Decision trees and ensemble methods

Lecture 11 of “Mathematics and AI”

Outline

1. Decision trees

   Regression trees, growing and pruning trees, classification trees

2. Ensemble methods

   Bagging, boosting, random forests, BART

Decision trees

Decision trees

- Segmentation of the feature space into segments R_k (typically boxes)
- Prediction y_i is identical for all $X_i \in R_k$
  - Regression: Regression to the mean $\bar{y}_k$ in R_k
  - Classification: Majority vote in R_k

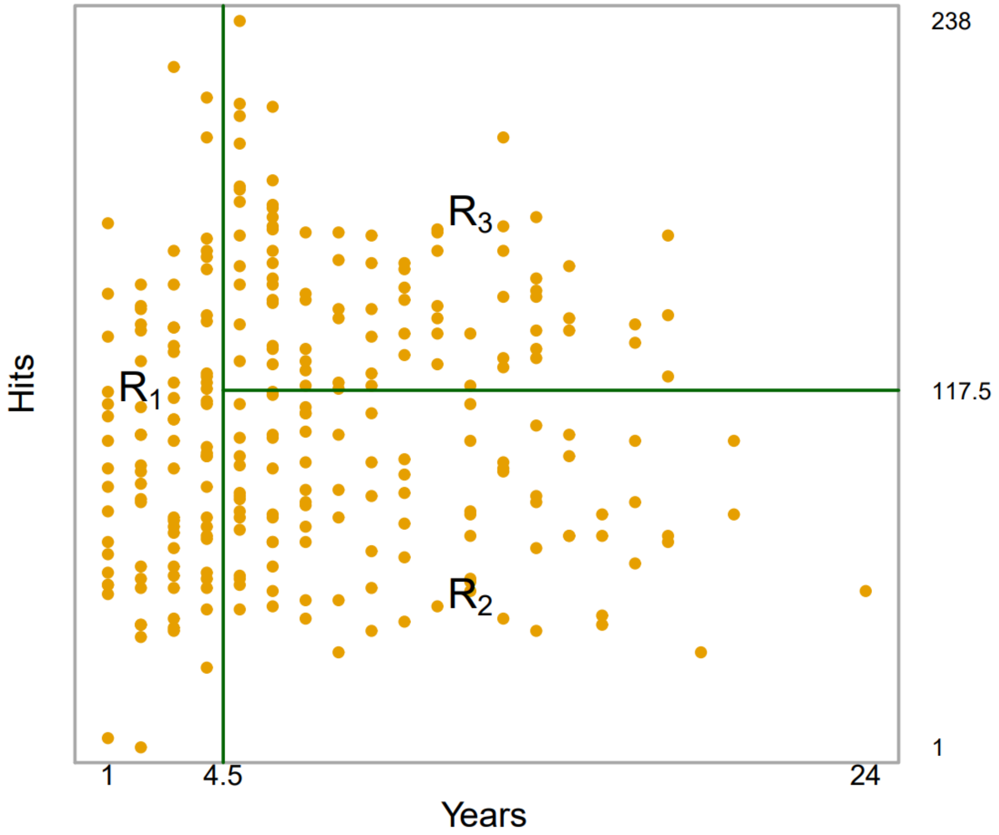
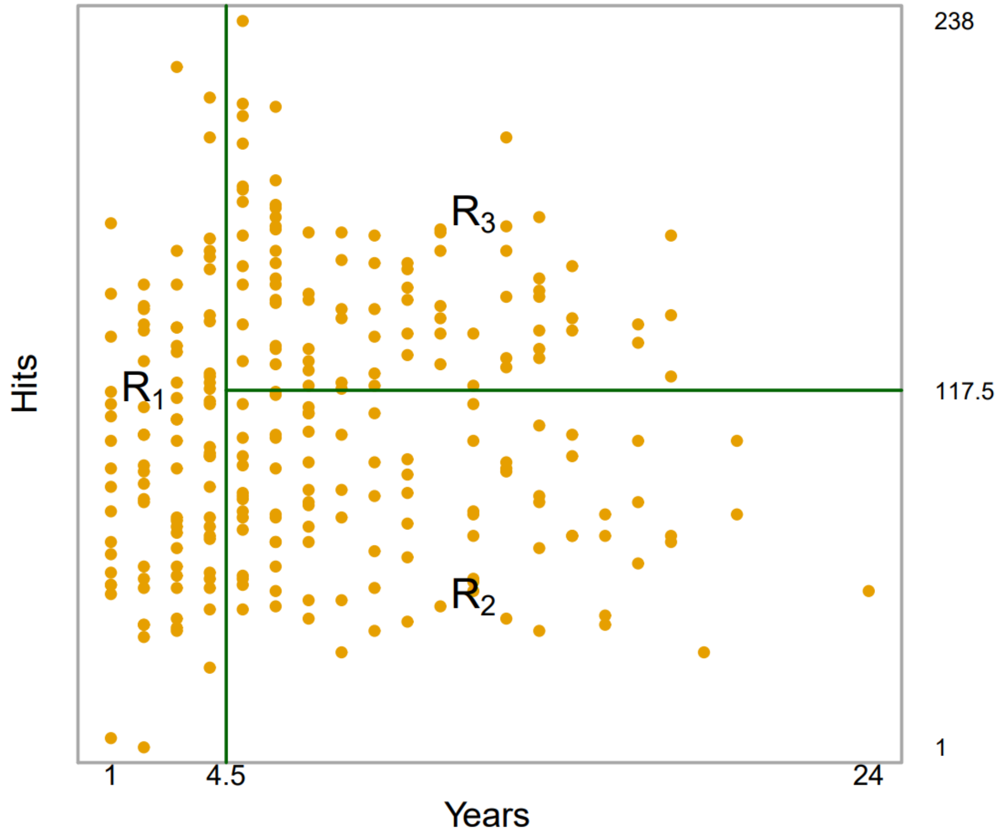


