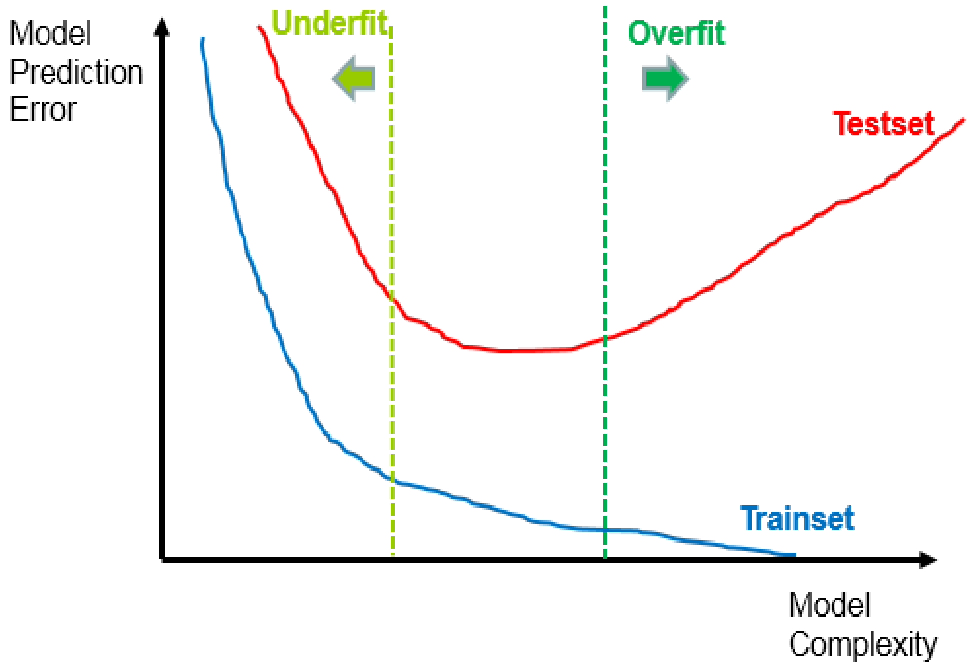
Complexity Penalty Parameter and Pruning the Tree

CART

Based on Chew C. H. (2020) textbook: AI, Analytics and Data Science. Vol 1., Chap 8.

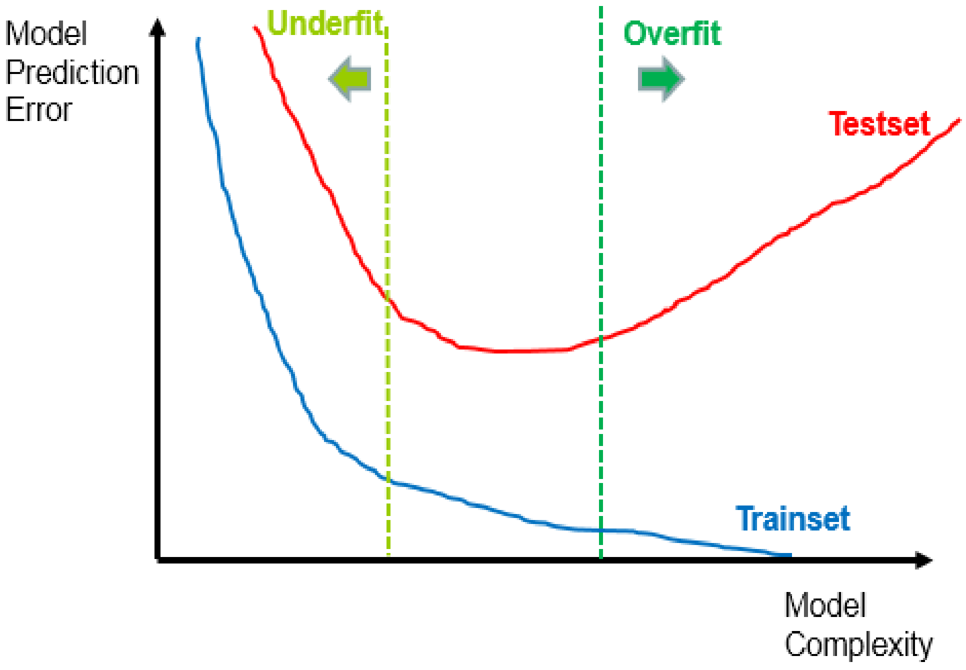
The Key to Pruning is to incorporate Complexity Penalty explicitly into Total Cost of using the Model

