Pruning Trees

Ways to prevent overfitting

Decision trees are easy to overfit

- Early Stopping
- Pruning
- Ensembling

Decision Tree Pruning

Issue: Sometimes a good split can happen later

Idea: Grow the tree fully then prune

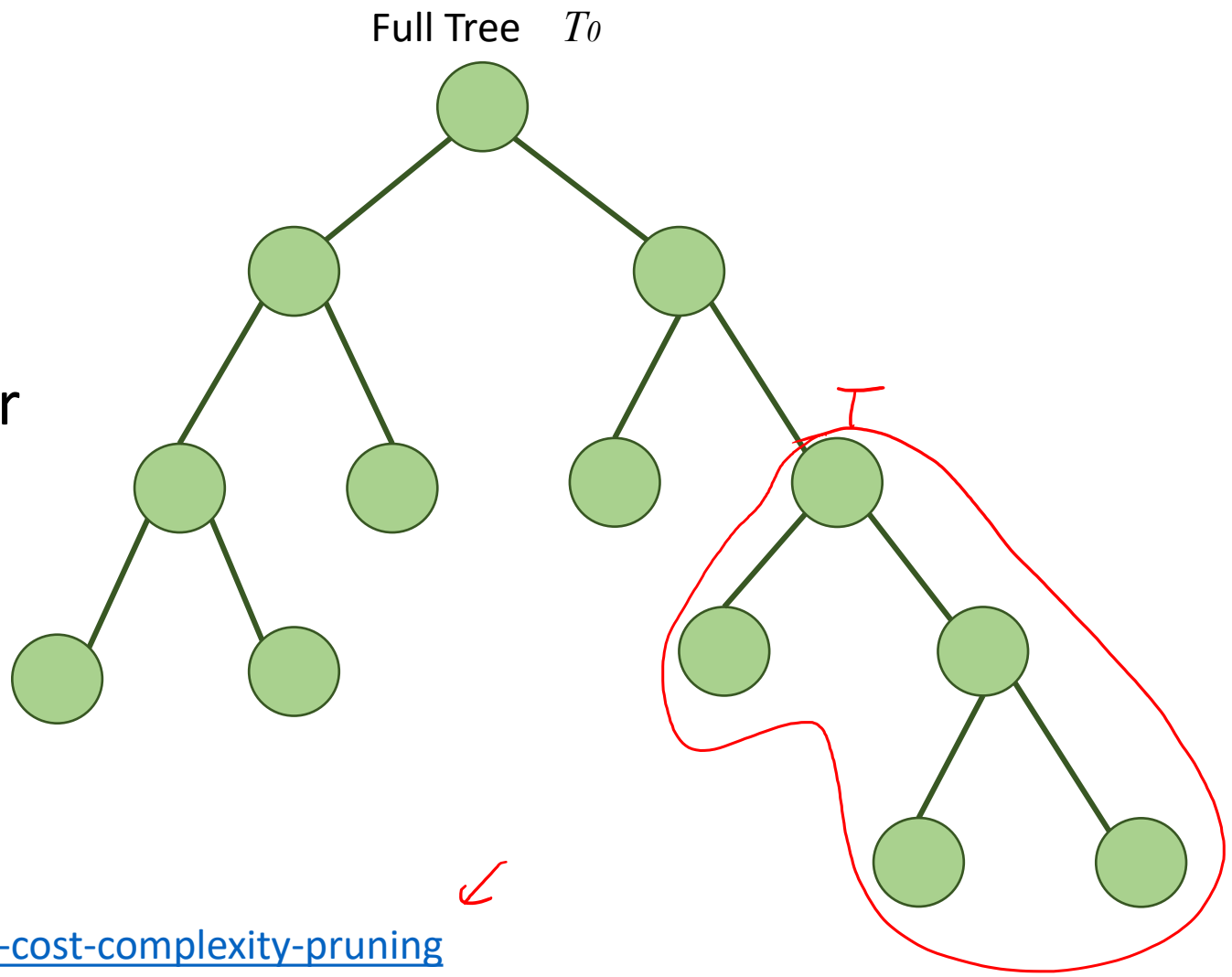
How: Minimal Cost-Complexity Pruning

