Breiman 2001

Random Forest Algorithm

Repeat k times:

- Draw a bootstrap sample from the dataset
- Train a decision tree

  Until the tree is maximum size

    Choose next leaf node

    Select *m* attributes at random from the *p* available

    Pick the best attribute/split as usual
- Measure out-of-bag error
  - Evaluate against the samples that were not selected in the bootstrap
  - Provides measures of strength (inverse error rate), correlation between trees (which increases the forest error rate), and variable importance

Make a prediction by majority vote among the *k* trees

Breiman 2001

Random Forests: Variable Importance

- Key Idea: If you scramble the values of a variable and the accuracy of your tree doesn't change much, then the variable isn't very important
- Measure the error increase
- Random Forests are more difficult to interpret than single trees; understanding variable importance helps
  - Ex: Medical applications can't typically rely on black box solutions

Gini Coefficient

- Entropy captured an intuition for "impurity"
  - We want to choose attributes that split records into pure classes
- The gini coefficient measures inequality

$$\mathrm{Gini}(T) = 1 - \sum_{i=1}^{n} p_i^2$$

Random Forests on Big Data

- Easy to parallelize
  - Trees are built independently
- Handles “small *n* big *p*” problems naturally
  - A subset of attributes are selected by importance

Summary: Decision Trees and Forests

- Representation
  - Decision Trees
  - Sets of decision trees with majority vote
- Evaluation
  - Accuracy
  - Random forests: out-of-bag error
- Optimization
  - Information Gain or Gini Index to measure impurity and select best attributes