- The risk of Overfitting is omnipresent in all models.
- Overfitting is far less obvious than underfitting.
- The model cannot see the Testset during Tree growing and Tree Pruning.



The Key to Pruning is to incorporate Complexity Penalty explicitly into Total Cost of using the Model

- Observe the trade-off when Overfitting begins.
- Overfitting:
  - As Model increases in Complexity,
  - Trainset error decreases,
  - Testset error increases.
- Such trade-off can be incorporated into a formula.



Trade-offs incorporated into a formula via a Complexity Penalty Parameter α

Total Cost of CART = Misclassification Error + Total Complexity Cost

$$R_{\alpha}(T) = R(T) + \alpha \, | T \, |$$

Note:

In Breiman (1984) textbook, Complexity of tree |T| = number of terminal nodes.

- As model complexity |T| increases, R(T) decreases but α × |T| increases.
- α is a positive constant.
  - The penalty per unit model complexity.

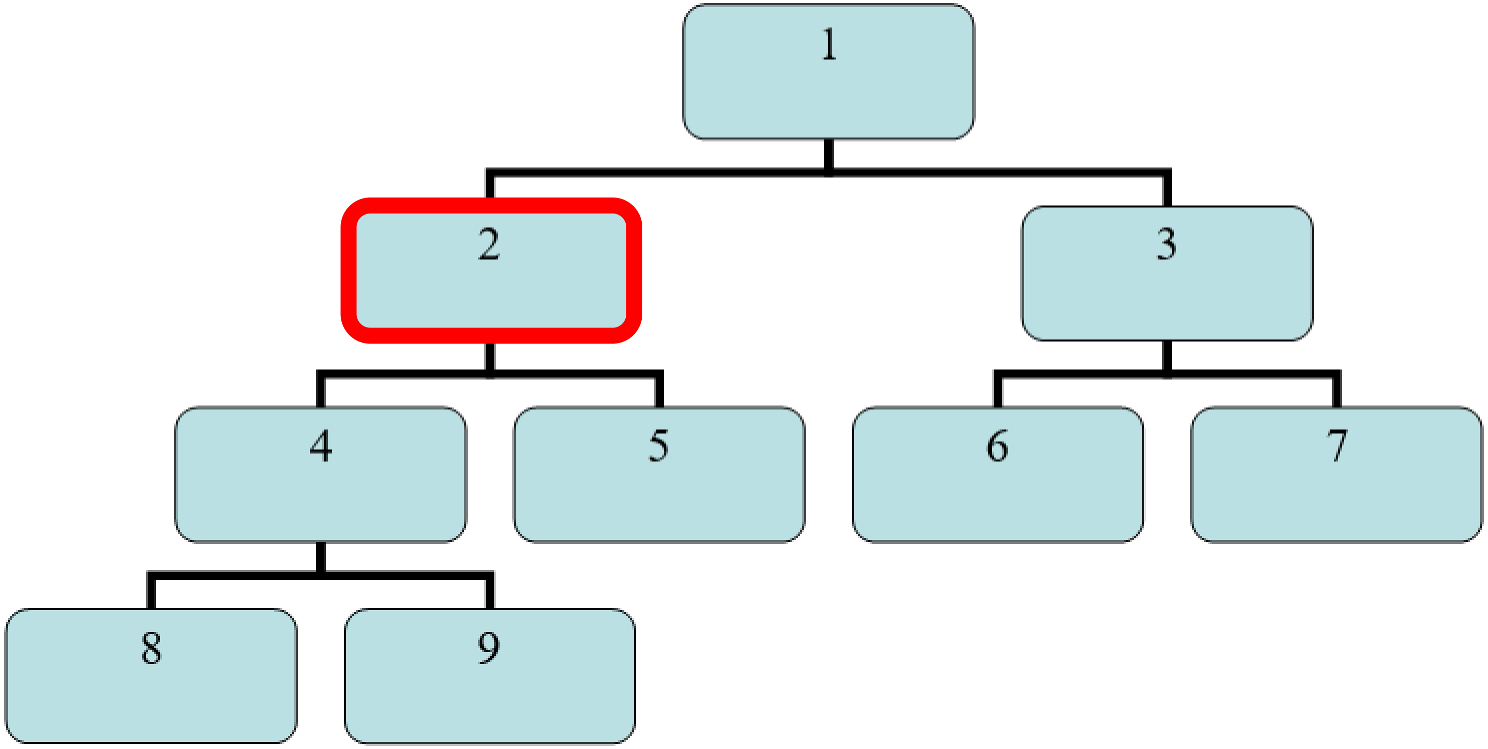
Pruning the Maximal Tree with Complexity Penalty

