Tree-based Models and Ensembles

Lecture 11

Supervised Learning Techniques

Covered so far

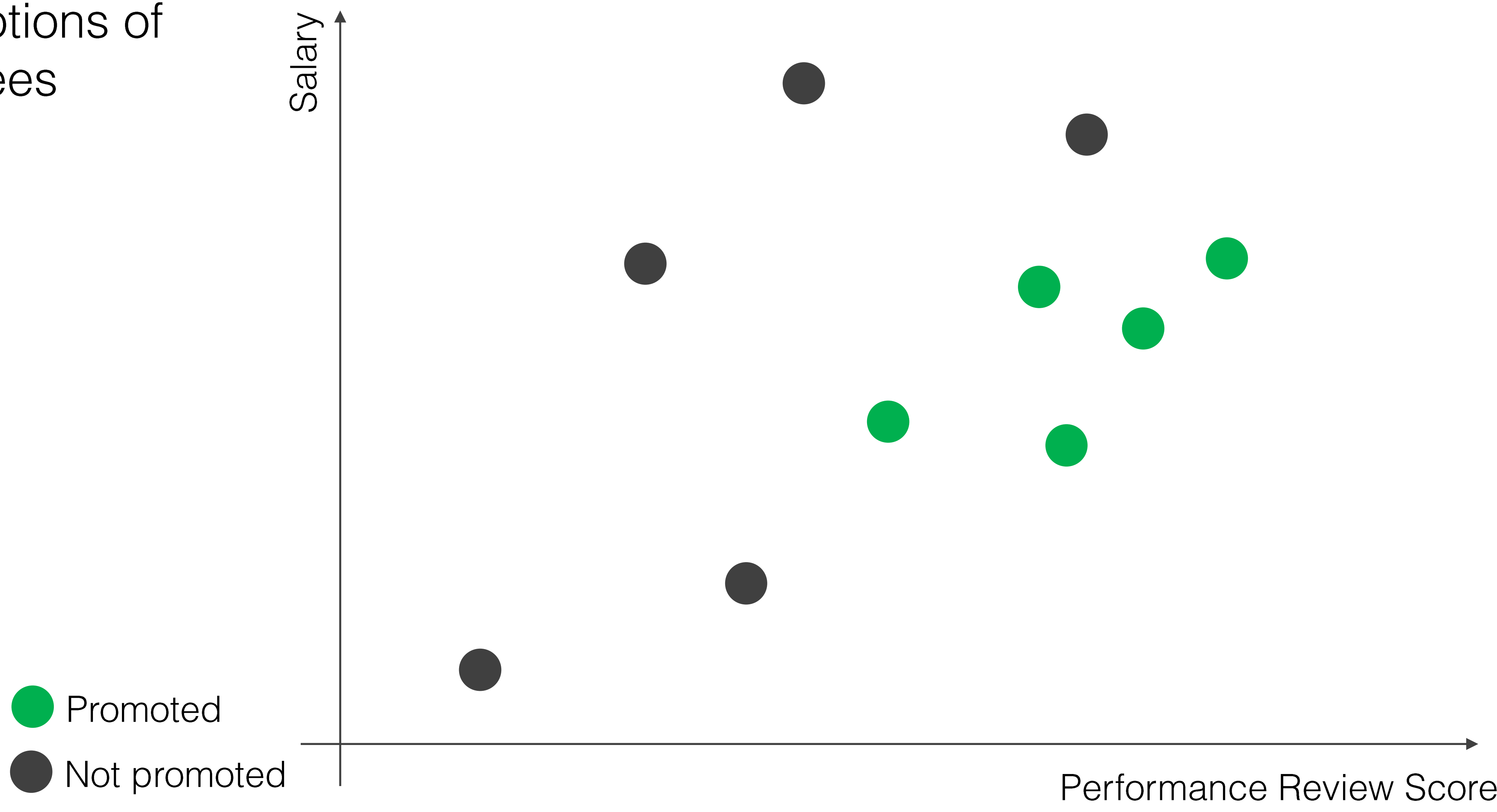
- Linear Regression
- K-Nearest Neighbors
- Perceptron
- Logistic Regression
- Fisher’s Linear Discriminant
- Linear Discriminant Analysis
- Quadratic Discriminant Analysis
- Naïve Bayes

- Decision Trees
- Ensemble methods (bagging and boosting)

Classification and Regression Trees (CART)

Classification trees = decision trees

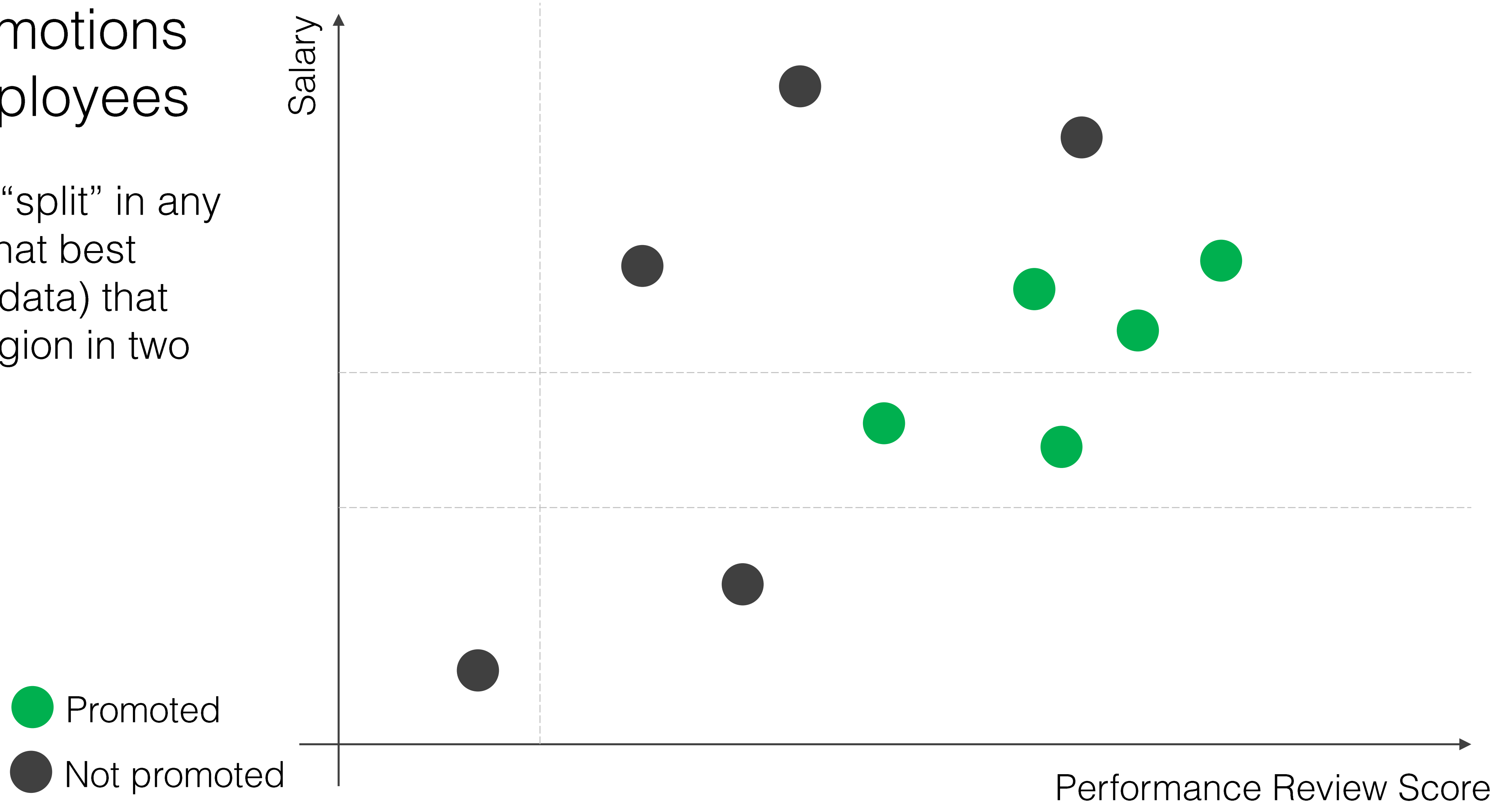
Predicting promotions of salaried employees



