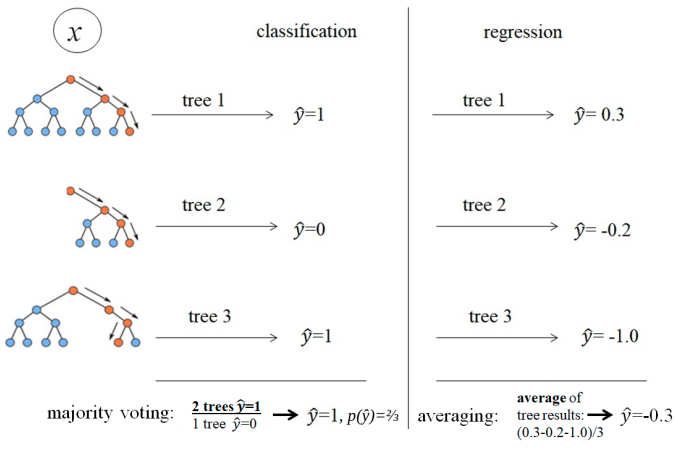
Introduction to Machine Learning

Random Forests: Bagging Ensembles



Learning goals

- Understand the basic idea of bagging
- Be able to explain the connection of bagging and bootstrap
- Understand how a prediction is computed for bagging
- Understand why bagging improves the predictive power

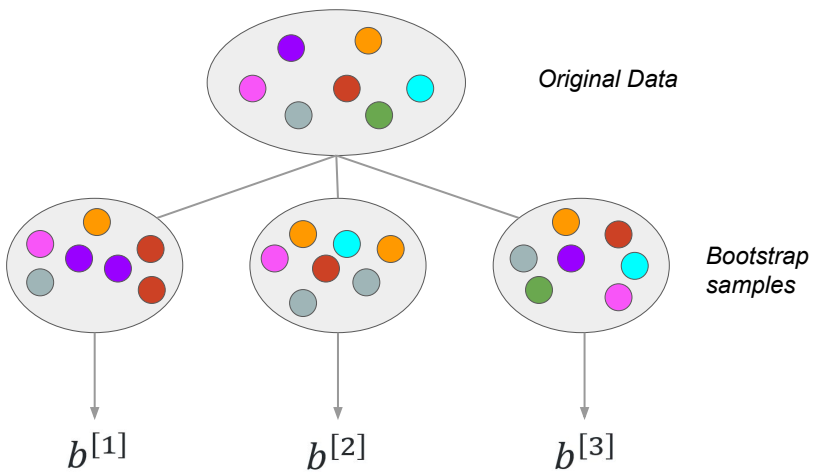
BAGGING

- Bagging is short for **B**ootstrap **Agg**regation.
- It's an **ensemble method**, i.e., it combines many models into one big "meta-model"
- Such model ensembles often work much better than their members alone would.
- The constituent models of an ensemble are called **base learners**

BAGGING

In a **bagging** ensemble, all base learners are of the same type. The only difference between the models is the data they are trained on. Specifically, we train base learners $b^{[m]}(\mathbf{x}), m = 1, \ldots, M$ on M **bootstrap** samples of training data $\mathcal{D}$:

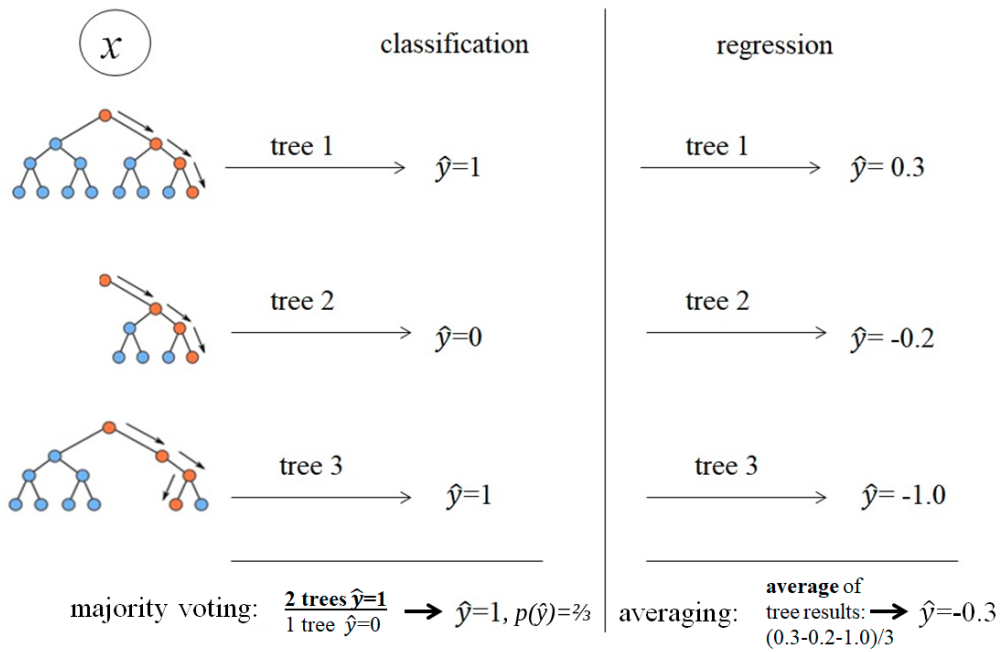
- Draw n observations from $\mathcal{D}$ with replacement
- Fit the base learner on each of the M bootstrap samples to get models $\hat{f}(x) = \hat{b}^{[m]}(\mathbf{x}), m = 1, \ldots, M$