- Given a big tree that probably overfits, need to “prune” and simplify the tree by applying penalty for tree complexity.
- The bigger the penalty, the smaller the tree.
- The complexity of the tree can be measured by:
  - Number of terminal nodes (leaves), or
  - Number of splits
- What do we mean by “pruning the tree”?

Pruning the Tree (Before)

Tree Diagram BEFORE Pruning Node 2.

Terminal Nodes are 5, 6, 7, 8, 9. Internal nodes are 1, 2, 3, 4.

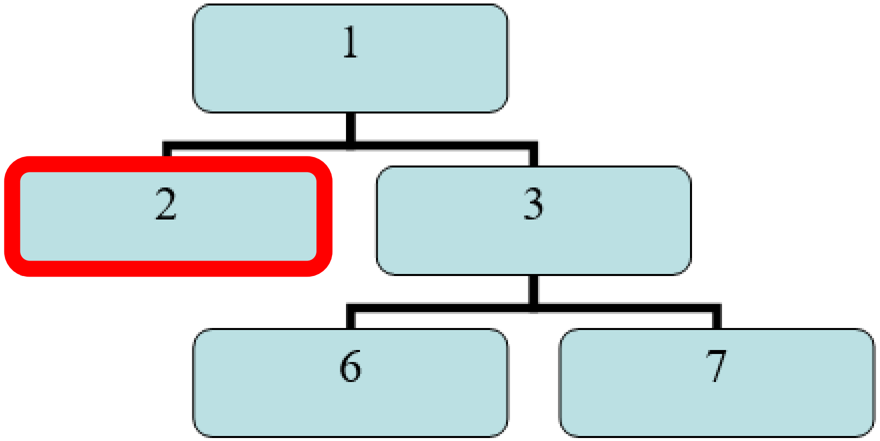


- Did you notice the node labelling convention? Implications?

Pruning the Tree (After)

Tree Diagram AFTER Pruning Node 2.

Terminal Nodes are 2, 6, 7. Internal nodes are 1, 3.

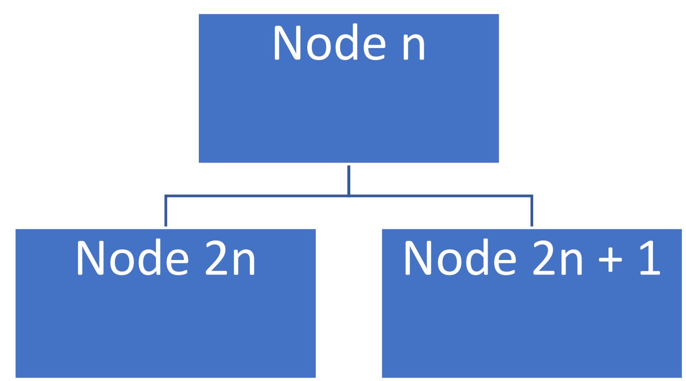


Thus, pruning at node t means all descendants of node t are cut off.

