Decision Trees

Venkat Reddy

What is the need of segmentation?

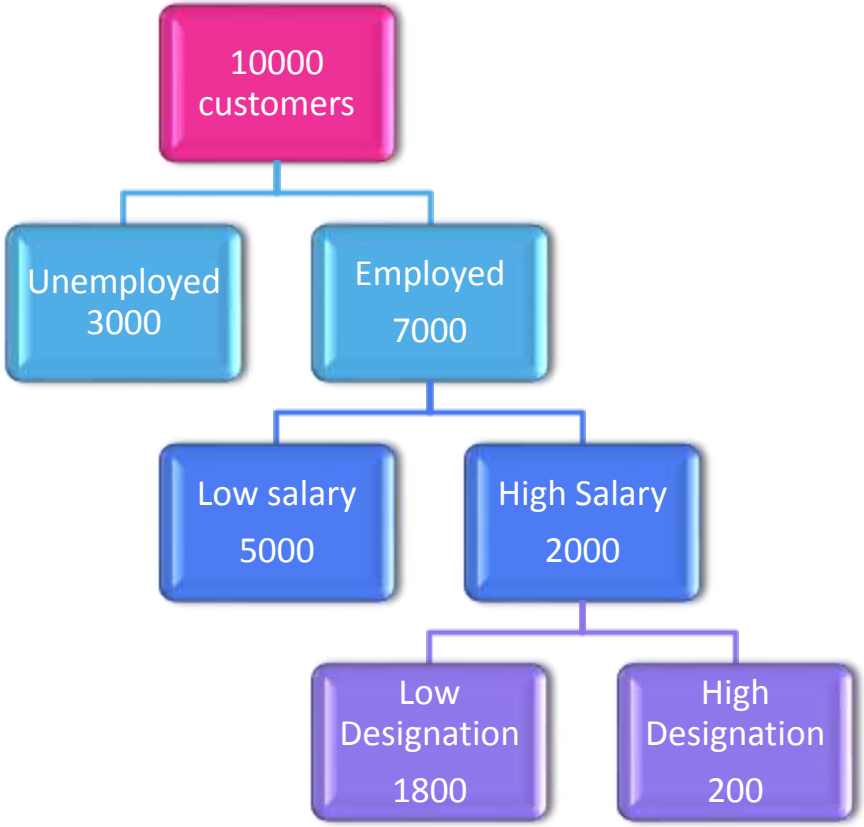
Problem:

- 10,000 Customers - we know their age, city name, income, employment status, designation
- You have to sell 100 Blackberry phones(each costs $1000) to the people in this group. You have maximum of 7 days
- If you start giving demos to each individual, 10,000 demos will take more than one year. How will you sell maximum number of phones by giving minimum number of demos?

What is the need of segmentation?

Solution

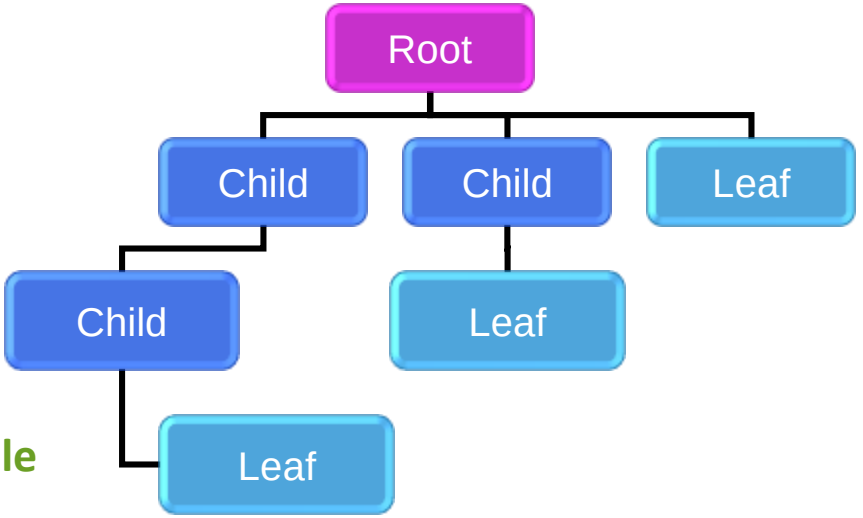
- Divide the whole population into two groups employed / unemployed
- Further divide the employed population into two groups high/low salary
- Further divide that group into high /low designation



