Machine Learning

Random Forests

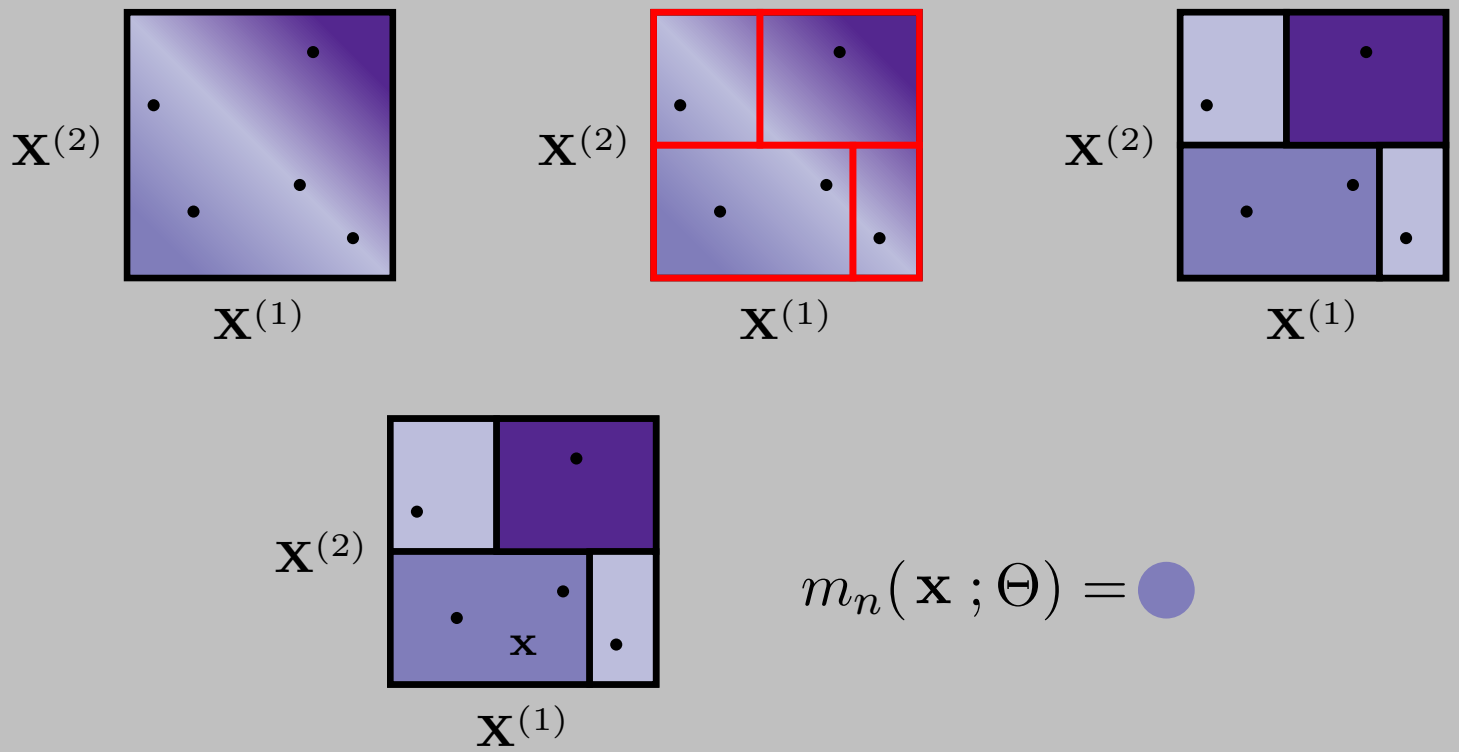
Irving Gómez Méndez

August-December, 2021

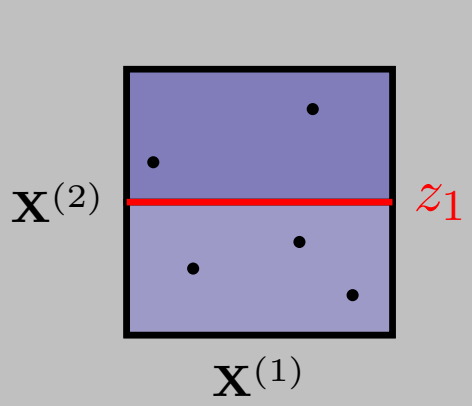
Regression Random Forests

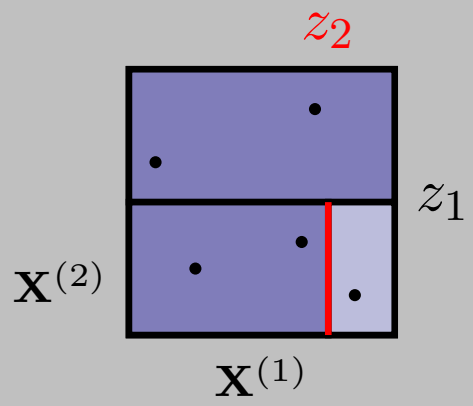
Estimation of the Regression Function

