LECTURE 27: DECISION TREES

- Objectives:
  - Why do we need a smart algorithm to reduce the number of parameters? On what type of information should this smart algorithm operate?
  - Basic concepts of *classification and regression trees* (CART)
  - How do we apply them to acoustic modeling? What are the benefits?

This lecture combines material from the course textbook:

> X. Huang, A. Acero, and H.W. Hon, *Spoken Language Processing - A Guide to Theory, Algorithm, and System Development*, Prentice Hall, Upper Saddle River, New Jersey, USA, ISBN: 0-13-022616-5, 2001.

and these MS project presentations:

- J. Ngan, "Information Theory Based Decision Trees for Data Classification," *Master of Science Special Project Presentation*, December 10, 1998 (available at Ngan: MS project presentation)
- A. Le, "Bayesian Decision Tree for Classification of Nonlinear Signal Processing Problems," *Master of Science Special Project Presentation*, November 12, 1998 (available at Le: MS project presentation)

LECTURE 27: DECISION TREES

- Objectives:
  - Why do we need a smart algorithm to reduce the number of parameters? On what type of information should this smart algorithm operate?
  - Basic concepts of *classification and regression trees* (CART)
  - How do we apply them to acoustic modeling? What are the benefits?

This lecture combines material from the course textbook:

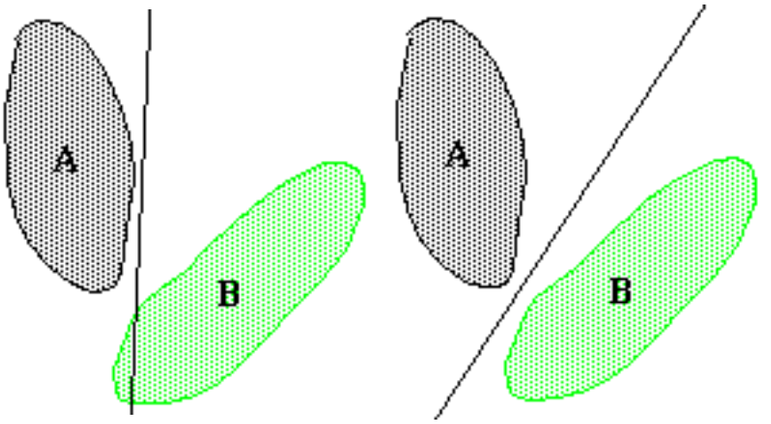
> X. Huang, A. Acero, and H.W. Hon, *Spoken Language Processing - A Guide to Theory, Algorithm, and System Development*, Prentice Hall, Upper Saddle River, New Jersey, USA, ISBN: 0-13-022616-5, 2001.

and these MS project presentations:

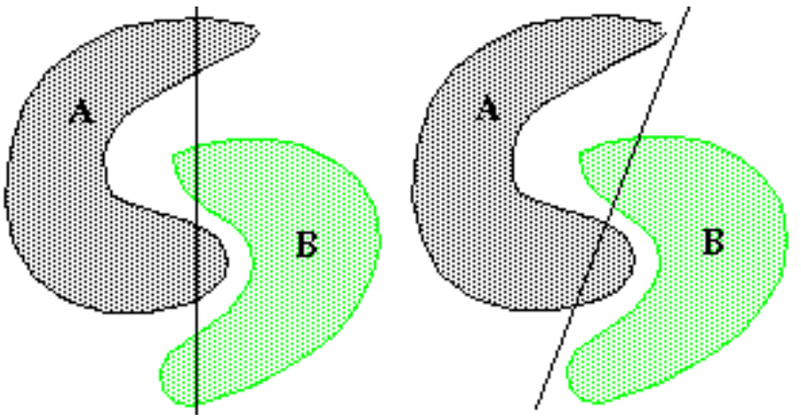
- J. Ngan, "Information Theory Based Decision Trees for Data Classification," *Master of Science Special Project Presentation*, December 10, 1998 (available at Ngan: MS project presentation)
- A. Le, "Bayesian Decision Tree for Classification of Nonlinear Signal Processing Problems," *Master of Science Special Project Presentation*, November 12, 1998 (available at Le: MS project presentation)