Decision Trees

Decision Tree Vocabulary

- Drawn top-to-bottom or left-to-right
- Top (or left-most) node = **Root Node**
- Descendent node(s) = **Child Node(s)**
- Bottom (or right-most) node(s) = **Leaf Node(s)**
- Unique path from root to each leaf = **Rule**



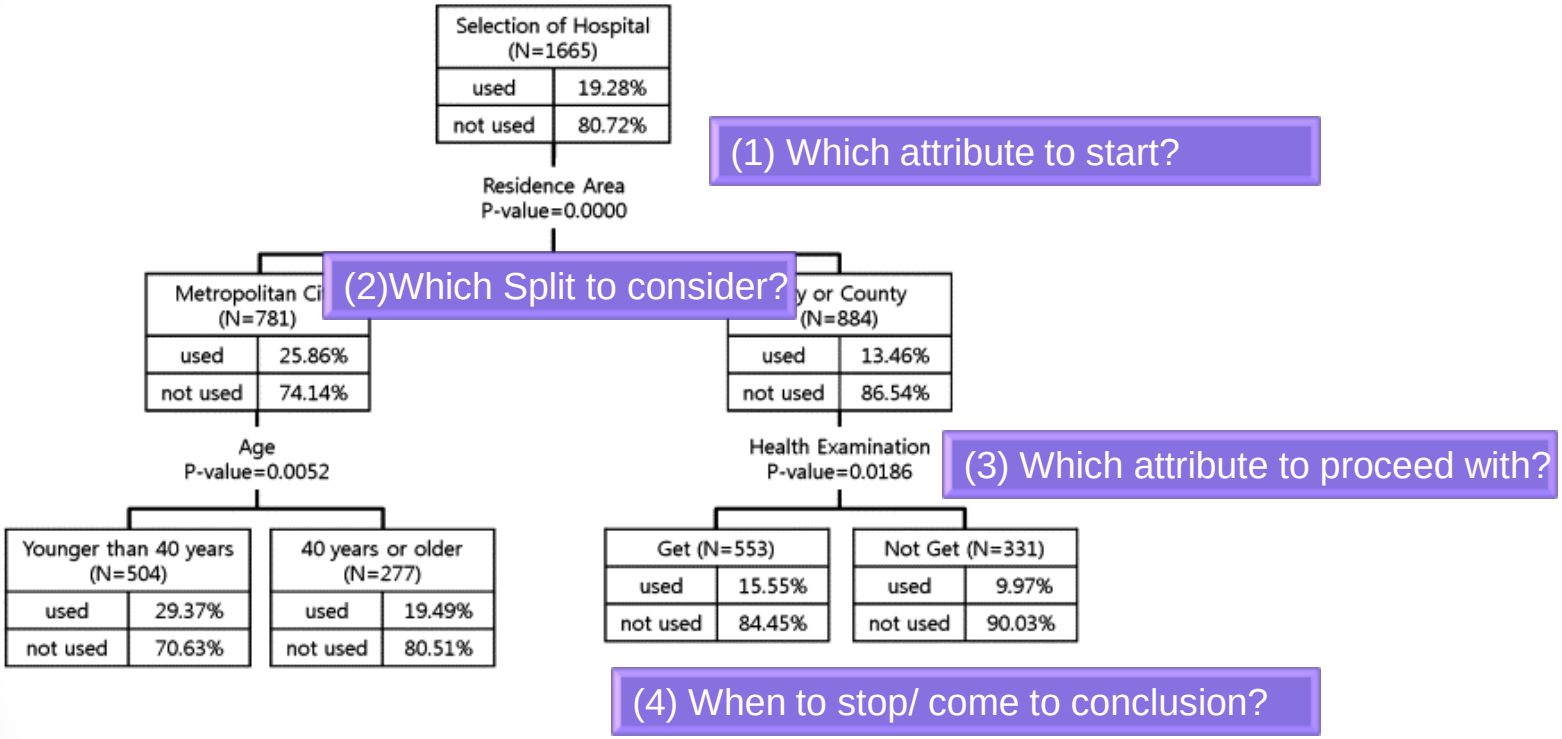
Decision Tree Types

- **Binary trees** – only two choices in each split. Can be non-uniform (uneven) in depth