Image source: Introduction to Statistical Learning in Python

Regression trees



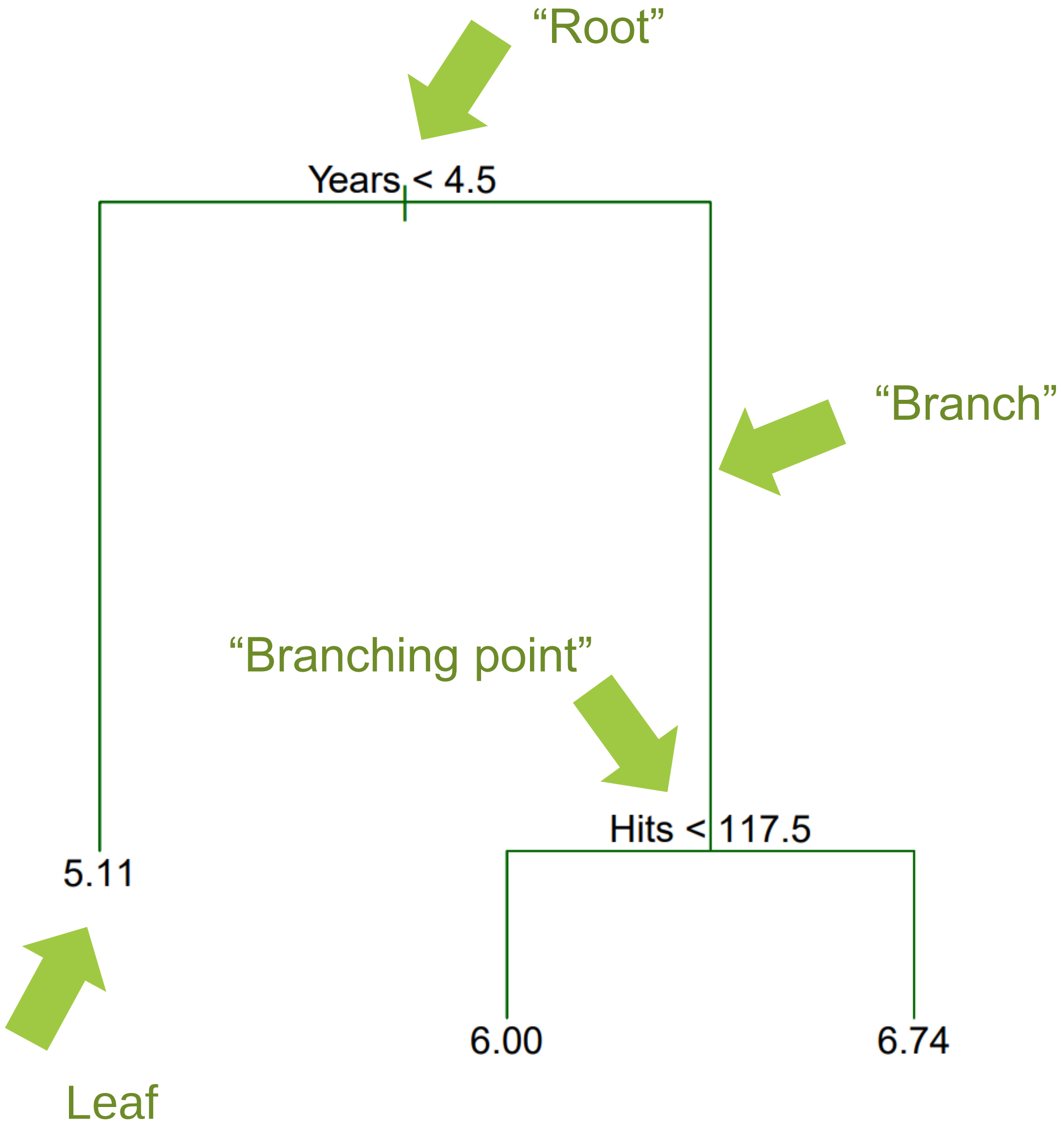


Image source: Introduction to Statistical Learning in Python

Growing a regression tree

- Quality of fit:

$$\mathrm{RSS} = \sum_{j=1}^{J} \sum_{i \in R_j} \left(y_i - \hat{y}_{R_j} \right)^2$$

- Recursive binary splitting:
  - Add one new segment at a time
  - Choose each segment s.t. it minimizes RSS

Image source: Introduction to Statistical Learning in Python

Growing a regression tree

- Quality of fit:

$$\mathrm{RSS} = \sum_{j=1}^{J} \sum_{i \in R_j} \left(y_i - \hat{y}_{R_j} \right)^2$$

- Recursive binary splitting:
    - Add one new segment at a time
    - Choose each segment s.t. it minimizes RSS

