Compressing Tree Ensembles through Level-wise Optimization and Pruning

Laurens Devos*, **Timo Martens***, Deniz Can Oruc, Wannes Meert, Hendrik Blockeel and Jesse Davis

KU LEUVEN LEUVEN.AI INSTITUTE

*equal contribution

Tree ensembles are easy to configure, fast to learn, performant, and solve real problems

The Guardian

News | Opinion | Sport | Culture | Lifestyle

Education Schools Teachers Universities Students

Football

The Belgian lab shaping modern soccer's data revolution

Leander Schaerlaeckens

Tue 3 Jun 2025 13.41 CEST

Share

arXiv > cs > arXiv:2106.03253

Computer Science > Machine Learning

[Submitted on 6 Jun 2021 (v1), last revised 23 Nov 2021 (this version, v2)]

Tabular Data: Deep Learning is Not All You Need

Ravid Shwartz-Ziv, Amitai Armon

RegCheck: A Real-Time Approach for Flagging Potentially Malicious Domain Name Registrations

Computer Science > Machine Learning

[Submitted on 18 Jul 2022]

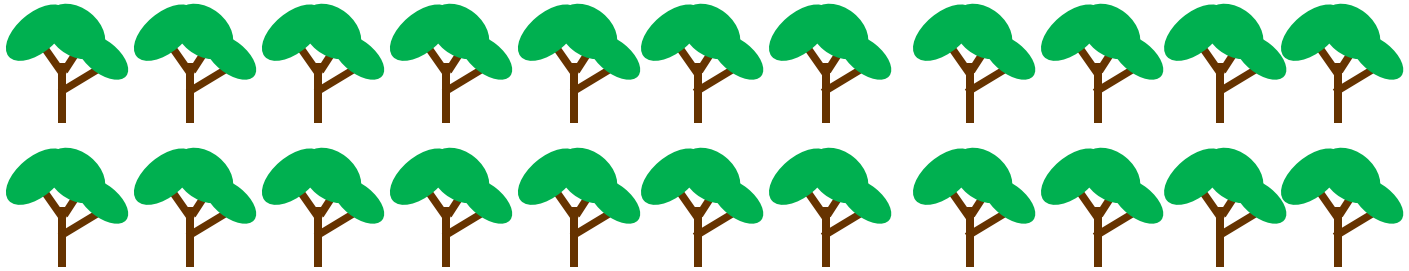
Why do tree-based models still outperform deep learning on tabular data?

Léo Grinsztajn (SODA), Edouard Oyallon (ISIR, CNRS), Gaël Varoquaux (SODA)

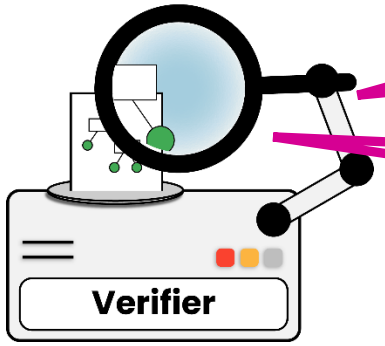
Evaluation of Automated Hypnogram Analysis on Multi-Scored Polysomnographies

Dries Van der Plas [1,2,3,*], Johan Verbraecken [3,4,5], Marc Willemen [4], Wannes Meert [2], Jesse Davis [2]

Complex ensembles increase performance, but yield other problems



1. Hard to verify



Robust to evasion attacks?

Fair, e.g., causal discrimination?

