Random Forests

CS534 - Machine Learning

Yubin Park, PhD

"If you can't beat 'em, join 'em."

To fight with the variance (or randomness),

we will adopt randomness to the limit.

Basic Decision Tree

- Start with a dataset: $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_n, y_n)\}$
- Grow a Decision Tree:
    1. Iterate over all possible splittig pairs: splitting variable and value
    2. Select the best splitting pair
    3. Split the data into two partitions based on the selected splitting pair
    4. Repeat the process till any stopping criterion is met

Bagged Trees

- Start with a dataset: $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_n, y_n)\}$
- **Boostrap datasets:** $\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_B$
- Grow a Decision Tree **for each bootstrapped dataset:**
    1. Iterate over all possible splittig pairs: splitting variable and value
    2. Select the best splitting pair
    3. Split the data into two partitions based on the selected splitting pair
    4. Repeat the process till any stopping criterion is met
- **Combine the trained decision trees**

Leo Breiman. Bagging Predictors, Machine Learning (1996)

Random Subspace

- Start with a dataset: $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_n, y_n)\}$
- Grow **multiple** Decision Tree using **the same training data**
  1. **Bootstrap features at each node**
  2. Iterate over all possible splittig pairs **within the bootstrapped features**
  3. Select the best splitting pair
  4. Split the data into two partitions based on the selected splitting pair
  5. Repeat the process till any stopping criterion is met
- **Combine the trained decision trees**

Tin Kam Ho. The Random Subspace Method for Constructing Decision Forests, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (1998)

Random Forests

- Start with a dataset: $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_n, y_n)\}$
- **Boostrap datasets:** $\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_B$
- Grow a Decision Tree **for each bootstrapped dataset:**
  1. **Bootstrap features at each node**
  2. Iterate over all possible splittig pairs **within the bootstrapped features**
  3. Select the best splitting pair
  4. Split the data into two partitions based on the selected splitting pair
  5. Repeat the process till any stopping criterion is met
- **Combine the trained decision trees**

[Leo Breiman. Random Forests, Machine Learning (2001)]

Extra-Trees

- Start with a dataset: $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_n, y_n)\}$
- Grow **multiple** Decision Tree using **the same training data**
    1. **Bootstrap features at each node**
    2. **Draw random split values for the bootstraped features**
    3. Iterate over all **the candidate splittig pairs**
    4. Select the best splitting pair
    5. Split the data into two partitions based on the selected splitting pair
    6. Repeat the process till any stopping criterion is met
- **Combine the trained decision trees**

Pierre Geurts, Damien Ernst, Louis Wehenkel. Extremely randomized trees, Machine Learning (2006)

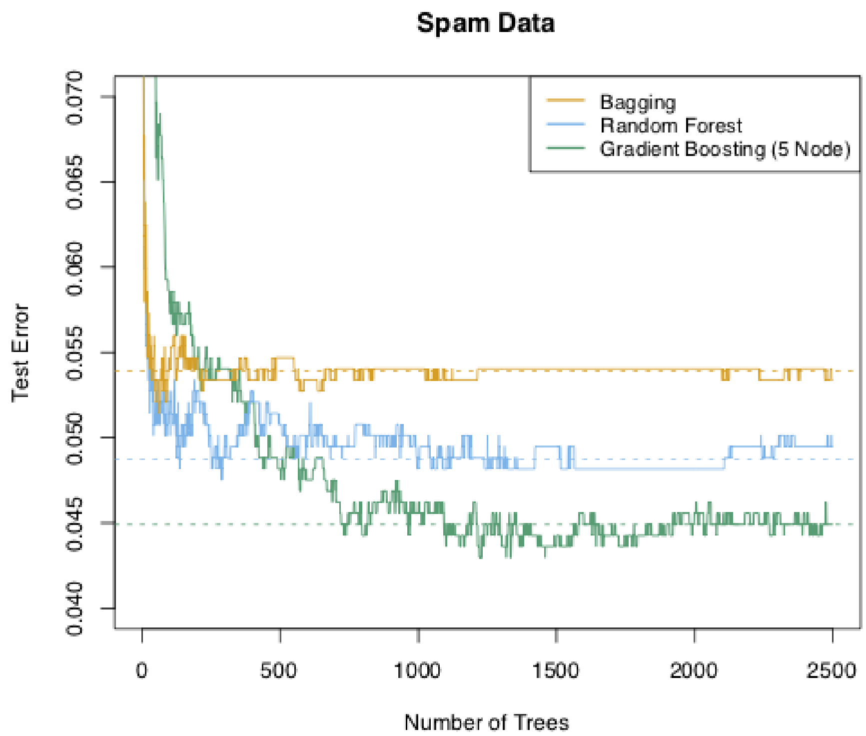

FIGURE 15.1. *Bagging, random forest, and gradient boosting, applied to the spam data. For boosting, 5-node trees were used, and the number of trees were chosen by 10-fold cross-validation (2500 trees). Each "step" in the figure corresponds to a change in a single misclassification (in a test set of 1536).*

