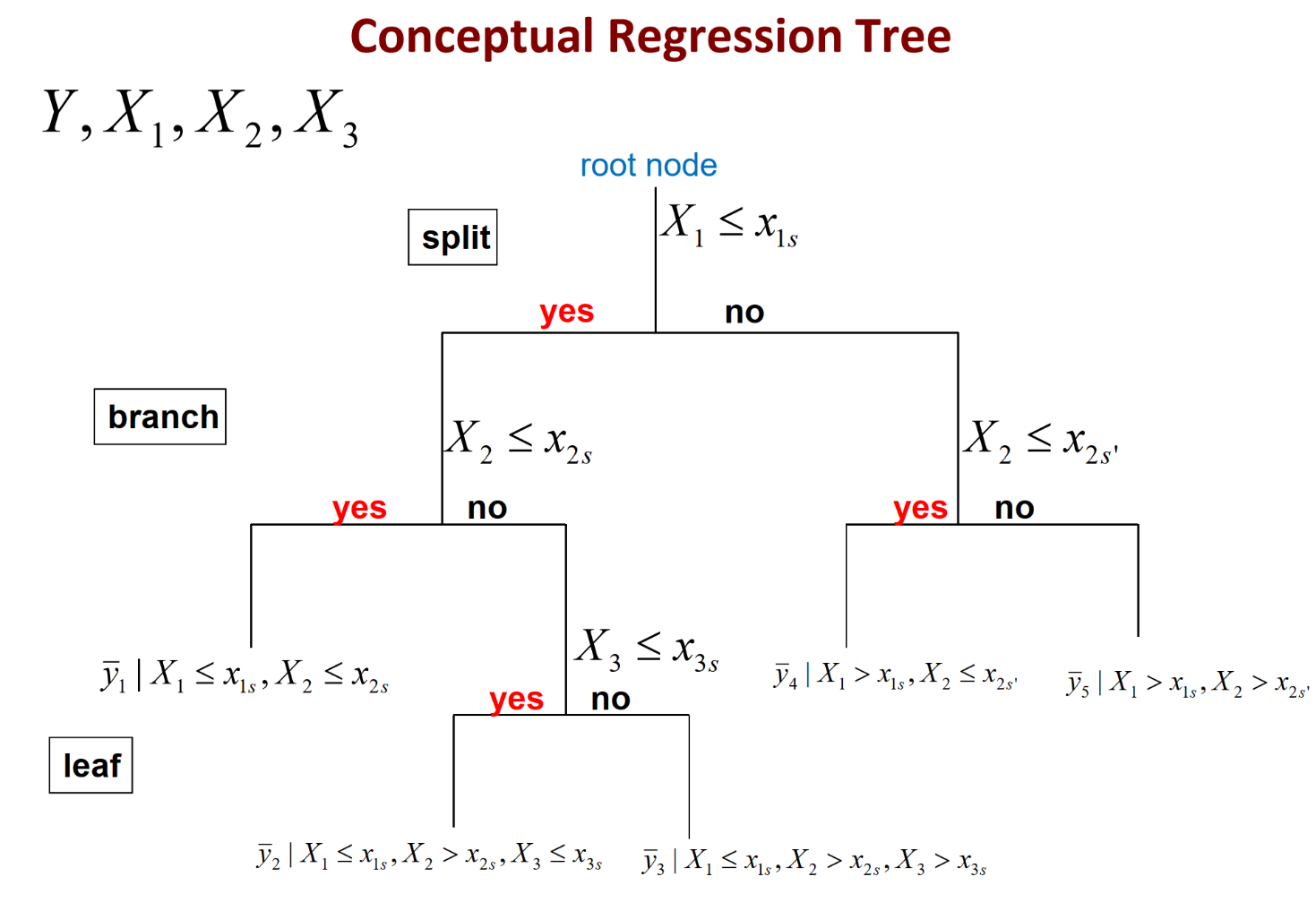
**Classification and Regression Trees (CART)**

stratify and segment the predictor space

use classification trees when the response variable is categorical

use regression trees when the response variable is continuous

**Regression Tree**



all the information is contained in the root node

divide the predictor space into distinct and nonoverlapping regions, , to minimize

fitted value in a region is the mean of the response values in this region

Recursive Binary Splitting

top-down greedy approach starts from the top of the tree and takes the best solution at each node

