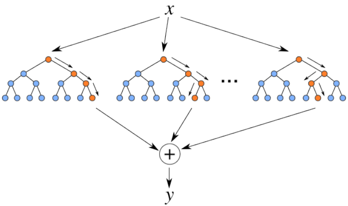
Introduction to Machine Learning

Random Forest: Introduction



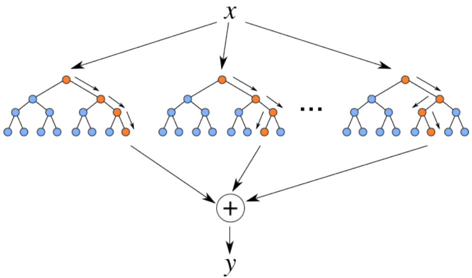
Learning goals

- Know how random forests are defined by extending the idea of bagging
- Understand that the goal is to decorrelate the trees
- Understand that the out-of-bag error is a way to obtain unbiased estimates of the generalization error during training

RANDOM FORESTS

Modification of bagging for trees proposed by Breiman (2001):

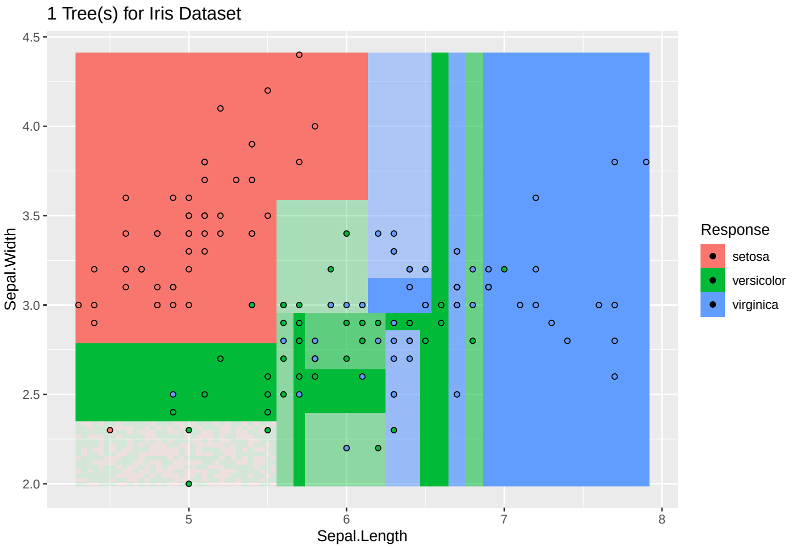
- Tree base learners on bootstrap samples of the data
- Uses **decorrelated** trees by randomizing splits (see below)
- Tree base learners are usually fully expanded, without aggressive early stopping or pruning, to **increase variance of the ensemble**



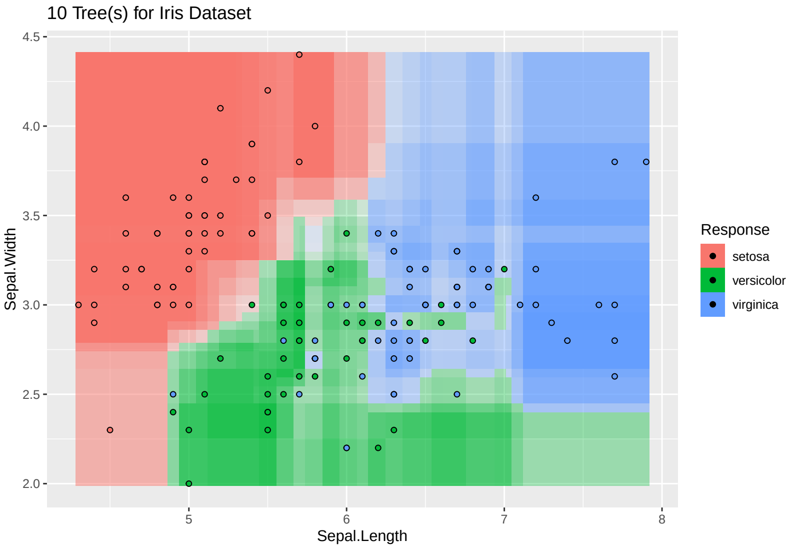
RANDOM FEATURE SAMPLING

- From our analysis of bagging risk we can see that decorrelating trees improves the ensemble
- Simple randomized approach:
  At each node of each tree, randomly draw mtry $\leq p$ candidate features to consider for splitting. Recommended values:
  - Classification: mtry $= \lfloor \sqrt{p} \rfloor$
  - Regression: mtry $= \lfloor p/3 \rfloor$

