Decision tree:

DT is simple and useful for interpretation. IN addition, no need to do the data transformation, like data standardization before modeling. But it is not competitive with other supervised methods like, linear regression with penalty, local regression, SVM… So it is often executed as a benchmark.

[That is, we consider all

predictors *X*1*, . . .,Xp*, and all possible values of the cutpoint *s* for each of

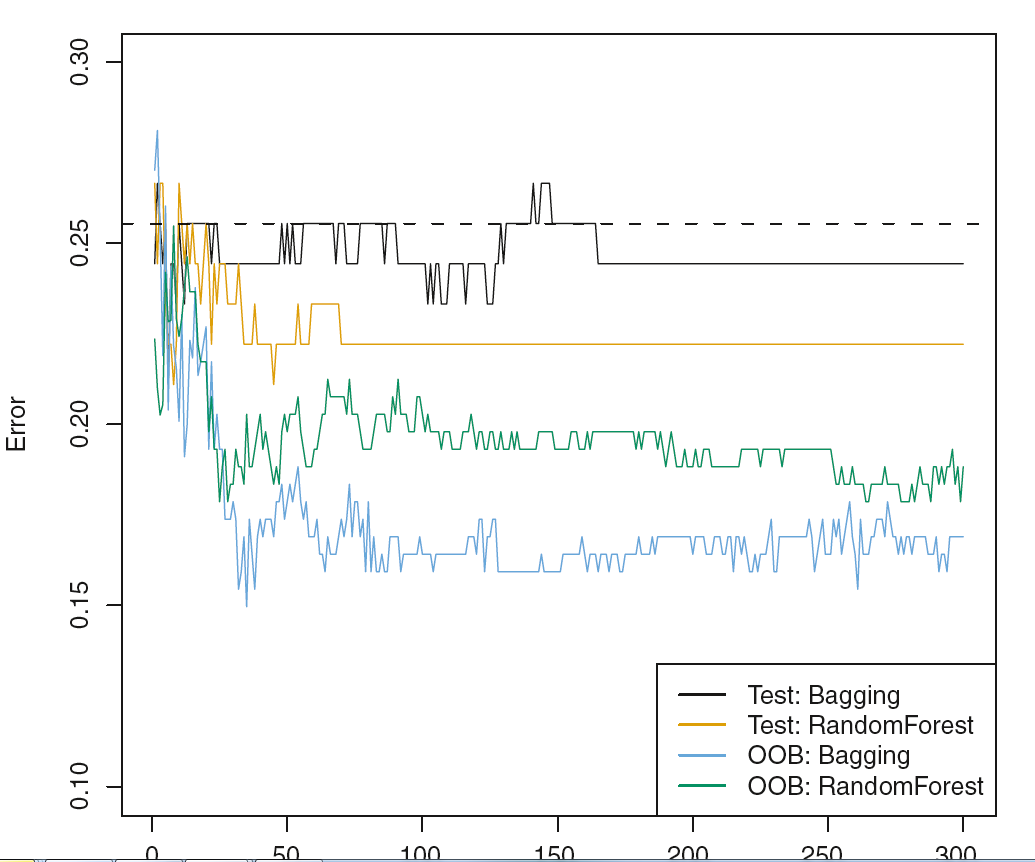
the predictors, and then choose the predictor and cutpoint such that the

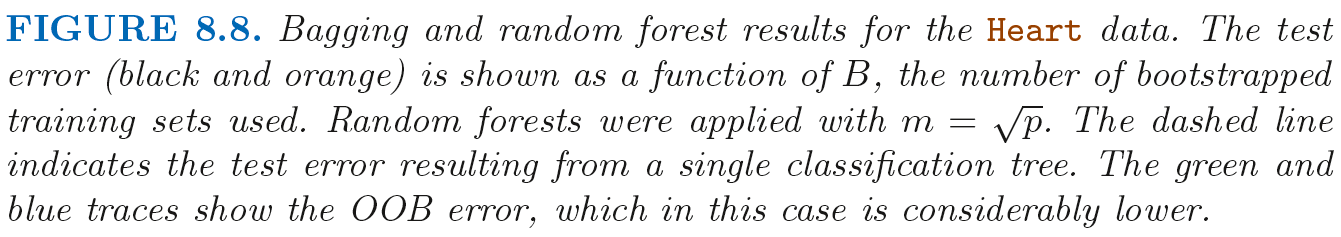
resulting tree has the lowest RSS(residual sum of squares)/test error rate.]