DECISION TREES: A POWERFUL
DATA-DRIVEN CLASSIFICATION ALGORITHM

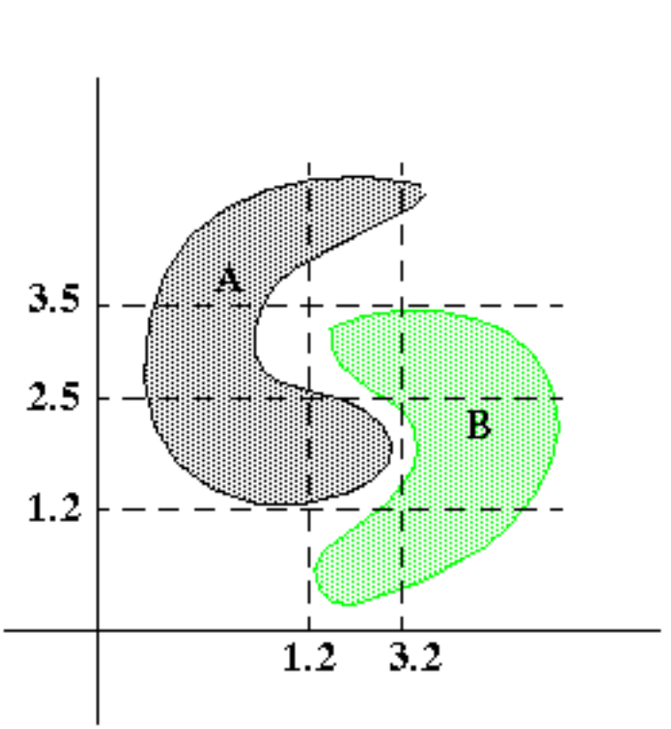
- When PCA fails:

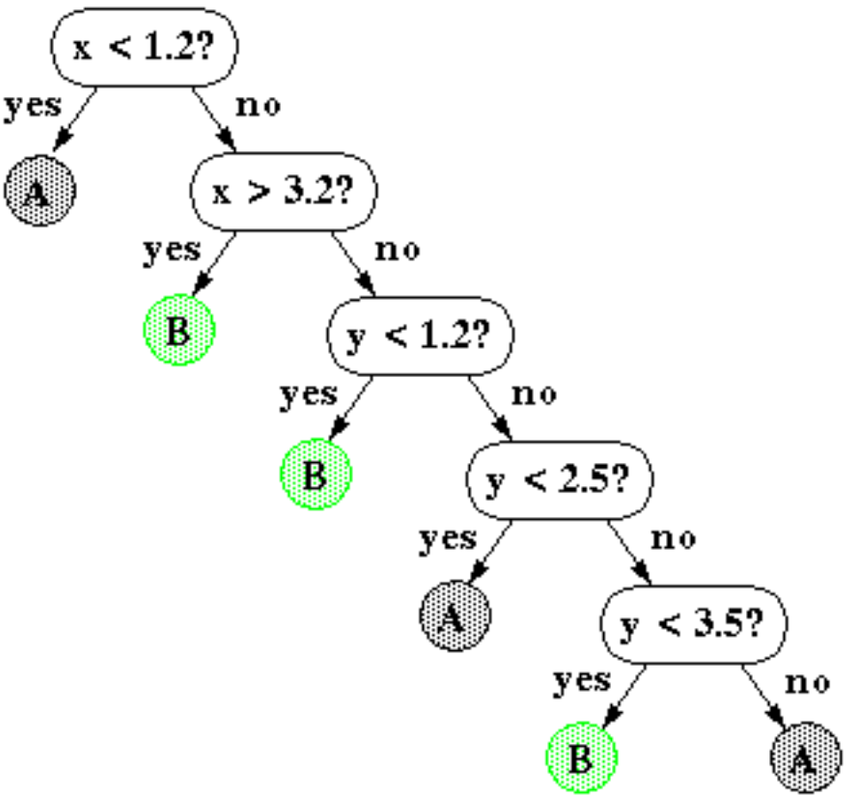


- and LDA fails:



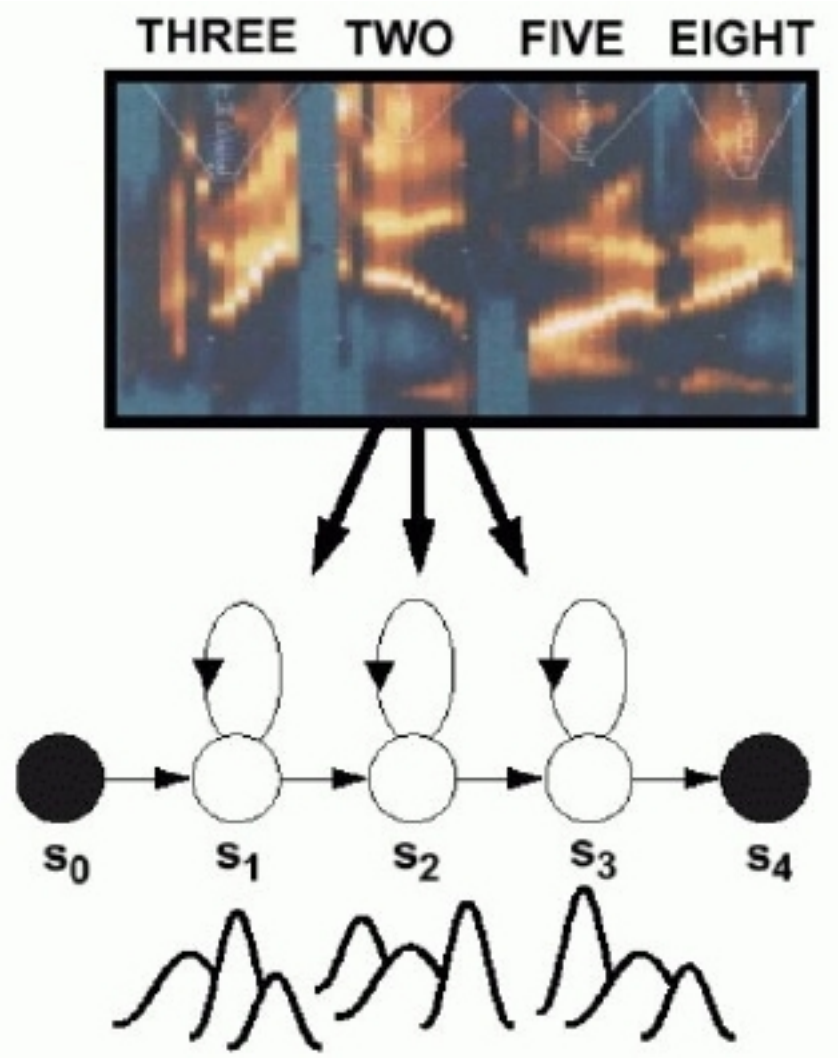
- we can imagine a more powerful data driven approach:





CONTROLLING PARAMETER COUNT IS AN IMPORTANT REALITY

- Acoustic models encode the temporal evolution of the features (spectrum).
- Gaussian mixture distributions are used to account for variations in speaker, accent, and pronunciation.
- Phonetic model topologies are simple left-to-right structures.
- Sharing model parameters is a common strategy to reduce complexity and avoid undertraining:

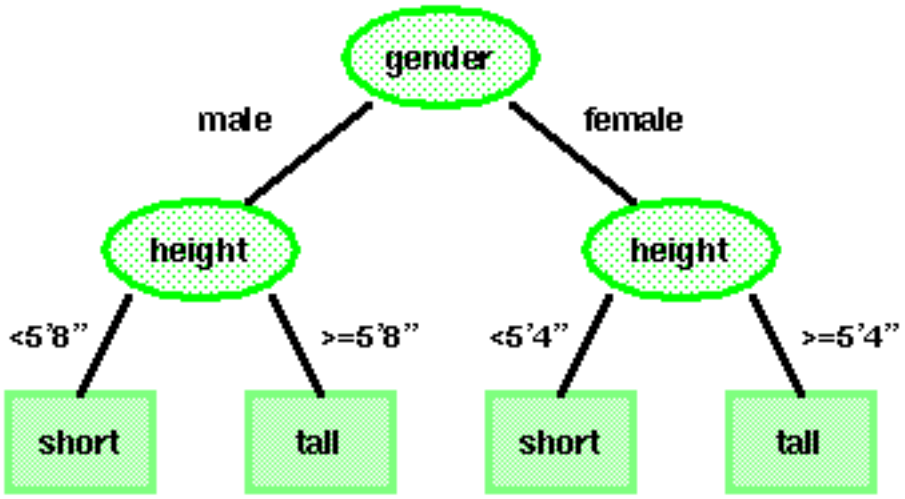


(39 features +
39 covariance values +
1 mixture weight) x
16 Gaussian per state x
3 states/phone x
80,000 CD phones =

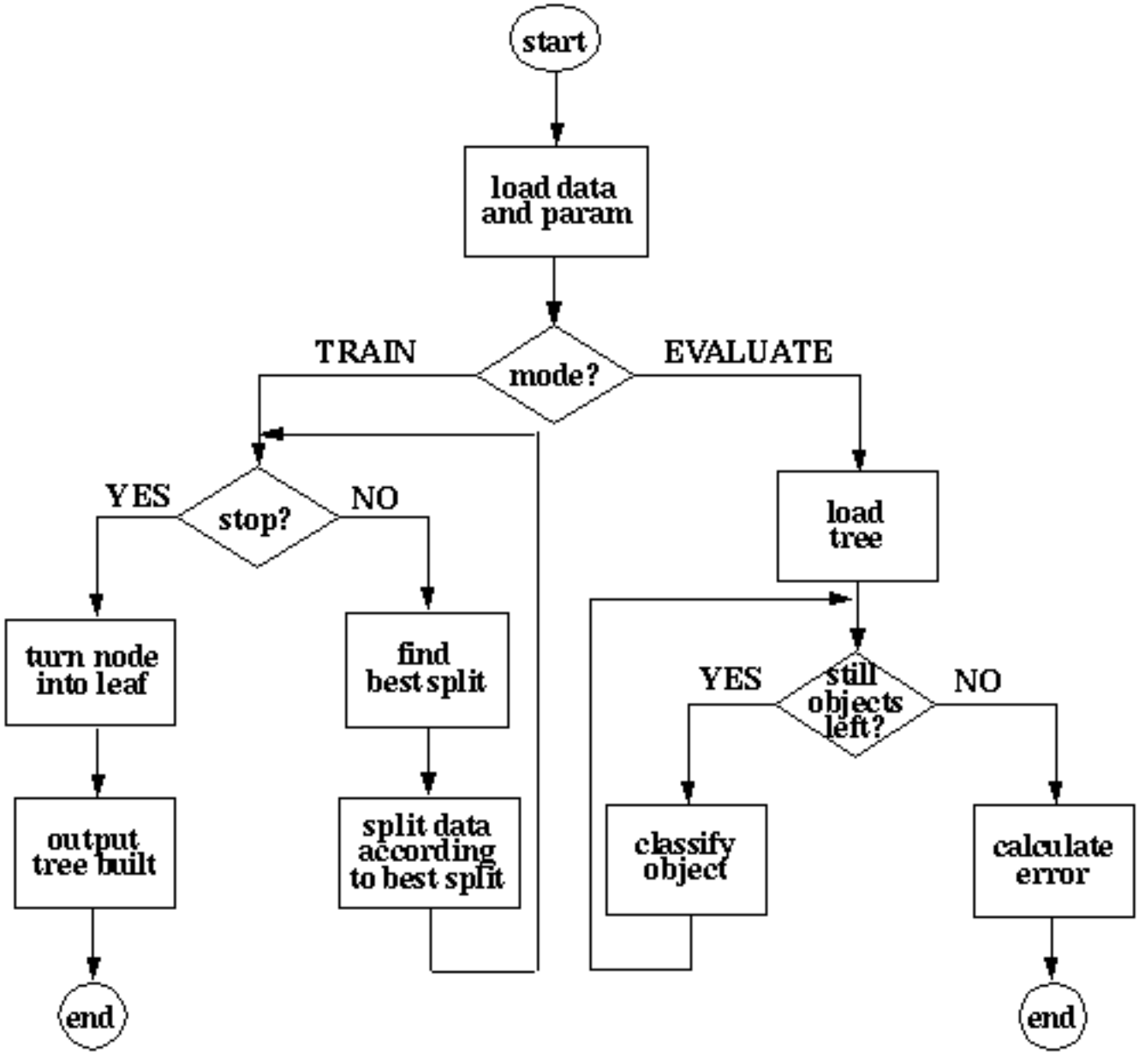
~300 x 10^6 parameters!

