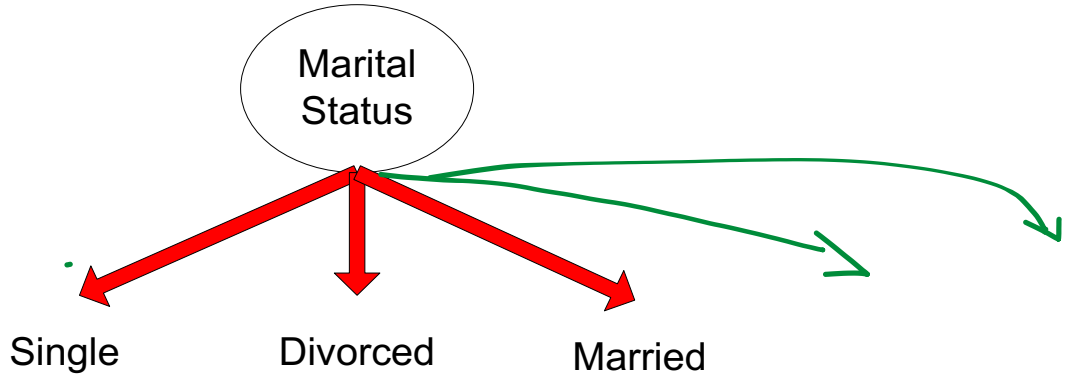
- How should the splitting procedure stop?
  - Stop splitting if all the records belong to the same class or have identical attribute values
  - Early termination

METHODS FOR EXPRESSING TEST CONDITIONS

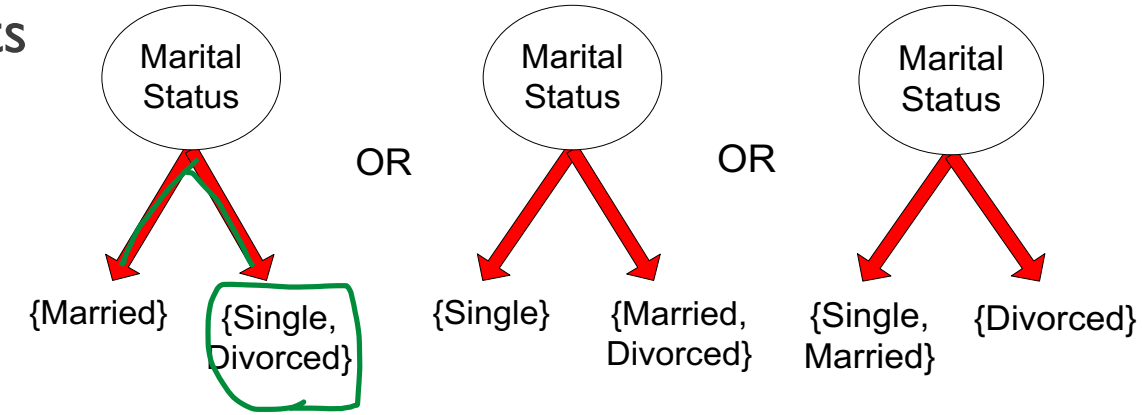
- Depends on attribute types
  - Binary 0 or 1 : Yes or No
  - Nominal {a, b, c, d}
  - Ordinal
  - Continuous R {1, 2, 3, ...}

