Aggregating Bootstrapped Results

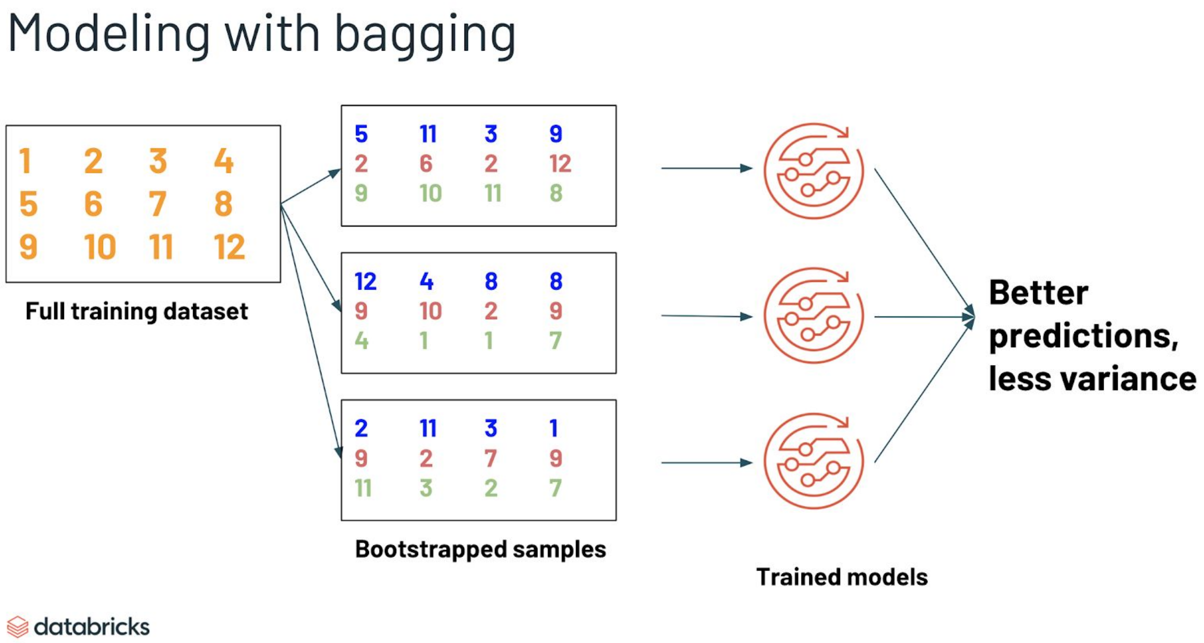
Estimated time: 7 minutes

We learned about the bootstrap method in the last video. In general, bootstrap aggregating (bagging) is a method for reducing the variance of a model: it is especially useful when applied to decision trees that often suffer from high variance - involves taking many sampled/bootstrapped training data sets, building separate models on each, and averaging the resulting predictions, to obtain a model with lower variance.

This works because there is variability in the training dataset, and therefore the fitted model is also subject to variability: if we had trained the model on another subsampled dataset from the population, we would have obtained a different model with different results.

Better predictive model

But we don’t usually have multiple training data sets - so we can artificially create them with bootstrapping. And then, we aggregate the results from the bootstrapped samples to arrive at better predictions.



