Data Mining

Classification I – Decision Tree (Part A)

Dr. Jason T.L. Wang, Professor

Department of Computer Science

New Jersey Institute of Technology

/

Classification: Definition and Applications

Definition

A classifier is trained by a given set of training data (in which each object has a label and belongs to a predefined class) and then is able to assign an unlabeled object to one of the classes or indicate the unlabeled object does not belong to any of the existing classes.

Applications: credit approval, target marketing, disease treatment

Classification (supervised learning) vs. clustering (unsupervised learning)

Decision Tree Induction (an example classifier)

- Decision tree
  - Internal node represents a test on an attribute
  - Branch represents an outcome of the test
  - Leaf nodes represent class labels
  - Good for categorical attributes with non-continuous values
  - ID3, C4.5
- How to generate a decision tree?
- How to use a decision tree?

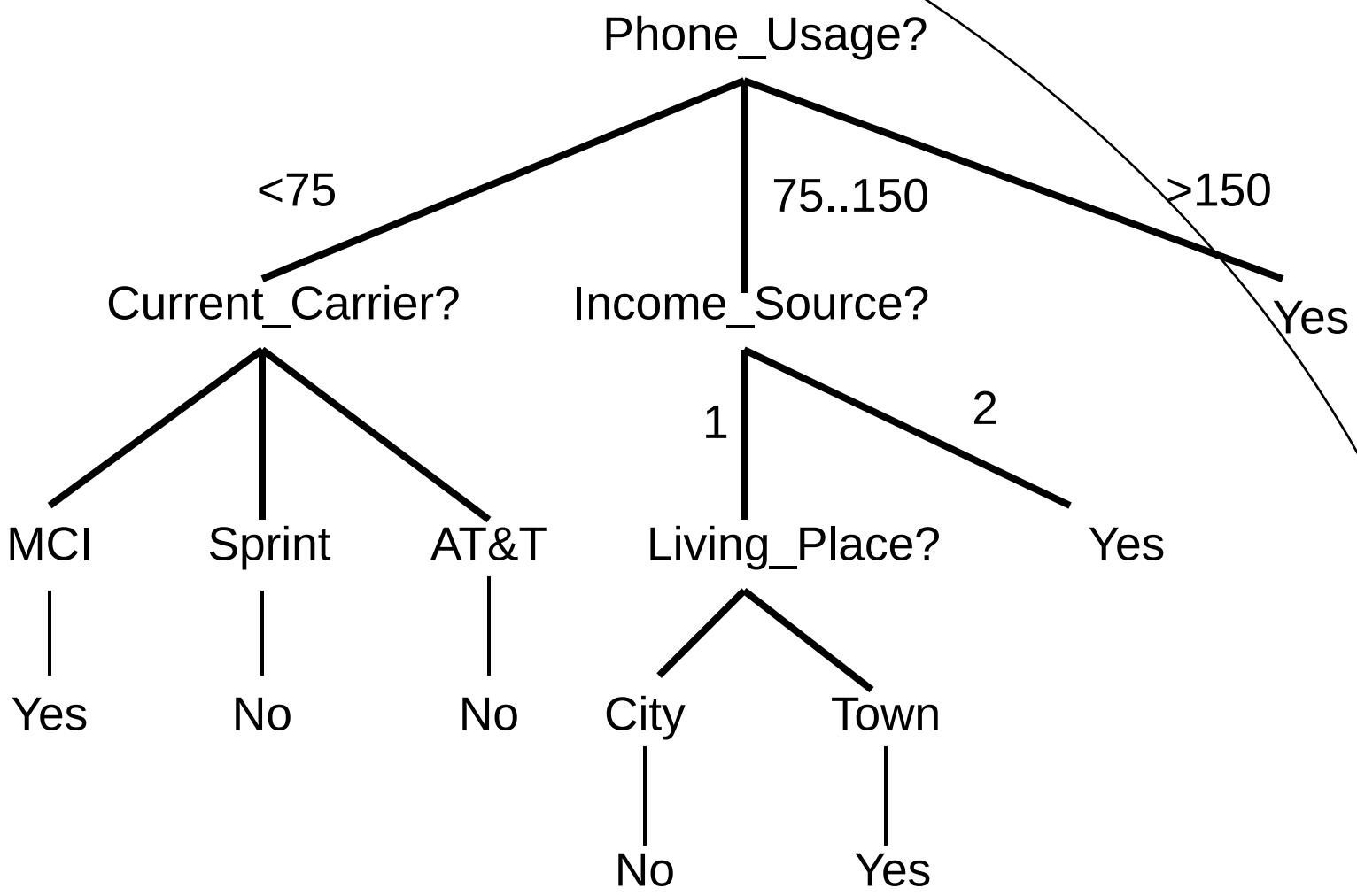
Training Dataset Example

“Change to new AT&T calling plan”

Phone_Usage	Income_Source	Living_Place	Current_Carrier	Change_Plan
<75	1	City	MCI	Yes
75..150	2	Town	MCI	Yes
<75	1	City	Sprint	No
>150	2	Town	AT&T	Yes
75..150	1	City	MCI	No
75..150	2	Town	AT&T	Yes
<75	2	Town	AT&T	No
>150	2	City	Sprint	Yes
<75	1	City	AT&T	No
75..150	1	Town	MCI	Yes
75..150	2	City	Sprint	Yes
>150	1	Town	AT&T	Yes
<75	1	Town	AT&T	No
<75	2	City	Sprint	No
75..150	2	City	MCI	Yes

Output: A Decision Tree for

“Change to new AT&T calling plan”



Algorithm

- At each level, select an attribute to branch on (always select the attribute with the highest information gain).
- Suppose there are two classes P (positive) and N (negative) where P contains p training examples and N contains n training examples.

$$I(p,n) = -\frac{p}{p+n}\log_2\frac{p}{p+n} - \frac{n}{p+n}\log_2\frac{n}{p+n}$$

Algorithm (cont.)

- Assume that using attribute A a set *S* will be partitioned into subsets $\{S_1, S_2, \ldots, S_v\}$
  - If S_i contains p_i examples of *P* and n_i examples of *N*, the entropy, or the expected information needed to classify objects in all subtrees S_i is

$$E(A) = \sum_{i=1}^{v} \frac{p_i + n_i}{p + n} I(p_i, n_i)$$

$$Gain(A) = I(p, n) - E(A)$$

End of
Decision Tree Module (Part A)