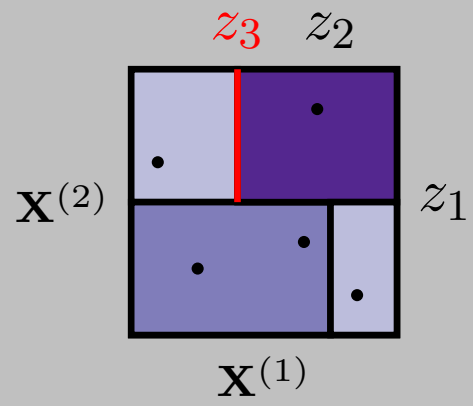
We can estimate m dividing (somehow) $\mathcal{X}$ in disjoint regions and predicting with a constant in each region.

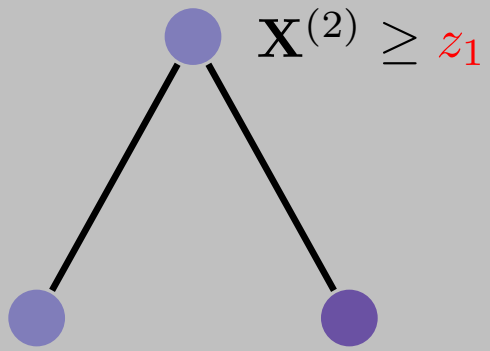


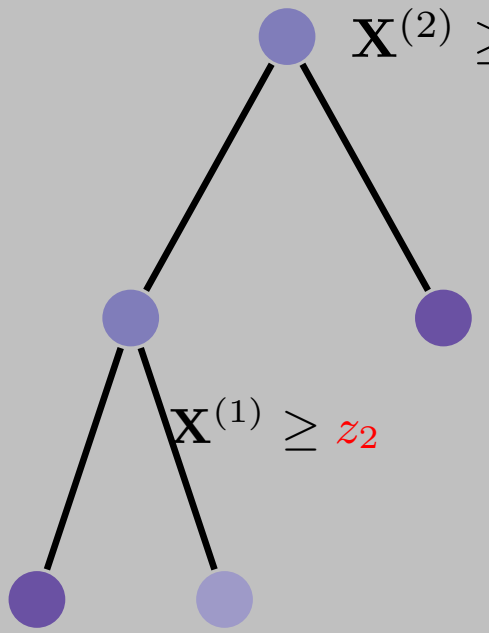
Recursive Trees / Creation of Disjoint Regions