BAGGING

Aggregate the predictions of the M fitted base learners to get the **ensemble model** $\hat{f}^{[M]}(\mathbf{x})$:

- Aggregate via averaging (regression) or majority voting (classification)
- Posterior class probabilities $\hat{\pi}_k(\mathbf{x})$ can be estimated by calculating predicted class frequencies over the ensemble



WHY/WHEN DOES BAGGING HELP?

In one sentence:

Because the variability of the average of the predictions of many base learner models is smaller than the variability of the predictions from one such base learner model.
If the error of a base learner model is mostly due to (random) variability and not due to structural reasons, combining many such base learners by bagging helps reducing this variability.

WHY/WHEN DOES BAGGING HELP?

Assume we use quadratic loss and measure instability of the ensemble with $\Delta\left(f^{[M]}(\mathbf{x})\right) = \frac{1}{M}\sum_m^M \left(b^{[m]}(\mathbf{x}) - f^{[M]}(\mathbf{x})\right)^2$:

$$\begin{aligned}
\Delta\left(f^{[M]}(\mathbf{x})\right) &= \frac{1}{M}\sum_m^M \left(b^{[m]}(\mathbf{x}) - f^{[M]}(\mathbf{x})\right)^2 \\
&= \frac{1}{M}\sum_m^M \left(\left(b^{[m]}(\mathbf{x}) - y\right) + \left(y - f^{[M]}(\mathbf{x})\right)\right)^2 \\
&= \frac{1}{M}\sum_m^M L(y, b^{[m]}(\mathbf{x})) + L(y, f^{[M]}(\mathbf{x})) \underbrace{-2\left(y - \frac{1}{M}\sum_{m=1}^M b^{[m]}(\mathbf{x})\right)\left(y - f^{[M]}(\mathbf{x})\right)}_{-2L\left(y, f^{[M]}(\mathbf{x})\right)}
\end{aligned}$$

So, if we take the expected value over the data's distribution:

$$\mathbb{E}_{xy}\left[L\left(y, f^{[M]}(\mathbf{x})\right)\right] = \frac{1}{M}\sum_m^M \mathbb{E}_{xy}\left[L\left(y, b^{[m]}(\mathbf{x})\right)\right] - \mathbb{E}_{xy}\left[\Delta\left(f^{[M]}(\mathbf{x})\right)\right]$$

$\Rightarrow$ The expected loss of the ensemble is lower than the average loss of the single base learner by the amount of instability in the ensemble's base learners.

The more accurate and diverse the base learners, the better.

IMPROVING BAGGING

How to make $\mathbb{E}_{xy}\left[\Delta\left(f^{[M]}(\mathbf{x})\right)\right]$ as large as possible?

$$\mathbb{E}_{xy}\left[L\left(y, f^{[M]}(\mathbf{x})\right)\right] = \frac{1}{M}\sum_{m}^{M}\mathbb{E}_{xy}\left[L\left(y, b^{[m]}(\mathbf{x})\right)\right] - \mathbb{E}_{xy}\left[\Delta\left(f^{[M]}(\mathbf{x})\right)\right]$$