BASIC TERMINOLOGY

- A decision tree consists of nodes and leaves, with each leaf denoting a class.
- Classes (tall or short) are the outputs of the tree.
- Attributes (gender and height) are a set of features that describe the data.
- The input data consists of values of the different attributes. Using these attribute values, the decision tree generates a class as the output for each input data.



DATA-DRIVEN OPERATION



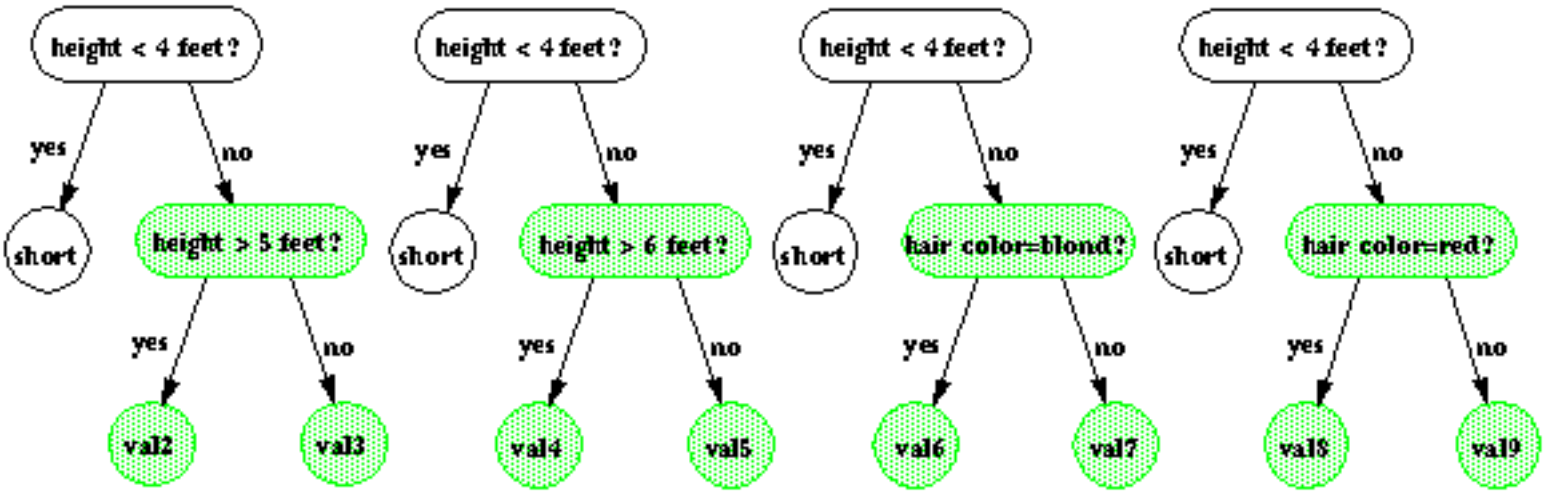
There are four important operations in constructing a decision tree:

- Question selection: choosing a set of questions to categorize your data (some algorithms can derive questions automatically).
- Splitting: partitioning data assigned to a node into N groups (N=2 for binary trees).
- Growing: expanding the tree to better represent the training data.
- Pruning: removing nodes to improve generalization.