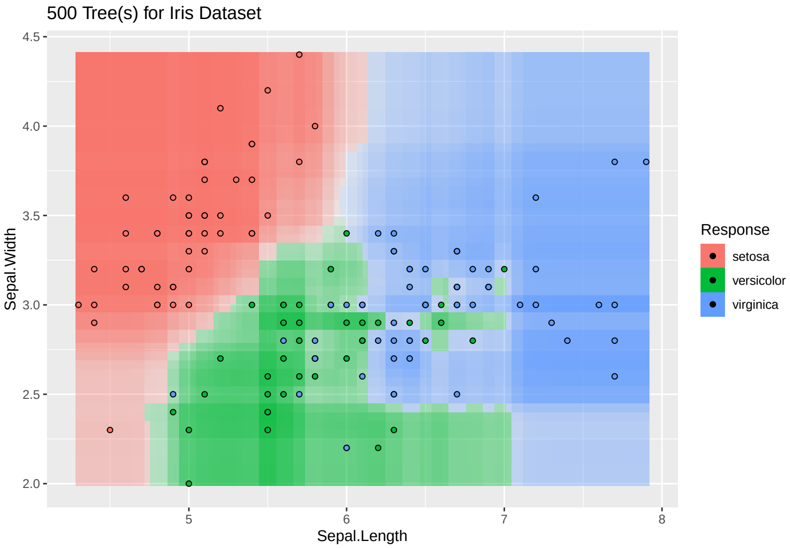
EFFECT OF ENSEMBLE SIZE



EFFECT OF ENSEMBLE SIZE

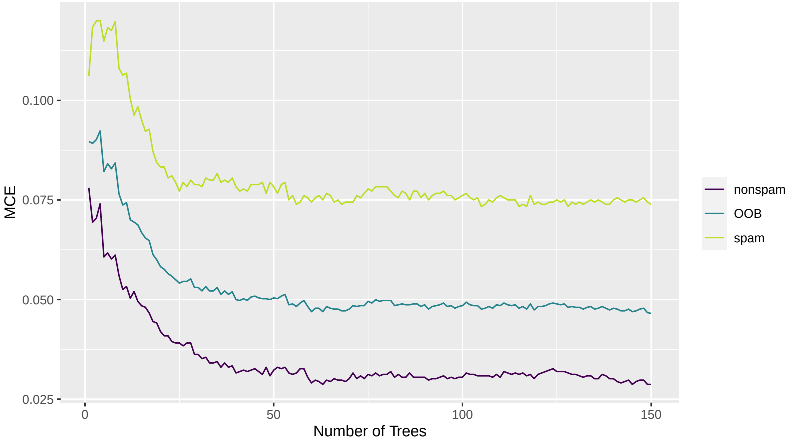


EFFECT OF ENSEMBLE SIZE



OUT-OF-BAG ERROR ESTIMATE

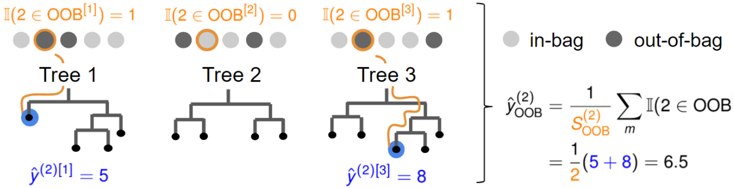
With the RF it is possible to obtain unbiased estimates of the generalization error directly during training, based on the out-of-bag observations for each tree:



OUT-OF-BAG PREDICTIONS

- For an estimation of the generalization error, we exploit the fact that the i-th observation acts as unseen test point for all trees in which it is OOB.
- Let $\text{OOB}^{[m]}$ denote the index set $\left\{i \in \{1, \ldots, n\} | (\mathbf{x}^{(i)}, y^{(i)}) \text{ is OOB for } b^{[m]}(\mathbf{x})\right\}$.
- The number of trees for which the i-th observation is OOB is then given by $S_{\text{OOB}}^{(i)} = \sum_{m=1}^{M} \mathbb{I}(i \in \text{OOB}^{[m]})$.
- We can compute the ensemble OOB prediction for each observation as:

$$\hat{y}_{\text{OOB}}^{(i)} = \begin{cases} \frac{1}{S_{\text{OOB}}^{(i)}} \sum_{m=1}^{M} \mathbb{I}(i \in \text{OOB}^{[m]}) \cdot \hat{y}^{(i)[m]} & \text{in regression,} \\ \left[\frac{1}{S_{\text{OOB}}^{(i)}} \sum_{m=1}^{M} \mathbb{I}(i \in \text{OOB}^{[m]}) \cdot \mathbb{I}(\hat{h}^{(i)[m]} = k)\right]_{k \in \{1, \ldots, g\}} & \text{in classification.} \end{cases}$$



$$\hat{y}_{\text{OOB}}^{(2)} = \frac{1}{S_{\text{OOB}}^{(2)}} \sum_{m} \mathbb{I}(2 \in \text{OOB}^{[m]}) \hat{y}^{(2)[m]} = \frac{1}{2}(5 + 8) = 6.5$$

OUT-OF-BAG ERROR

- Note that the ensemble OOB predictions $\hat{y}^{(i)}_{\text{OOB}}$ are scalars in regression and g-valued probability vectors in classification.
- Now we take the average of the resulting point-wise losses to estimate the OOB error of the forest:

$$\widehat{\text{err}}_{\text{OOB}} = \frac{1}{n}\sum_{i=1}^{n} L(y^{(i)}, \hat{y}^{(i)}_{\text{OOB}})$$

- Computing the probability of i being OOB in the m-th tree, we can see that the OOB error estimation is actually akin to 3-fold CV:

$$\mathbb{P}(i \in \text{OOB}^{[m]}) = \left(1 - \frac{1}{n}\right)^n \overset{n\to\infty}{\longrightarrow} \frac{1}{e} \approx 0.37$$

for $i \in \{1, \dots, n\}$.