INTUITION BEHIND DECORRELATION

- Since bootstrap samples are similar, models $\hat{b}^{[m]}$ are correlated, affecting the variance of an ensemble $\hat{f}$
- We would like variance to go down linearly with ensemble size, but because of correlation we cannot really expect that
- Assuming $\text{Var}(\hat{b}^{[m]}) = \sigma^2$, $\text{Corr}(\hat{b}^{[m]}, \hat{b}^{[j]}) = \rho$, semi-formal analysis, without proper analysis of prediction error:

$$\text{Var}\left(\hat{f}\right) = \text{Var}\left(\frac{1}{M}\sum_{m=1}^{M}\hat{b}^{[m]}\right) = \frac{1}{M^2}\left(\sum_{m=1}^{M}\text{Var}(\hat{b}^{[m]}) + 2\sum_{m<j}\text{Cov}(\hat{b}^{[m]}, \hat{b}^{[j]})\right)$$

$$= \frac{1}{M^2}\left(M\sigma^2 + 2\frac{M(M-1)}{2}\rho\sigma^2\right) = (1-\rho)\frac{\sigma^2}{M} + \rho\sigma^2$$

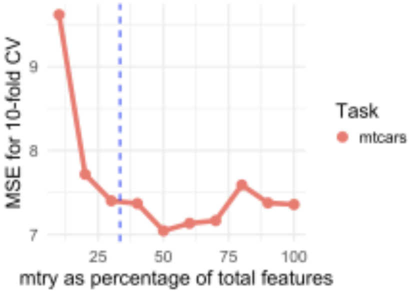
- Ensemble variance is "convex-combo of linear-reduction and no-reduction, controlled by ρ"
- Maybe we can decorrelate trees, to reduce ensemble variance? And get less prediction error?

RANDOM FEATURE SAMPLING

RFs decorrelate trees with a simple randomization:

- For each node of tree, randomly draw `mtry` $\leq p$ features (`mtry` = name in some implementations)
- Only consider these features for finding the best split
- Careful: Our previous analysis was simplified! The more we decorrelate by this, the more random the trees become! This also has negative effects!

EFFECT OF FEATURE SAMPLING

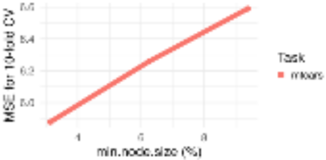


- Optimal `mtry` typically larger for regression than for classification
- Good defaults exist, but still most relevant tuning param
- Rule of thumb:
  - Classification: $\texttt{mtry} = \lfloor \sqrt{p} \rfloor$
  - Regression: $\texttt{mtry} = \lfloor p/3 \rfloor$

TREE SIZE

In addition to `mtry`, RFs have two other important HPs:

- Min. nr. of obs. in each decision tree node
  Default (ranger): `min.node.size = 5` ▸ Breiman 2001

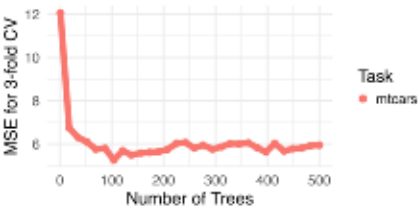


- Depth of each tree
  Default (ranger): `maxDepth` $= \infty$
- There are more alternative HPs to control depth of tree: minimal risk reduction, size of terminal nodes, etc.

RF usually use fully expanded trees, without aggressive early stopping or pruning, to further **increase variability of each tree.** ▸ Louppe 2015

CAN RF OVERFIT? Probst and Boulesteix 2018

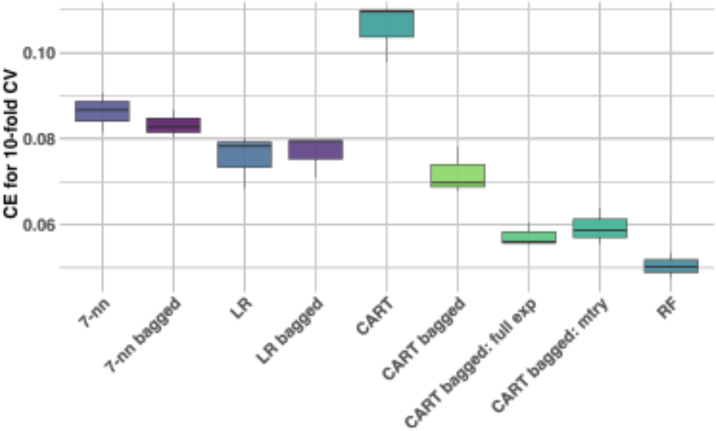
- Just like any other learner, RFs **can** overfit!
- However, RFs generally **less** prone to overfitting than individual CARTs.
- Overly complex trees can *still* lead to overfitting!
  If most trees capture noise, so does the RF.
- But randomization and averaging helps.



Since each tree is trained *individually and without knowledge of previously trained trees*, increasing `ntrees` generally reduces variance **without increasing the chance of overfitting!**

RF IN PRACTICE

Benchmarking bagged ensembles with 100 BLs each on `spam` versus RF (`ntrees` $= 100$, `mtry` $= \sqrt{p}$, `minnode` $= 1$), we see how well RF performs!



$\Rightarrow$ RFs combine the benefits of random feature selection and fully expanded trees.