Ensembles and Random Forests

Ryan Miller

Limitations of Decision Trees

- Decision trees are easy to interpret and don't require much computation to train
  - However, capturing a complex relationship using a decision tree requires the tree be deep (lots of splits)
  - Deep trees are high variance model and are prone to overfitting
- This presentation will introduce *random forests* as an extension of decision tree modeling

Bagging

Random forests rely on a strategy known as *bagging*, or "bootstrap aggregation":

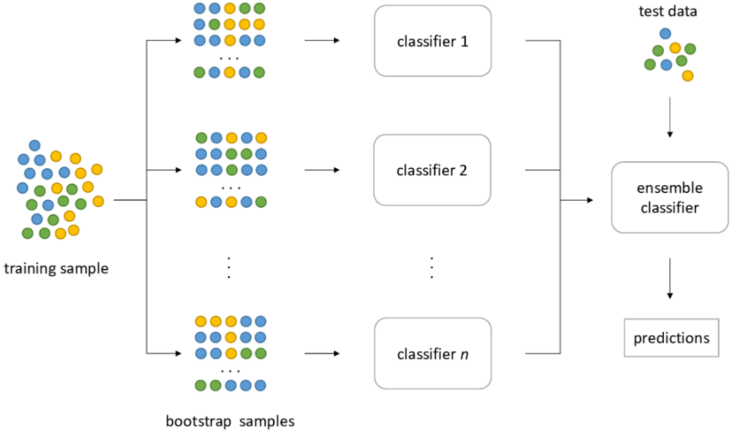


Image credit: https://hudsonthames.org/bagging-in-financial-machine-learning-sequential-bootstrapping-python/

Ensembles

- Bagging produces an *ensemble model* comprised of many different base models
  - Each base model contributes towards the final prediction of the ensemble, either by majority/weighted voting (classification) or simple/weighted averaging (regression)

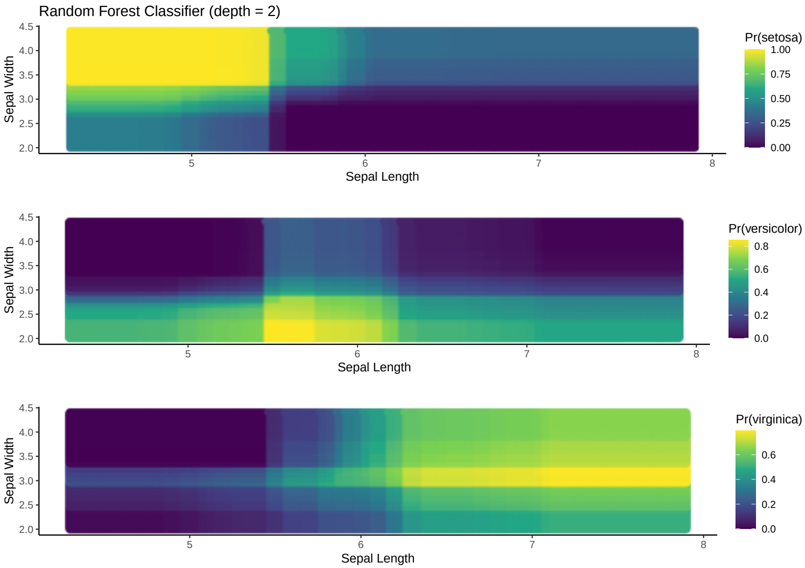
Ensembles

- Bagging produces an *ensemble model* comprised of many different base models
  - Each base model contributes towards the final prediction of the ensemble, either by majority/weighted voting (classification) or simple/weighted averaging (regression)
- Random forests are an ensemble model built using bagging, where each base model is a decision tree
  - What would happen if bagging were not used?