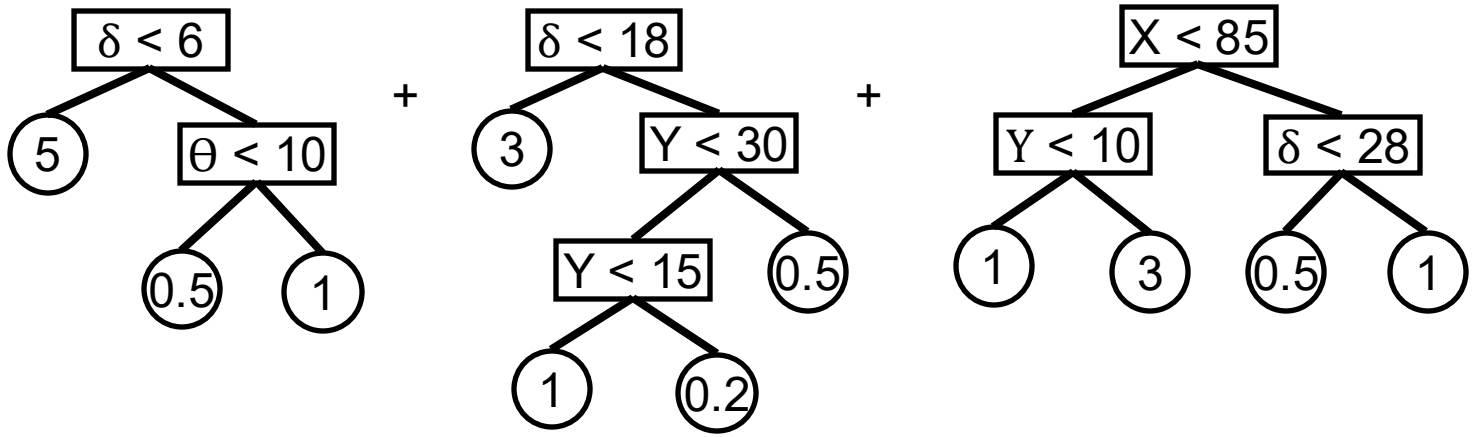
2. Not ideal for resource constrained devices

Solution: Simplify the ensemble by pruning (sub)trees!

LOP: Level-wise Optimization and Pruning
[Devos et al. ICML'2025]

Given: Learned tree ensemble

Do: Compress the model s.t. lose < 0.5 percentage points of performance on validation data by removing

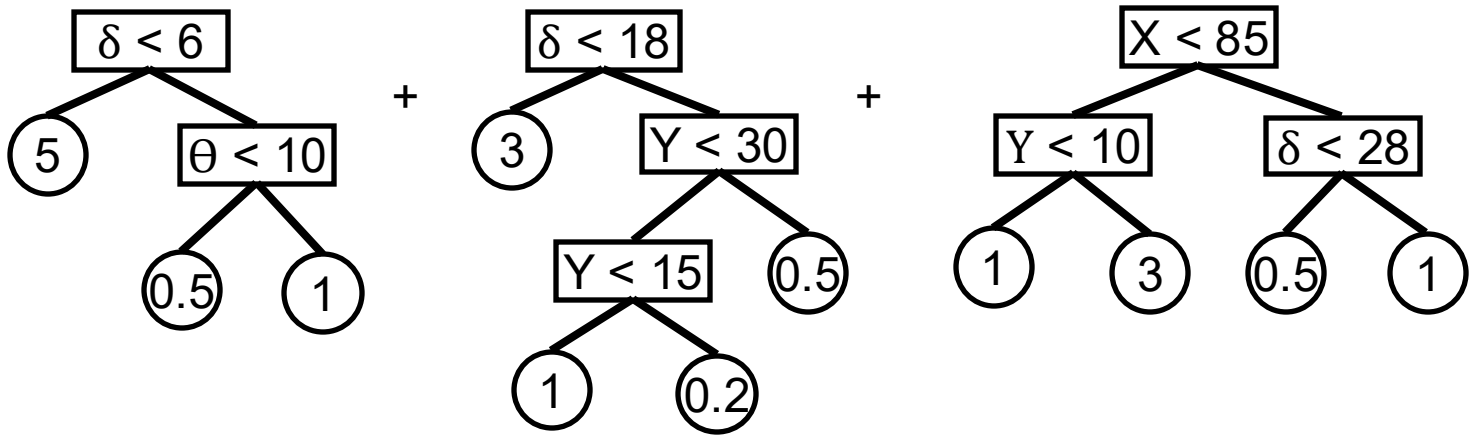


LOP: Level-wise Optimization and Pruning
[Devos et al. ICML'2025]

Given: Learned tree ensemble

Do: Compress the model s.t. lose < 0.5 percentage points of performance on validation data by removing