Classification and Regression Trees (CART)

Predicting promotions of salaried employees

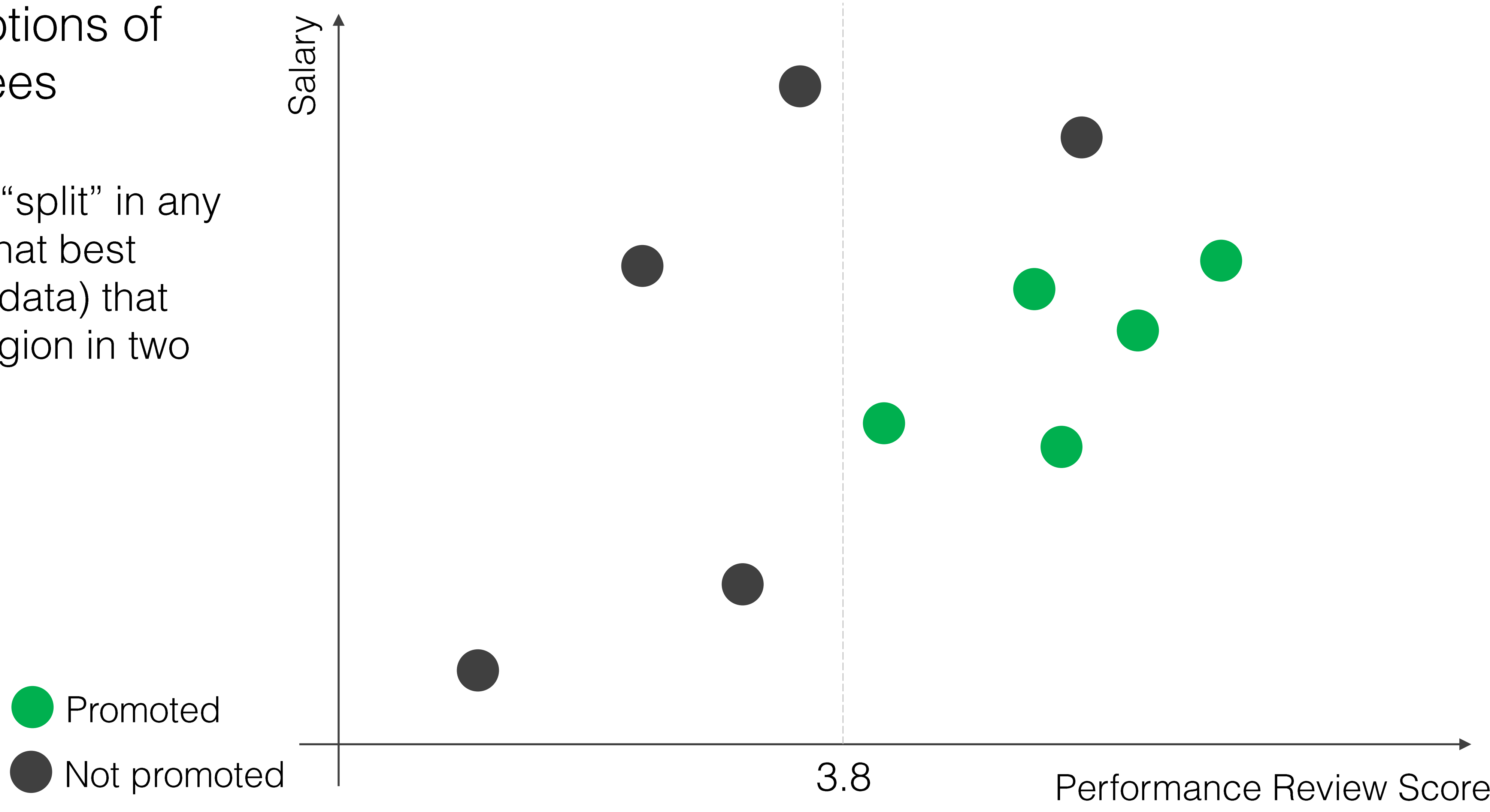
1 Find the best "split" in any one feature (that best classifies the data) that divides the region in two



Classification and Regression Trees (CART)

Predicting promotions of salaried employees

1 Find the best "split" in any one feature (that best classifies the data) that divides the region in two



Classification and Regression Trees (CART)

