

Random Forest and Generalization

A Probabilistic Analysis of Decision Trees and Bootstrap Methods

Güneş Akbaş

Department of Mathematics
Bilkent University

April 9, 2025

1. What is Supervised Learning?
2. Decision Trees
3. Bootstrap Method

Supervised Learning Overview

Training Data:

We assume a training set

$$\mathcal{D} = \{(x_i, y_i) \mid i = 1, 2, \dots, n\},$$

where:

- $x_i \in \mathcal{X}$ is the feature vector,
- $y_i \in \mathcal{Y}$ is the target value.

Target Function and Noise:

Supervised learning posits an unknown target function

$$f : \mathcal{X} \rightarrow \mathcal{Y},$$

and we assume the observed target values are generated by

$$y_i = f(x_i) + \varepsilon_i,$$

where ε_i models noise (e.g., measurement error or inherent randomness).

Supervised Learning Overview

Learning Objective:

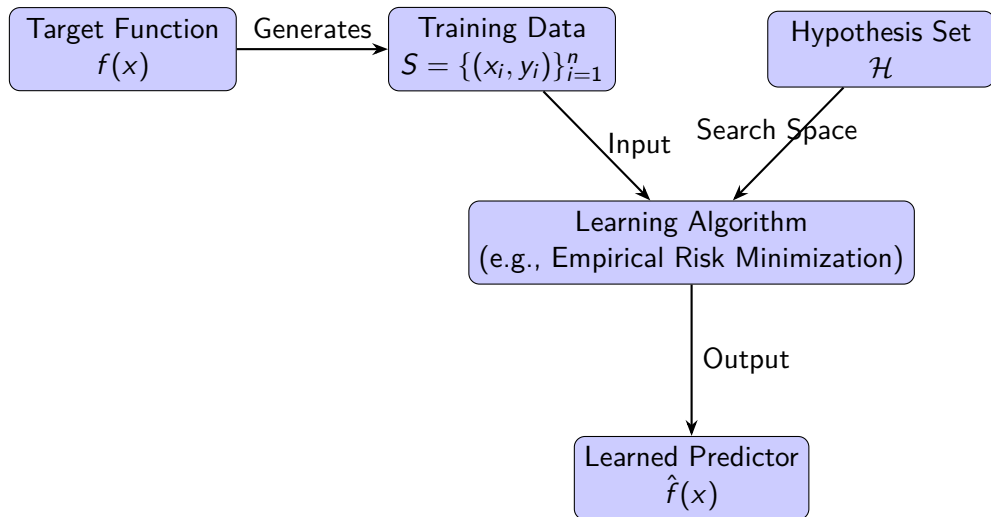
Our goal is to select a hypothesis f from a family \mathcal{H} (e.g., linear functions, trees, etc.) that approximates f^* well. This is often accomplished by minimizing a loss function. For example, in regression with squared error loss:

$$\min_{\hat{f} \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^n (y_i - \hat{f}(x_i))^2.$$

Interpretation:

ε_i represents the *noise* in our target samples, and f is the ideal target function that we wish to approximate through learning.

Schematic Overview: Learning Process



Decision Trees

Definition of a Decision Tree:

Decision Tree

A decision tree is a measurable function

$$T : \mathcal{X} \rightarrow \mathcal{Y},$$

constructed by partitioning \mathcal{X} into a finite collection of disjoint measurable subsets (called *leaves*)

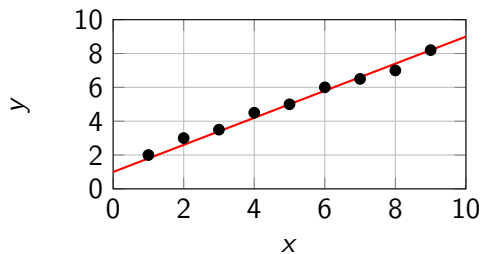
$$\{A_1, A_2, \dots, A_K\},$$

$$\mathcal{X} = \bigcup_{k=1}^K A_k, \quad \text{with } A_i \cap A_j = \emptyset \text{ for } i \neq j.$$

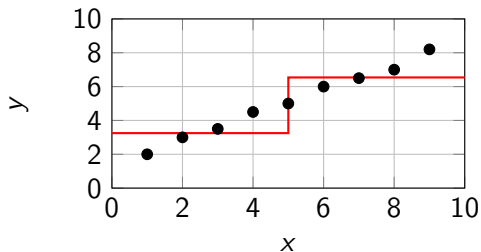
$$T(x) = \sum_{k=1}^K a_k \mathbf{1}_{A_k}(x),$$

Linear Regression vs. Regression Tree (Step Function)

Linear Regression



Regression Tree (Step Function)



Bias–Variance Decomposition

- The mean squared error (MSE) at any x is decomposed as:

$$\mathbb{E}[(\hat{f}(x) - f(x))^2] = \underbrace{(\mathbb{E}[\hat{f}(x)] - f(x))^2}_{\text{Bias}^2} + \underbrace{\text{Var}[\hat{f}(x)]}_{\text{Variance}}.$$

- **Bias:**

- Originates from the hypothesis set/model class.
- A simpler or mis-specified model may not capture the true function $f(x)$, leading to high bias.