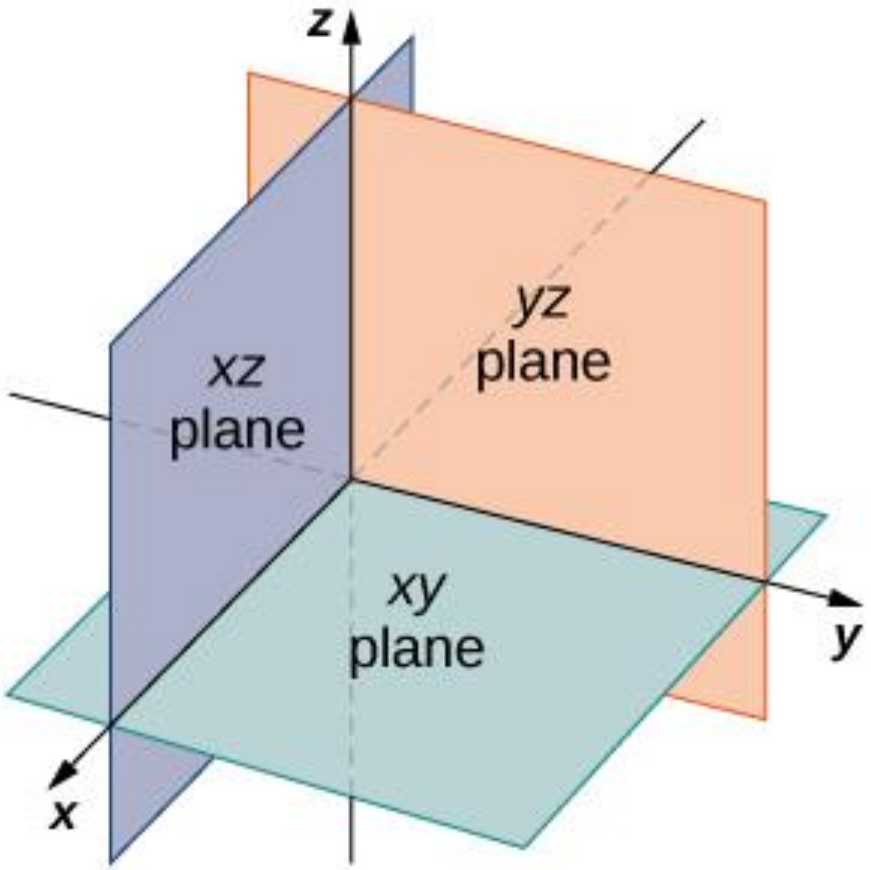
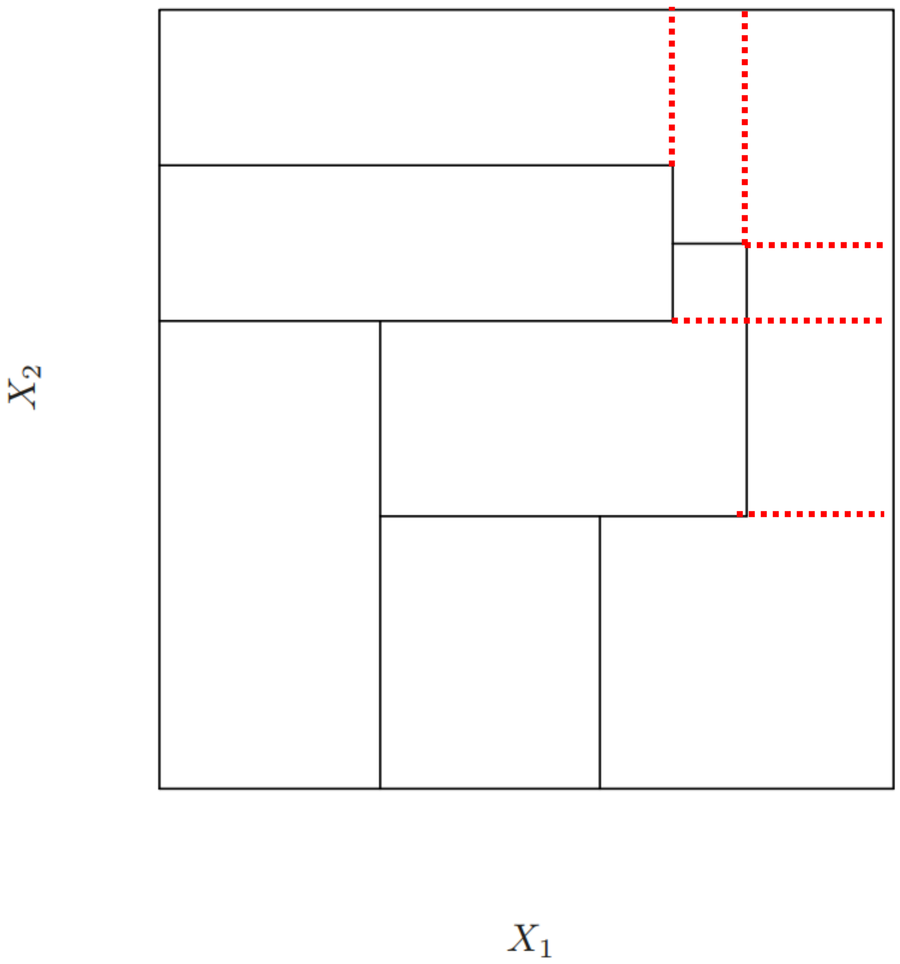