Modeling with bagging

1	2	3	4
5	6	7	8
9	10	11	12

Full training dataset

5	11	3	9
2	6	2	12
9	10	11	8

12	4	8	8
9	10	2	9
4	1	1	7

2	11	3	1
9	2	7	9
11	3	2	7

Bootstrapped samples

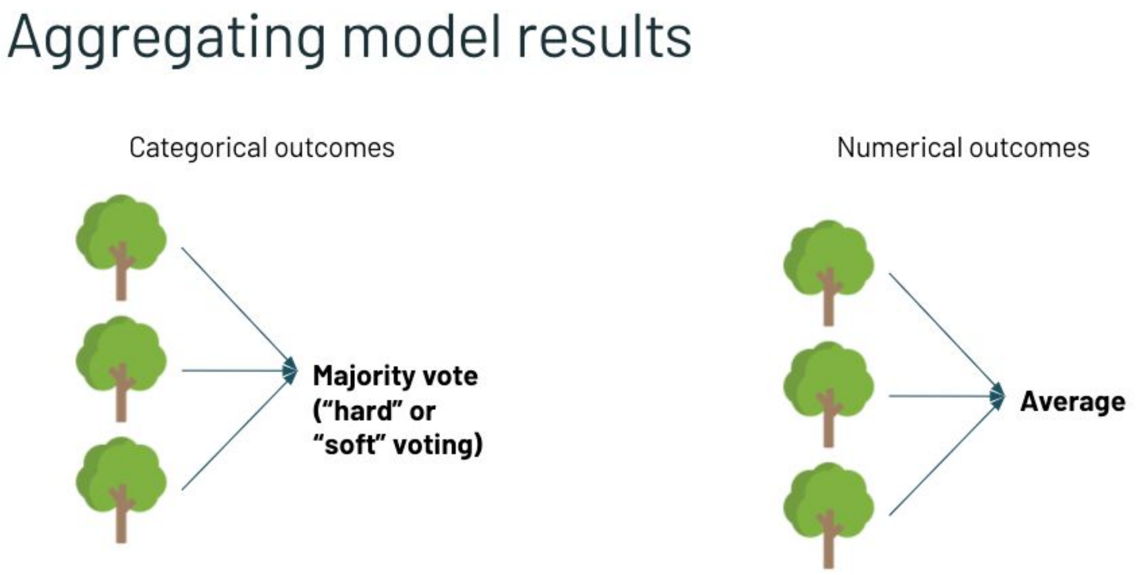
Trained models

Better predictions, less variance

databricks

How do we actually aggregate the results of different models into one output or result? For categorical there are two methods - with “hard voting”, we take the majority vote of all the models and assign the unknown data point to that class. Another method, called soft voting, is where we consider the probabilities of each class returned by all the models, average these probabilities and keep the class with the highest average probability.

For numerical, average the results of each individual model and apply that to the unknown data point to get the result of the ensemble model.



Aggregating model results

Categorical outcomes

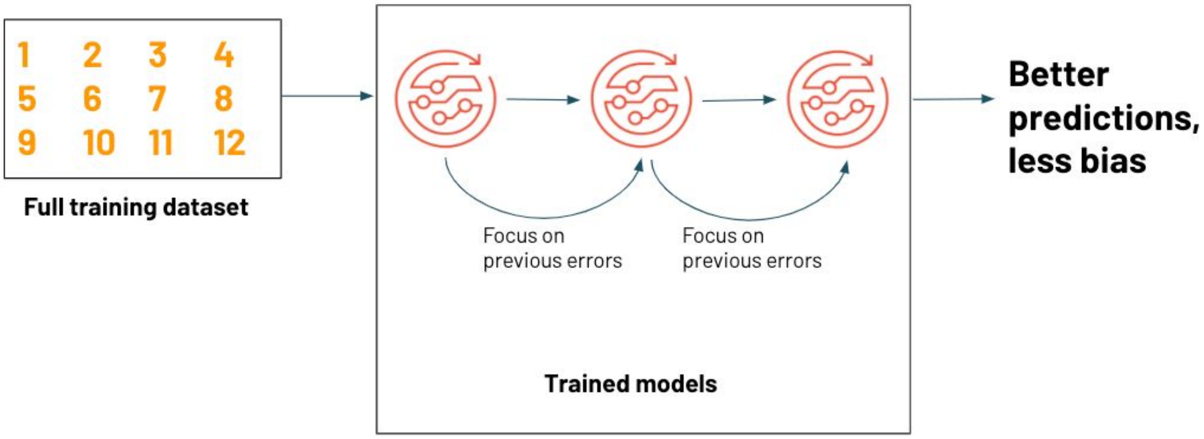
Majority vote ("hard" or "soft" voting)

Numerical outcomes

Average

We mentioned boosting as another type of ensemble model in a previous video. Different from bagging - instead of combining multiple independent trees in parallel, it iteratively combines trees sequentially into one model. The trees are not independent - each tree attempts to correct the errors of the previous tree, by giving more weight to observations in the dataset that were incorrectly predicted by the previous models in the sequence. The resulting ensemble model will have less underfitting and bias than the individual weak learner models. One commonly used boosting technique is Gradient Boosting Trees. Another is called xgboost. These are often used to win data science competitions.

Modeling with boosting



1 2 3 4
5 6 7 8
9 10 11 12

Full training dataset

Focus on previous errors

Focus on previous errors

Trained models

Better predictions, less bias