Introduction to Artificial Intelligence
MNIST and CIFAR10 with Boundary Trees

Jay Ricco, David Van Chu

August 4, 2017

1 MNIST

Changing the maximum branching factor, k, played a large role in both accuracy as well as the speed of training and querying. As the branching factor grows, it takes longer to train and query, although the accuracy is higher, so it may be worth it depending on the use case. For example:

1. with branching factor = ∞, averaged over 10 runs
    - accuracy = 88%
    - training = 140.49 seconds
    - testing = 36.80 seconds
2. with branching factor = 5, averaged over 10 runs
    - accuracy = 85%
    - training = 72.54 seconds
    - testing = 14.19 seconds

After running the tests, I noticed that setting the branching factor to 5-10 seemed to be the best if both time and accuracy is a concern, as anything less than 5 will result in lower accuracy while taking about the same amount of time to train and test.

Another interesting point is that with a larger branching factor, the training time varies by quite a bit, ranging from 140.47 seconds to 223.79 seconds, whereas tests ran with a lower branching factor resulted in consistent training times.

2 CIFAR10

We see similar results when using Boundary Trees on the CIFAR10 dataset, although it never achieves over 27.6% accuracy. Using a smaller branching factor quickens the training time significantly compared to the MNIST dataset results.

1. with branching factor = ∞, averaged over 10 runs
    - accuracy = 27.3%
    - training = 125.91 seconds
    - testing = 52.22 seconds
2. with branching factor = 5, averaged over 10 runs
    - accuracy = 26.5%
    - training = 32.36 seconds
    - testing = 10.46 seconds