- **N-way trees** or ternary trees – three or more choices in at least one of its splits (3-way, 4-way, etc.)

Decision Tree Algorithms

- Hunt's Algorithm (one of the earliest)
- CART
- ID3
- C4.5
- SLIQ
- SPRINT
- **CHAID**

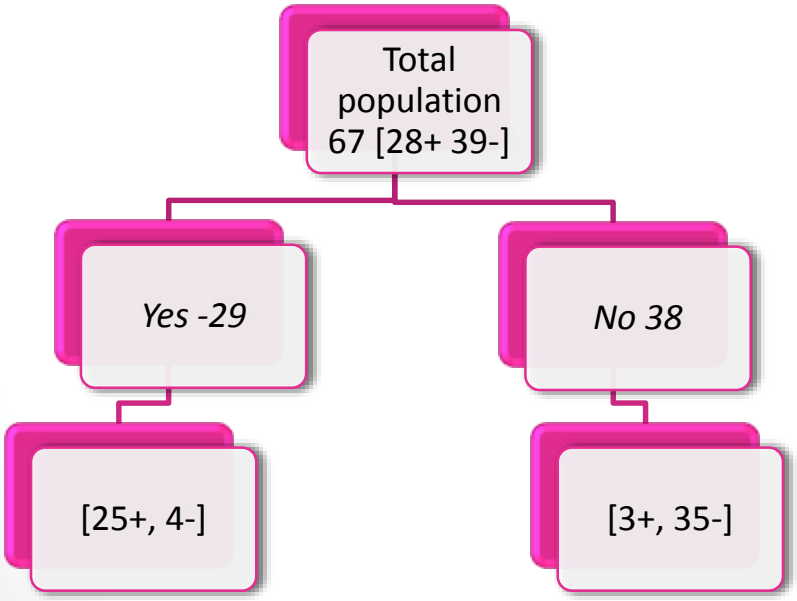
Decision Trees Algorithm – Answers?



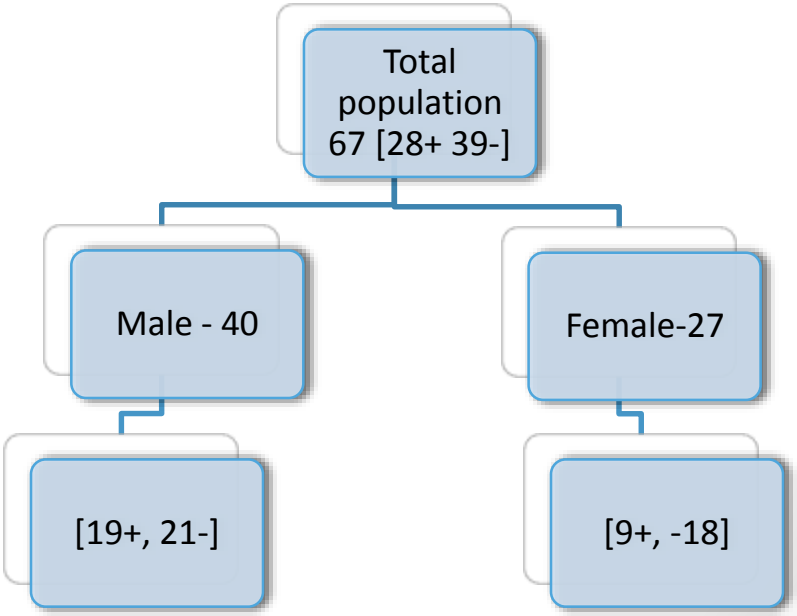
Example: Splitting with respect to an attribute