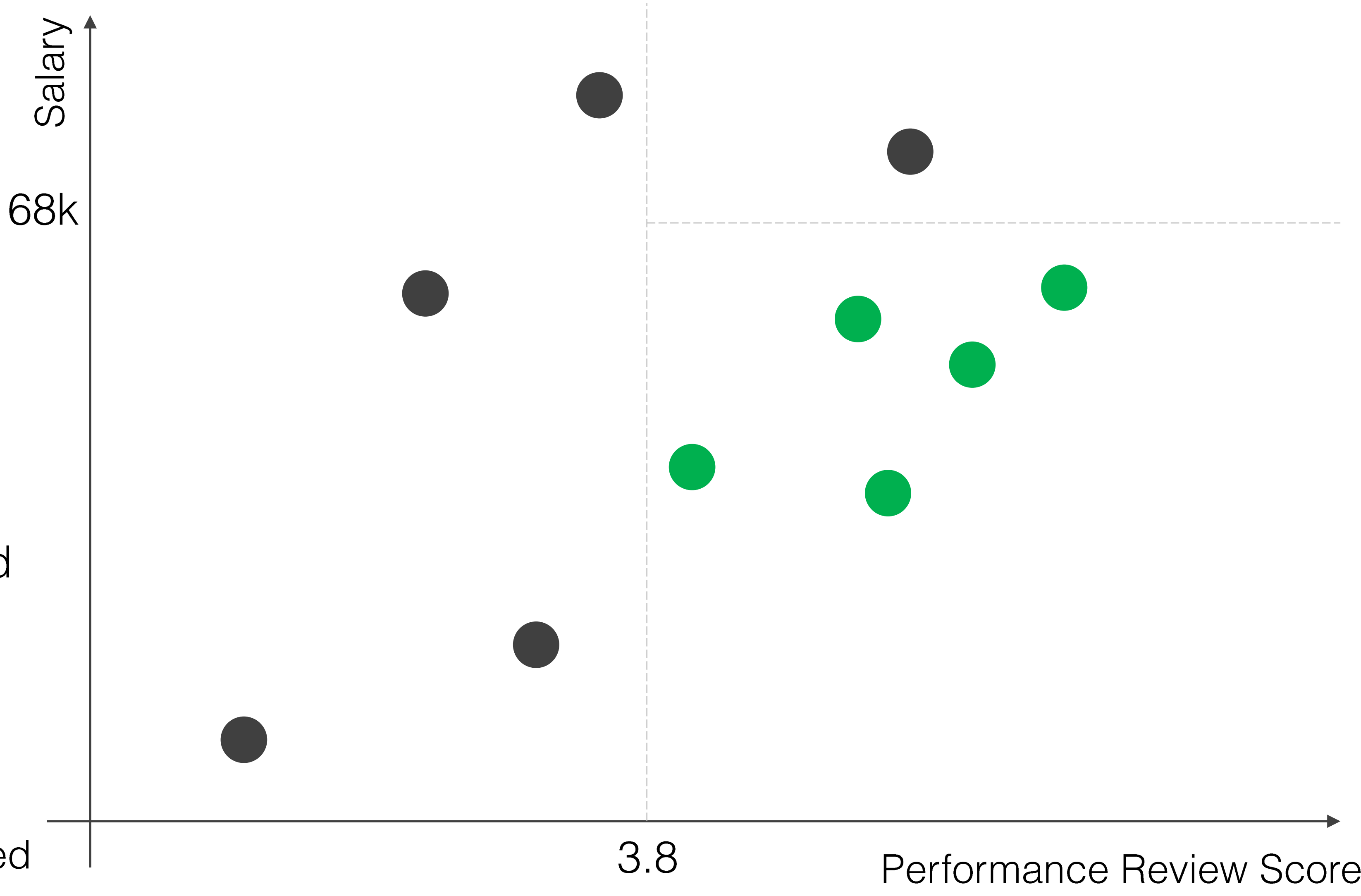
Predicting promotions of salaried employees

1. Find the best "split" in any one feature (that best classifies the data) that divides the region in two
2. Continue splitting regions (1 feature at a time) until a stopping criterion is reached (e.g. there are at most N samples in any region