TEST CONDITION FOR NOMINAL ATTRIBUTES

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

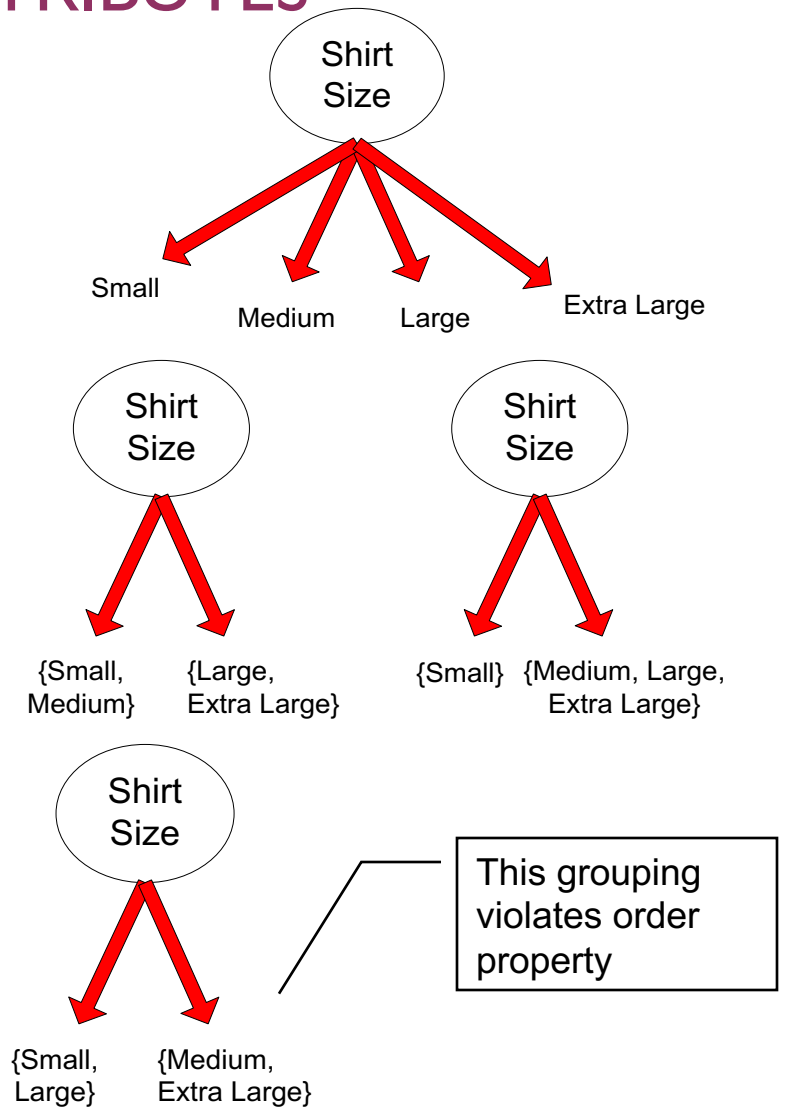


- Binary split:
  - Divides values into two subsets

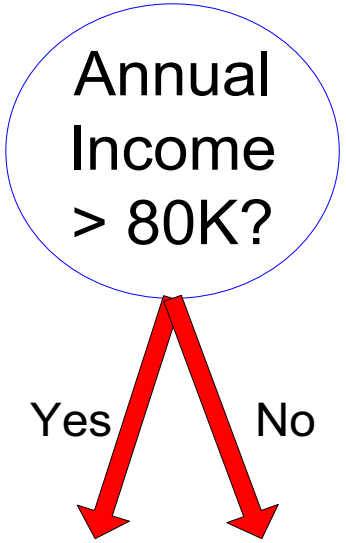


TEST CONDITION FOR ORDINAL ATTRIBUTES

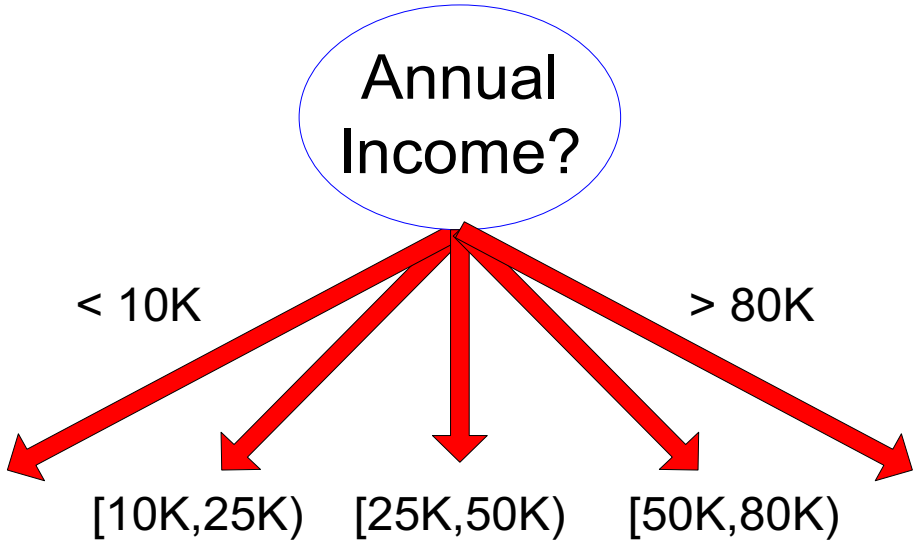
- Multi-way split:
  - Use as many partitions as distinct values
- Binary split:
  - Divides values into two subsets
  - Preserve order property among attribute values