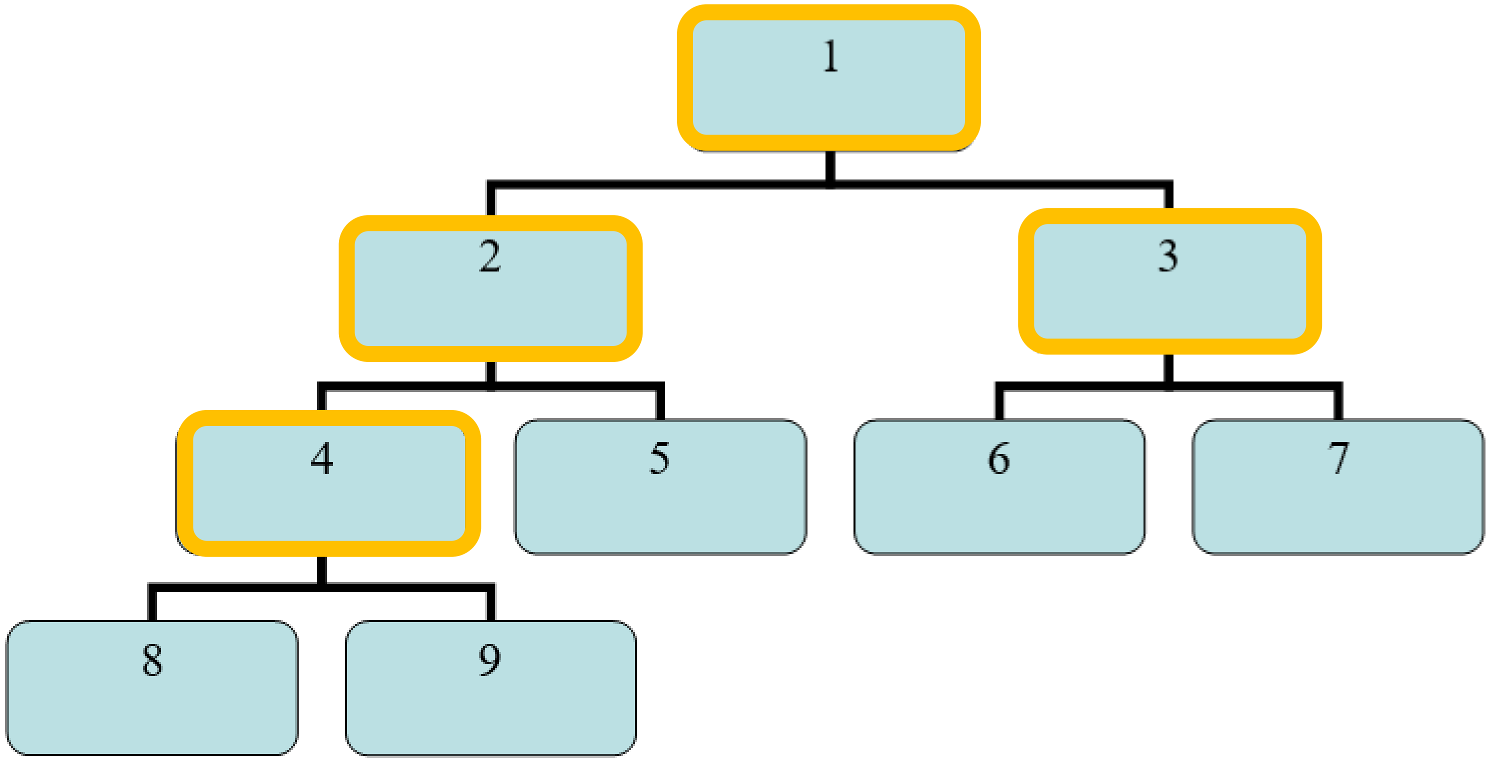
Node t still remains in the tree and becomes a terminal node.