Chapter 15 of [ESLII]

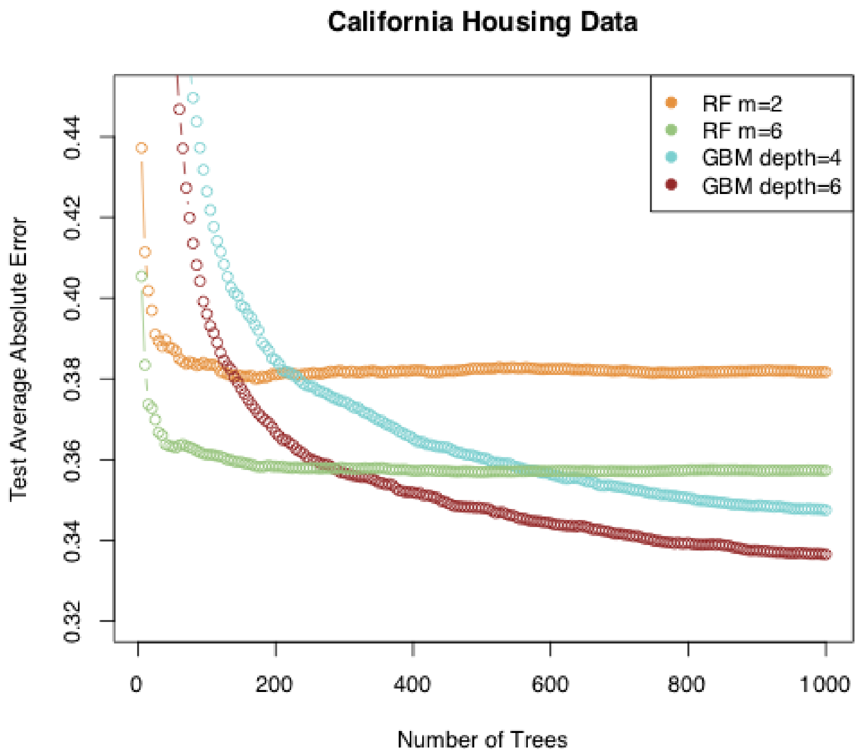

FIGURE 15.3. *Random forests compared to gradient boosting on the California housing data. The curves represent mean absolute error on the test data as a function of the number of trees in the models. Two random forests are shown, with $m = 2$ and $m = 6$. The two gradient boosted models use a shrinkage parameter $\nu = 0.05$ in (10.41), and have interaction depths of 4 and 6. The boosted models outperform random forests.*

Chapter 15 of [ESLII]

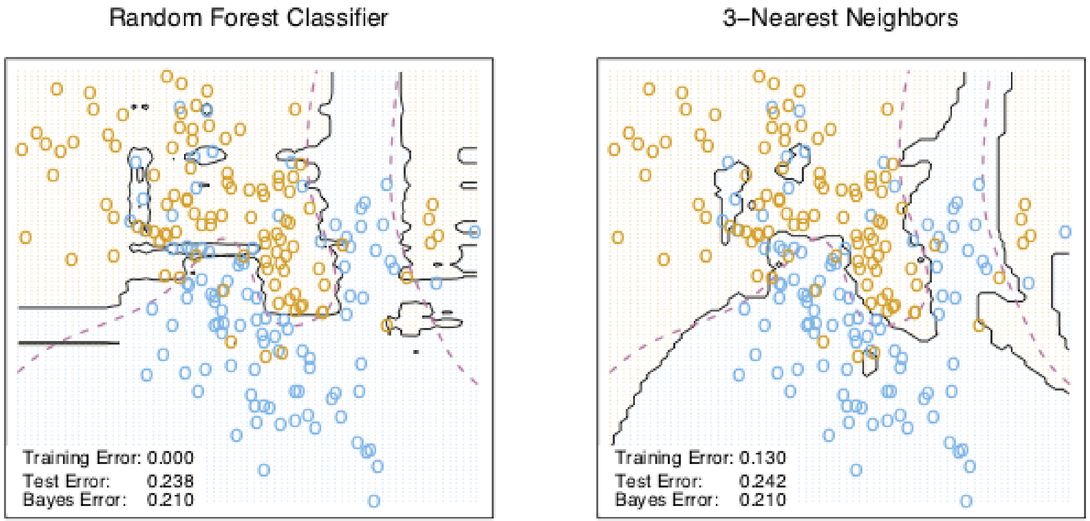

FIGURE 15.11. *Random forests versus 3-NN on the mixture data. The axis-oriented nature of the individual trees in a random forest lead to decision regions with an axis-oriented flavor.*

Chapter 15 of [ESLII]

Questions?