Greedy, recursive binary tree

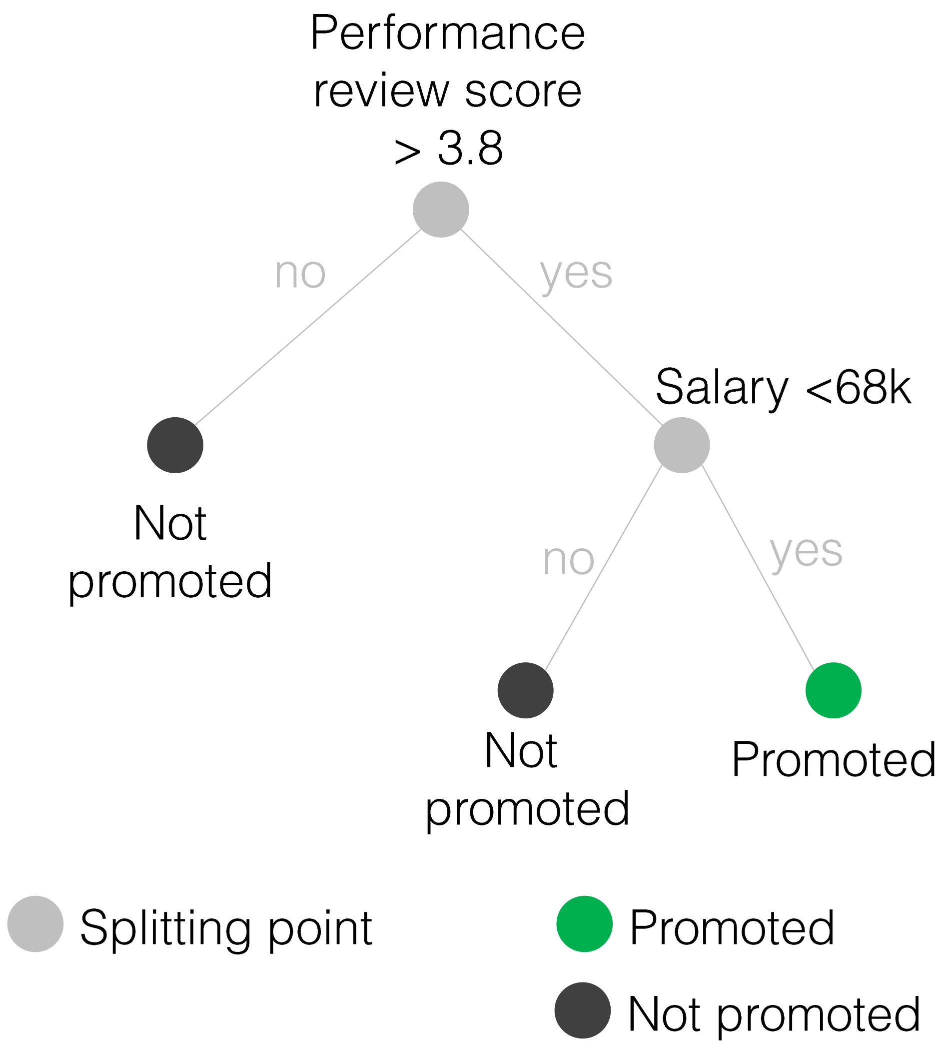
Promoted

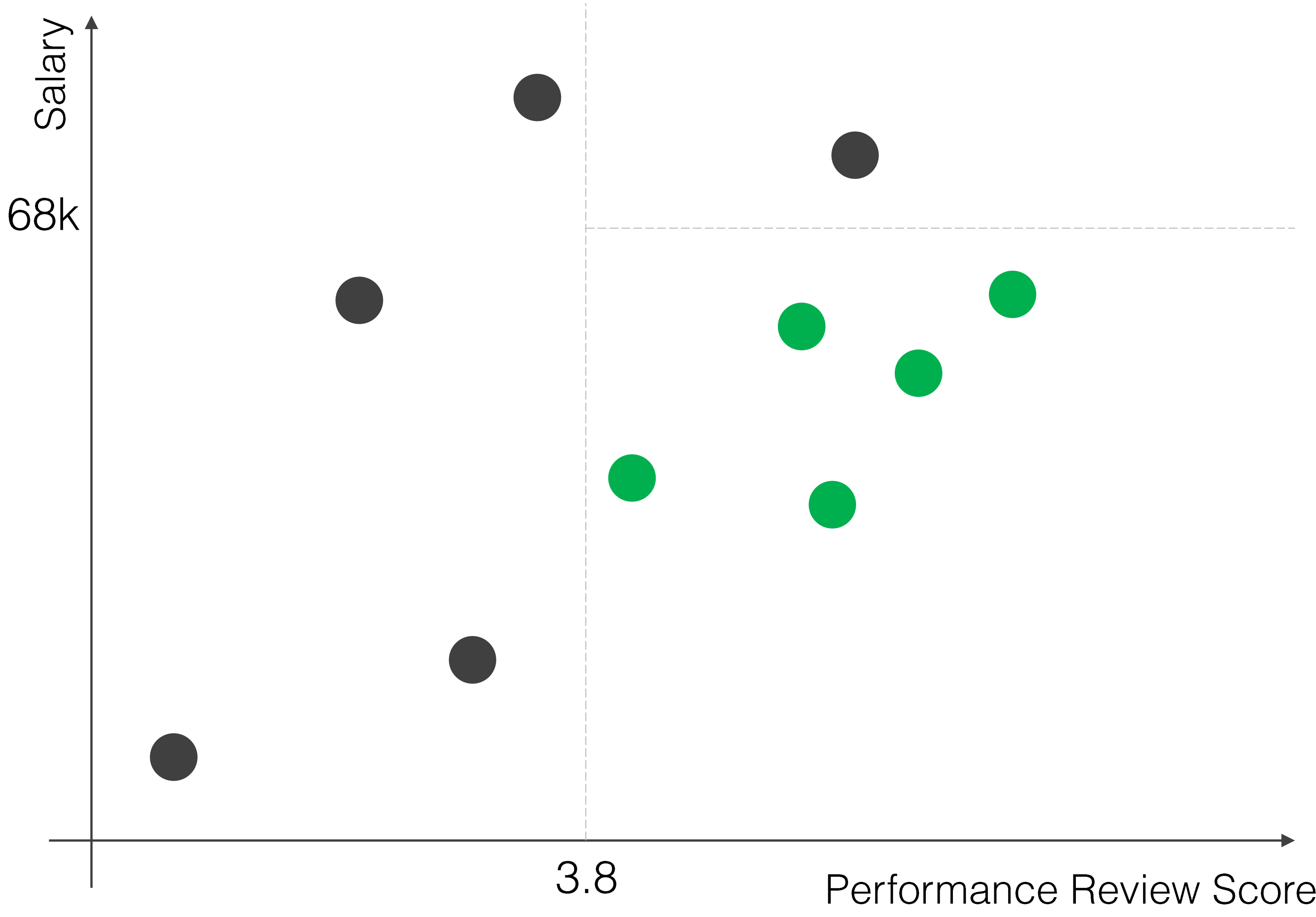
Not promoted



Classification and Regression Trees (CART)

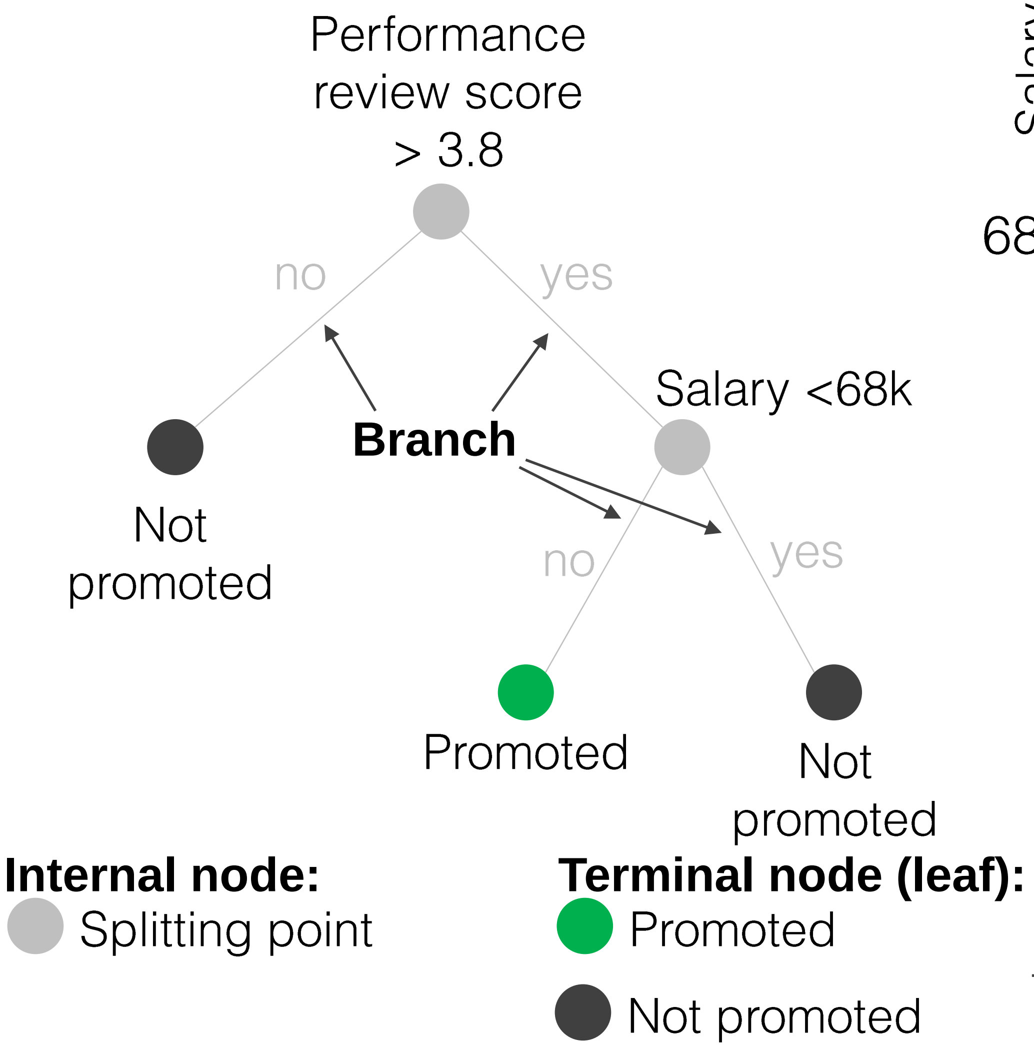
Tree representation:

