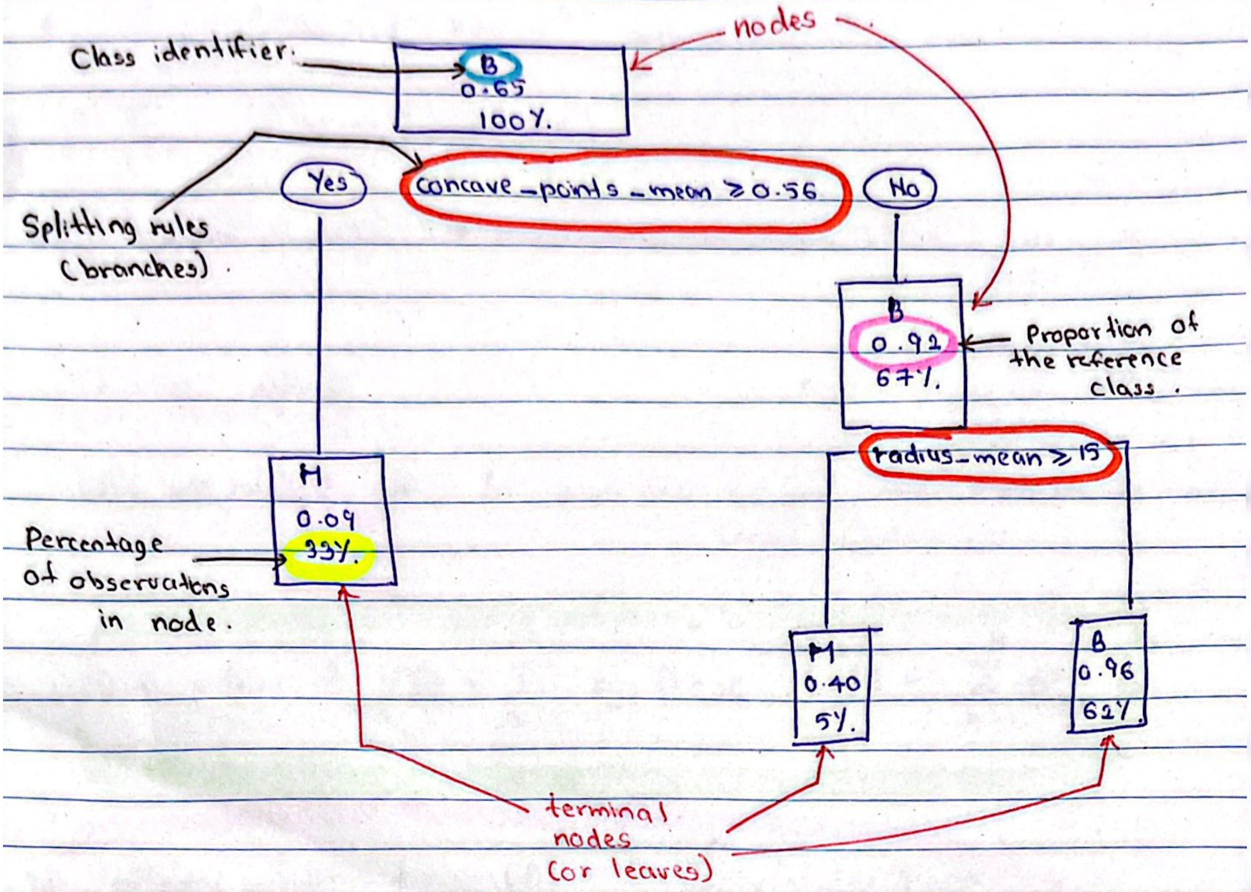
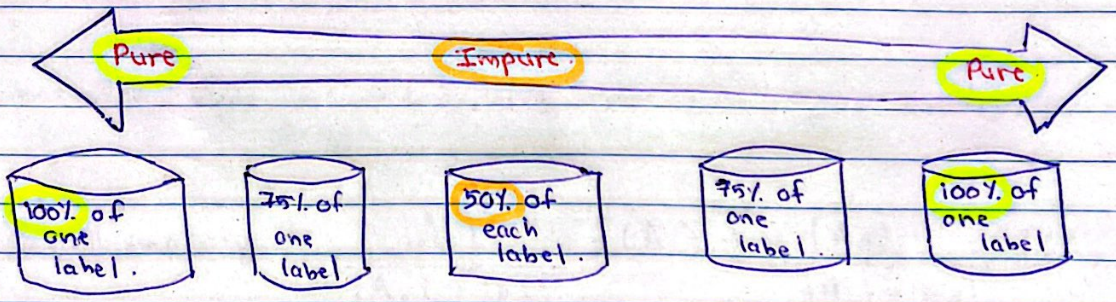
Classification tree

* A recursive two-way partition (or branches) of predictors.



Class identifier.
nodes
B
0.65
100%.
Yes
concave_points_mean ≥ 0.56
No
Splitting rules
(branches).
B
0.92
67%.
Proportion of the reference class.
radius_mean ≥ 15
M
0.09
33%.
Percentage of observations in node.
M
0.40
5%.
B
0.96
62%
terminal nodes (or leaves)

⟹ Colors indicate class, with a darker color indicating lower impurity.