Image source: OpenStax Calculus 3

Possible segmentations



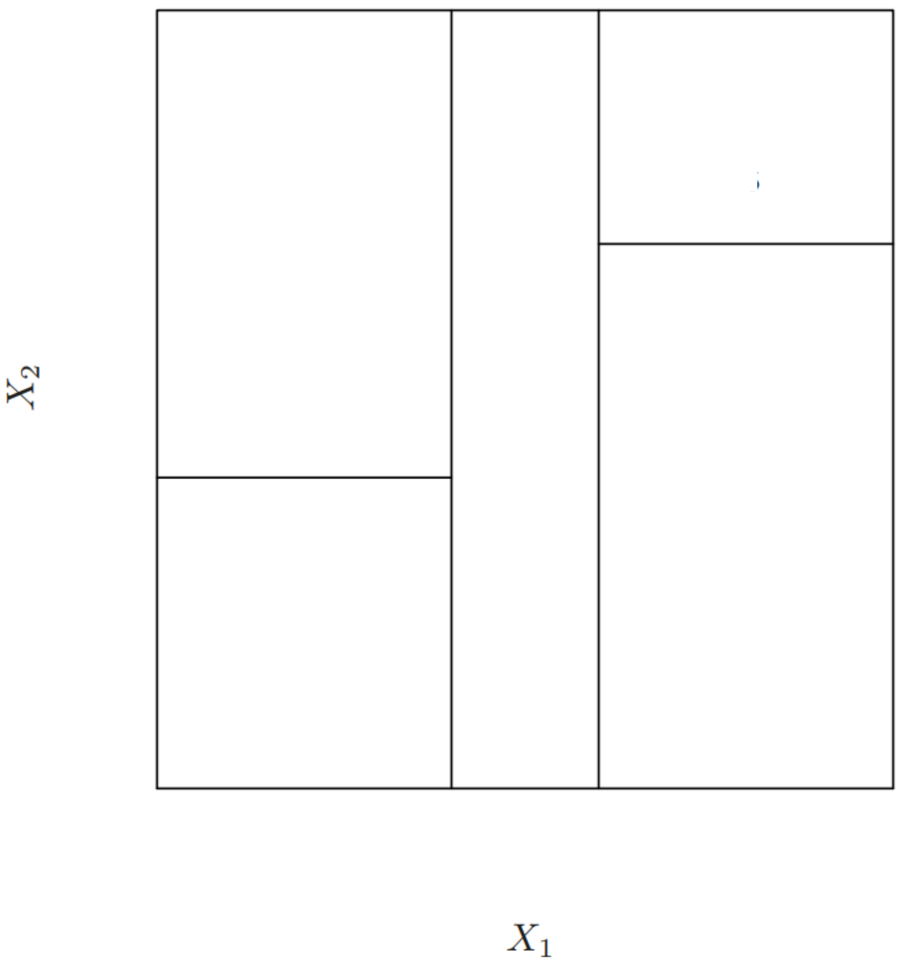


Image source: Introduction to Statistical Learning in Python

Pruning trees

- Minimizing RSS without additional stopping criterion creates a tree with n segments and perfect within-sample performance
- Reduce model complexity and overfitting by removing (i.e. “pruning”) segmentation lines (i.e., “leaves”)

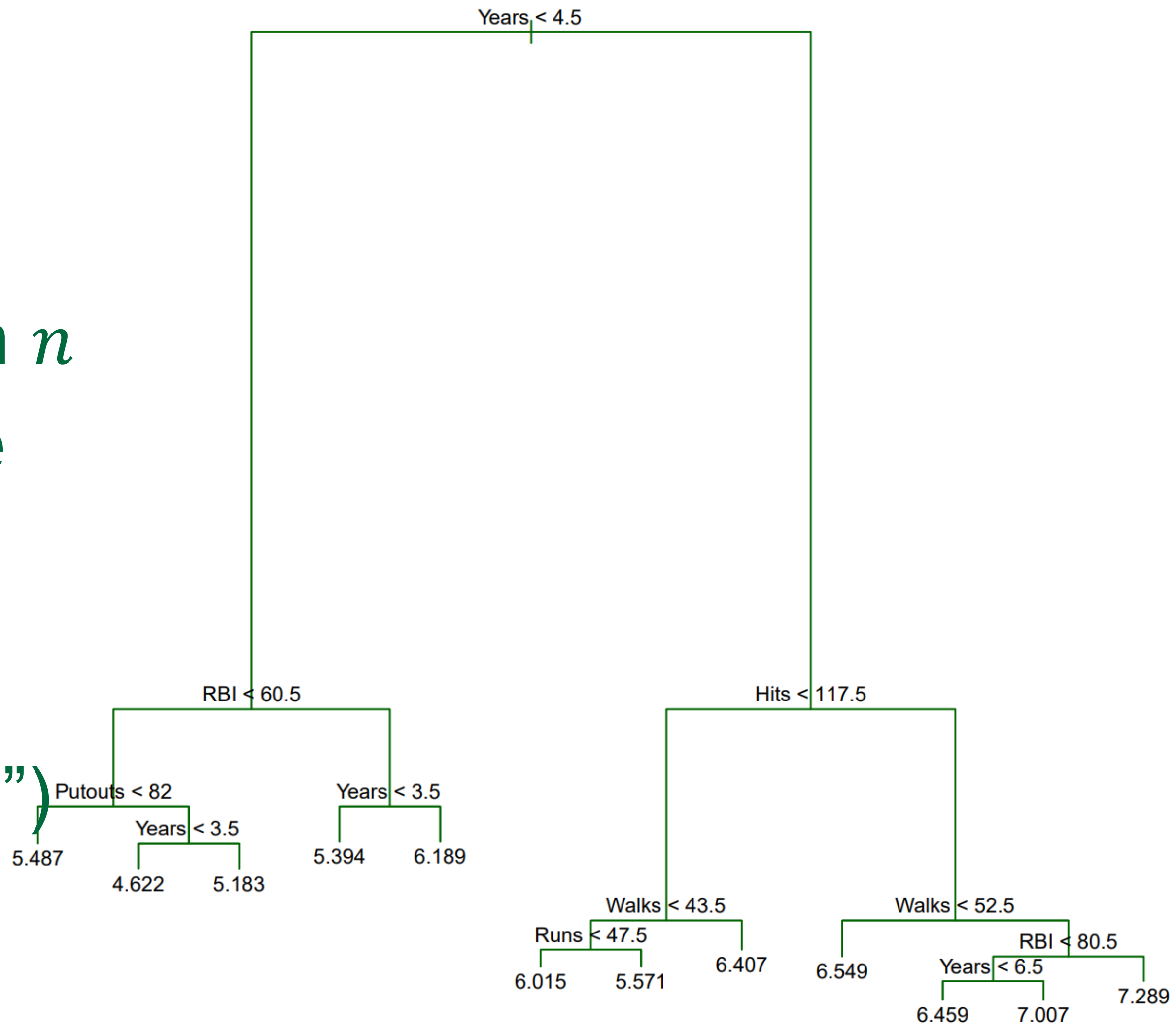


Image source: Introduction to Statistical Learning in Python

Pruning trees

- Adjusted quality of fit:

$$\text{RSS} + \text{L1 penality} = \sum_{j=1}^{|T|} \sum_{i \in R_j} \left(y_i - \hat{y}_{R_j} \right)^2 + \alpha |T|$$

- Tuning α leads to sequence of trees of decreasing complexity

The full pipeline

1. Grow tree using full training set
   - Large complex tree with high variance, low (no) bias
2. Apply L1 penalty, prune leaves successively with increasing α
   - Sequence of “best trees” with descending tree size / model complexity
3. Grow trees using k-fold crossvalidation for various values of α
   - Best value for α
4. Retrieve tree with corresponding α value from sequence of best trees
   - Tree with best variance-bias tradeoff

