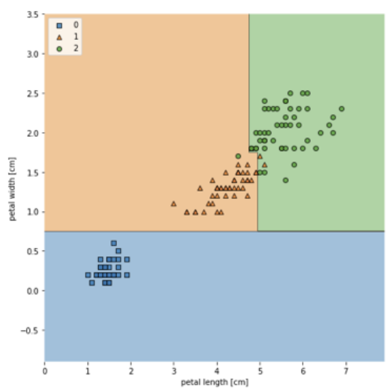
CAS CS 440
Lec 21

Supervised Learning III

1. Tree Complexity
    a. Trees can get crazy big
        i. The bigger the tree: the more complex the decision surface
    b. Last time: more complex functions = more likely to memorize
    c. How can we build in a way to keep things “simple”
        i. Only make a bigger tree if absolutely necessary
    d. Tree pruning!
        i. 

2. Decision Tree Pruning
    a. Two flavors of pruning:
        i. Pre-pruning:
            1. Prune while the decision tree is being constructed
            2. Stop growing the tree when some criteria is met
        ii. Post-pruning
            1. Make a decision tree
            2. Prune subtrees afterwards
            3. Two flavors of this:
                a. Top-down pruning
                b. Bottom-up pruning
3. Top-Down Post Pruning
    a. Pessimistic Error Pruning:
        i. When at node ‘n’ that has children c1, c2, …, ck
        ii. Delete a child one at a time
            1. Measure the % error
                a. If the % error increases: leave this child alone
                b. If the % error decreases: prune this child
        iii. Repeat for all surviving child nodes

4. Bottom-Up Post Pruning
    a. Reduced Error Pruning
    b. Minimum Error Pruning
    c. Minimum Cost Complexity Pruning
        i. Also build sequence of trees (from initial tree)
        ii. Each tree (in sequence) created by removing a subtree (from prev. tree in sequence)
            1. How to pick subtree to remove?
            2. Pick subtree ‘n’ that minimizes:
        iii. $$\frac{e\left(prune(T,\ n),\ \ \cdot\ \right) - e(T,\ \ \cdot\ )}{\left|n \in leaves(T)\right| - |leaves(prune(T,\ n))|}$$
5. Minimum Description Length (MDL)
    a. Often used to measure complexity of a function:
    $$MDL\left(h\right) = \#\ bits\left(h\right) + \#\ bits(D\ \left|\ h\right)$$
    b. For decision trees:
    $$MDL(T) = size(T) + \#misclassifications(T)$$
    c. Can use MDL as a criteria!
        i. Tradeoff between tree size and training errors!
        ii. Prefers smaller trees (when # misclassifications is constant)