for each variable , has variability

after splitting data by

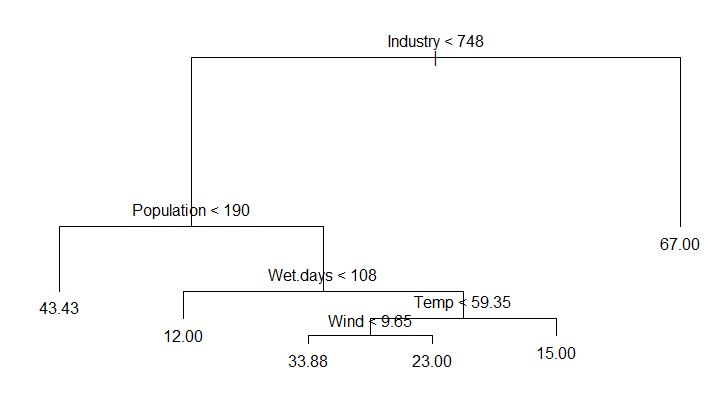
split each to minimize the new

repeat in each region recursively

= initial tree

= number of leaves/terminal nodes in initial tree

= mean of the training observations in region



length of branch is inversely related to its variance

to make a prediction for a given observation, use the mean or the mode of the training observations in the region it belongs to

e.g. the average pollution level of a city with less than 748 industry, population of less than 190,000, fewer than 108 wet days, average temperature of less than 59.35°, and wind of less than 9.65 is predicted to be 33.88

**Pruning**

recursive binary splitting usually overfits the data and creates too many splits, creating an overly complicated model that’s hard to interpret

a smaller tree with fewer splits can lead to better interpretation at the cost of higher residual deviance

build a large tree and fold the terminal branches back

Maximum Number of Leaves

specify total number of terminal nodes

compare mean deviance of new tree to mean deviance of original tree

Cost-Complexity

= penalty given to additional leaves

= penalty given to additional leaves

= number of leaves/terminal nodes in subtree

for each , choose the subtree nested in that minimizes

controls a trade-off between the subtree’s complexity and its fit to the training data

when , because measures training error

the higher is, the higher the penalty for having many terminal nodes so is minimized for smaller subtrees

as increases from 0, branches get pruned from the tree in a nested and predictable fashion

K-fold Cross-Validation

divide the number of training observations into folds

are training observations, are testing observations

for each fold 1) use recursive binary splitting to grow a large tree on the training data, stopping when each terminal node has fewer than a minimum number of observations

2) apply cost complexity pruning to the large tree to obtain a sequence of best subtrees as a function of for each fold

calculate the mean squared prediction error on the data in the last fold as a function of

average the results for each value of

choose and corresponding subtree with the least number of terminal nodes that minimize

**Tree vs. Linear Models**

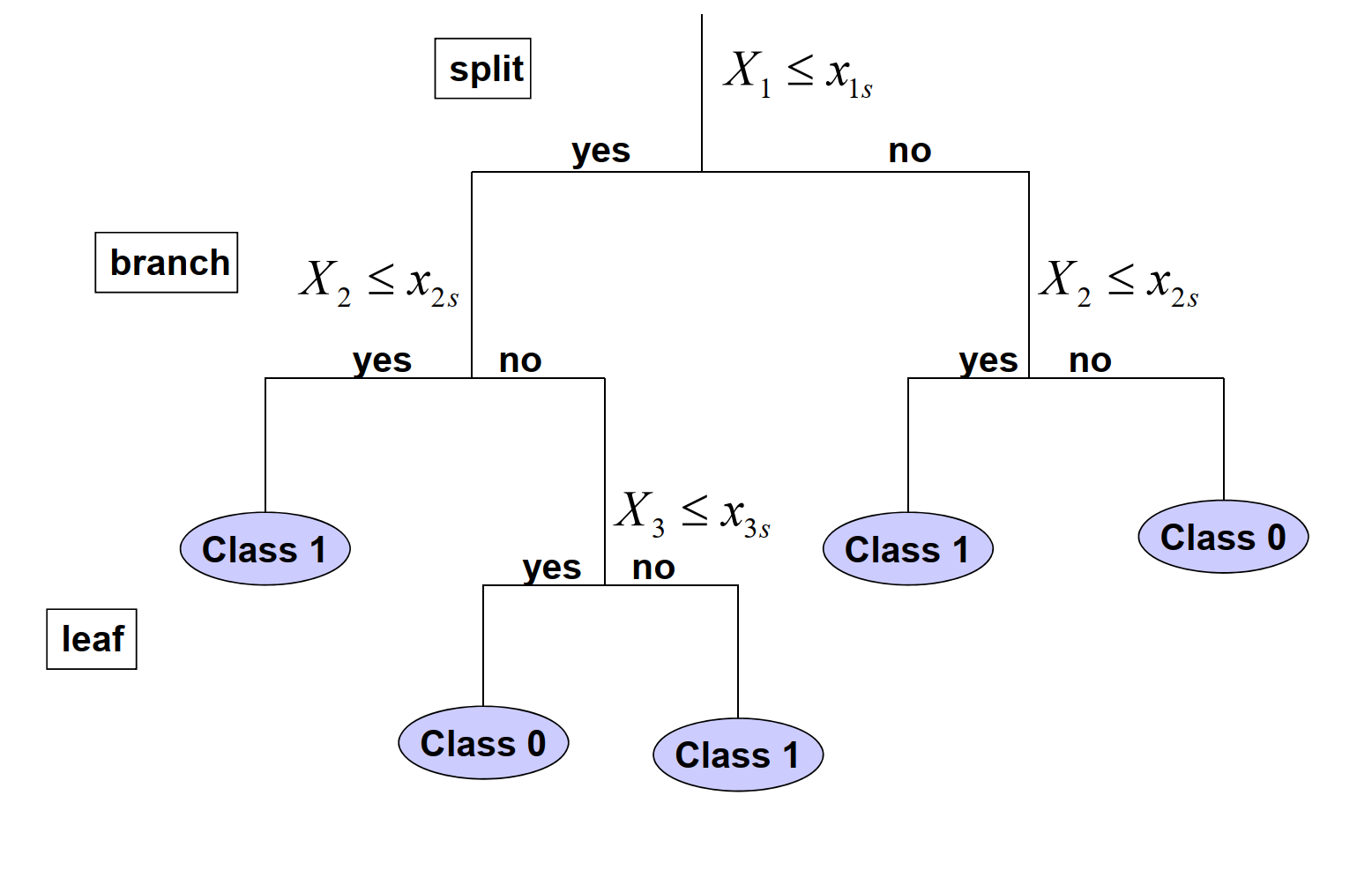
linear regression is appropriate if there is a linear relationship between covariates and response variable