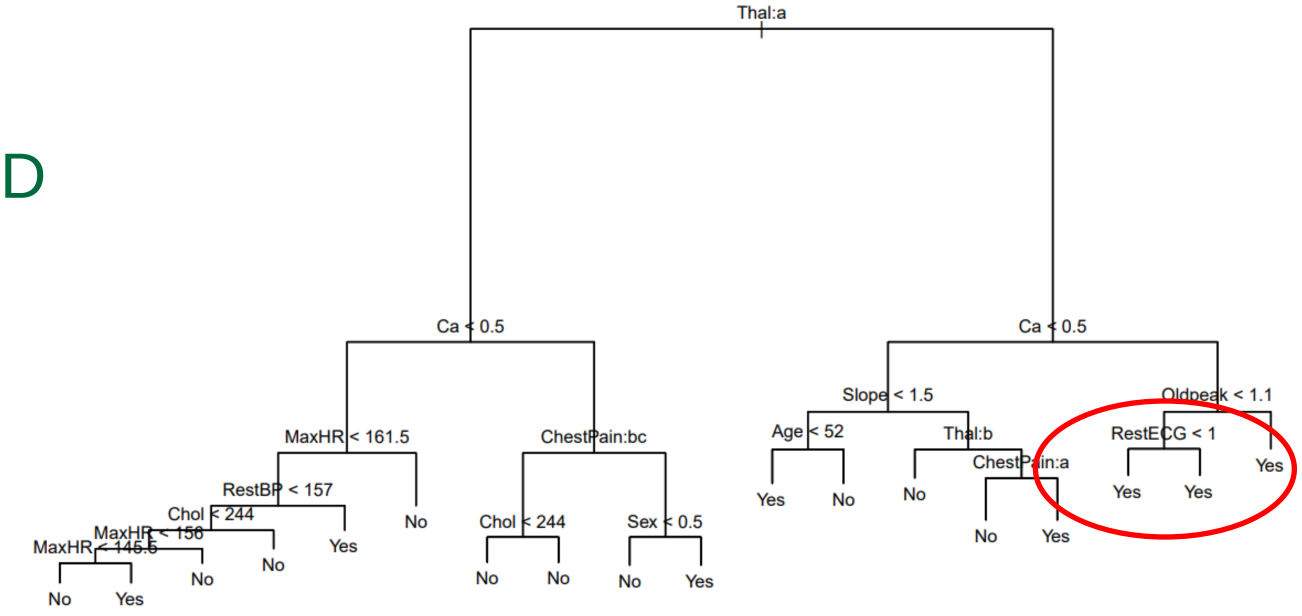
A SISO* experiment

- $f(x) = x$
- $f(x) = x^2$
- $f(x) = \operatorname{sign}(x)$
- $f(x) = \cos(x)$

*SISO stands for “Single input single output”

Classification trees

- Quality of fit
  - ➢ Error rate is hard to optimize via GD
  - ➢ Gini index: $G = \sum_{j=1}^{J} \hat{p}_{m_j} \left(1 - \hat{p}_{m_j}\right)$
  - ➢ Entropy $H = -\sum_{j=1}^{J} \hat{p}_{m_j} \log \hat{p}_{m_j}$
- G , H optimize “node purity”



Ensemble methods

“The crowd's wisdom often surpasses that of even the most knowledgeable expert.”

James Surowiecki (Author of “The Wisdom of Crowds”)

Ensemble methods

- Idea: Combine many “weak learners” to create a “strong learner”
- Approaches:
  - Bagging,
  - random forests,
  - boosting,
  - Bayesian additive regression trees (BART)

Bagging

1. Bootstrap the training set B times
   - B (somewhat independent) training sets
2. Train a model on each training set
   - B models $\hat{f}^{(b)}$, $b \in \{1, \ldots, B\}$ that yield (possibly different) predictions
3. For each query X, the ensemble prediction is mean (or majority vote) among B predictions $\hat{f}^{(b)}(X)$