1. Full trees

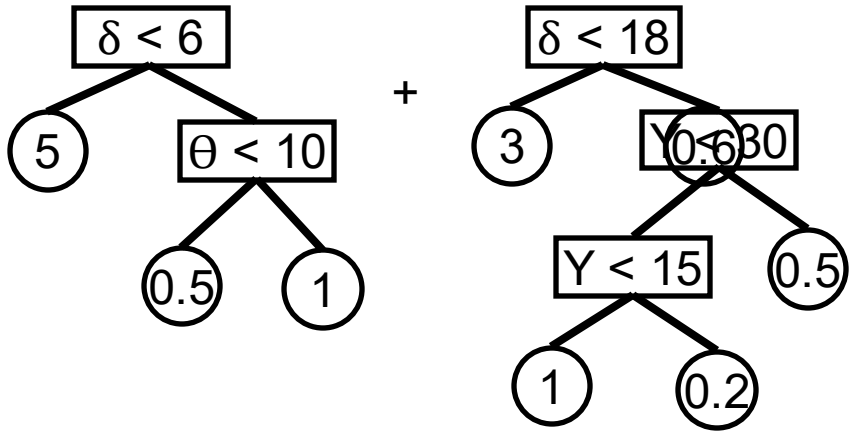


LOP: Level-wise Optimization and Pruning
[Devos et al. ICML'2025]

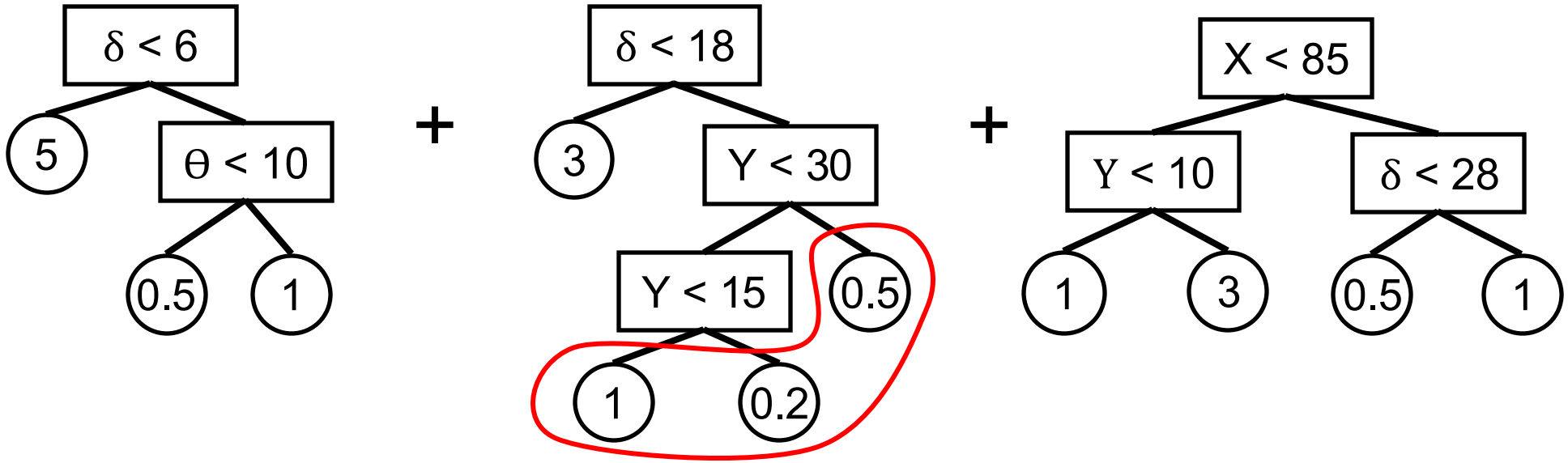
Given: Learned tree ensemble

Do: Compress the model s.t. lose < 0.5 percentage points of performance on validation data by removing

1. Full trees
2. Subtrees



Intuition of our approach



- If these leave values were nearly identical: Replace Y < 30 with a leaf
- **Problem:** Doing so would decrease this tree's performance
- **Solution:** Fine-tune leaves in other trees to compensate

