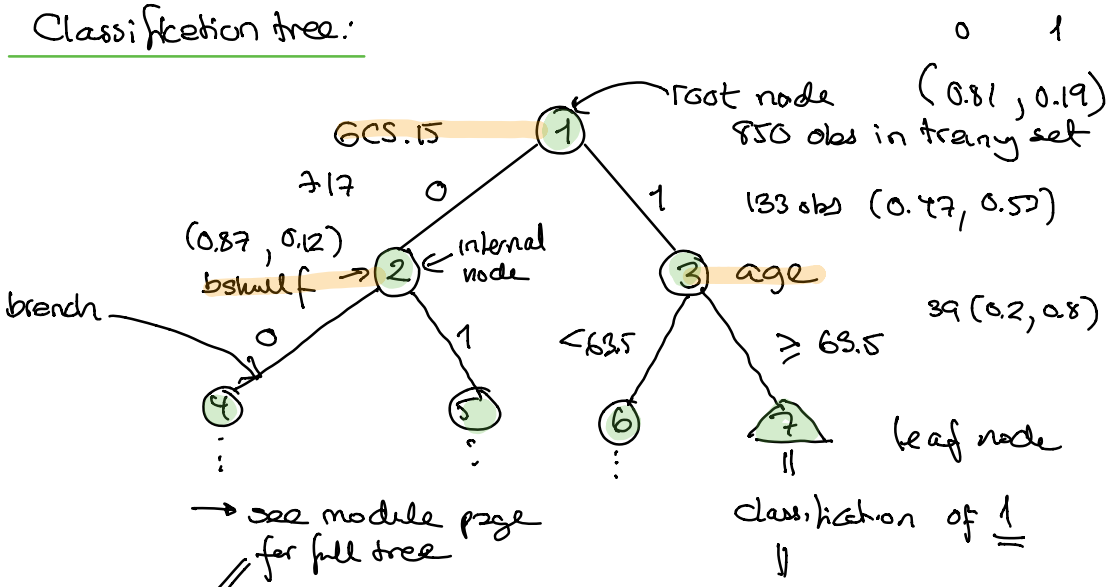
M8L1: Trees (and forests) THA4268, 04.03.2019

Main idea: derive a set of rules (binary splits) for segmenting the predictor space into a number of non-overlapping regions.

Classification tree:



Here we had $J = 11$ leaf nodes

Trees are easy to interpret and visualize.

describes the region in predictor space where

GCS.15 = 1 and age ≥ 63.5

$R_j, \hat{y}_{R_j} = 1$

Constructing a regression tree

(x_i, y_i) $\quad i = 1, \dots, n$ $\quad Y = f(x) + \epsilon$

x_i: p-dim; y_i: univariate, continuous

1) Divide the predictor space into J non-overlapping regions $R_1, \dots, R_J$.

2) Prediction in R_j is $\hat{y}_{R_j}$ = mean of the train obs. that fall into R_j

How to decide on $R_1, \dots, R_J$:

$$RSS = \sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2 \quad \leftarrow \text{minimize this?}$$

Greedy approach → recursive binary splitting

At the top node j split into

$$R_1(j,s) = \{x \mid x_j < s\} \text{ and } R_2(j,s) = \{x \mid x_j \geq s\}$$

by choosing j and s to minimize

$$\sum_{i: x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i: x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2$$

$\hat{y}_{R_1}$: mean of train samples in region R_1

$\hat{y}_{R_2}$

⇒ our tree has one split with two branches

repeat

Classification tree → see module pages

1) Prediction
2) Splitting criterion: Gini, cross entropy

BeeringPoint: NEW: gain curve { correct e given percentage of custom (top 2%?) → in some ranked order

Why ROC curve: because most of set cut-off like top 2% on prob. driven

Spend 80-98% of time on data cleaning ≈ data wrangling
data science:

data warehouse knowledge is important