- **Example:**We want to sell some appartments. The population contains 67 persons. We want to test response based on the spilts given two attributes 1)Owning a car 2)gender

Split With Respect to 'Owning a car'



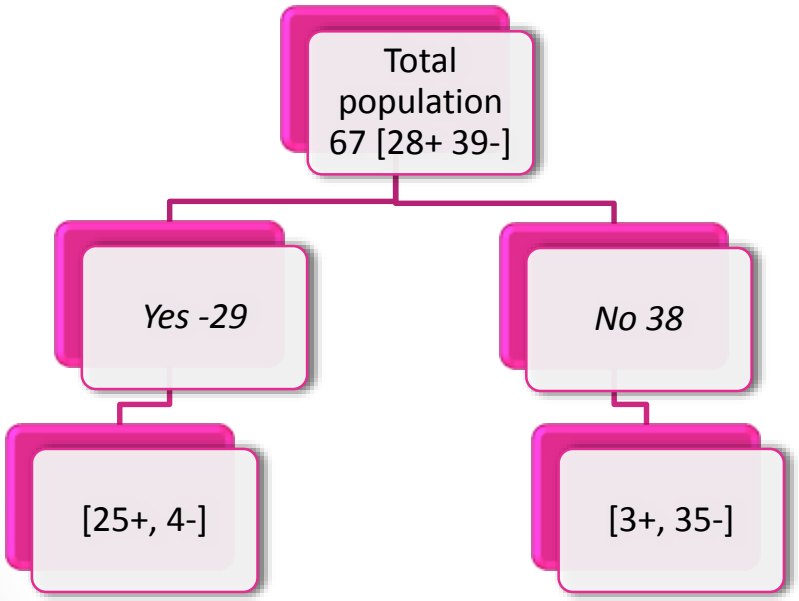
Split With Respect to 'Gender'



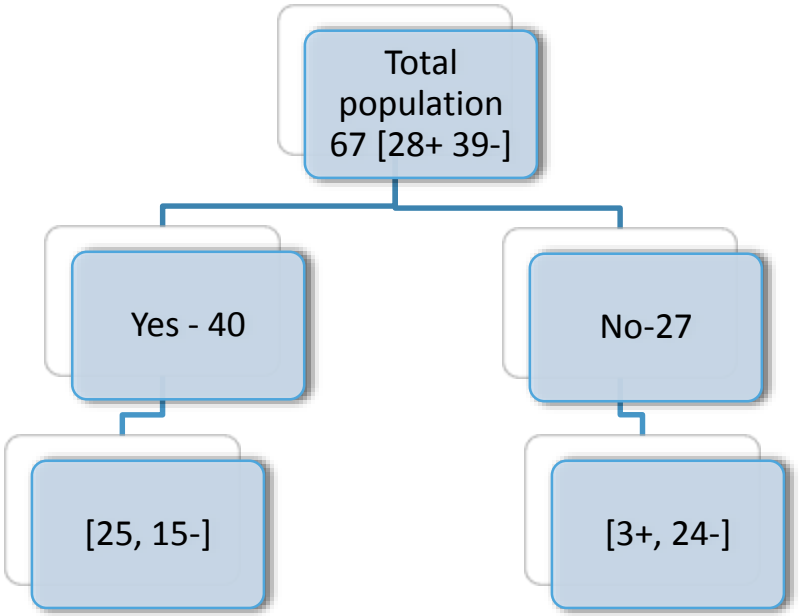
- In this example there are 21 positive responses from people owning a car & 8 positive responses from people who doesn't own a car