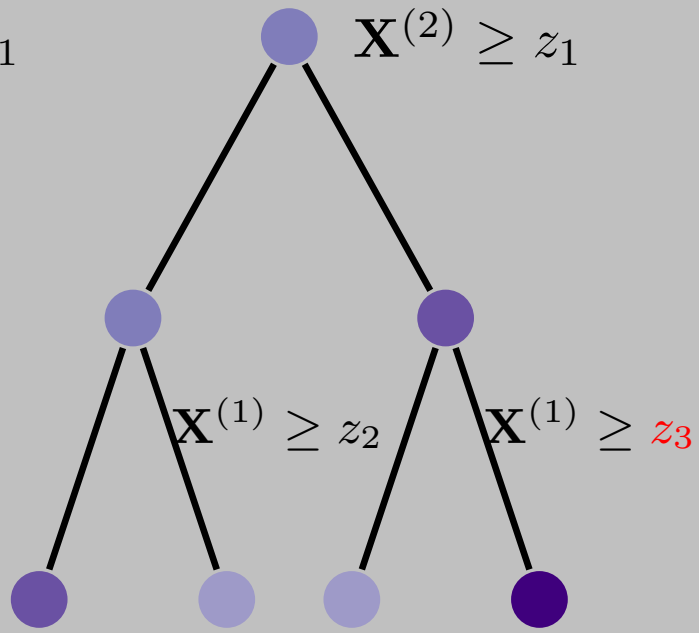












CART-split Criterion

Let $d = (h, z)$ be a cut in direction h at position z,

$$
\begin{aligned}
L_n(A, d) =& \frac{1}{N(A)} \sum_{i=1}^{n} \left(Y_i - \bar{Y}_A \right)^2 \mathbb{1}_{\mathbf{X}_i \in A} \\
& - \frac{1}{N_n(A)} \sum_{i=1}^{n} \left(Y_i - \bar{Y}_{A_L} \mathbb{1}_{\mathbf{X}_i^{(h)} < z} - \bar{Y}_{A_R} \mathbb{1}_{\mathbf{X}_i^{(h)} \geq z} \right)^2 \mathbb{1}_{\mathbf{X}_i \in A} \\
& \vdots \\
=& \frac{N(A_L) N(A_R)}{N(A) N(A)} \left(\bar{Y}_{A_L} - \bar{Y}_{A_R} \right)^2 .
\end{aligned}
$$

The **CART-criterion** attempts to **minimize the variance inside the cells**, making the cells as distinct as possible (in terms of the value of Y) but **maintaining the balance in the number of points** in the cells.
We construct the cells maximizing the CART criterion over all possible cuts in cell A,

$$\widehat{d} = (\widehat{h}, \widehat{z}) \in \arg\max_{d \in \mathcal{C}_A} L_n(A, d),$$

where $\mathcal{C}_A$ is the set of all possible cuts in node A.

Random Forest

A random forest is an ensemble of trees, i.e.

$$m_{M,n}(\mathbf{x}; \Theta_1, \ldots, \Theta_M) = \frac{1}{M} \sum_{k=1}^{M} m_n(\mathbf{x}; \Theta_k).$$

We expect the **predictor**:

- To be more stable
- To have less variance

Conditions:

- Independent or non-correlated trees

However the trees are not independent because they use the same data $\mathcal{D}_n$.

Solution:

- Introduce sources of randomness

Parameters

We add two sources of randomness in each tree:

1. We **select randomly** a_n (with or without replacement) **observations** prior to the construction of each tree.
2. We **select randomly** `mtry` **candidate directions** to perform the cut.

These are parameters of the random forest together with:

1. M, which is the **number of trees**. It is only restricted by computational power.
2. `nodesize`, which is the **maximum number of points in a final cell**.

Classification Random Forests

Split Criteria for Classification

For classification, the split criterion usually takes the form of an impurity function to be minimized, like the misclassification error, the Gini index or the criteria known as ID3 and C4.5 which minimize the Shannon entropy and replace binary splits on categorical variables with multiway splits.

To define the impurity functions for the classification tasks, assume that $\mathcal{Y} = \{c_1, \ldots, c_J\}$, fix a cell A of the tree and denote by $N(A)$ the number of observations belonging to that cell, let be

$$p_{c_j} = \frac{1}{N(A)} \sum_{i=1}^{n} \mathbb{1}_{Y_i = c_j, \mathbf{X}_i \in A}$$

the proportion of observations of the class c_j in node A. We classify the observations in cell A to class $\hat{c}_A = \arg\max_{c_j} p_{c_j}$, the majority class in the cell.

