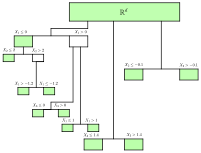
Interpretable Machine Learning

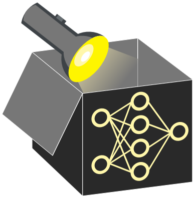
Random Planted Forests



Learning goals

- Motivation for RPFs
- Understand node types and restricting interactions in decision trees
- Understand planted trees: non-binary decision trees and inner leaves

RANDOM PLANTED FORESTS (RPF) ▸ Hiabu et al. 2023

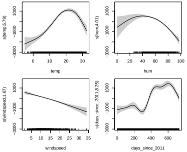


Goal: Create a powerful tree ensemble, but still interpretable

Idea:

- GAMs easily interpretable, because no interaction ⇝ Plot 1D functions

$$\hat{f}(x) = \theta_0 + f_1(x_1) + f_2(x_2) + \ldots + f_p(x_p),$$



- Same for function containing interactions between max. 2 features ⇝ function of 2 features, Plot 2D functions (i.e. 3D plot)

$$\hat{f}(x) = \theta_0 + f_1(x_1) + f_2(x_2) + \ldots + f_p(x_p) + f_{1,2}(x_1, x_2) + \ldots + f_{1,p}(x_1, x_p) + \ldots + f_{p-1,p}(x_{p-1}, x_p),$$

⇝ Visualize single functions $f_1, f_2, f_{1,2}(x_1, x_2), f_{1,3}(x_1, x_3) \ldots$

⇒ Interpretability possible via restricting degree of interactions

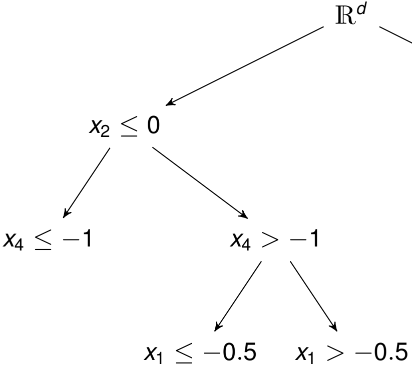
Problem: How to know degree of interactions?

Solution: Easy to determine for trees / tree ensembles!

RPF: DETERMINE INTERACTION TYPE IN TREES

Define the *interaction type t* of a node as the subset of features involved in constructing this node.

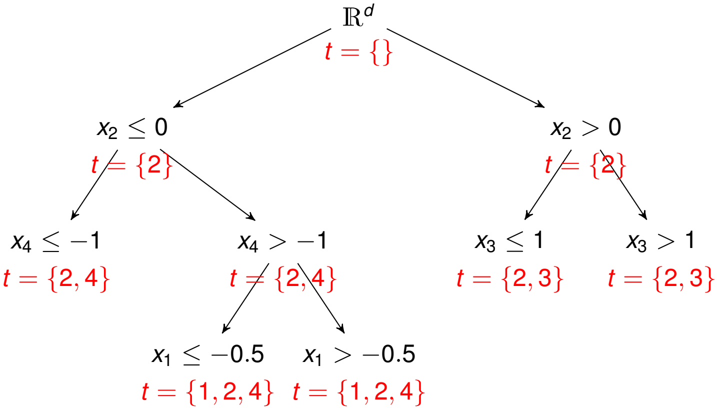
Example:



RPF: DETERMINE INTERACTION TYPE IN TREES

Define the *interaction type t* of a node as the subset of features involved in constructing this node.

Example:



$\mathbb{R}^d$

$t = \{\}$

$x_2 \leq 0$

$t = \{2\}$

$x_2 > 0$

$t = \{2\}$

$x_4 \leq -1$

$t = \{2, 4\}$

$x_4 > -1$

$t = \{2, 4\}$

$x_3 \leq 1$

$t = \{2, 3\}$

$x_3 > 1$

$t = \{2, 3\}$

$x_1 \leq -0.5$

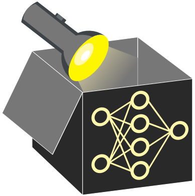
$t = \{1, 2, 4\}$

$x_1 > -0.5$

$t = \{1, 2, 4\}$

$\Rightarrow$ Degree of interaction in each node is $|t|$.

RPF: BOUNDED INTERACTION ORDER + PLANTED TREES



Goal: restrict this interaction degree
⇝ In RPFs:

- Always keep track of interaction type in each node
- For each new split, make sure max. degree of interactions is not exceeded ⇒ When max. number of features reached, no new features are allowed

Problem: For small interaction order, single trees quickly limited

- E.g. interaction order 1: Every tree only one feature

⇒ Many trees needed for more complex model

Idea: Allow inner nodes to split again
Define *Planted Trees*: Decision trees in which each inner nodes can be *leaves*:

- Add prediction to final output
- Can be split again ⇒ several splits possible

RPF: EXAMPLE

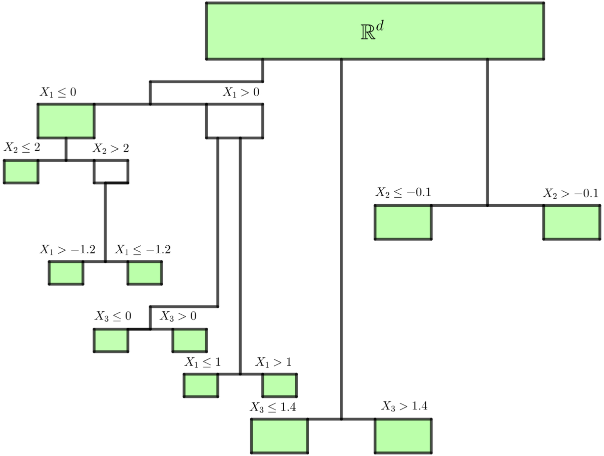


Figure: Example of a single fully grown planted tree, green nodes: “leaves”

RPF: EXAMPLE

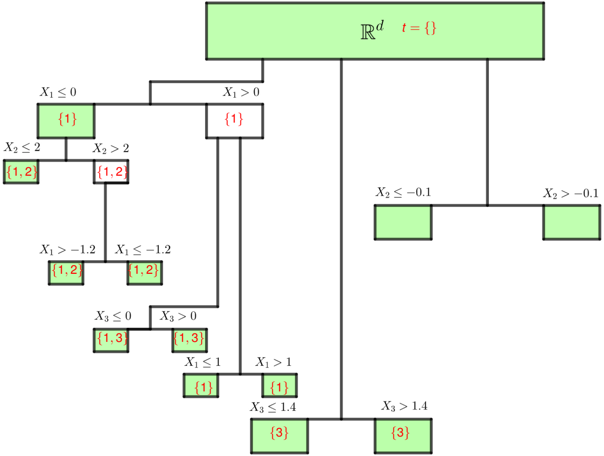
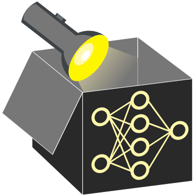


Figure: Example of a single fully grown planted tree, green nodes: “leaves”

RPF: ALGO



- Max. interaction degree is a hyperparameter
- Total number of trees is a hyperparameter
- End growing tree after max. total number of splits instead of max. depth (min. number of samples also possible, but then higher nodes would split too often)
- Randomization as in Random Forests:
  - Only optimize over subset of features, randomly chosen
  - Only optimize over subset of possible split values
- Make an inner leaf an inner node (i.e. delete “leaf” property), if it has children with the same type

RPF: EXAMPLE RESULTS AND INTERPRETATION

RPF: CONCLUSION