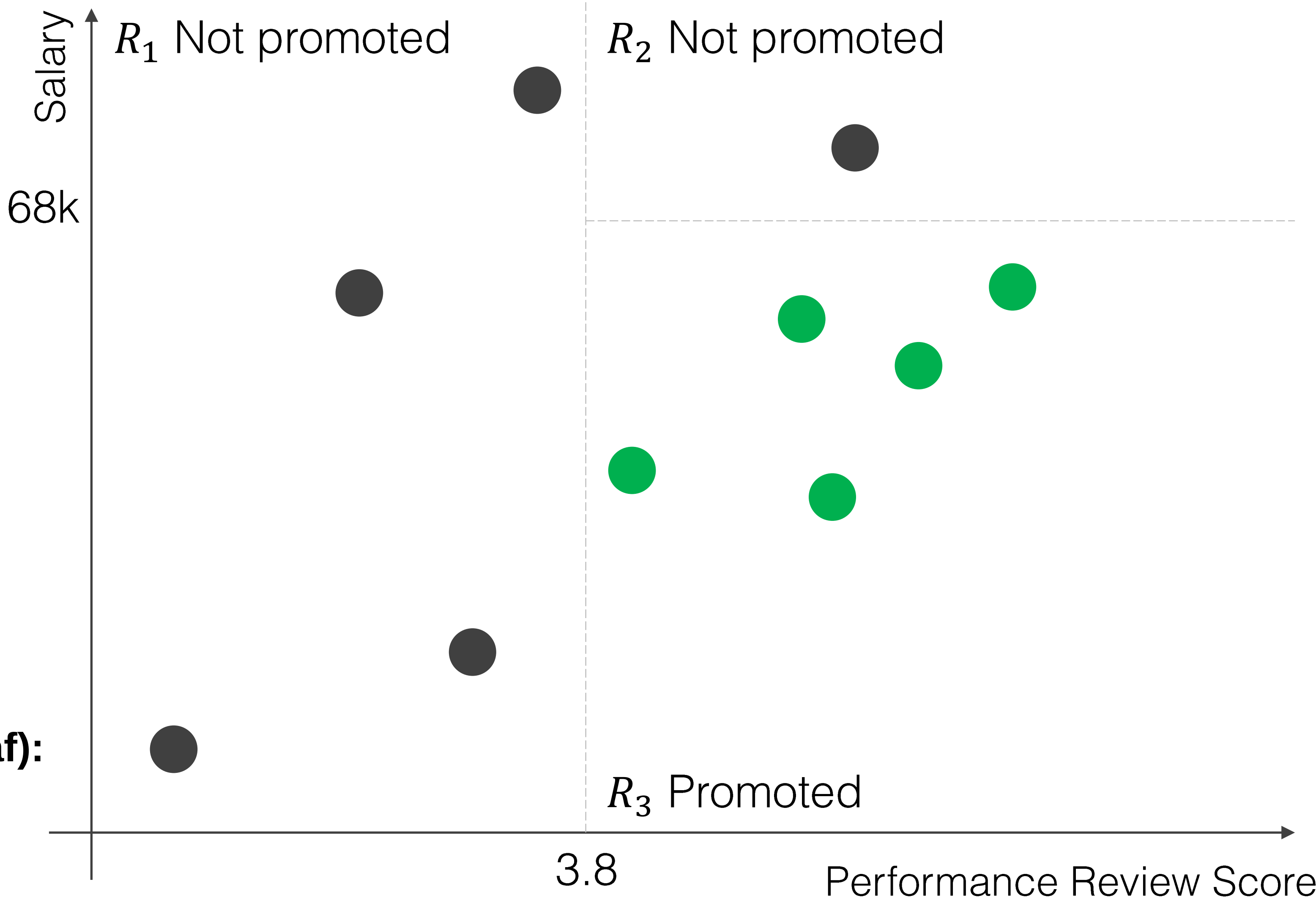




Classification and Regression Trees (CART)

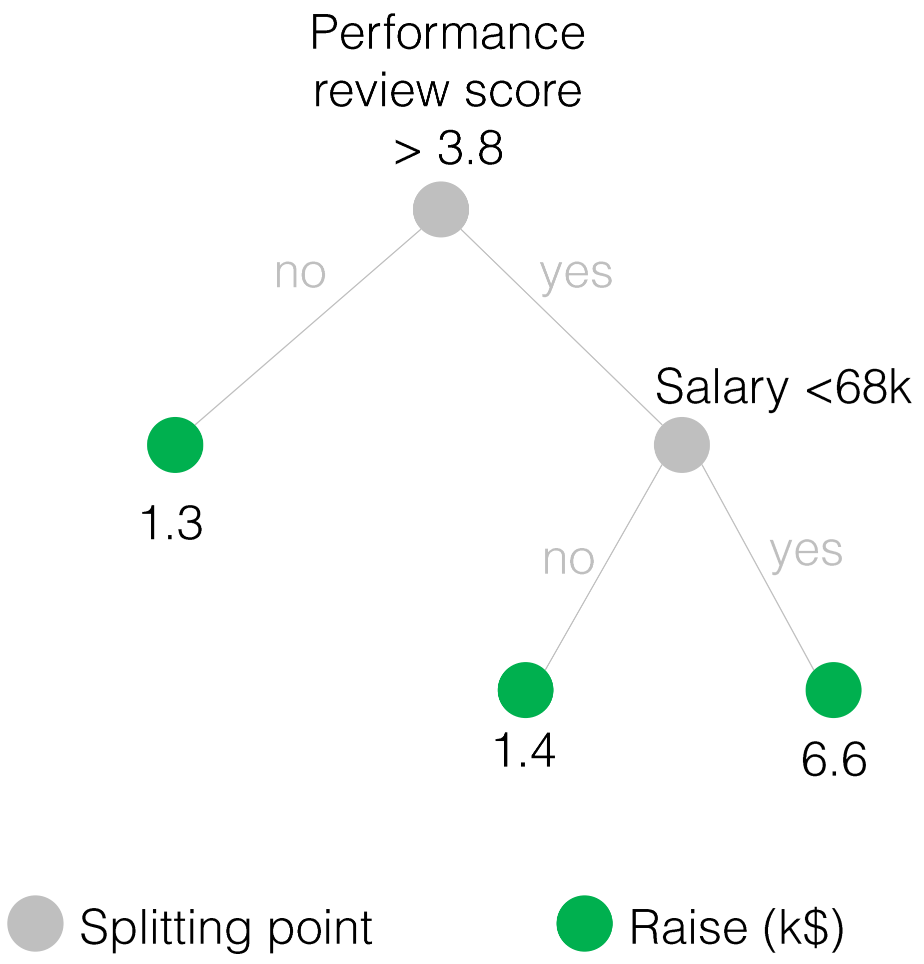
Tree representation:

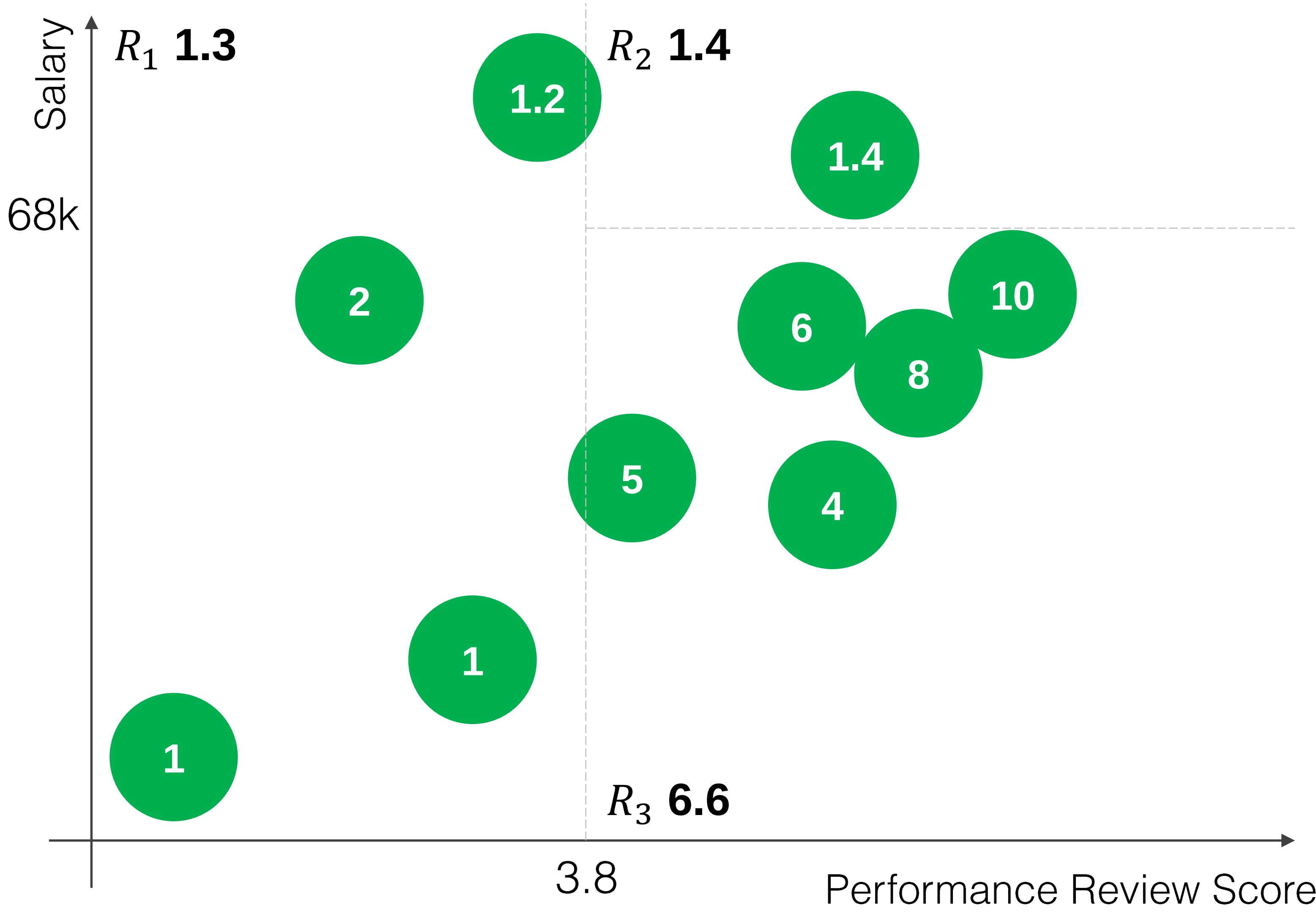




The Regression Setting

In this case, each region is represented by an average of the values it contains





How do we determine which split to make?

Pick the split that reduces the error/cost criterion most after the split

Splitting criterion

$$C = \sum_{r=1}^{R_{tot}} Q(r)$$

Regression

Mean square error

$$Q_{MSE}(r) = \sum_{i \in R_r} \left(y_i - \hat{y}_{R_r}\right)^2$$

y_i = training data response i

$\hat{y}_{R_r}$ = mean value in region r,

(where R_{tot} is the total # of regions)