The previous impurity functions are then defined as

$$\text{Missclassification error:} \quad \frac{1}{N(A)} \sum_{i=1}^{n} \mathbb{1}_{Y_i \neq \hat{c}_A, \mathbf{X}_i \in A} = 1 - p_{\hat{c}_A}$$

$$\text{Gini index:} \quad \sum_{j=1}^{J} p_{c_j}(1 - p_{c_j})$$

$$\text{Shannon entropy:} \quad -\sum_{j=1}^{J} p_{c_j} \log_2 p_{c_j}$$

For two classes, if p is the proportion in the second class, these three measures are $1 - \max\{p, 1-p\}$, $2p(1-p)$ and $-p\log_2(p) - [(1-p)\log_2(1-p)]$.

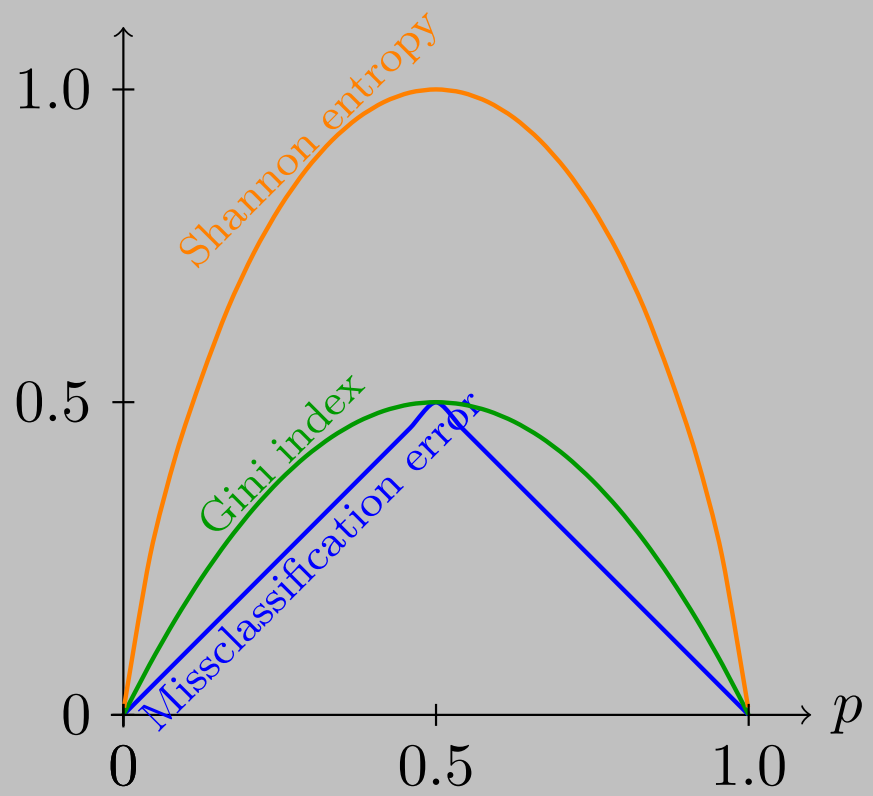


Figure: Node impurity measures for two-class classification, as a function of the proportion p in the second class.

Properties of the Criteria

All three are similar, but Shannon entropy and Gini index are differentiable, and hence more amenable to numerical optimization. In addition, Shannon entropy and Gini index are more sensitive to changes in the node probabilities than the misclassification rate.
However, while they are robust and reliable impurity functions they are not exempt of defects, like the end-cut preference, that is the tendency to favor unbalanced splits (in the number of points belonging to each child node) in which p is near 1 or zero, resulting in deep and uninterpretable trees. Furthermore, as some authors have pointed out, they have the propensity of preferring splits based on input variables with many outcomes.

Conditional Inference Trees

Facing this problem, conditional inference trees were introduced. Splits are performed in two steps. In the first step the relation of a variable to the response in assessed by permutation tests. After the strongest relation is found by minimal p-value of the permutation tests it is checked if the significance to a certain level is still present after adjustment for multiple testing. Finally, in the second step the best cutpoint is determined for the variable chosen. The growth of the tree stops when no further significant relations are found.