TEST CONDITION FOR CONTINUOUS ATTRIBUTES



(i) Binary split

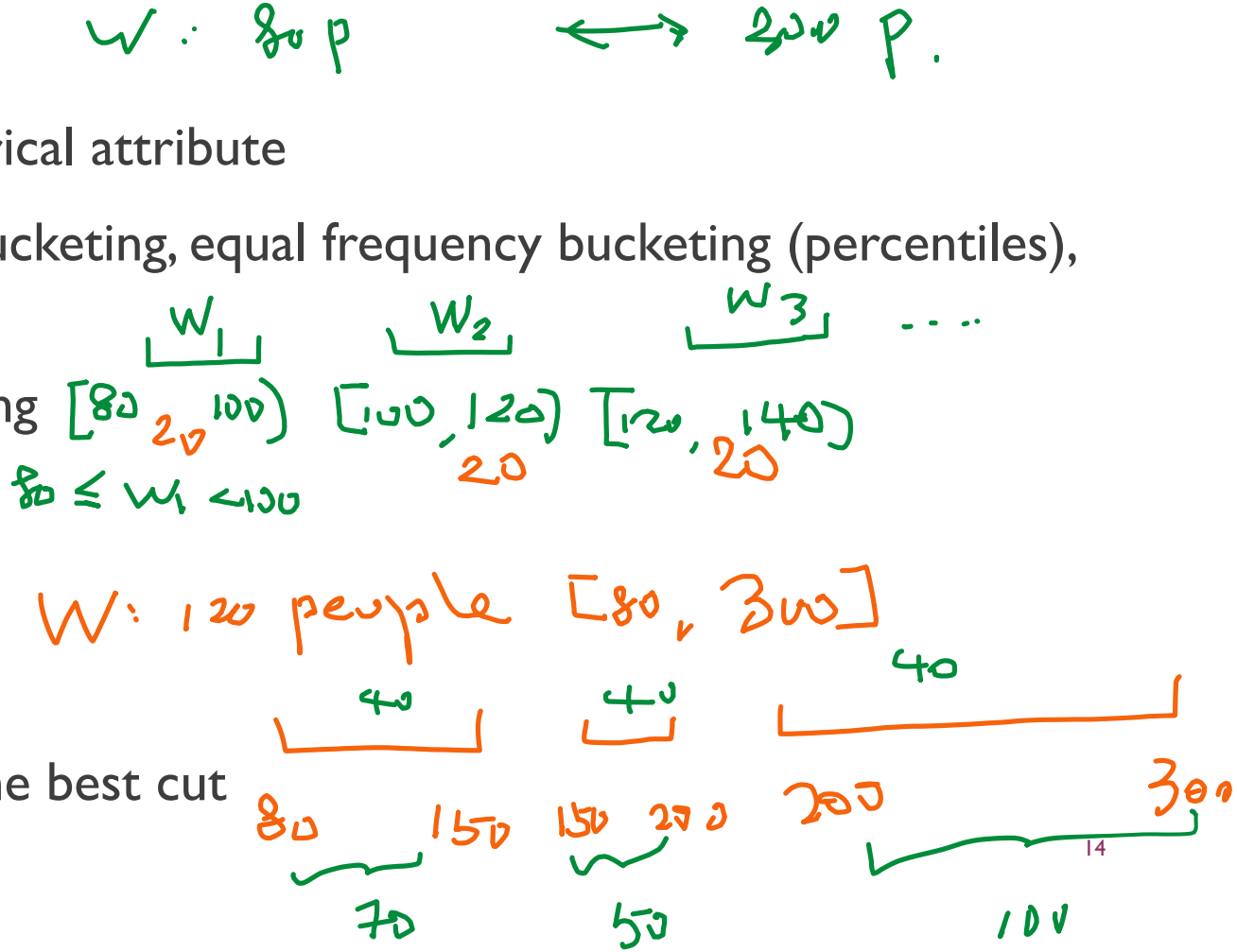


(ii) Multi-way split

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 ≥ v)
    - consider all possible splits and finds the best cut
    - can be more compute intensive