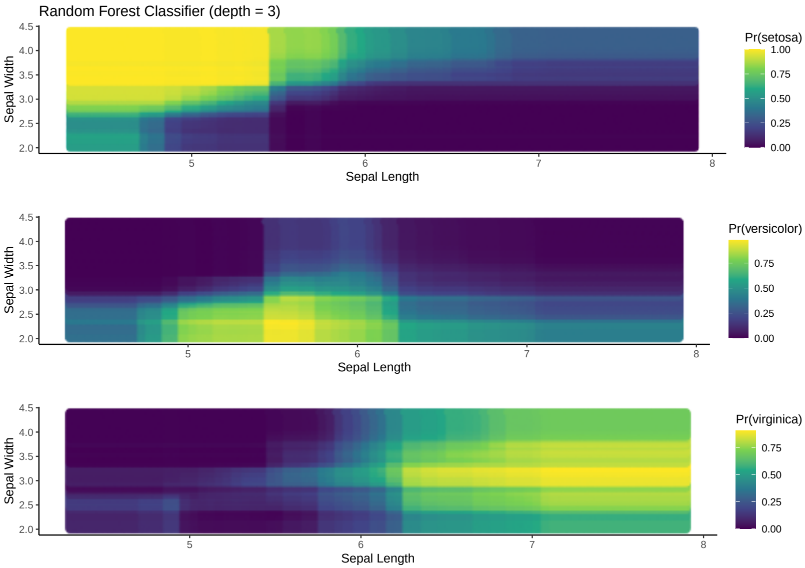
Random Forest

- Bagging is one strategy used by random forests to address the limitations of a single decision tree
- Another is *predictor subsampling*, or the random selection of limited candidate pool of predictors to be considered at each split
  - What might happen if predictor subsampling were not used?

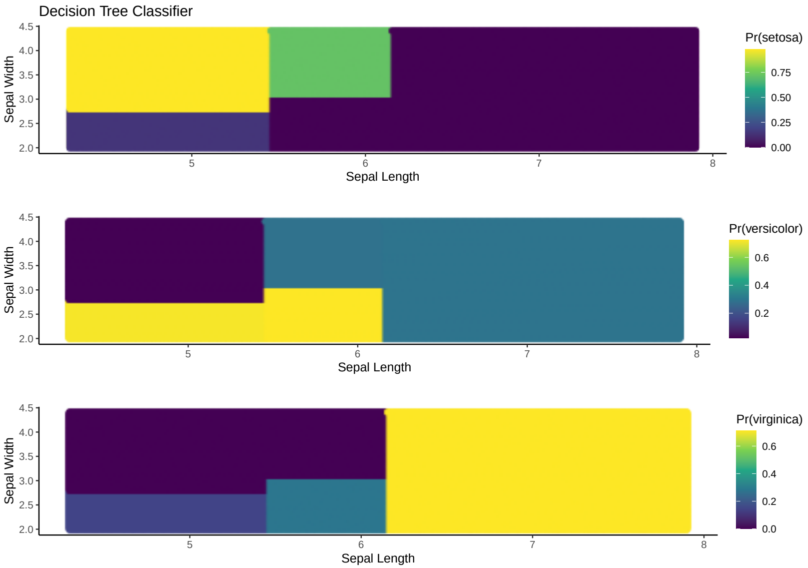
Random Forest (depth = 2)



Random Forest (depth = 3)



Single Decision Tree (depth = 3)



Final Remarks

- Random forests will generally offer better predictive performance than a single decision tree
  - The primary downside is that random forests are not easily interpretable
- Important tuning parameters are `max_depth`, `min_samples_split`, and `max_features` (the fraction of predictors considered at each split)
- The number of trees in the forest is also important, but including more trees past a certain point will not improve the ensemble.

Final Remarks (cont.)

- `max_depth` and `min_samples_split` help prevent base models from being overfit
  - By using an ensemble approach, random forests can be flexible without using deep trees
  - Thus, relative to a single decision tree, you should consider using a smaller `max_depth` and larger `min_samples_split`
- `max_features` governs the degree of correlation between base models
  - Smaller values reduce correlations between trees (at the expense of predictive power within individual trees)
  - Default recommendations are $\sqrt{p}$ for classification and $p/3$ for regression