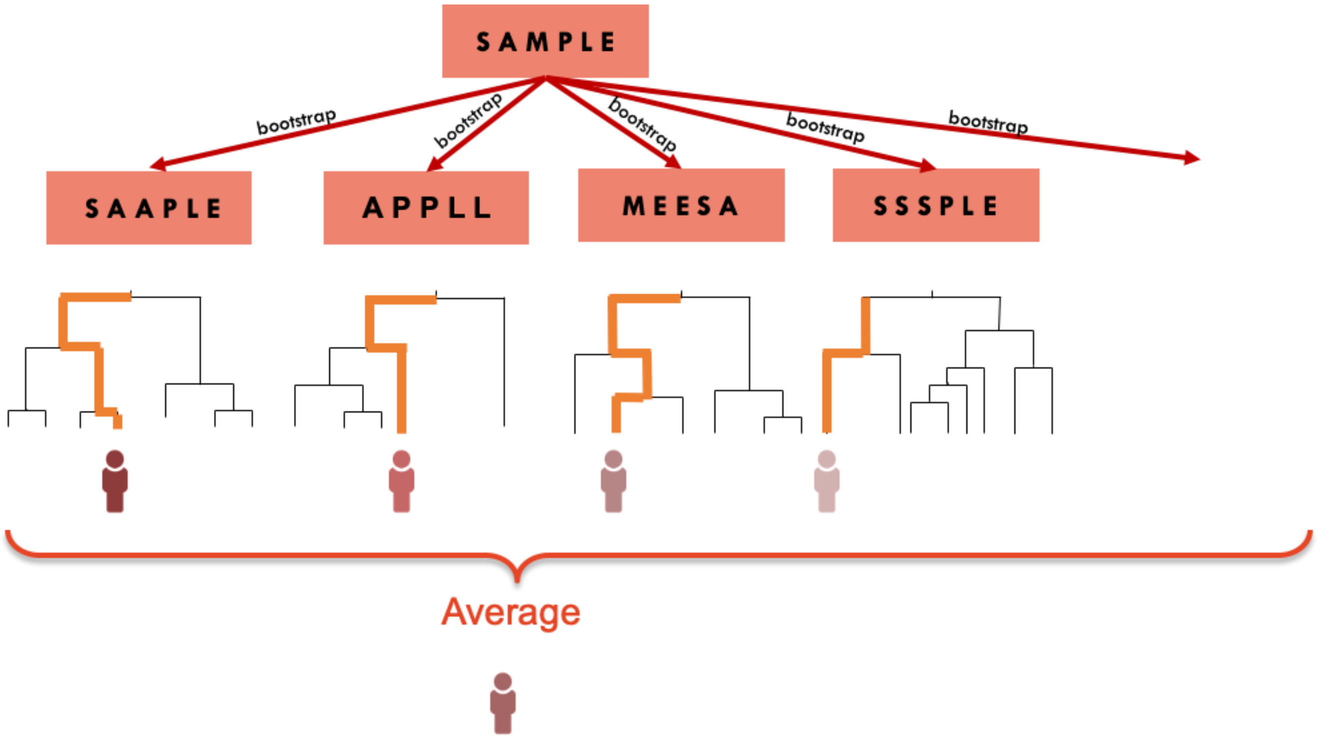


Image source: https://bookdown.org/tpinto_home/Beyond-Additivity/bagging.html

Random forests

1. Select B random subset of variables
   - B ("very" independent) training sets
2. Train a model on each training set
