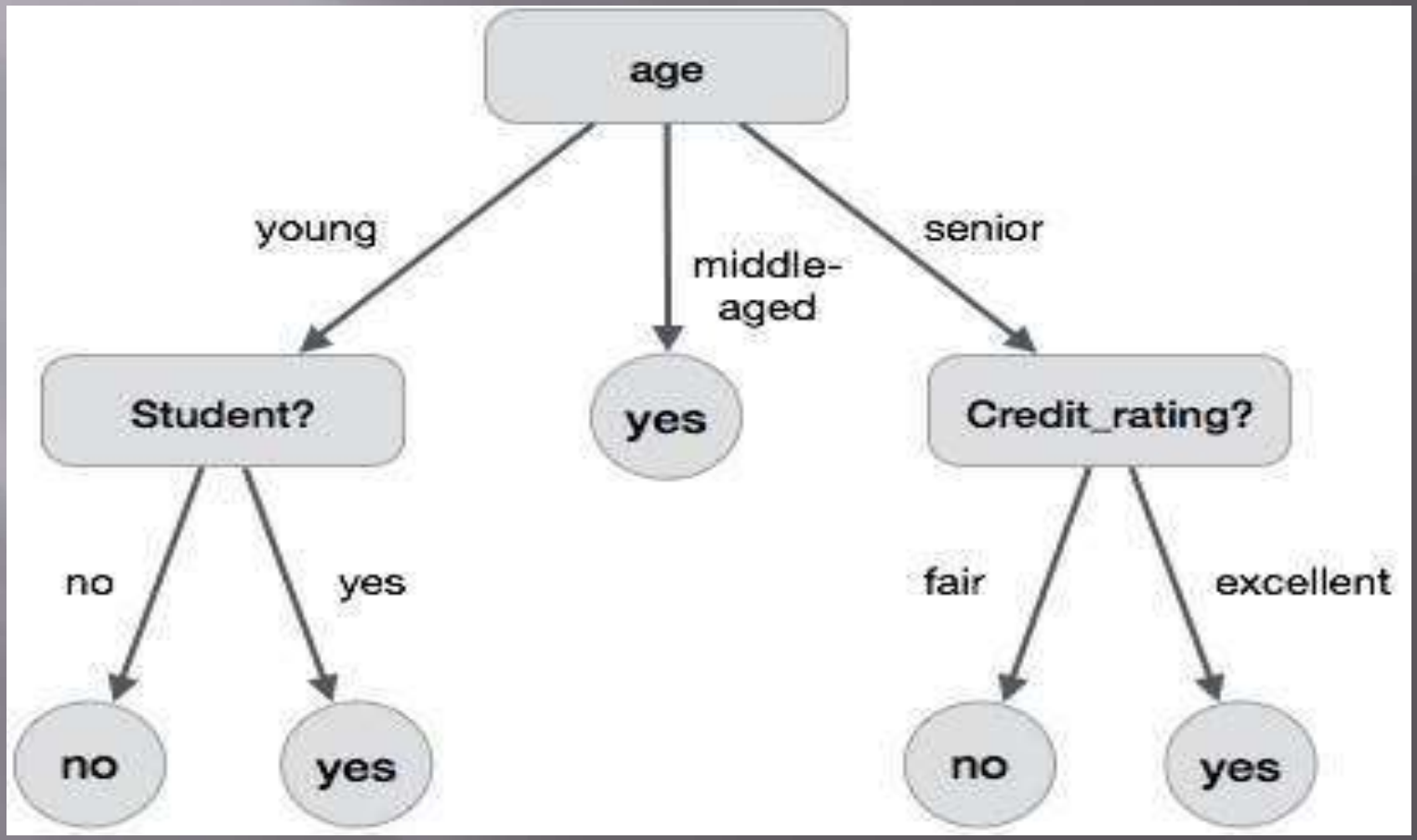
# Data Mining - Decision Tree Induction

# Decision Tree

A decision tree is a structure that includes a root node, branches, and leaf nodes. Each *internal node* denotes a *test on an attribute*, each *branch* denotes the *outcome of a test*, and each *leaf node* holds a *class label*. The topmost node in the tree is the root node.

**The following decision tree is for the concept buy_computer that indicates whether a customer at a company is likely to buy a computer or not. Each internal node represents a test on an attribute. Each leaf node represents a class.**



# Decision Tree Induction Algorithm

## Generating a decision tree form training tuples of data partition D

Algorithm : Generate_decision_tree

Input:

Data partition, D, which is a set of training tuples
and their associated class labels.
attribute_list, the set of candidate attributes.
Attribute selection method, a procedure to determine the
splitting criterion that best partitions that the data
tuples into individual classes. This criterion includes a
splitting_attribute and either a splitting point or splitting subset.

Output: A Decision Tree

**Method**

```
create a node N;
if tuples in D are all of the same class, C then
  return N as leaf node labelled with class C;
 if attribute list is empty then
  return N as leaf node with labelled
  with majority class in D;|| majority voting
apply attribute_selection_method(D, attribute list)
to find the best splitting criterion;
label node N with splitting criterion;
if splitting attribute is discrete-valued and
  multiway splits allowed then  // no restricted to binary trees
attribute list = splitting attribute; // remove splitting attribute
for each outcome j of splitting criterion
  // partition the tuples and grow subtrees for each partition
  let Dj be the set of data tuples in D satisfying outcome j; // a partition
  if Dj is empty then
    attach a leaf labelled with the majority
    class in D to node N;
  else
    attach the node returned by Generate
    decision tree(Dj, attribute list) to node N;
  end for
return N;
```

## Tree Pruning

Tree pruning is performed in order to remove anomalies in the training data due to noise or outliers. The pruned trees are smaller and less complex.

## Tree Pruning Approaches

There are two approaches to prune a tree −

- **Pre-pruning −** The tree is pruned by halting its construction early.
- **Post-pruning** - This approach removes a sub-tree from a fully grown tree.

## Cost Complexity

The cost complexity is measured by the following two parameters −

- Number of leaves in the tree, and

*Next class Bayesian Classification*

**Thank You**