Assume $\mathbb{E}_{xy}\left[b^{[m]}(\mathbf{x})\right] = 0$ for simplicity, $\text{Var}_{xy}\left[b^{[m]}(\mathbf{x})\right] = \mathbb{E}_{xy}\left[(b^{[m]}(\mathbf{x}))^2\right] = \sigma^2$,
$\text{Corr}_{xy}\left[b^{[m]}(\mathbf{x}), b^{[m']}(\mathbf{x})\right] = \rho$ for all m, m'.

$$\implies \text{Var}_{xy}\left[f^{[M]}(\mathbf{x})\right] = \frac{1}{M}\sigma^2 + \frac{M-1}{M}\rho\sigma^2 \qquad \left(\ldots = \mathbb{E}_{xy}\left[(f^{[M]}(\mathbf{x}))^2\right]\right)$$

$$\begin{aligned}
\mathbb{E}_{xy}\left[\Delta\left(f^{[M]}(\mathbf{x})\right)\right] &= \frac{1}{M}\sum_{m}^{M}\mathbb{E}_{xy}\left[\left(b^{[m]}(\mathbf{x}) - f^{[M]}(\mathbf{x})\right)^2\right] \\
&= \frac{1}{M}\left(M\mathbb{E}_{xy}\left[(b^{[m]}(\mathbf{x}))^2\right] + M\mathbb{E}_{xy}\left[(f^{[M]}(\mathbf{x}))^2\right] - 2M\mathbb{E}_{xy}\left[b^{[m]}(\mathbf{x})f^{[M]}(\mathbf{x})\right]\right) \\
&= \sigma^2 + \mathbb{E}_{xy}\left[(f^{[M]}(\mathbf{x}))^2\right] - 2\frac{1}{M}\sum_{m'}^{M}\underbrace{\mathbb{E}_{xy}\left[b^{[m]}(\mathbf{x})b^{[m']}(\mathbf{x})\right]}_{=\text{Cov}_{xy}\left[b^{[m]}(\mathbf{x}), b^{[m']}(\mathbf{x})\right] + \mathbb{E}_{xy}\left[b^{[m]}(\mathbf{x})\right]\mathbb{E}_{xy}\left[b^{[m']}(\mathbf{x})\right]} \\
&= \sigma^2 + \left(\frac{1}{M}\sigma^2 + \frac{M-1}{M}\rho\sigma^2\right) - 2\left(\frac{M-1}{M}\rho\sigma^2 + \frac{1}{M}\sigma^2 + 0\cdot 0\right) \\
&= \frac{M-1}{M}\sigma^2(1-\rho)
\end{aligned}$$

IMPROVING BAGGING

$$\mathbb{E}_{xy}\left[L\left(y, f^{[M]}(\mathbf{x})\right)\right] = \frac{1}{M}\sum_{m}^{M}\mathbb{E}_{xy}\left[L\left(y, b^{[m]}(\mathbf{x})\right)\right] - \mathbb{E}_{xy}\left[\Delta\left(f^{[M]}(\mathbf{x})\right)\right]$$

$$\mathbb{E}_{xy}\left[\Delta\left(f^{[M]}(\mathbf{x})\right)\right] \cong \frac{M-1}{M}\mathrm{Var}_{xy}\left[b^{[m]}(\mathbf{x})\right]\left(1 - \mathrm{Corr}_{xy}\left[b^{[m]}(\mathbf{x}), b^{[m']}(\mathbf{x})\right]\right)$$

- $\Rightarrow$ **better base learners** are better (... duh)
- $\Rightarrow$ **more base learners** are better (theoretically, at least...)
- $\Rightarrow$ **more variable base learners** are better (as long as their risk stays the same, of course!)
- $\Rightarrow$ **less correlation between base learners** is better:
  bagging helps more if base learners are wrong in different ways so that their errors “cancel” each other out.

BAGGING: SYNOPSIS

- Basic idea: fit the same model repeatedly on many **bootstrap** replications of the training data set and **aggregate** the results
- Gains performance by reducing the variance of predictions, but (slightly) increases the bias: it reuses training data many times, so small mistakes can get amplified.
- Works best for unstable/high-variance base learners, where small changes in the training set can cause large changes in predictions: e.g., CART, neural networks, step-wise/forward/backward variable selection for regression
- Works best if base learners' predictions are only weakly correlated: they don't all make the same mistakes.
- Can degrade performance for stable methods like k-NN, LDA, Naive Bayes, linear regression