Classification

Misclassification rate

$$Q_{Misclass} = 1 - \max_k \left(\hat{p}_{rk}\right)$$

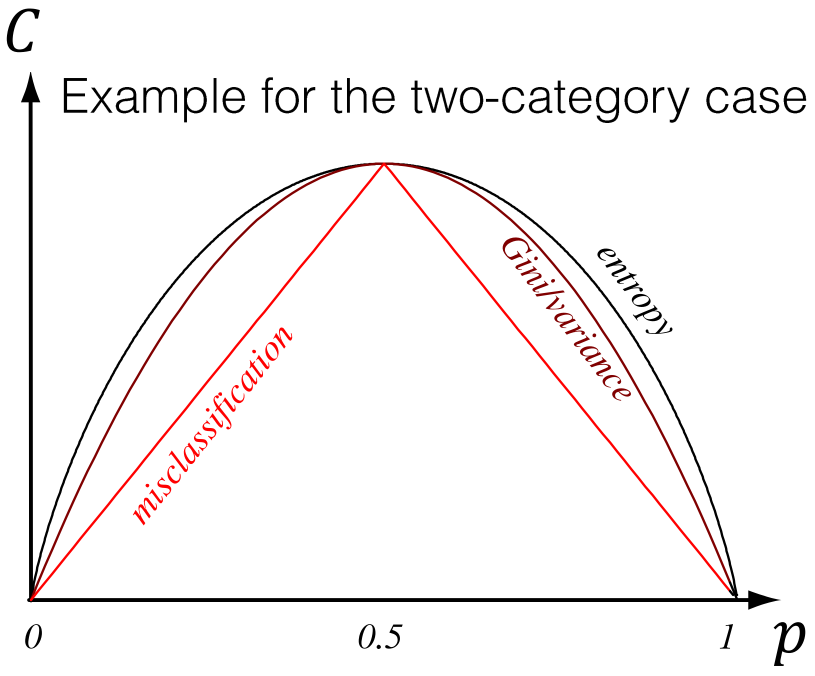
Gini impurity

$$Q_{Gini} = \sum_{k=1}^{K} \hat{p}_{rk}(1 - \hat{p}_{rk})$$

Cross-entropy

$$Q_{entropy} = -\sum_{k=1}^{K} \hat{p}_{rk} \log \hat{p}_{rk}$$

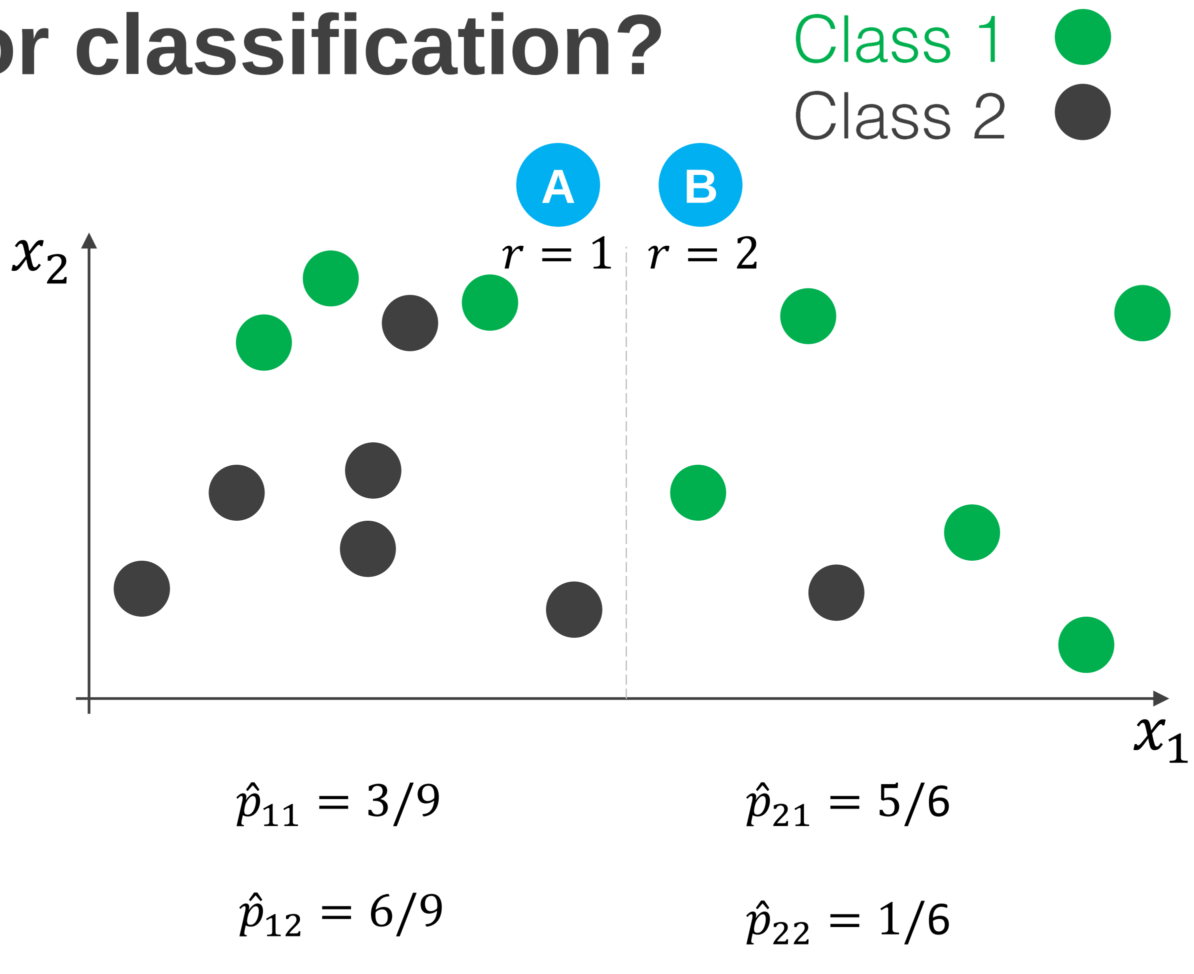
$\hat{p}_{rk}$ = proportion of training observations in the r^{th} region from the k^{th} class



Duda, Hart, and Stork., Pattern Classification

How to measure quality of split for classification?

$\hat{p}_{rk}$ = proportion of training observations in the r^{th} region from the k^{th} class



For each region:

	A	B
Misclassification rate $Q_{Misclass} = 1 - \max_k (\hat{p}_{rk})$	0.333	0.167
Gini impurity $Q_{Gini} = \sum_{k=1}^{K} \hat{p}_{rk}(1 - \hat{p}_{rk})$	0.444	0.278
Cross-entropy $Q_{entropy} = -\sum_{k=1}^{K} \hat{p}_{rk} \log \hat{p}_{rk}$	0.912	0.650

Tree Pruning

Trees have the tendency to overfit the data

Consider the stopping rule: stop splitting once there is only 1 observation in each region (leads to complete overfit)

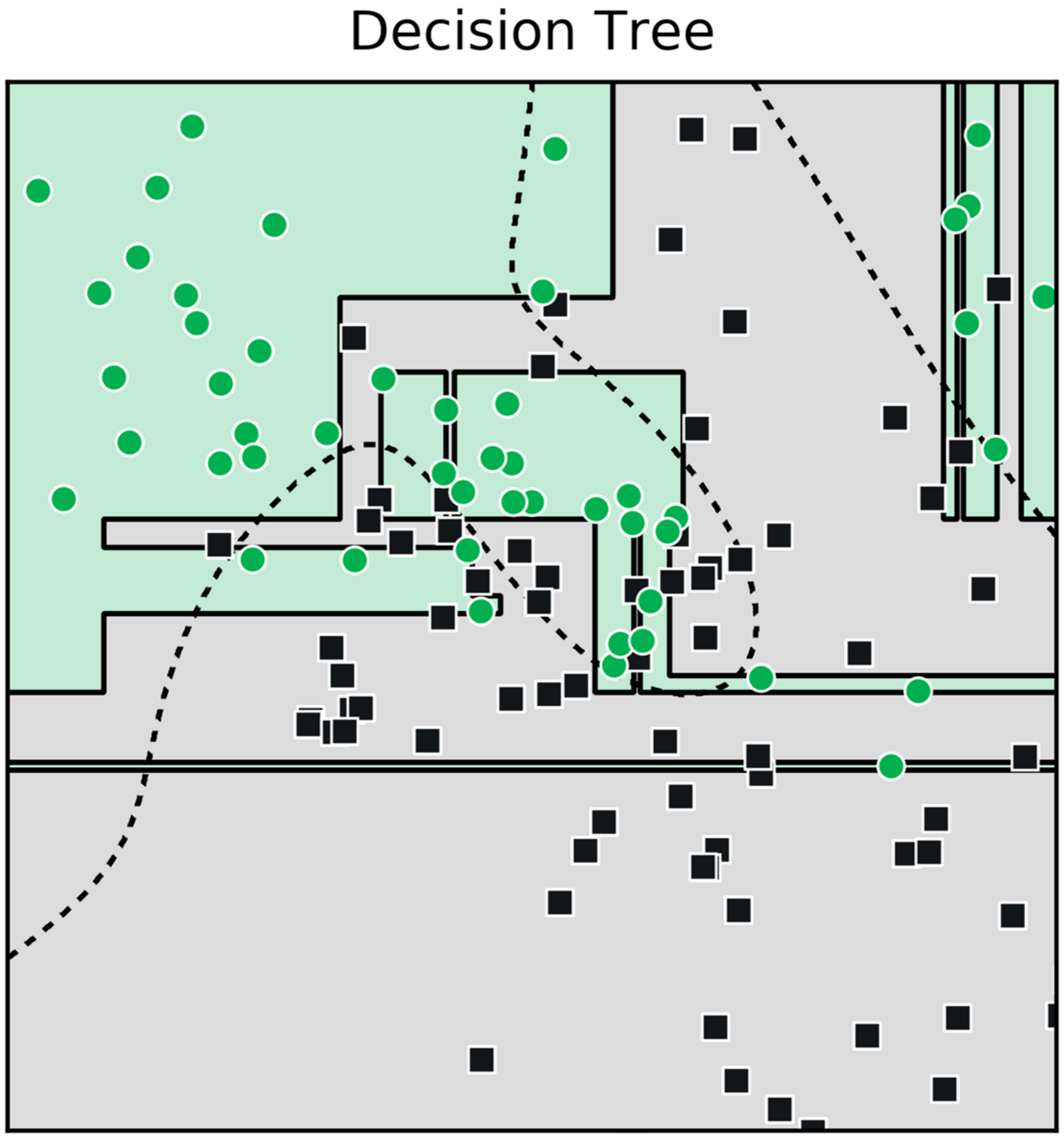
Pruning the tree back reduces this overfit (removing splits after the tree is formed)

Pruning can be optimized through a penalty on the number of terminal nodes:

$$C_{Prune} = \sum_{j=1}^{T} \sum_{i \in R_j} \left(y_i - \hat{y}_{R_j}\right)^2 + \alpha T$$