Example: Splitting with respect to an attribute

Split With Respect to 'Owning a car'



Split With Respect to 'marital status'



- Which is the best split attribute? Owing a car / Gender/ Marital status?
- The one which removes maximum impurity

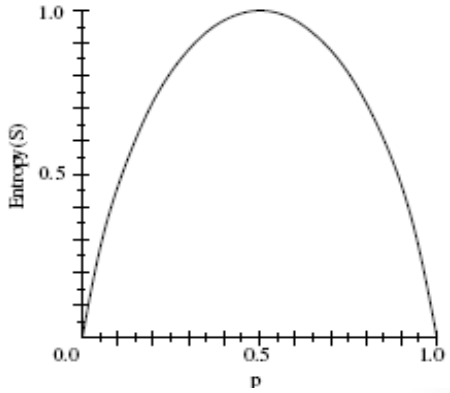
Best Splitting attribute

- The splitting is done always based on the binary objective variable(0/1 type)
- The best split at root(or child) nodes is defined as one that does the best job of separating the data into groups **where a single class**(either 0 or 1) **predominates** in each group
- Measure used to evaluate a potential split is **purity**
- The best split is one that increases purity of the sub-sets by the greatest amount

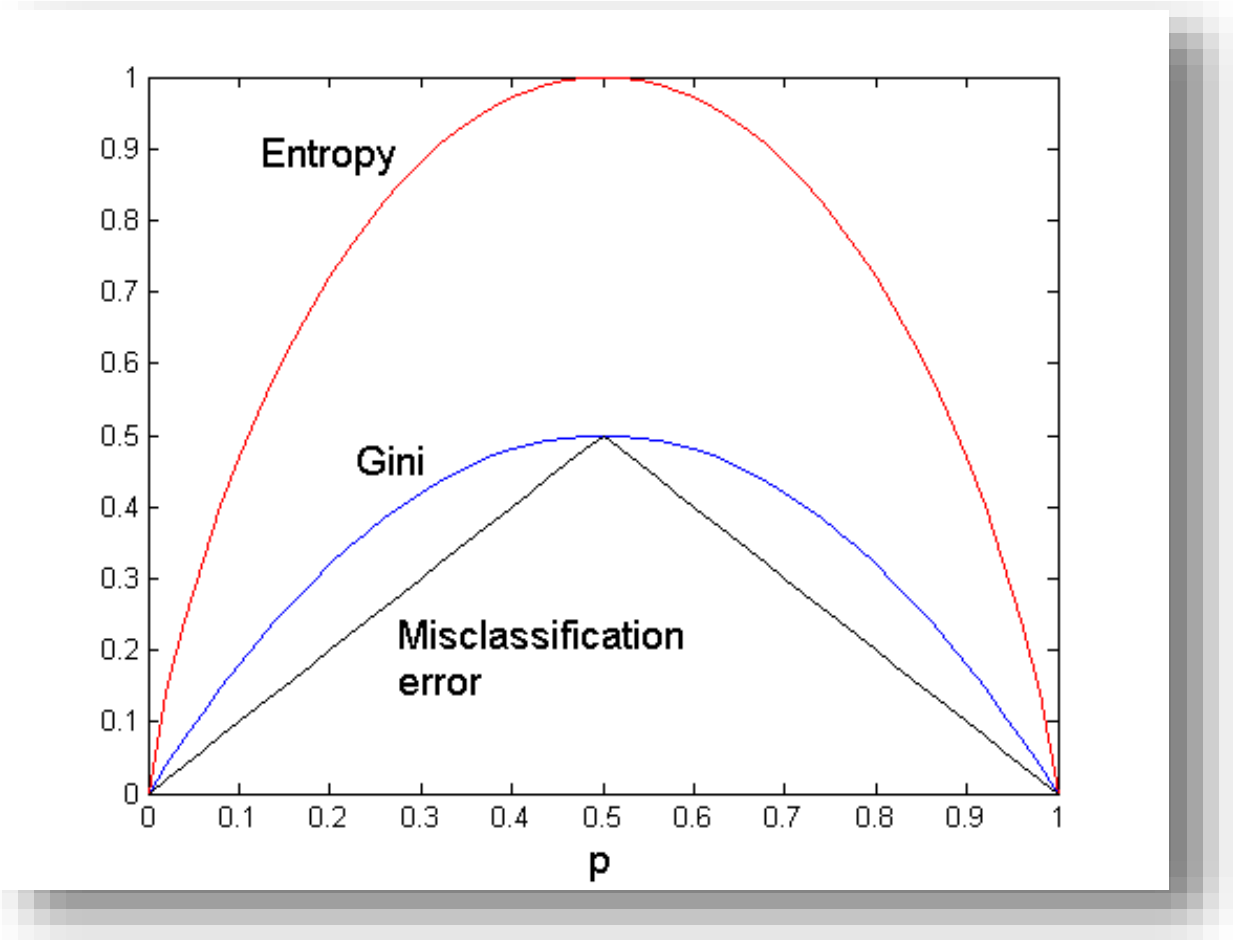
Purity (Diversity) Measures:

- **Entropy:** Characterizes the impurity/diversity of segment (an arbitrary collection of observations)
  - Measure of uncertainty/Impurity
  - Expected number of bits to resolve uncertainty
  - Entropy measures the information amount in a message
  - S is a sample of training examples, p_+ is the proportion of positive examples, p_- is the proportion of negative examples
  - Entropy(S) = $-p_+ \log_2 p_+ - p_- \log_2 p_-$
  - General formula for Entropy(S) = $-\Sigma p_j \times \log_2(p_j)$
  - Entropy is maximum when p=0.5

- Chi-square measure of association
- Gini Index : Gini(T) = $1 - \Sigma p_j^2$
- Information Gain Ratio
- Misclassification error



All Diversity Measures are maximum when p=0.5



Deciding the best split

Using Entropy

- Entropy([28+,39-]) Ovearll = -28/67 $\log_2$ 28/67 – 39/67 $\log_2$ 39/67 = 98% (Impurity)
- Entropy([25+,4-]) Owing a car = 57%
- Entropy([3+,35-]) No car = 40%
- Entropy([19+,21-]) Male= 99%
- Entropy([9+,18-]) Female = 91%
- Entropy([25+,15-]) Married= 95%
- Entropy([3,24-]) Unmarried = 50%

- Information Gain= entropyBeforeSplit – entropyAfterSplit
- Easy way to understnd Information gain= (ovearll entropy) – (sum of weighted entopy at each node)
- Attribute with maximum information is best split attribute

Using Chi Square Measure for association/Degree of independence

- Chi-square for owning a car **= 2.71**
- Chi square for Gender **= 0.09**
- Chi square for marital status =**1.19**