In speech recognition, we operate on continuous-valued feature vectors, and use likelihood computations derived directly from HMM training. This is a major reason why decision trees are so popular in speech recognition systems - the implementation is very elegant.

SPLITTING CRITERIA

To split data at a node, we need to find the question that results in the greatest entropy reduction (removes uncertainty in the data):



In speech recognition, we can show this amounts to maximizing the increase in likelihood:

dL = L(parent) - L(left child) - L(right child)

These likelihoods can be computed from the state occupancies computed during training (see decision tree-based state tying for a detailed derivation and the important references).

GROWING THE TREE

We typically grow the tree by successively splitting each node until nodes can no longer be split. Though this is locally optimal, it is not globally optimal. Nevertheless this produces useful trees with minimum computational complexity.

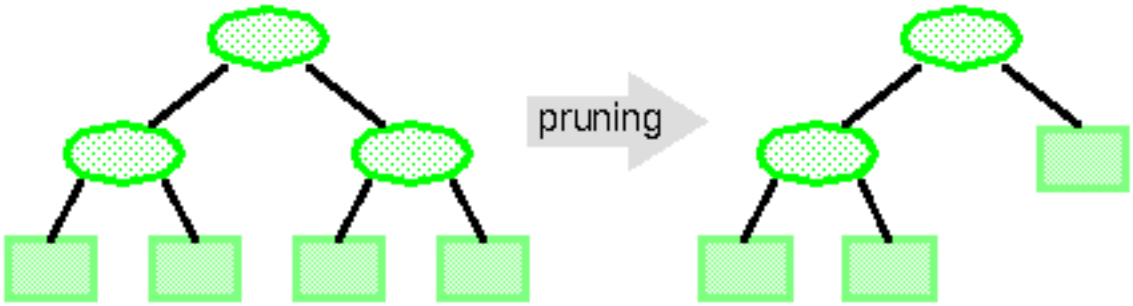
We can continue splitting nodes until:

- No more splits are possible (all samples at a node belong in the same class).
- The greatest likelihood increase (entropy reduction) falls below our pre-set threshold.
- The number of data samples falling in a leaf node falls below some threshold.

Nodes which can no longer be split are declared **terminal nodes**. When all active nodes are terminal nodes, tree growing terminates.

PRUNING A TREE IMPROVES GENERALIZATION

The most fundamental problem with decision trees is that they "overfit" the data and hence do not provide good generalization. A solution to this problem is to prune the tree:



Cost-complexity pruning is a popular technique for pruning. Cost-complexity can be defined as:

$$R_\alpha(t) = R(T) + \alpha|\tilde{T}|$$

where $|\tilde{T}|$ represents the number of terminal nodes in the subtree.

Each node in the tree can be classified in terms of its impact on the cost-complexity if it were pruned. Nodes are successively pruned until certain thresholds (heuristics) are satisfied.

