Data Science
Bagging – Random Forest

Séverine Affeldt

Université de Paris
Centre Borelli, CNRS

2022 – 2023

BAGGING (M2...)

Idea of Bagging

Make a *vote* from B trees built from bootstrap samples.

Learning

Input number of models B, learning algorithme $ALGO$, data $\Omega = \{\{X_p\}, Y\}$, $|\Omega| = \text{n}$

Iteration

$MODELS = \{\}$

For b in 1 to B, do

   Draw n samples (with replacement), Ω_b

   Built a model M_b on Ω_b with $ALGO$

   Add M_b to $MODELS$

end For

Output $MODELS$

Classifying

For $i^\star$ to be classified

   Apply each model to get an ensemble of $\hat{y}_b(i^\star)$

   Make a simple vote $\hat{y}_{bag}(i^\star) = argmax_k \left[\sum_b^B I\Big(\hat{y}_b(i^\star) = y_k\Big) \right]$

Bagging - Pros

- **Cooperation** Multiple models with *various* predictions are cooperating (Caution: you should not use too similar models)
- **Variance** Individual variance is compensated by the cooperation of the ensemble
- **No overfitting** A large B does not imply overfitting (In practice: $B \geq 100$)

Bagging - Cons

- **Bias** Bagging does not manage the underfitting (but, you can reduce the bias with deep trees)
- You should avoid **weak classifiers** (cf. Boosting)

We want to estimate an a posteriori probability for $Y \in \{+, -\}$, ie. $P(Y = +|\mathbf{X})$

Using the vote frequency

$$P(Y = +|\mathbf{X}) = \frac{\sum_{b=1}^{B} I(\hat{y}_b = +)}{B}$$

Using the probability P_b of each model M_b

$$P(Y = +|\mathbf{X}) = \frac{\sum_{b=1}^{B} \hat{P}_b(Y = +|\mathbf{X})}{B}$$

NB: Better when only few models

Issue

We have now B models

$\Rightarrow$ cannot inspect all trees to identify the most important features

Straightforward solution

Averaging the measure (eg., Tschuprow T, Gini) over all M_b models for all attributes.

Out-of-bag (OOB) error estimation

The error can be measured **during** the bootstrap **training**.

Each model M_b is based on n samples drawn **with replacement**.
Hence, some samples are not used.

$\rightarrow$ We can use these samples as test set ($\approx 36.8\%$ of the samples).

Why the tests set can be estimated to 36.8%

Sampling with replacement $\sim$ sequence of binomial trials with success $=$ being chosen

For n samples, $P(success) = 1/n$ and $P(failure) = (n-1)/n$

For a subsample of size b, the odds of selecting a sample x times are

$$P(x, b, n) = \left(\frac{1}{n}\right)^x \left(\frac{n-1}{n}\right)^{b-x} \binom{b}{x}$$

For bootstrap, $b = n$; if we have enough samples ($n \rightarrow +\inf$),

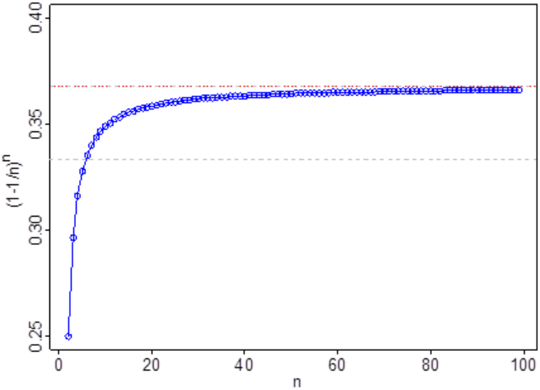
$$P(x, b, n) = \lim_{n \rightarrow +\infty} \left(\frac{1}{n}\right)^x \left(\frac{n-1}{n}\right)^{n-x} \binom{n}{x} = \frac{1}{ex!}$$

Finally, if $x = 0$ (never selected),

$$P(x = 0, b = n, n) = \lim_{n \rightarrow +\infty} \left(\frac{n-1}{n}\right)^n = \frac{1}{e} \approx 0.368$$

So, with the bootstrap approach, each model M_b is learnt on $\approx 63.2\%$ of the dataset.

What is a large n?



For $n \geq 11$, you already reach $1/3$

Definition of margin (for bagging)

The margin is the difference between the proportion of *vote* for the true class and the proportion of *vote* for another. When $Y \in \{+,-\}$,
For one sample $Y(i) = y_{k^\star}$

$$m = \frac{\sum_{b=1}^{B} I(\hat{y}_b(i) = k^\star)}{B} - max_{k \neq k^\star} \frac{\sum_{b=1}^{B} I(\hat{y}_b(i) = k)}{B}$$

A **large margin** means a **sharp decision** and a **small variance**
$\Rightarrow$ the ensemble approaches increase the margin

RANDOM FOREST (M2...)

Efficient bagging

- Individually efficient trees
- Deep trees (weak bias) – min leaf size $= 1$
- **Decorrelated trees** – strongly different trees that be complementary

The idea of Random Forest

Introduce a stochastic perturbation when learning the models M_b by considering only m variables out of p, randomly selected for each split.

Learning *(modified bagging)*

Input number of models B, learning algorithme $ALGO$, data $\Omega = \{\{X_p\}, Y\}$, $|\Omega| = n$, m (default: $m = \sqrt{(p)}$)

Iteration

$MODELS = \{\}$
For b in 1 to B, do
  Draw n samples (with replacement), Ω_b
  Built a model M_b on Ω_b with $ALGO$
  For each split, do
    Choose m variables **at random** among $\{X_p\}$
    Split with the best variables among m
  Add M_b to $MODELS$
end For

Output $MODELS$

Classifying *(same as bagging)*

For $i^\star$ to be classified
  Apply each model to get an ensemble of $\hat{y}_b(i^\star)$
  Make a simple vote $\hat{y}_{bag}(i^\star) = argmax_k \left[\sum_b^B I\left(\hat{y}_b(i^\star) = y_k\right)\right]$

Pros

- Good prediction performance in general
- Few parameters (B and m)
- When large B, no overfitting
- Variables ranking
- Error evaluation while learning (OOB)
- Parallelization

Cons

- Bad performance if few good predictors X ($\Rightarrow$ each M_b might be bad...)

Bibliography

- Rakotomalala, Ricco. “Arbres de décision.” Revue Modulad 33 (2005) : 163 – 187.
- Tan, Pang-Ning, Michael Steinbach, and Vipin Kumar. *Classification: basic concepts, decision trees, and model evaluation.* Introduction to data mining 1 (2006) : 145 – 205.
- Gonzales, Pierre-Louis.“Segmentation”, 2010
- Zighed, D. A., and R. Rakotomalala. “Graphes d’induction”, Hermès. Annexe A (2000).

Temporary page!

LaTeX was unable to guess the total number of pages correctly. A
was some unprocessed data that should have been added to the
this extra page has been added to receive it.

If you rerun the document (without altering it) this surplus page
away, because LaTeX now knows how many pages to expect for t
document.