- **Variance:**

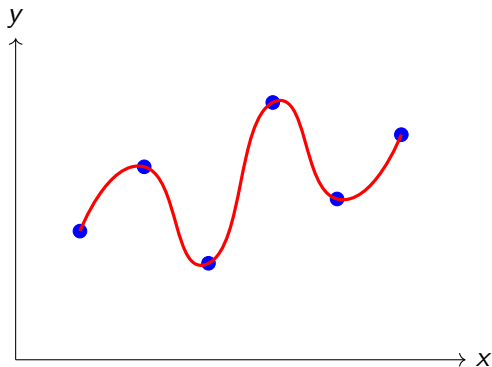
- Arises from the randomness in the training dataset.
- With fewer data points, $\hat{f}(x)$ may vary significantly from sample to sample.

- Together,

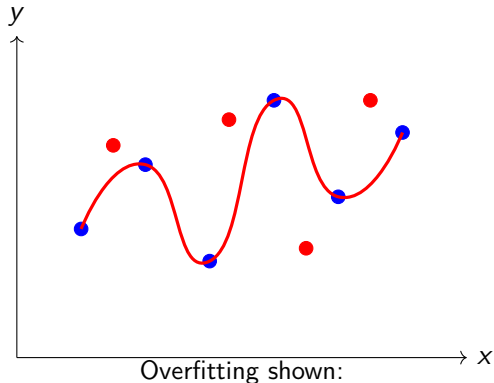
$$\text{MSE} = \text{Bias}^2 + \text{Variance},$$

highlighting a trade-off: richer hypothesis sets (lower bias) can lead to higher variance unless controlled by more data.

Overfitting in Polynomial Regression



Polynomial touches all
blue training data points



Overfitting shown:
New red data points are not touched by the
overfitted polynomial

Bootstrap Aggregation for Regression Trees

- **Observed Data:** We have a dataset

$$S = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}, \quad x_i \in \mathbb{R}^d, y_i \in \mathbb{R}.$$

- **Empirical Distribution:** The empirical distribution of S is

$$\hat{P}(x, y) = \frac{1}{n} \sum_{i=1}^n \delta_{(x_i, y_i)}(x, y),$$

which places mass $1/n$ on each data point.

- **Bootstrap Sampling:** For each bootstrap replicate $b = 1, \dots, B$:
 - Draw a sample $S^{*(b)}$ of size n **with replacement** from S .
 - Note: Some points may be repeated, while about 37% of the original points will be left out (out-of-bag).

Bagging

Training Regression Trees: For each bootstrap sample $S^{*(b)}$, train a regression tree to obtain an estimator

$$\hat{f}^{(b)}(x).$$

Aggregation: The bagged (aggregated) estimator is

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

Benefits:

- **Variance Reduction:** Averaging reduces the variance of individual trees.
- **Out-of-Bag (OOB) Error:** OOB samples provide an internal and nearly unbiased estimate of the generalization error.

Bagging: Theoretical Variance Reduction

- Consider B regression trees, with each tree $\hat{f}_b(x)$ having variance

$$\text{Var}[\hat{f}_b(x)] = v.$$

- Let $\epsilon_b(x) = \hat{f}_b(x) - \mathbb{E}[\hat{f}_b(x)]$ be the error for tree b .
- Assume a constant pairwise correlation ρ between tree errors:

$$\text{Cov}(\epsilon_b(x), \epsilon_{b'}(x)) = \rho v, \quad b \neq b'.$$

- The bagged (aggregated) predictor is

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

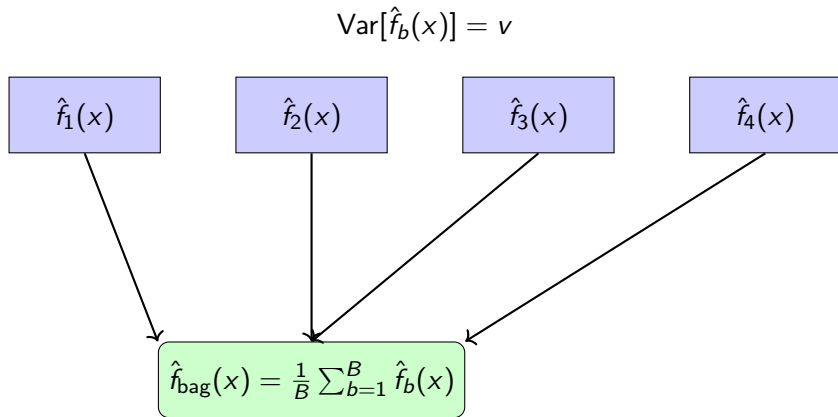
- Its variance is computed as

$$\text{Var}[\hat{f}_{\text{bag}}(x)] = \frac{v}{B} + \rho v \left(1 - \frac{1}{B}\right).$$



- In the limit of large B , this is approximately

$$\text{Var}[\hat{f}_{\text{bag}}(x)] \approx \rho v,$$

Visual Illustration of Bagging Variance Reduction



$$\text{Var}[\hat{f}_{\text{bag}}(x)] \approx \frac{v}{B} + \rho v \left(1 - \frac{1}{B}\right) \approx \rho v;$$

-  Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin, *Learning from Data: A Short Course in Machine Learning*, AMLBook, 2012.
-  T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed., Springer, 2009.

The End