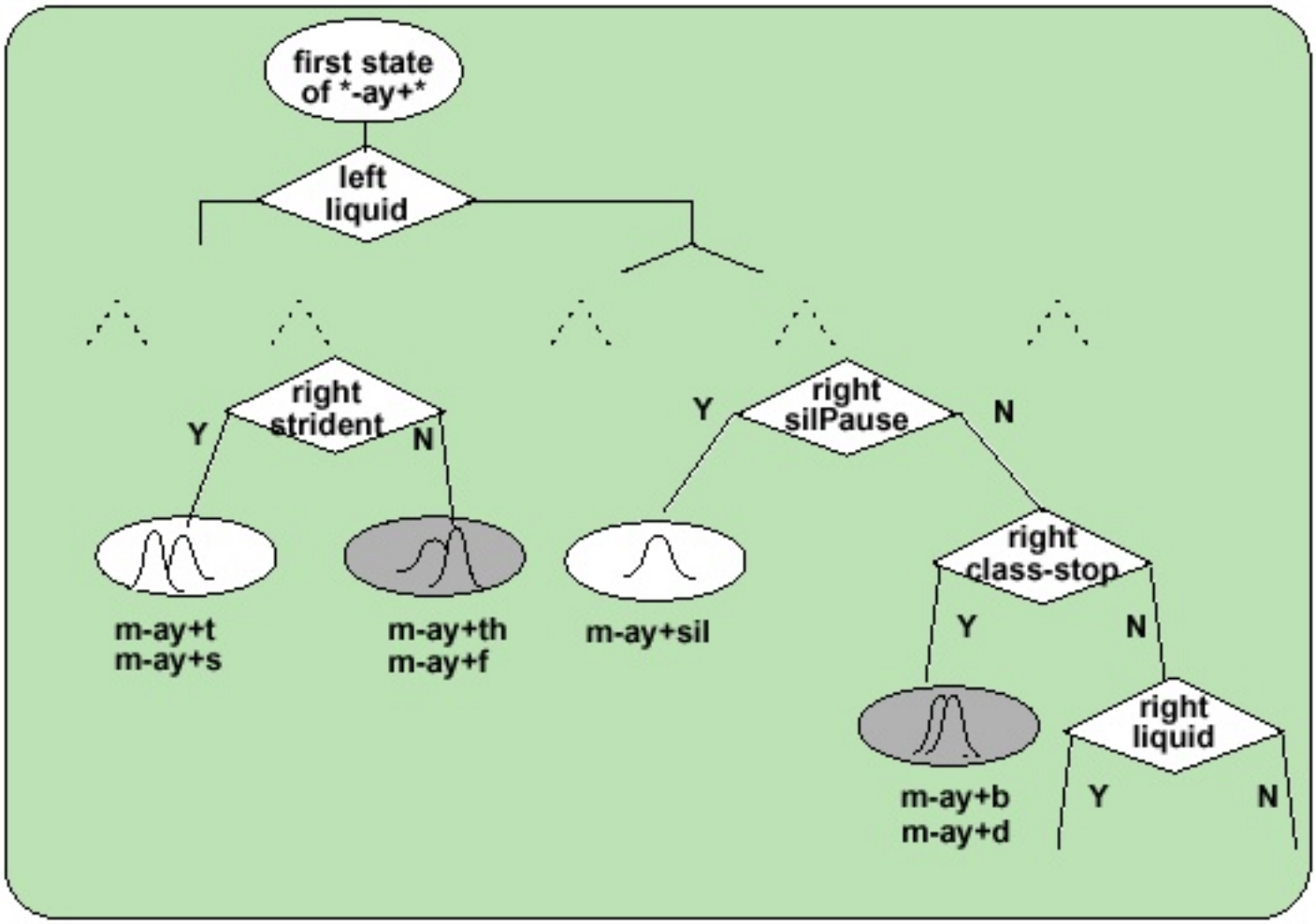
By pruning the nodes that are far too specific to the training set, it is hoped the tree will have better generalization. In practice, we use techniques such as cross-validation and held-out training data to better calibrate the generalization properties.

THE CART ALGORITHM

The classification and regression tree (CART) algorithm can be summarized as follows:

1. Create a set of questions that consists of all possible questions about the measured variables (phonetic context).
2. Select a splitting criterion (likelihood).
3. Initialization: create a tree with one node containing all the training data.
4. Splitting: find the best question for splitting each terminal node. Split the one terminal node that results in the greatest increase in the likelihood.
5. Stopping: if each leaf node contains data samples from the same class, or some pre-set threshold is not satisfied, stop. Otherwise, continue splitting.
6. Pruning: use an independent test set or cross-validation to prune the tree.

APPLICATION: ACOUSTIC MODELING



APPLICATION: PRONUNCIATION MODELING

Goal: Condition mappings from baseforms to pronunciations using as much linguistic information as possible (e.g., syllable boundaries). Train using hand-labeled data.