   - B models $\hat{f}^{(b)}$, $b \in \{1, \dots, B\}$ that yield (possibly different) predictions
3. For each query X, the ensemble prediction is mean (or majority vote) among B predictions $\hat{f}^{(b)}(X)$

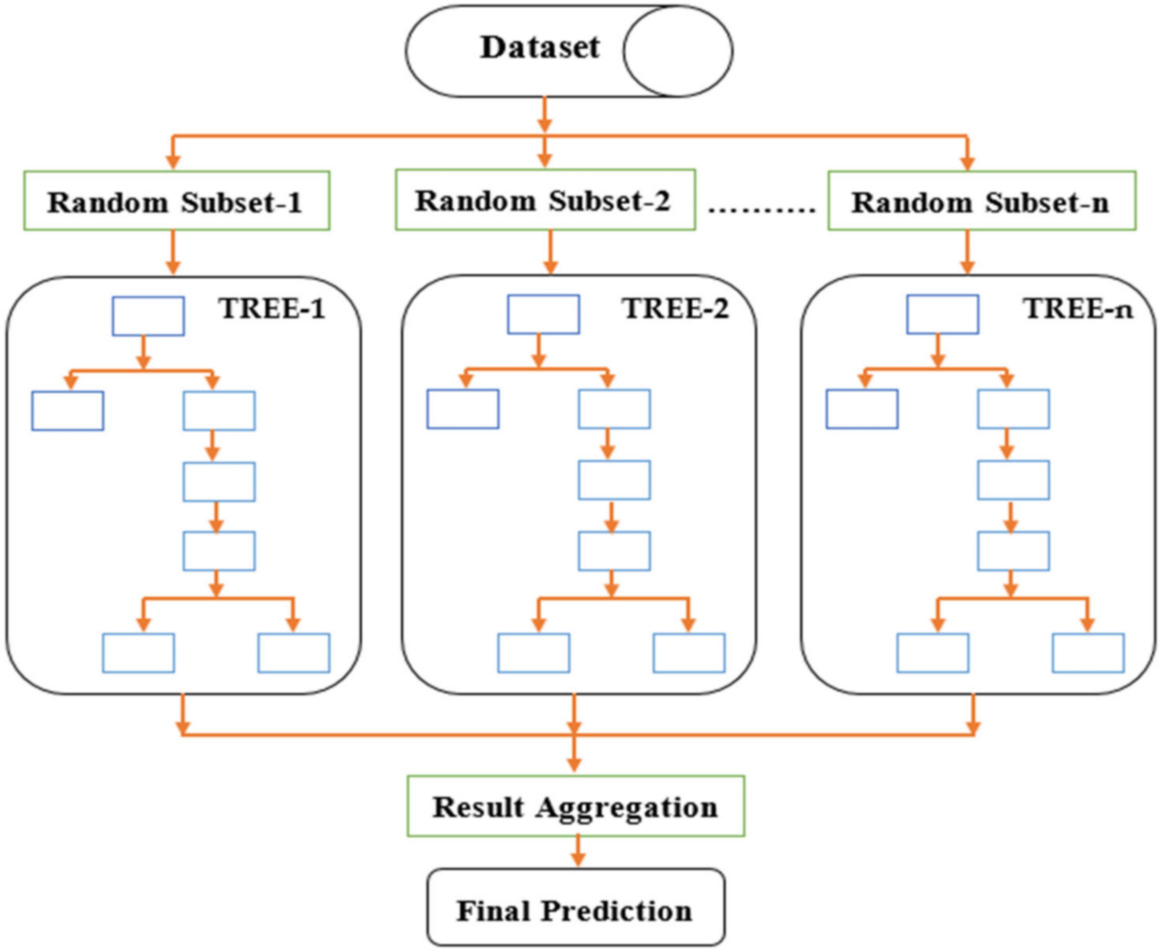
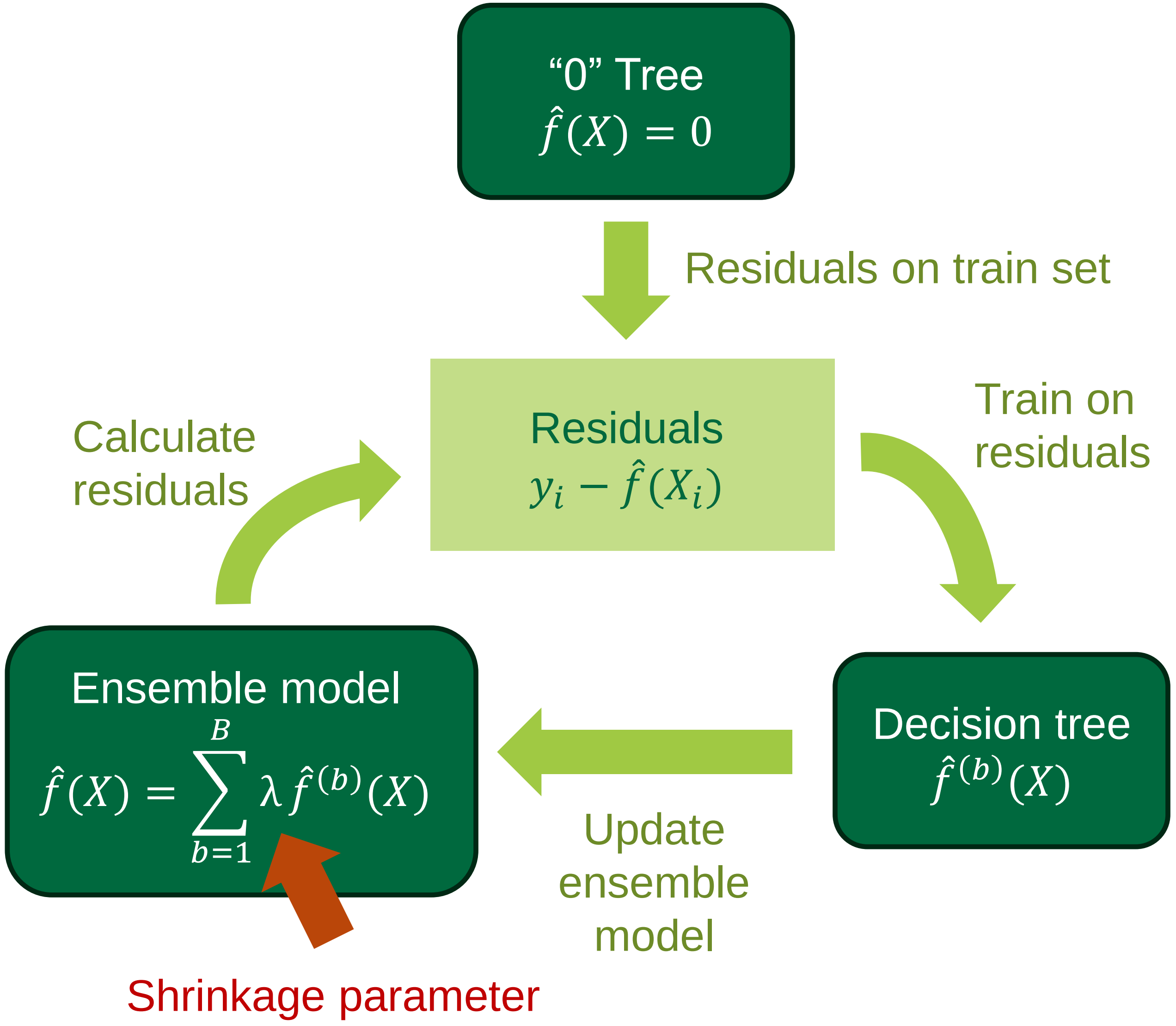