Node Labelling Convention

- Especially important if you have a big tree.
- A big tree cannot be visualized with the details at each node.
- By standardizing the node labelling convention, given any node, we know:
  - Who is the left child node (if any)
  - Who is the right child node (if any)
  - Who is the parent node (if any)
- i.e. no need to see the big tree to trace the ancestry.



Where to prune?



- How to determine the “best” node to prune away?
- Ans: Weakest link pruning.
- How to determine which link is weakest?

Total Cost, Adjusted for Model Complexity

Define $\tilde{T}$ to be the set of all **terminal nodes** in the tree T.

Using r(t) and p(t), where p(t) is the proportion of all cases in terminal node t, define the contribution of terminal node t to the total misclassification error as

$$R(t) = r(t) \times p(t) \qquad\qquad t \in \tilde{T}$$

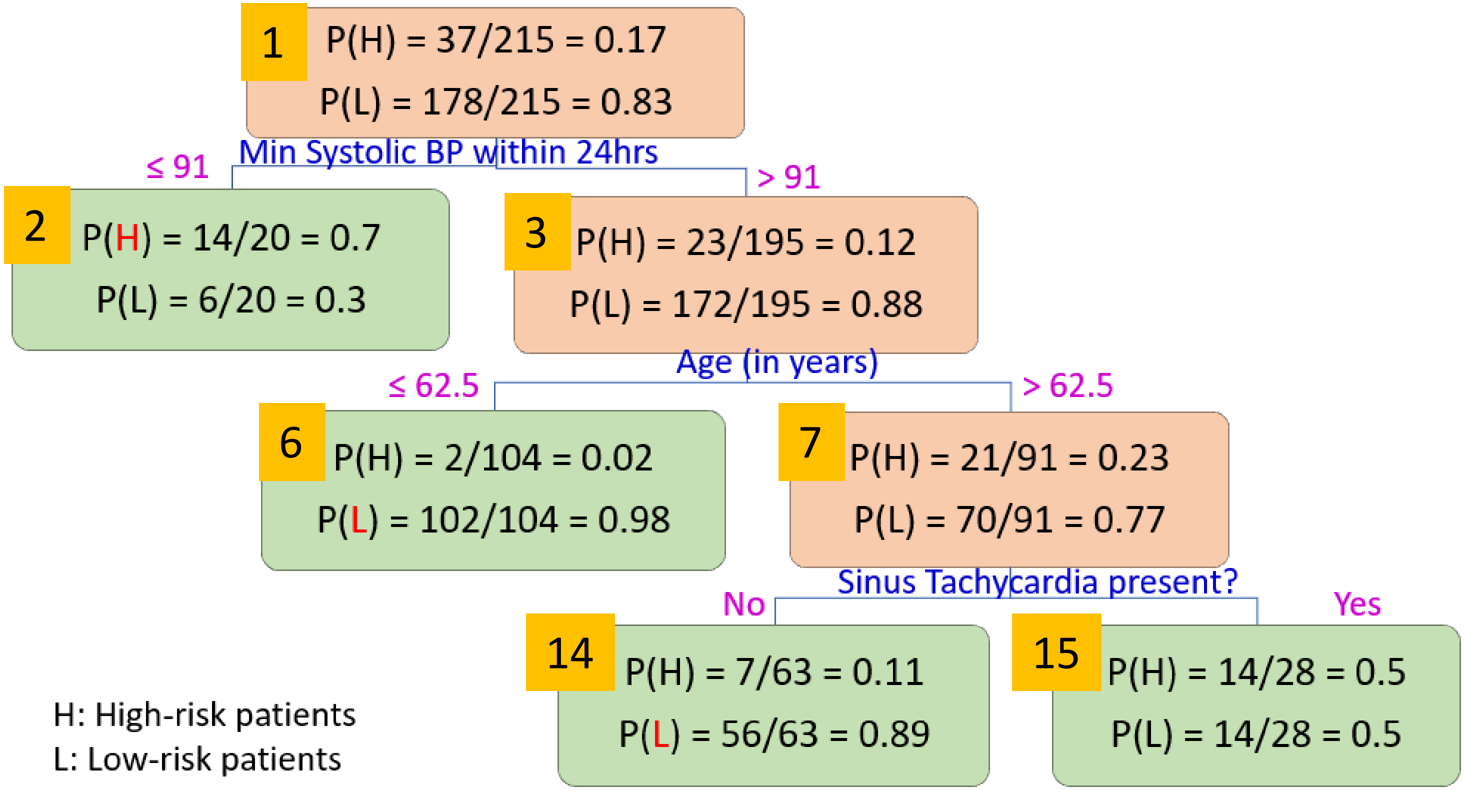
Overall Misclassification Error of the Tree T:

$$R(T) = \sum_{t \in \tilde{T}} R(t)$$

Complexity Adjusted Total Cost of the Tree T:

$$R_\alpha(T) = R(T) + \alpha\,|T|$$

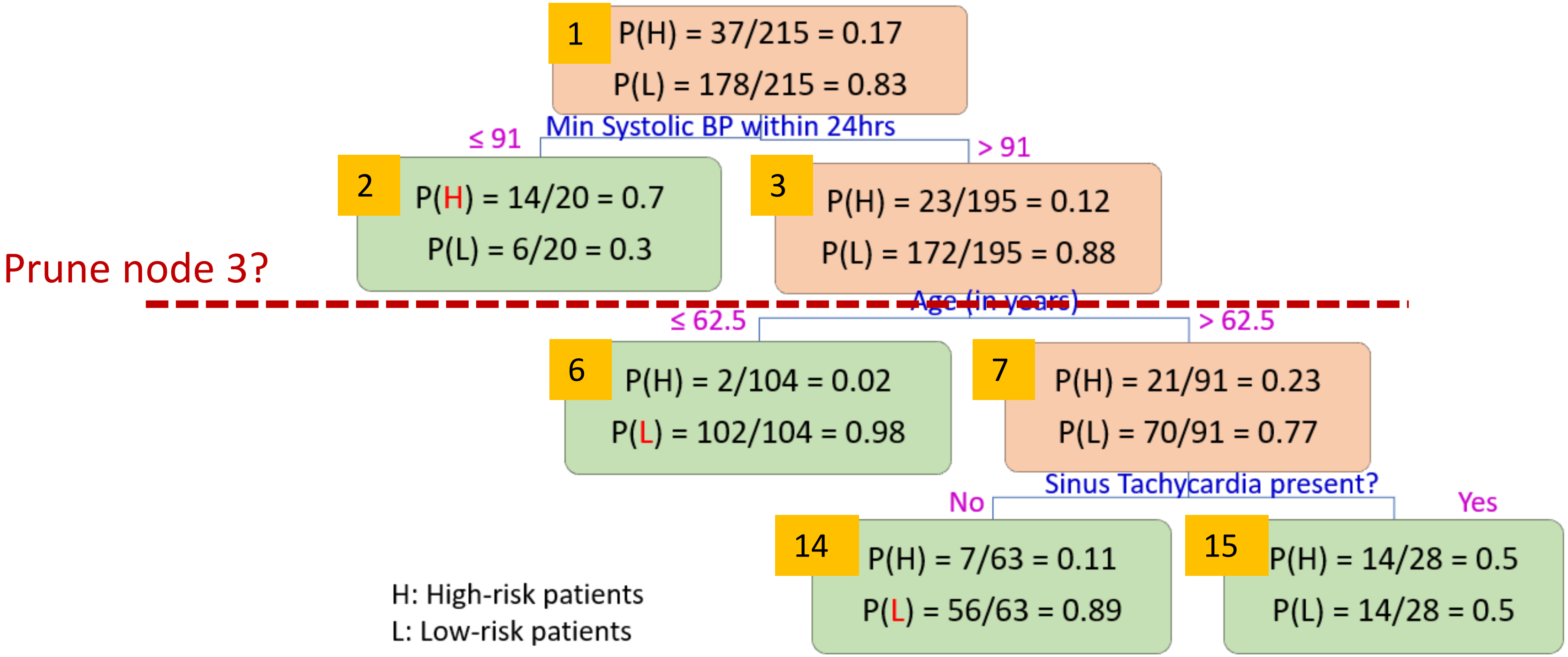
Where $|T|$ represents the **size of the tree** e.g. number of terminal nodes, and α represents the **penalty cost per unit complexity**. This is a costing mechanism to penalize complexity. Large α result in small tree.