- The attribute with maximum chi square is the best split attibute

The Decision tree algorithm

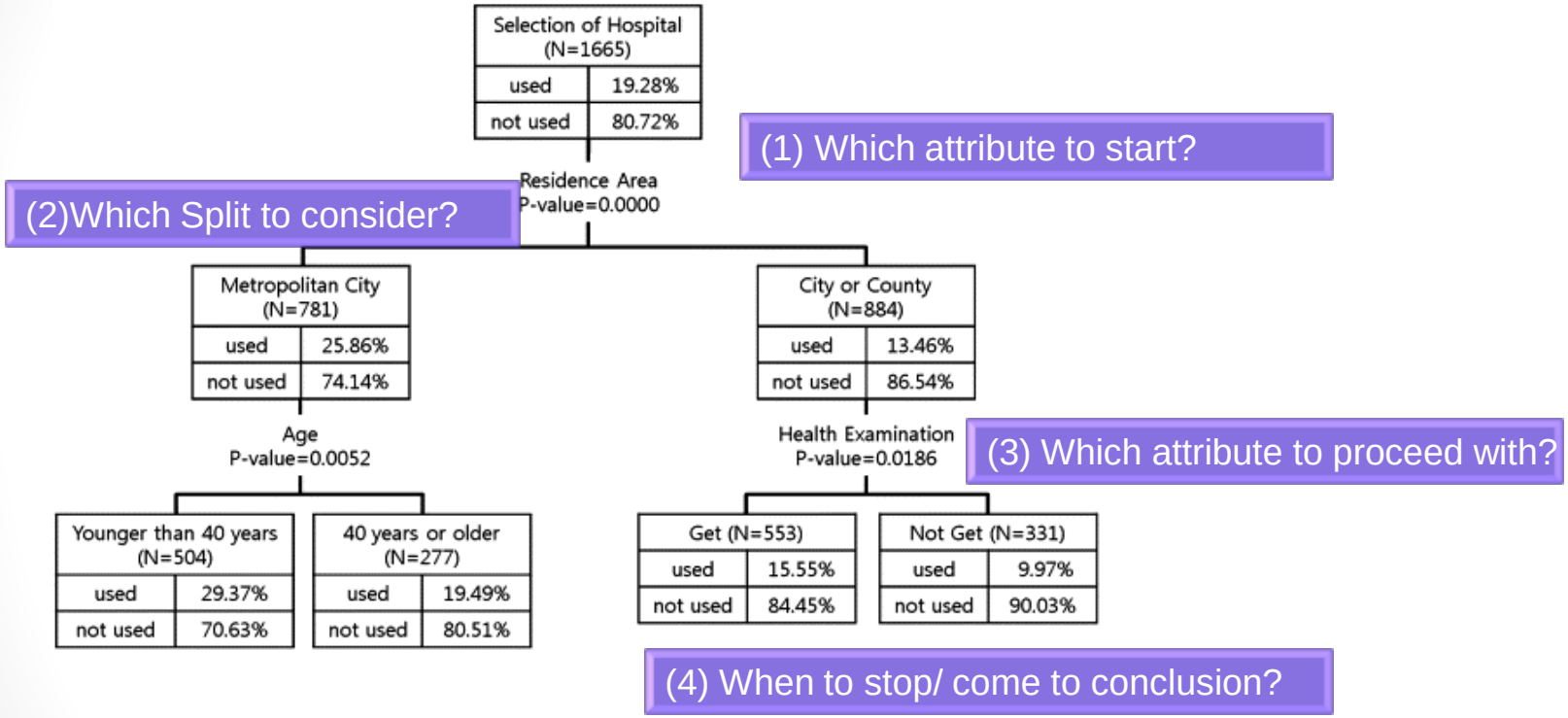
Until stopped:

1. Select a leaf node
2. Select one of the unused attributes
   - Partition the node population and calculate information gain.
   - Find the split with maximum information gain for a this attribute
3. Repeat this for all attributes
   - Find the best splitting attribute along with best split rule
4. Spilt the node using the attribute
5. Go to each child node and repeat step 2 to 4

Stopping criteria:

- Each leaf-node contains examples of one type
- Algorithm ran out of attributes
- No further significant information gain

Decision Trees Algorithm – Answers?



Tree validation

- Confusion Matrix:

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

Tree validation

- Sometimes cost of misclassification is not equal for both good and bad.
- We use a cost matrix along with confusion matrix
- C(i|j): Cost of misclassifying class j example as class i

		PREDICTED CLASS	
ACTUAL CLASS	C(i|j)	**Class=Yes**	**Class=No**
	Class=Yes	C(Yes|Yes)	C(No|Yes)
	Class=No	C(Yes|No)	C(No|No)

Tree Validation

- Model-1 and Model-2 which one of them is better?