Notations

* s = the value to split in variable j.

* $A_L = \{\{y_i, x_i\} : x_{ij} < s\}$ and

$A_R = \{\{y_i, x_i\} : x_{ij} \geqslant s\}$.

* p_{kA} = the proportion of observations in class k in a set A.

Impurity metrics

Selected threshold.

selected predictor

Why we need? ⇒ An algorithm to find the optimal $[j^*, s^*]$ (split and variable).

Gini index: $f_{Gini}(A) = \sum_{k=1}^{K} p_{kA}(1 - p_{kA}) = 1 - \sum_{k=1}^{K} p_{kA}^2$

entropy index: $f_{enpropy}(A) = -\sum_{k=1}^{K} p_{kA} \log(p_{kA})$.

controls how a decision tree decides where to split the data.

* smaller values of the impurity index means higher purity.

Overall impurity = $\frac{|A_L|}{|A_L| + |A_R|} f(A_L) + \frac{|A_R|}{|A_L| + |A_R|} f(A_R)$.

Objective.

$$\{j^*, s^*\} = \underset{j \in \{1,\dots,p\},\, s \in \mathbb{R}}{\arg\min} \quad \text{Overall impurity } (A_L, A_R).$$

⇒ We have to repeat the process until, we reach the stopping rule.

* minsplit : the minimum number of observations in any non-terminal node.

* minbucket : the minimum number of observations allowed in a terminal node.

* cp : complexity parameter – minimum difference between impurity values required to continue splitting.

rel error = in-sample error (always decreases with more split).

Xerror = the cross-validation error.

Xstd = the standard deviation of the cross-validation error.

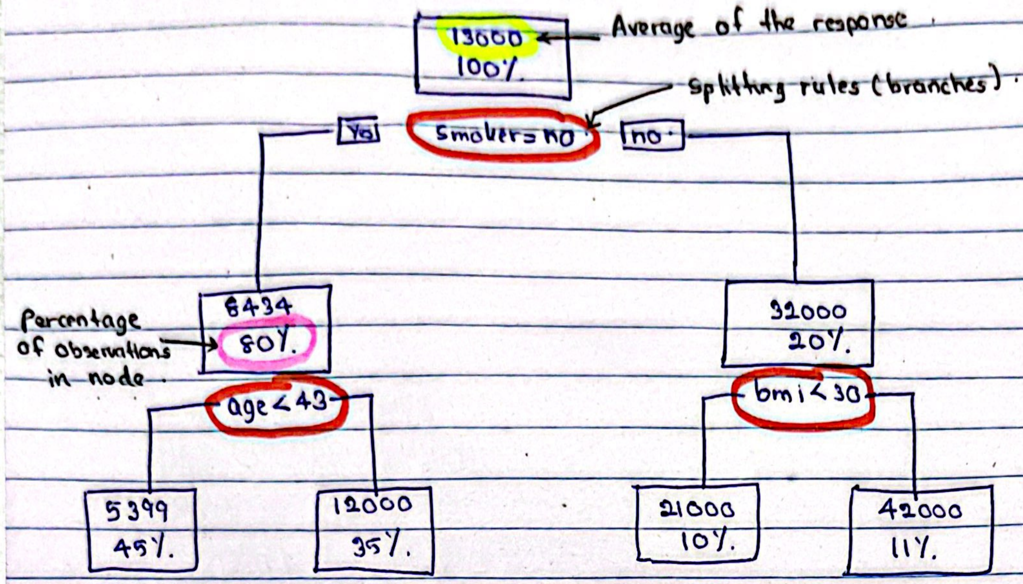
Accuracy measures.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}.$$

$$\text{Balanced accuracy} = \frac{TPR + FPR}{2} = \frac{TP}{2(TP+FN)} + \frac{FP}{2(FP+TN)}$$

$$\left.\begin{array}{r}\text{Cohen's kappa} \\ \text{coefficient}\end{array}\right\} = \frac{2 \times (TP \times TN - FN \times FP)}{(TP+FP)(TN+FP) + (TP+FN)(TN+FN)}.$$

Regression tree



* Colors indicate class, with a darker color indicating lower impurity.

* The prediction is the average of the response in the terminal node.

* For a binary variable, A_L and A_R is based on its class.
* Then find the average responses, $\bar{y}_L$ and $\bar{y}_R$, for A_L and A_R, respectively.

* We can use the residual sum of squares, to measure the impurity.

$$RSS(A_L) = \sum_{i \in A_L} (y_i - \bar{y}_L)^2.$$

* The overall impurity is $RSS(A_L) + RSS(A_R)$.

Bagging (More on next tutorial)

* Multiple subsets are created from the original data set with equal tuples, selecting observations with replacement.

* A base model is created on each of these subsets.

* Each model is learned in parallel with each training set and independent of each other.

* The final predictions are determined by combining the predictions from all the models.