Machine Learning

Dr.Hajialiasgari

Tehran University
Of
Medical Science

February 10, 2025

TEHRAN UNIVERSITY
OF
MEDICAL SCIENCES

1. Introduction
2. Bagging

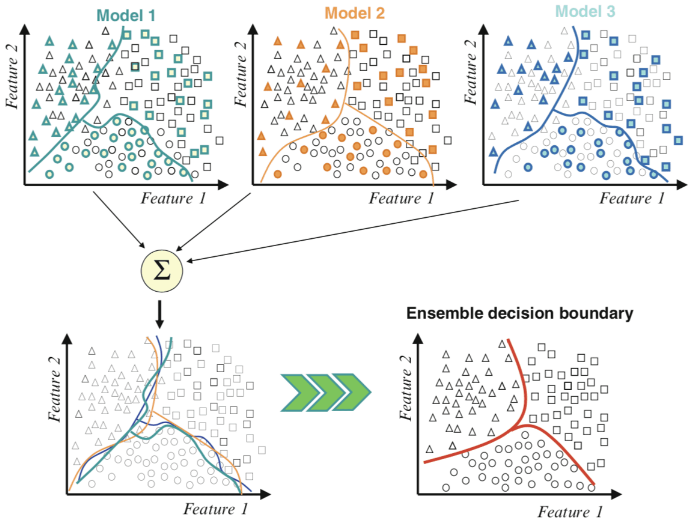
1. Introduction
    - Ensemble Learning
    - Ensemble Methodes
2. Bagging

Ensemble Learning

Ensemble learning is a technique that combines multiple models to improve predictive performance.

- **Improved Accuracy** Combining multiple models results in better approximations.
- **Reduced Overfitting** Averaging multiple models minimizes overfitting.
- **Increased Stability** Reduces sensitivity to small variations in data.

How Ensemble Learning Work



Parallel Ensemble Methods

In parallel ensemble methods, models are trained independently, and their outputs are combined. Examples include:

- Bagging (Bootstrap Aggregating)
- Random Forest

These methods aim to reduce variance.

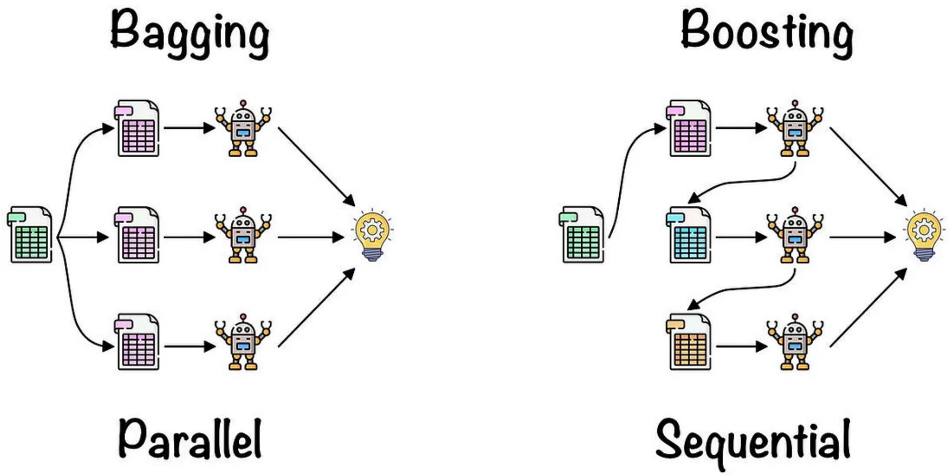
Sequential Ensemble Methods

In sequential ensemble methods, models are trained in sequence, with each model correcting the errors of the previous one. Examples include:

- Boosting (AdaBoost, Gradient Boosting, XGBoost)
- Stacking (Stacked Generalization)

These methods aim to reduce bias.

