An Introduction to Random Forests

November 28, 2024

Figure: A forest... but not a random one.

🌲 Refresher: Decision Trees

Example: Howell data (783 observations).

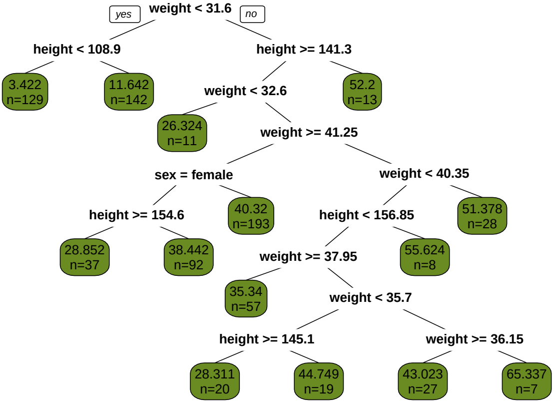

Figure: Regression tree on Howell data. Age is predicted based on sex, height (cm) & weight (cm)

🌳 Random Forests: Motivation

CARTs (Classification And Regression Trees) are very powerful but:

- They are prone to **overfitting**.
- They are **unstable**.
- They are **noisy**.
- They may struggle to detect **complex** and often **non-linear** patterns.

$\rightarrow$ **Random Forests** address the above shortcomings!

Random Forests: Definition

Definition

A Random Forest (RF) is an **ensemble learning** algorithm for regression and classification tasks. It is based on the concept of **bootstrap aggregation (bagging)** applied on decision trees.

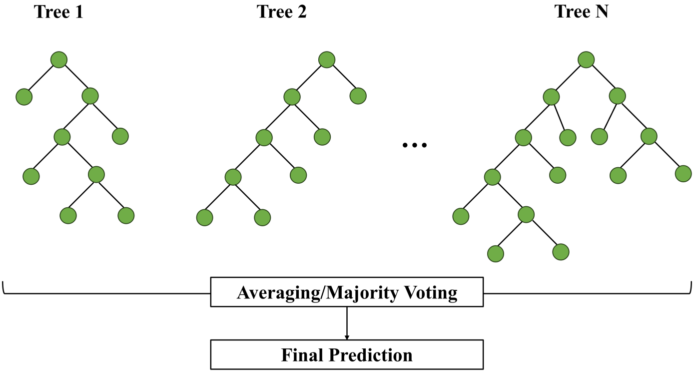

Figure: Illustration of a random forest as a collection of decision trees.

🌲 Bootstrap Aggregation (Bagging)

Bagging consists of 2 main steps:

1. **Bootstrapping:** For given training data $\mathbf{Z} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$, draw B samples $\mathbf{Z}_1, \ldots, \mathbf{Z}_B$ with replacement and obtain predictions $\hat{f}_1(\boldsymbol{x}), \ldots, \hat{f}_B(\boldsymbol{x})$ (regression) or $\hat{C}_1(\boldsymbol{x}), \ldots, \hat{C}_B(\boldsymbol{x})$ (classification).
2. **Aggregating:** Aggregate bootstrap predictions to obtain the bagging estimate.
   - For regression:

$$\hat{f}_{\text{bag}}(\boldsymbol{x}) = \frac{1}{B} \sum_{b=1}^{B} \hat{f}_b(\boldsymbol{x}).$$

   - For classification:

$$\hat{C}_{\text{bag}}(\boldsymbol{x}) = \text{majority vote}\{\hat{C}_b(\boldsymbol{x})\}_{b=1}^{B}.$$

Why does bagging work?

Bagging aims to **reduce the variance** of the predictions.

- For B identically distributed (**not** independent) random variables with positive pairwise correlation ρ and common variance σ^2, the variance of their sample average is given by $\rho\sigma^2 + \sigma^2(1-\rho)/B$. (Exercise)
- When $B \to \infty$, the above Expression depends only on the pairwise correlation and the variance.
- Random forests make use of this result by reducing ρ, without increasing σ^2 too much.

Decorrelating trees

- Decision trees are known to be highly correlated.
- Dominant variables for the regression/classification problem are a main source of this issue.
- Growing decorrelated trees involves an extra step of random variable selection for splitting.

🌲 Training a Random Forest: Out-Of-Bag Samples

- Hyperparameter tuning and training a RF is done using **Out-Of-Bag (OOB)** samples:
  - For each bootstrap sample $\mathbf{Z}_b$, compute the error of the predictions using data not in $\mathbf{Z}_b$ and then average over all B samples.
  - This is very similar to Cross Validation!

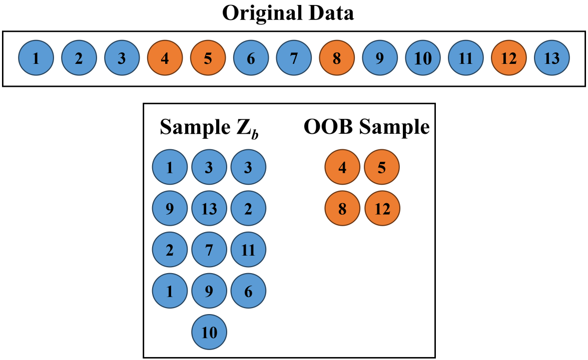

Figure: Illustration of OOB sampling.

🌳 Example: Howell data revisited

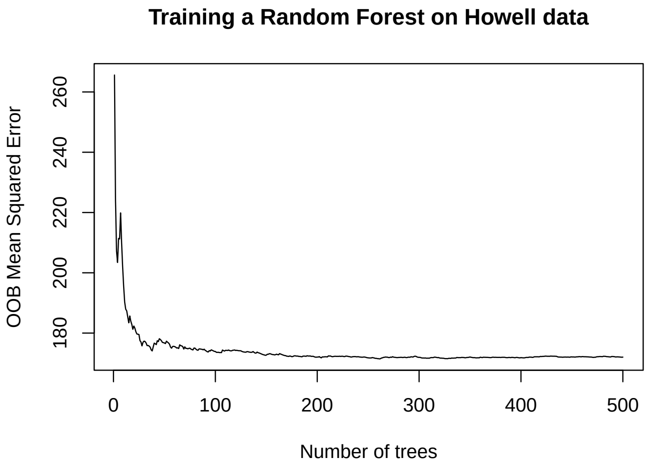

Figure: Training a Random Forest on the Howell data

Summary

Random Forest Algorithm:

1. For $b = 1, \ldots, B$:
   1. Draw a bootstrap sample $\mathbf{Z}_b$ of size n from the training data.
   2. Grow a tree T_b using $\mathbf{Z}_b$ by repeating the following three steps until minimum node size of maximum tree depth is reached:
      1. Select m variables randomly from the p predictors.
      2. Pick the best variable & splitting point among the m.
      3. Split the node into two daughter nodes.
2. Output the tree ensemble $\{T_b\}_{b=1}^{B}$.

Given a new vector of predictors $\boldsymbol{x}^*$, make a prediction by averaging (regression) or taking the majority vote (classification) of tree predictions.

🌲 Useful Resources

- R package randomForest provides a good implementation of Random Forests.
- Code to reproduce plots in earlier slides can be found in: https://github.com/EfthymiosCosta/Example-Lecture-RandomForests.