Image source: Saeed MS, Mustafa MW, Sheikh UU, Jumani TA, Mirjat NH. Ensemble Bagged Tree Based Classification for Reducing Non-Technical Losses in Multan Electric Power Company of Pakistan. *Electronics*. 2019; 8(8):860. https://doi.org/10.3390/electronics8080860

Boosting

Idea:

- Build a sequence of trees of desired complexity.
- Each tree addresses the shortcomings of the previous tree.



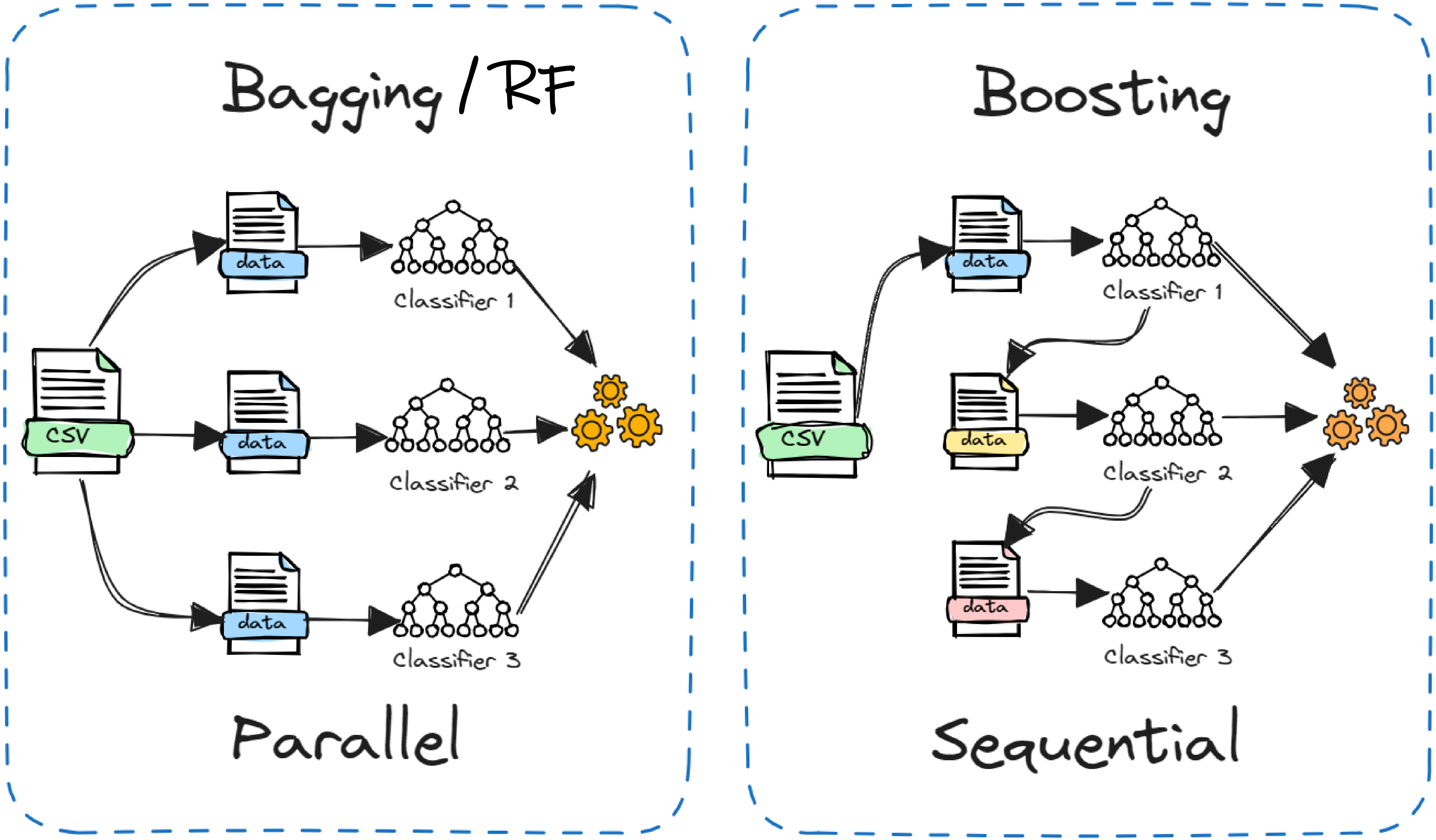
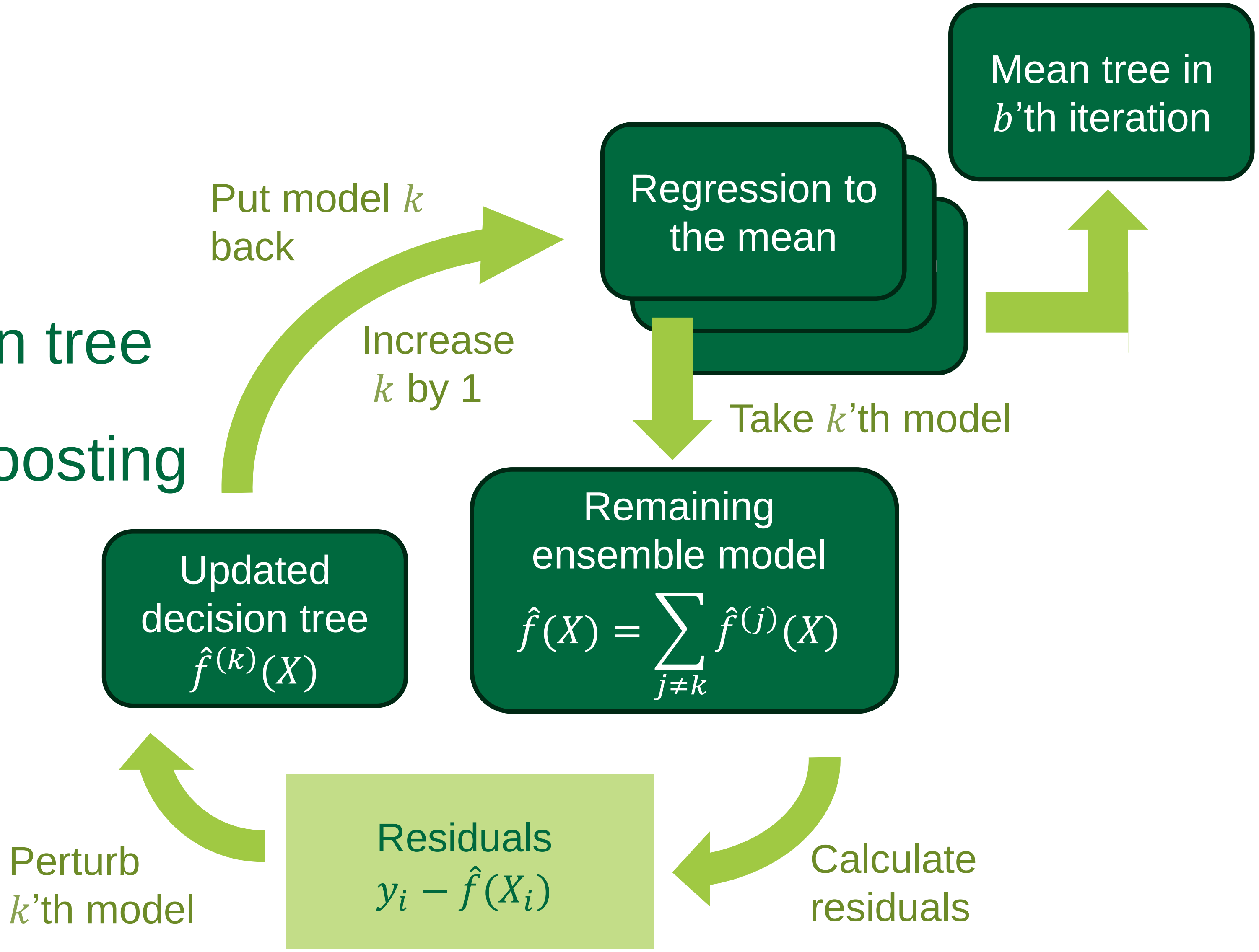


Image source: https://www.datacamp.com/tutorial/what-bagging-in-machine-learning-a-guide-with-examples

BART

- Bayesian additive regression tree
- Idea: Combine bagging & boosting



- Cycle through stack B times
- Ensemble is mean of $B - L$ mean trees with "burn in" rounds L