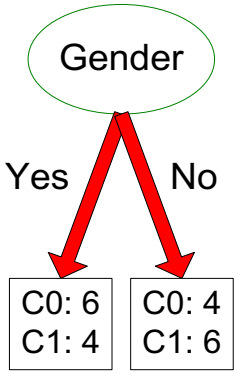


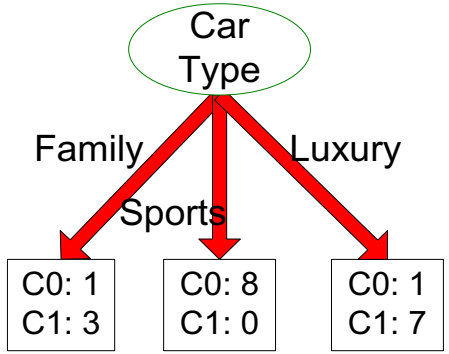
HOW TO DETERMINE THE BEST SPLIT

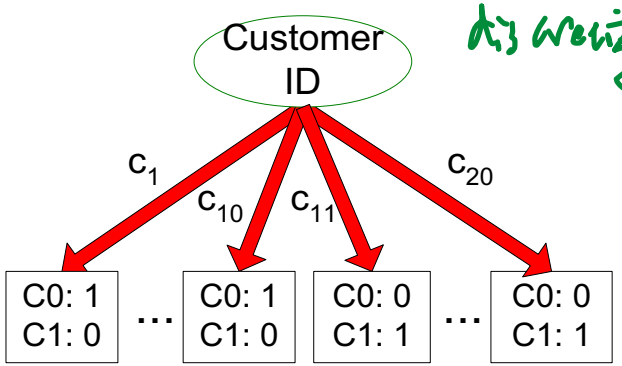
Before Splitting: 10 records of class 0 (c0),
10 records of class 1 (c1)

What are the values of the label for this data? How many cases / records for each label.

Learn the type of each attribute / feature, their values.







Which test condition is the best?

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

HOW TO DETERMINE THE BEST SPLIT

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

How to define what is purer or not

- Need a measure of node impurity:

C0: 5
C1: 5

High degree of impurity

C0: 9
C1: 1

Low degree of impurity

MEASURES OF NODE IMPURITY

- Gini Index

$$Gini\ Index = 1 - \sum_{i=0}^{c-1} p_i(t)^2$$

Where $\boldsymbol{p_i(t)}$ is the frequency of class $\boldsymbol{i}$ at node t, and $\boldsymbol{c}$ is the total number of classes

- Entropy

$$Entropy = -\sum_{i=0}^{c-1} p_i(t) log_2 p_i(t)$$

- Misclassification error (confusing matrix for decision tree)

$$Classification\ error = 1 - \max[p_i(t)]$$

FINDING THE BEST SPLIT

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

highest G via low M

Gain = P - M

$G = -M$

or equivalently, lowest impurity measure after splitting (M)

GINI INDEX

- **What is Gini Index?**
- Gini index or Gini impurity measures the degree or probability of a particular variable being wrongly classified when it is randomly chosen.

$$Gini\ Index = 1 - \sum_{i=0}^{c-1} p_i(t)^2$$

- **But what is actually meant by 'impurity'?**
- If all the elements belong to a single class, then it can be called pure.
- The degree of Gini index varies between 0 and 1,
  0: all elements belong to a certain class or if there exists only one class, and
  1: the elements are randomly distributed across various classes.
- A Gini Index of 0.5 denotes equally distributed elements into some classes.

EXAMPLE OF GINI INDEX

Past Trend	Open Interest	Trading Volume	Return
Positive	Low	High	Up
Negative	High	Low	Down
Positive	Low	High	Up
Positive	High	High	Up
Negative	Low	High	Down
Positive	Low	Low	Down
Negative	High	High	Down
Negative	Low	High	Down
Positive	Low	Low	Down
Positive	High	High	Up

P(Past Trend=Positive): 6/10 $\frac{6}{10} = \frac{\#\ of\ positive\ cases\ in\ attribute\ pas}{\#\ of\ cases\ for\ attribute\ past\ trend}$

P(Past Trend=Negative): 4/10

If (Past Trend = Positive & Return = Up), probability = 4/6

If (Past Trend = Positive & Return = Down), probability = 2/6

Gini index = 1 - ((4/6)^2 + (2/6)^2) = 0.45

If (Past Trend = Negative & Return = Up), probability = 0

If (Past Trend = Negative & Return = Down), probability = 4/4

Gini index = 1 - ((0)^2 + (4/4)^2) = 0

Weighted sum of the Gini Indices can be calculated as follows:

Gini Index for Past Trend = (6/10)*0.45* + *(4/10)*0 = 0.27

$if(PT=P\ \&\ R=U) = \frac{\#\ PT=P\ U\ \#R=u}{\#\ PT=P} = \frac{4}{6}$