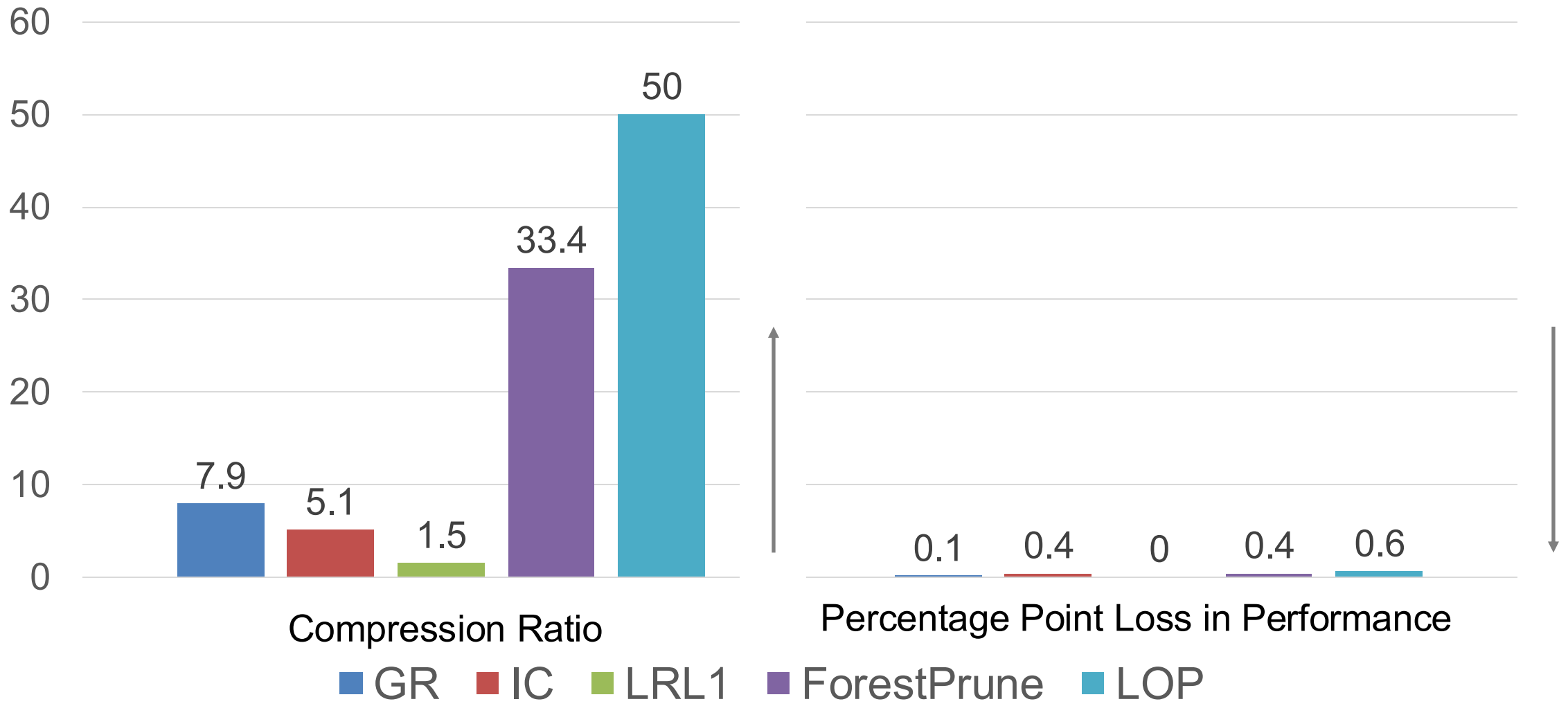
Our idea: Formulate global optimization problems to encourage such compression and compensation!

Q1: How does LOP compare to existing approaches?

- Trained multiple models on each of 14 binary classification datasets
  - XGBoost: Varied depth, number of trees, & learning rate
  - (RandomForest and regression results are in the paper)

- Compressed each model using:
  - Global refinement (GR): Recursively merges leaves with similar values
  - Individual contributions (IC): Selects a subset of trees
  - LRL1: Prunes only whole trees
  - ForestPrune: Prune whole (sub)trees at a specific level
  - LOP: Our approach

LOP yields higher compression ratios and is equivalently performant as the XGBoost model



Intrigued? Come talk to us

Laurens

@laudevs.bsky.social

Timo

- Funding:

Funded by the European Union

Artificial Intelligence Flanders