However, it turns out that classification

error is not sufficiently sensitive for node purity, we can also use Gini index. Gini index can be seen as a measure of node purity, a small value indicates purity node.

We tend to choose those split that can increasing node purity.





Variable Importance Measures

Thus, bagging improves prediction

accuracy at the expense of interpretability.

Decision tree has a good advantage, simply method of interpretation, especially by some graphic display. For Bagging, we can try to get the predictor importance to try to do some interpretation. How to do that? In regression tree, we can use the decreased RSS in all the bags to measure which split is the best split, that is which predictor should be used for this split. So similarly, for bagging classification trees, we can use the decreased Gini Index across all bags to measure which predictor is the most important for this split. That is the variable importance.

RandomForest

square root

in RF, instead of using all the predictors in each split of each tree, we just select a subset predictors as the candidates. The aim is to make the trees as irrelative as possible. That leads to decrease the overfitting compared with Bagging classification trees.

Using a small value of mtry in building RF will be helpful when we have a large number of **correlated predictors**

Boosting, unlike a single decision tree, which fits the data harder and potentially overfitting. Boosting learns slowly. Using the small trees to fit the residual, we slowly improve the fitting in area where is does not perform well. The shrinkage parameter lambda even further slow down the process of fitting

XGboost:

Objective function is comprised of loss function and regularization term.

Loss function is used to measure how the model fits the observation

Regularization term is used to control the complexity of the model, which help to avoid the overfitting.

XGboost and GBM(gradient boosted machines) are all based on gradient boosting. Specially, XGBoost uses a more regularized model formalization to control the overfitting , which gets it better performance.

Actually, xgboost is more suitable for regularized gradient boosting.

Xgboost vs deep net

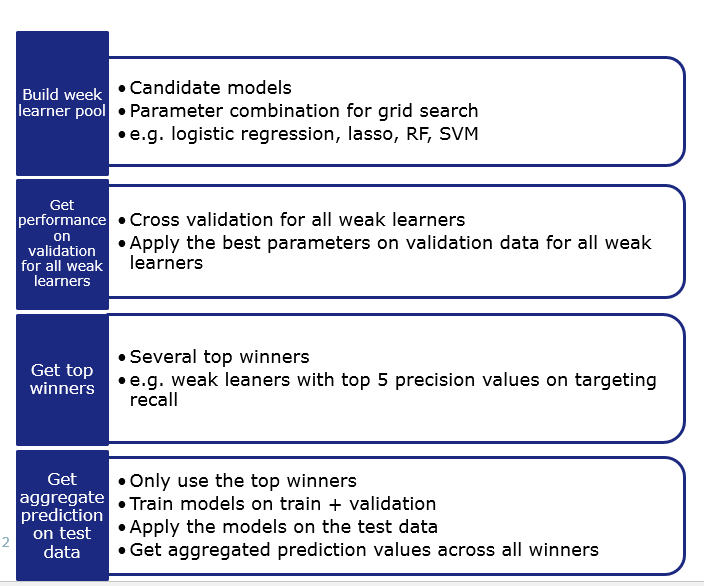
Deep net is good at capturing things like image, vadio…, tree based models like xgboost can deal with tabula data quite well. Also tree based models have good properties deep net does not have, (e.g. easy to interpret, not affected by input data scale, and easy to tune)

Xgboost runs much faster than many other gradient boosting

GBM vs random forest

Boosted trees usually have a better performance than random forest if tuning the parameters carefully.

1. Fitting term: boosted trees creates new tree based on the existing trees, so can get better fit using less number of trees.
2. Ideas of subsampling and bagging in rf can also be incorporated in boosted trees.
3. Speed. Rf can use penalization to speed the training. Which can also be employed in boosted trees.
4. Be generated to other weak learners. We literately correct the previous model using the new fitted model, which is used to fit the previous residual. So here the new fitted model is not required to be tree-based model. Gradient boosts is a framework to iteratively improve any weak learners. The number of iterative is decided by cross-validataion



MAP maximize a posterior estimation is a method to get the estimation based on empirical data and also involved the prior distribution, So MAP is a special case of MLE.