Cost Matrix	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	+	-
	+	-1	100
	-	1	0

Model M_1	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	150	40
	-	60	250

Accuracy = 80%

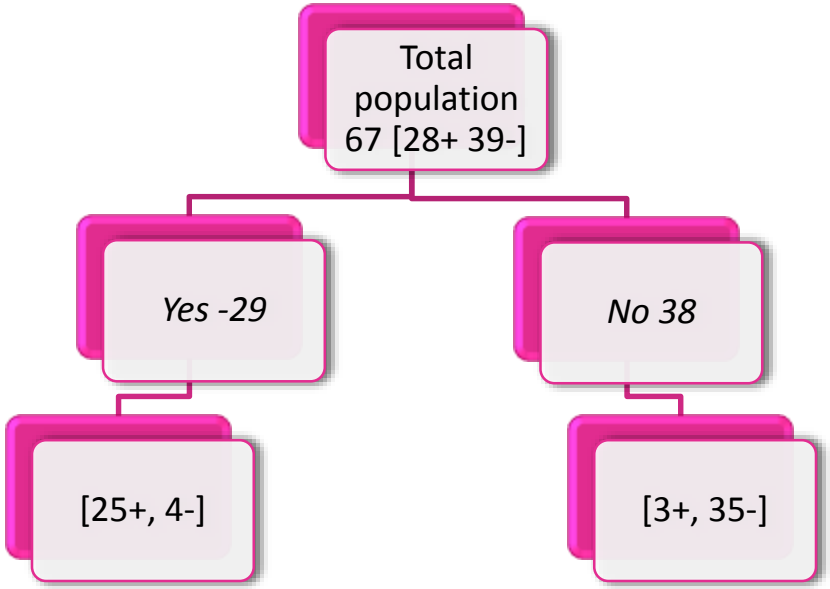
Cost = 3910

Model M_2	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	250	45
	-	5	200

Accuracy = 90%

Cost = 4255

Validation - Example



If having a car is the criteria for buying a house then

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	25 (TP)	3 (FN)
	Class=No	4 (FP)	35 (TN)

$$\text{Accuracy} = \frac{a+d}{a+b+c+d}$$

$$\text{Accuracy} = \frac{60}{67} = 90\%$$

CHAID Segmentation

- CHAID- Chi-Squared Automatic Interaction Detector
- CHAID is a non-binary decision tree.
- The decision or split made at each node is still based on a single variable, but can result in multiple branches.
- The split search algorithm is designed for categorical variables.
- Continuous variables must be grouped into a finite number of bins to create categories.
  - A reasonable number of “equal population bins” can be created for use with CHAID.
  - ex. If there are 1000 samples, creating 10 equal population bins would result in 10 bins, each containing 100 samples.

- A Chi-square value is computed for each variable and used to determine the best variable to split on.

CHAID Algorithm

Until stopped:

1. Select a node
2. Select one of the unused attributes
   - Partition the node population and calculate Chi square value
   - Find the split with maximum Chi square for a this attribute
3. Repeat this for all attributes
   - Find the best splitting attribute along with best split rule
4. Spilt the node using the attribute
5. Go to each child node and repeat step 2 to 4

Stopping criteria:

- Each leaf-node contains examples of one type
- Algorithm ran out of attributes
- No further significant information gain

Over fitting

- Model is too complicated
- Model works well on training data and performs very badly on test data
- Over fitting results in decision trees that are more complex than necessary
- Training error no longer provides a good estimate of how well the tree will perform on previously unseen records
- Need new ways for estimating errors

Avoiding Over fitting-Pruning

- Pre-Pruning (Early Stopping Rule)
  - **Stop the algorithm** before it becomes a fully-grown tree
  - Typical stopping conditions for a node:
    - Stop if all instances belong to the same class
    - Stop if all the attribute values are the same
  - More restrictive conditions:
    - Stop if number of instances is less than some user-specified threshold
    - Stop if expanding the current node does not improve impurity measures (e.g., Gini or information gain).
- Post-pruning
  - Grow decision tree to its entirety
  - Trim the nodes of the decision tree in a bottom-up fashion
  - If generalization error improves after trimming, replace sub-tree by a leaf node.