- $\tilde{T} = \{2, 6, 14, 15\}$
- r(2) = 0.3, p(2) = 20/215, R(2) = 0.3*20/215 ≈ 0.027907
- r(6) = 0.02, p(6) = 104/215, R(6) = 0.02*104/125 ≈ 0.009674
- r(14) = 0.11, p(14) = 63/215, R(14) = 0.11*63/215 ≈ 0.03223
- r(15) = 0.5, p(15) = 28/215, R(15) = 0.5*28/215 ≈ 0.065116
- R(T) ≈ 0.1349
- $R_\alpha(T) = 0.1349 + 4\alpha$

Pruning triggered at certain values of alpha

- Only when alpha is big enough will the weakest link be pruned away.
- Chicken Rice Analogy:
  - Current cost in food centre: \$3 per pack. Consume 10 packs per month.
  - What if cost increase to \$3.10 per pack?
  - What if cost increase to \$5 per pack?
  - What if cost increase to \$5.20 per pack?
  - What if cost increase to \$20 per pack?
- CART:
  - Current cost: $\alpha = 0$ units per terminal node. i.e. no penalty for model complexity.
  - What if $\alpha = 0.1$?
  - What if $\alpha = 0.3$?
- As alpha increases, pruning is triggered only at certain few values of alpha, until only the root node is left.



- Is α = 0.0001 sufficient to prune node 3?
- Without pruning node 3: $R_\alpha(T) = 0.1349 + 4\alpha \approx 0.1353$
- If prune at node 3, node 3 becomes terminal node, and r(3) = 0.12, p(3) = 195/215, R(3) ≈ 0.1088, R(T) = R(2) + R(3) ≈ 0.1367, $R_\alpha(T) = 0.1367 + 2\alpha = 0.1369$
- Do not prune node 3 since total cost is lower without pruning.

- Is α = 0.02 sufficient to prune node 3?
- Without pruning node 3: $R_\alpha(T) = 0.1349 + 4\alpha \approx 0.2149$
- If prune at node 3, node 3 becomes terminal node, $R_\alpha(T) = 0.1367 + 2\alpha = 0.1767$
- Prune node 3 since total cost is lower with pruning.
- i.e. we compare the consequences of the two scenarios.

What minimum value of alpha will trigger the pruning?

- Compare the two equations from the two scenarios and solve for alpha.
- Total cost without pruning at node 3 = Total cost if prune at node 3

$$0.1349 + 4\alpha = 0.1367 + 2\alpha$$

$$2\alpha = 0.0018$$

$$\alpha = 0.0009$$

- α = 0.0009 is the minimum value of α required to prune at node 3. i.e. the prune trigger. [Note that this value is only an approximation as we used only 4 decimal places in the first equation.]
- Any value of alpha above the minimum value will also prune away node 3.
- Repeat the same analysis at all other internal nodes (i.e. nodes 1, 7) to get their prune triggers.
- The weakest link will be the lowest value among all the prune triggers. E.g. at node X.
- Prune away node X. Recompute $R(T)$, $R_\alpha(T)$, and repeat the analysis at all internal nodes to get the second weakest link, and so on….
- The pruning sequence is thus specified and completely determined by the data.

Next: rpart demonstration on a dataset

- CART implemented in Rpackage rpart.
- Phrase 1: Grow Tree to max.
- Phrase 2: Prune Tree to min.
- Pruning sequence determines a sequence of sub trees.
- One of the sub-tree is the optimal tree for CART model prediction.
- How to use the CART model to make predictions.