Pruning option is available since sklearn 0.22

Minimal Cost-Complexity Pruning

$R(T)$

α : complexity parameter

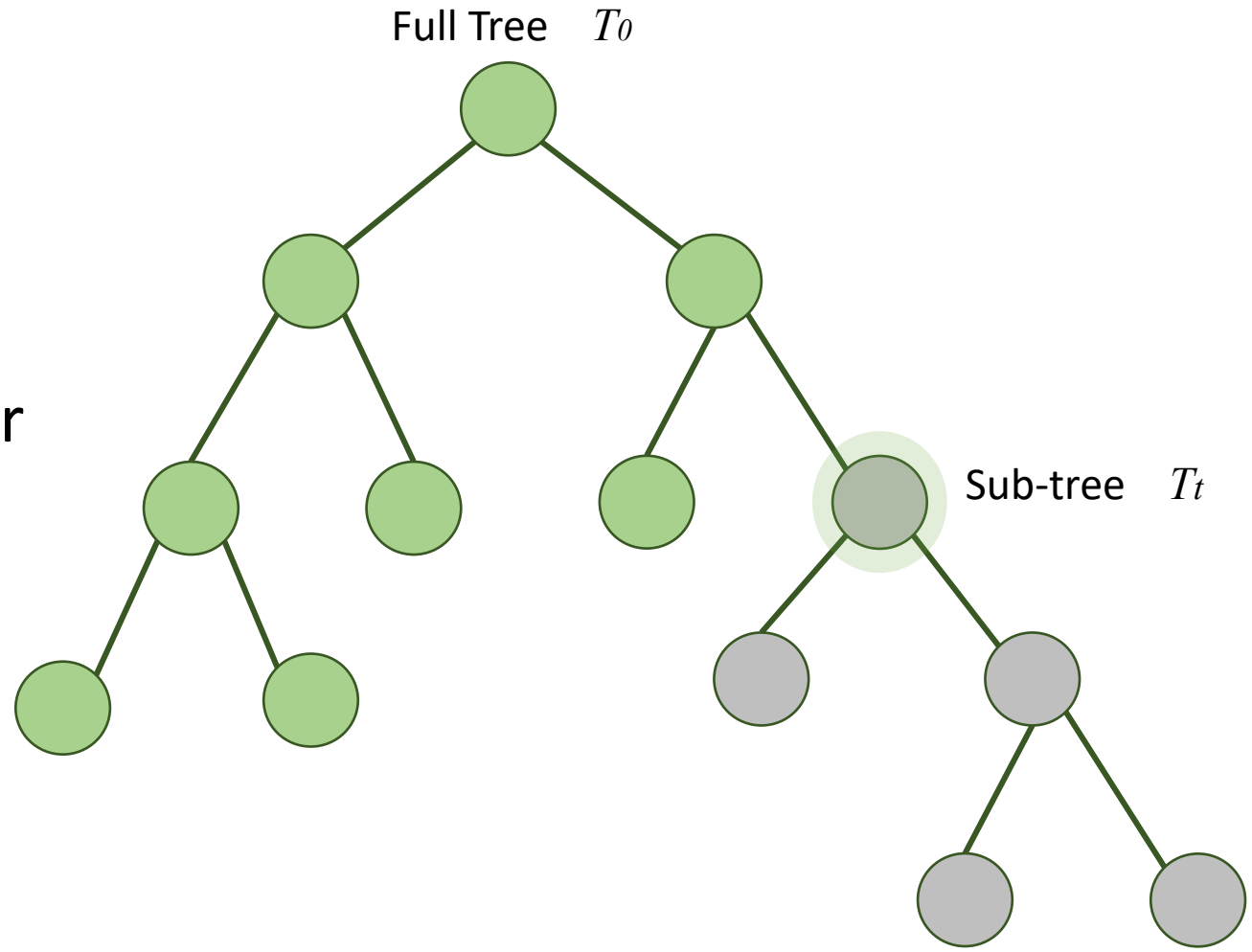


https://scikit-learn.org/stable/modules/tree.html#minimal-cost-complexity-pruning
https://scikit-learn.org/stable/auto_examples/tree/plot_cost_complexity_pruning.html