CART works well for highly non-linear problems

regression trees are multiple linear models with dummy variables representing the branches

**Classification Tree**

**root node**



class labels for different samples

label at each terminal node is the modal value, value with the largest probability

use binary splits to minimize deviance/misclassification error and maximize information gain

= number of observations in the mth terminal node that belongs to class

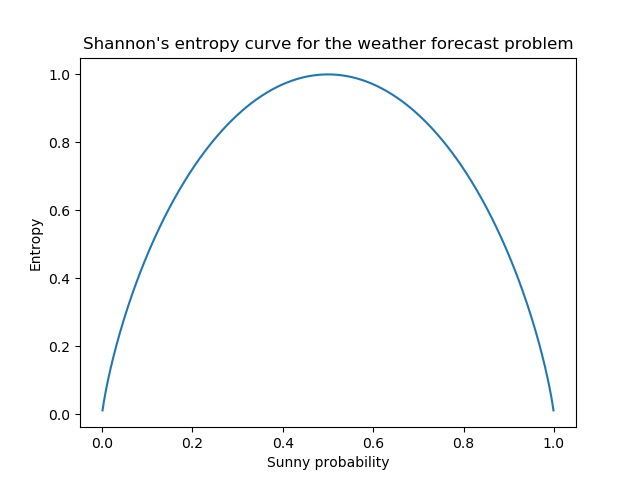
**Misclassification Error**

Shannon Entropy

measures the lack of information contained in a binary variable

the closer is to 0 or 1, the smaller the entropy is and the less random the data is

entropy is largest at , where using classification tree would be no better than randomly guessing



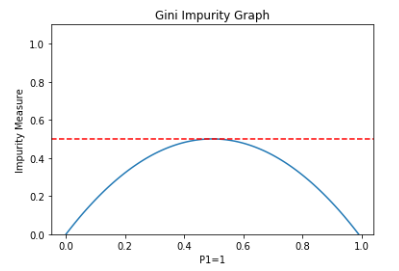
Gini’s Impurity Index

probability of an incorrect classification of a new instance

the closer is to 0 or 1, the purer the data is and the lower the likelihood of misclassification

maximum impurity of 0.5 at , where using classification tree would be no better than randomly guessing

Gini’s impurity index is used for growing a tree and misclassification error is used for pruning a tree



**Advantages and Disadvantages of Trees**

trees are easy to explain than linear regression

decision trees closely mirror human decision making than regression analysis

trees can be displayed graphically and are easily interpreted by non-experts

trees can easily handle qualitative predictors without the need to create dummy variables

trees don’t always make precise predictions, so need to aggregate many decision trees using methods like bagging and random forests

**Ensemble Methods**

bagging and random forests aim to reduce the complexity of models that overfit the training data

bagging and random forests improve prediction at the cost of interpretability

boosting tries to increase the complexity of models that suffer from high bias and underfit the training data

Bagging

generate many subsets of the data by taking bootstrap samples of the data

for each subset, generate a tree without pruning

average the predictions from all the trees to get final prediction

Random Forest

generate many subsets of the data by taking bootstrap samples of the data

for each subset, randomly select a set of covariates to build the tree

cannot make a generalization for all the trees

Boosting

build one tree then train the next tree based on the of

grow a family of trees sequentially