

The “cp” parameter

- “cp” stands for “**complexity parameter**”
- Recall the first tree we made using LAT/LON had many splits, but we were able to trim it without losing much accuracy.
- Intuition: having too many splits is bad for generalization, so we should penalize the **complexity**

The “cp” parameter

- Define **RSS**, the **residual sum of squares**, the sum of the square differences
$$RSS = \sum_{i=1}^n (y_i - f(x_i))^2$$
- Our goal when building the tree is to minimize the RSS by making splits, but we want to penalize too many splits. Define **S** to be the number of splits, and λ (lambda) to be our penalty. Our goal is to find the tree that minimizes
$$\sum_{Leaves} (RSS \text{ at each leaf}) + \lambda S$$

The “cp” parameter

- λ (lambda) = 0.5

Splits	RSS	Total Penalty
0	5	5
1	$2 + 2 = 4$	$4 + 0.5*1 = 4.5$
2	$1+0.8+2 = 3.8$	$3.8 + 0.5*2 = 4.8$

The “cp” parameter

$$\sum_{Leaves} (\text{RSS at each leaf}) + \lambda S$$

- If pick a large value of λ , we won't make many splits because we pay a big price for every additional split that outweighs the decrease in “error”
- If we pick a small (or zero) value of λ , we'll make splits until it no longer decreases error.

The “cp” parameter

- The definition of “cp” is closely related to λ
- Consider a tree with no splits – we simply take the average of the data. Calculate RSS for that tree, let us call it **RSS(no splits)**

$$c_p = \frac{\lambda}{\text{RSS}(\text{no splits})}$$