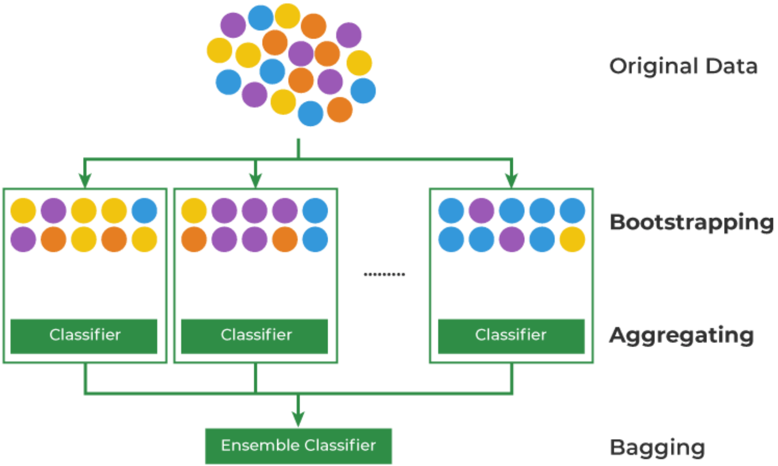
Sequential Vs Parallel



Basic Idea:

- Bagging is an ensemble learning technique designed to reduce variance and improve stability.
- It creates multiple independent models by training them on different subsets of the original dataset.
- The predictions of all models are then combined to form a final output.

How Bagging Work



Bagging Algorithm

Algorithm: Bagging (Bootstrap Aggregation)

Input: Training dataset $\mathscr{D} = \{(x_i, y_i)\}_{i=1}^{N}$, number of base models B

Output: Final prediction $f_{\text{bag}}(x)$

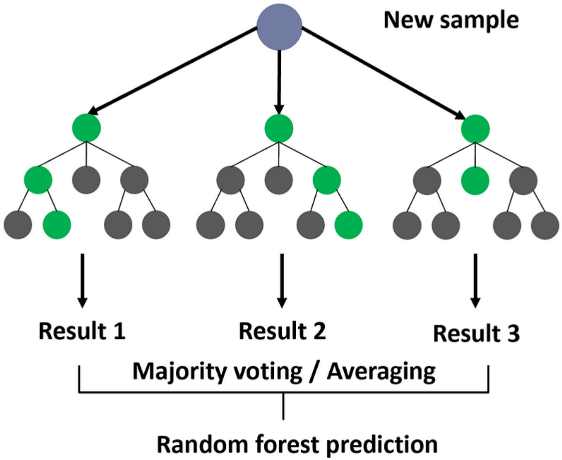
1. For each $b = 1$ to B:
   - Draw a bootstrap sample $\mathscr{D}_b$ of size N from $\mathscr{D}$ (sampling with replacement)
   - Train a base model $f_b(x)$ on $\mathscr{D}_b$
2. Compute the final prediction:
   - **For classification:** Majority vote of $f_b(x)$
   - **For regression:** Average of $f_b(x)$

Decision Trees in Bagging (Random Forest)

Random Forest is an ensemble learning method that builds multiple decision trees using the bagging (Bootstrap Aggregating) technique. Instead of training a single deep decision tree, Random Forest:

- Uses multiple decision trees as base models.
- Trains each tree on a random subset of data (bootstrap sampling).
- Selects a random subset of features at each split.
- Aggregates predictions using majority voting (classification) or averaging (regression).

Random Forest



Random Forest Algorithm

Steps:

1. Select the number of trees B in the forest.
2. For each tree b:
   - Draw a bootstrap sample D_b from dataset D.
   - Select a random subset of m features (usually $m <$ total features).
   - Grow a decision tree using D_b with selected features.
   - Do not prune trees to keep them fully grown.
3. **For prediction:**
   - **Classification:** Use majority voting across all trees.
   - **Regression:** Use the average prediction from all trees.

Advantages of Random Forest

- Reduces overfitting compared to a single decision tree.
- Handles high-dimensional data well.
- Works for both classification and regression tasks.
- Robust to missing data and noise.
- Provides feature importance ranking.

Disadvantages of Random Forest

- Computationally expensive compared to a single decision tree.
- Less interpretable due to multiple trees.
- Requires careful tuning of hyperparameters (e.g., number of trees, number of features per split).

For more information and code check the related notebook

End of Ensemble Learning Part 1