* Meta learning approaches don’t innovate on the learning or training algorithms
* Why would ensemble methods work?
* Bagging reduces variance of low-bias models of simulating training multiple data sets
* Boosting reduces the bias of low-variance models
* A shallow decision tree is one that only has a few nodes
* Decision stumps have only one split
* Shallow trees learn roughly the same model
* Decision stumps are weak learnings because they have error rates that are slightly better than random guessing
* Weak learners are low variance and high bias
* Train a second regressor on the residuals of the first training
* The overall model (the ensemble) is the sum of all the models trained so far
* Gradient boosting is the example on 22
* In the first round of learning, you train on the original data set and you train on the stump
* Then, the predictions are used to create a new training set based on the residuals
* We’re creating a sequence of training sets and then we combine these models together to create a more complex model
* The actual training algorithm is unimportant
* This is sort of like gradient descent but only allowing axis-aligned update directions
* In axis aligned gradient descent, you’re descending along a specific coordinate; project your descent vector to the axis/coordinate that has the largest drop
* Linear model: one coefficient per feature
* Ensemble model: one coefficient per model
* Every axis is a weak model
* Find the h basis vector that reduces error the most
* Somehow there are coefficients for the different models; for now don’t worry about where the coefficients come from
* Multiple ways of deriving the same algorithm
* Can we also classification for linear classifiers
* Some weak learning ali,b
* Effectively a python for loop around your different functions
* Using a exponential for error measure
* The loss function I also exponential loss
* The scoring function 9 (target function)
* Trains weak classifier on reweighted data
* Iteratively finds weak classifier to minimize residual
* Exponential loss is very sensitive to outliers
* Bagging and boosting are both ensemble models; one for simple and one for complex models
* Validation error is used as a validation
* In ensemble selection, we don’t care and just try to reduce validation error
* The validation set must be large enough; you need a validation set to do this!
* There is a risk overfitting to the validation set