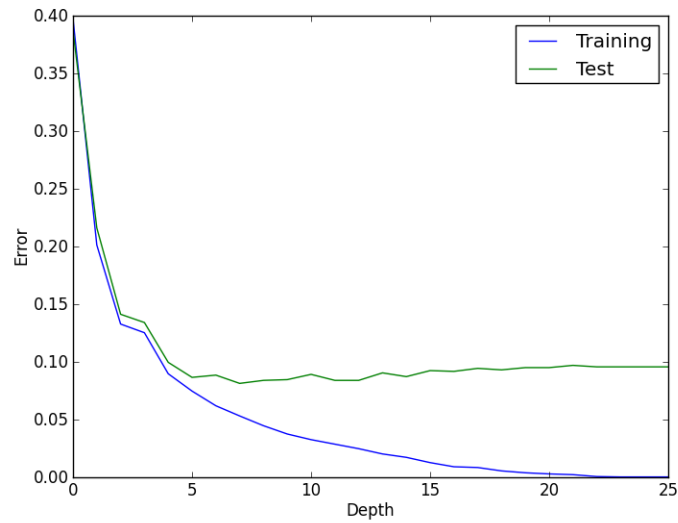


# CS189–Spring 2013 — Homework 5

Sharad Vikram, Gerald Fong, Kevin Yanofsky, section ,

April 18, 2013

## Decision Tree (with no pruning)

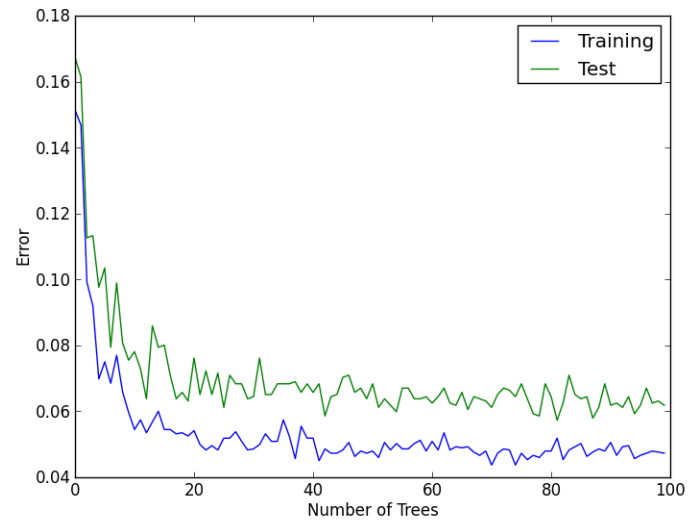


Depth	0	1	2	3	4	5	6	7	8
Training Error	0.397	0.201	0.133	0.125	0.0897	0.0747	0.062	0.0531	0.0447
Test Error	0.387	0.216	0.141	0.134	0.0996	0.0866	0.0885	0.0813	0.084

Best error rate:  $e = 0.0813$  with depth 7

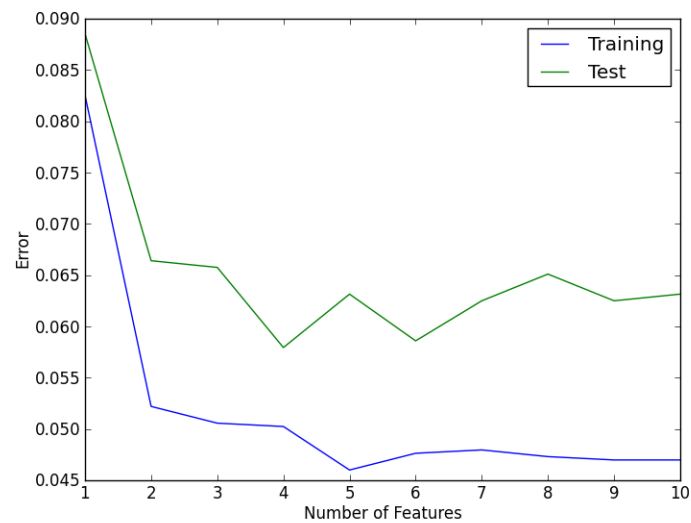
The decision tree has only one parameter which is the maximum depth that a leaf can be at. Once the recursive algorithm reaches a given depth, it will immediately stop splitting and will use a leaf node instead.

## Random Forests



7 features, 400 training points, 3 depth

Number of Trees	1	10	20	30	40	50	60	70	80	90	100
Training Error	0.15	0.06	0.05	0.048	0.052	0.047	0.048	0.048	0.048	0.048	0.047
Test Error	0.167	0.076	0.063	0.064	0.068	0.064	0.064	0.063	0.068	0.068	0.062



100 trees, 400 training points, 3 depth

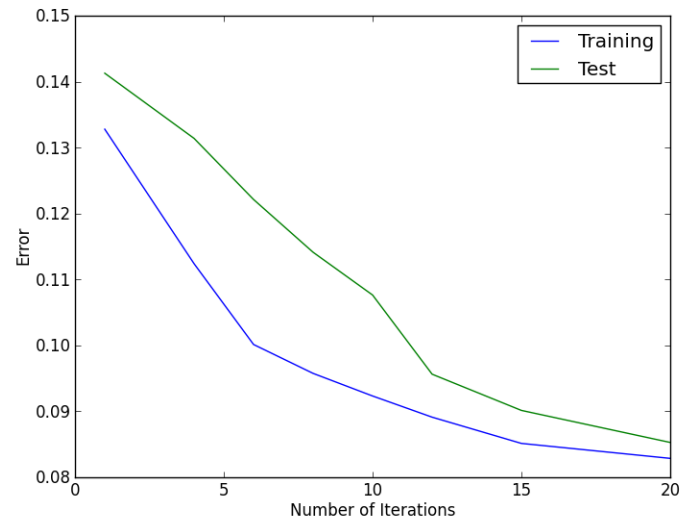
Number of Features	1	2	3	4	5	6	7	8	9	10
Training Error	0.083	0.052	0.051	0.05	0.046	0.048	0.048	0.047	0.047	0.047
Test Error	0.089	0.066	0.066	0.058	0.063	0.059	0.063	0.065	0.063	0.063

Best error rate:  $e = 0.059$  with depth 5, 200 points, 100 trees, 6 features

The randomized forest classifier had 4 parameters: maximum tree depth, number of features to split by, number of subsamples of training set to train on, and number of trees.

Tweaking with the parameters led to the result that a low training set number was never beneficial, whereas a lower number of features helped to prevent overfitting and thus improved the error rate. Another trend was that after reaching a certain number of trees, the error rate would not improve. Similarly, a moderate amount of depth in each tree would also help the error rate.

## AdaBoost



Iterations	1	4	6	8	10	12	15	20
Training Error	0.133	0.112	0.1	0.096	0.092	0.089	0.085	0.083
Test Error	0.141	0.131	0.122	0.114	0.108	0.096	0.09	0.085

Best error rate:  $e = 0.067$  with 100 iterations

In each iteration, we generated a weak learner (decision tree of depth 5) and trained it using the weighted training set. The learner would split nodes using a weighted entropy measure, and each iteration would train a new learner. The main parameter to be trained was the number of iterations, and a large number of iterations did not hurt.