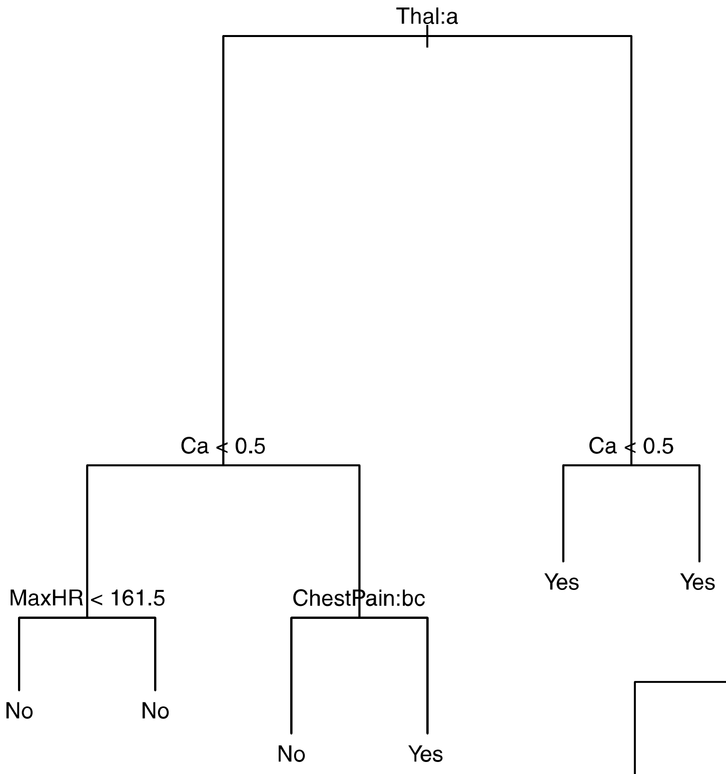
penalty on number of terminal nodes

number of terminal nodes

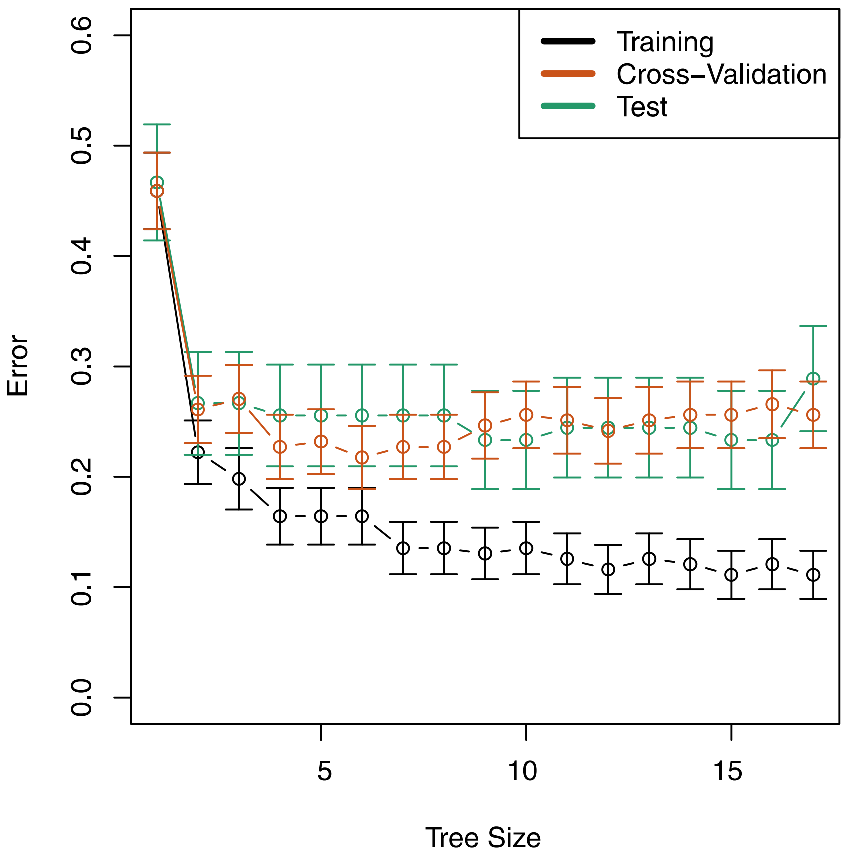


Pruning example

Pruned Tree

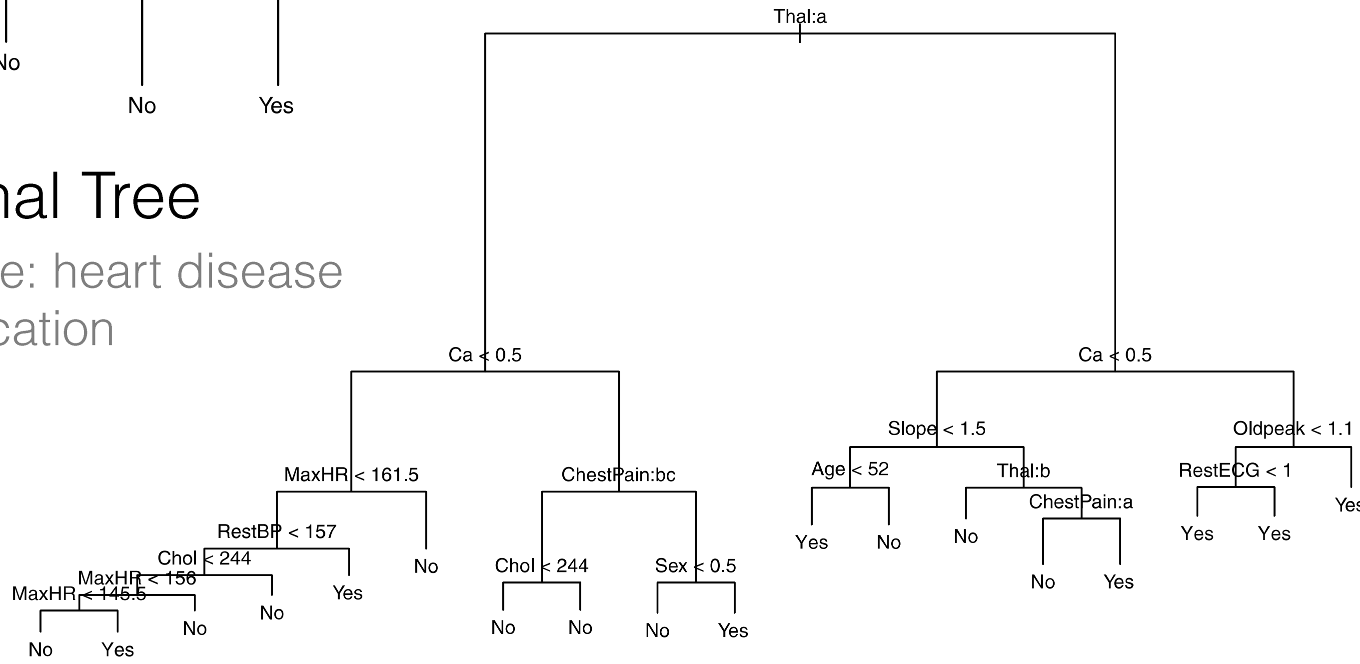


Performance



Original Tree

Example: heart disease classification



James et al., An Introduction to Statistical Learning