Minimal Cost-Complexity Pruning

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

α : complexity parameter



Minimal Cost-Complexity Pruning

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

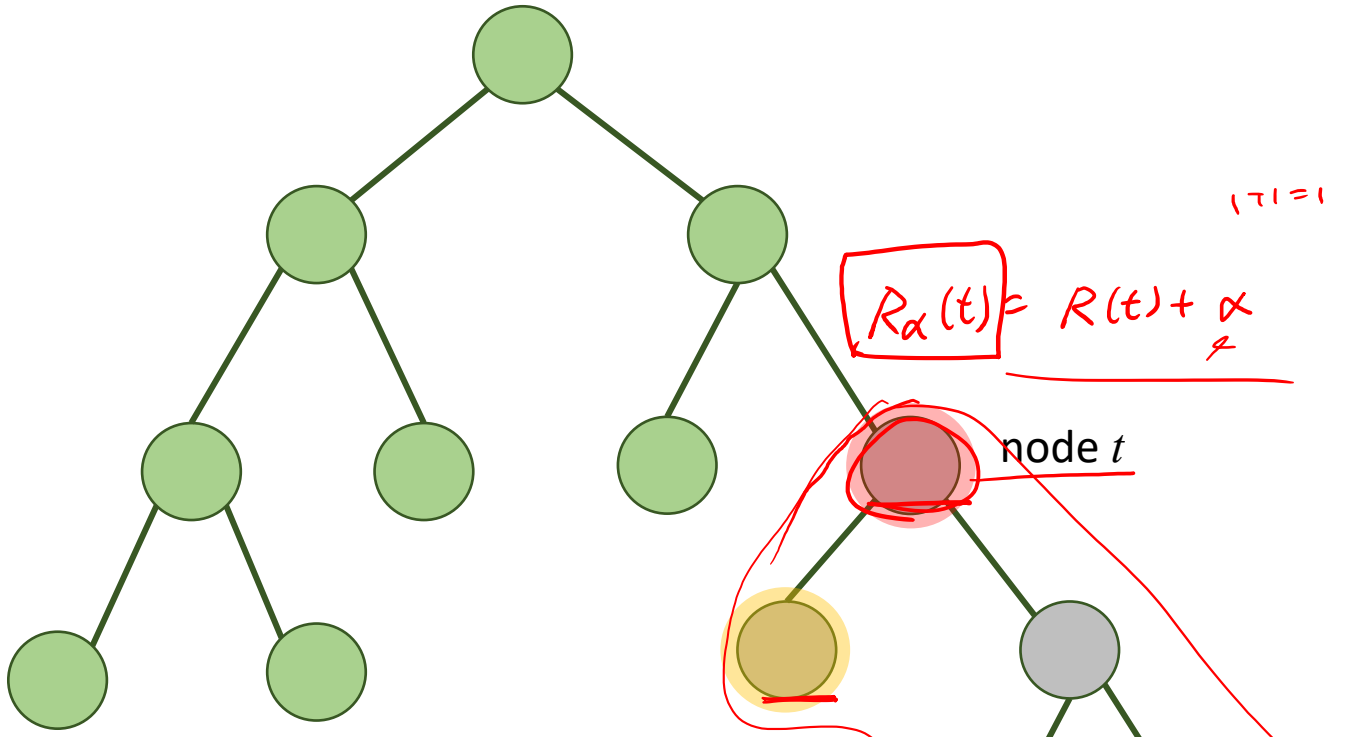
α : complexity parameter

|T|: number of leaf nodes of the subtree

Impurity at the node t

$$R(T_t) < R(t)$$

Sum of the impurities at the leaf nodes of the subtree Tt



$$R_\alpha(T_t) = R_\alpha(t)$$

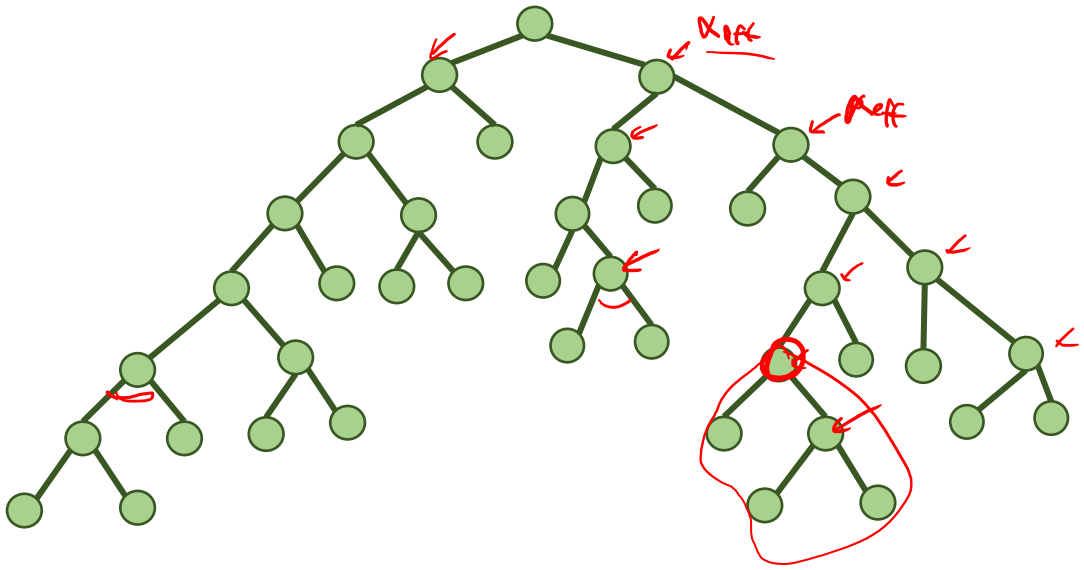
$$\alpha_{eff}(t) = \frac{R(t) - R(T_t)}{|T| - 1}$$

$R_\alpha(t) = R(t) + \alpha$

$R_\alpha(T_t)$

$= R(T_t) + \alpha \cdot |T|$

Minimal Cost-Complexity Pruning



Iteratively removes the weakest link

When does it stop pruning?

Stop when $\min(\alpha_{eff}) > \alpha_{ccp}$

α_{ccp} : cost complexity parameter, "ccp_alpha"

Decision Tree Pros and Cons

Trees are easy to understand

Trees don't suffer collinearity

Trees are good for non-linear features

Trees handle categorical variables easily

Trees are weak-learner

Trees have high variance in general

Trees can overfit easily

Linear regression is a better choice if features are linear

Tree's performance can be greatly improved when ensembled