Pushing the Envelope of Gradient Boosting Forests via Globally-Optimized Oblique Trees

Magzhan Gabidolla and Miguel Á. Carreira-Perpiñán

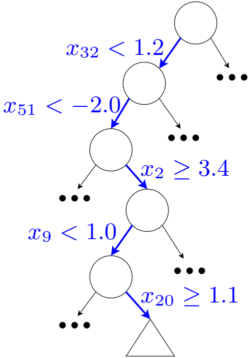
Dept. of Computer Science & Engineering
University of California, Merced

CVPR 2022

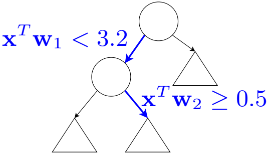
Gradient Boosting (GB) Forests

- Ensembles of decision trees have long been established as some of the most powerful, off-the-shelf machine learning models.
- In recent years, one type of forest, Gradient Boosting (GB), has gained prominence due to their:
  - Strong empirical performance on many problems
  - The development of extremely efficient implementations such as XGBoost or LightGBM.
- They typically require little effort on hyperparameter tuning and are thus considered "off-the-shelf".
- Given the tremendous effort put on the development and refinement of the popular GB toolkits, how can we further improve GB forests?

Modeling high-order feature interactions: Axis-aligned vs Oblique trees



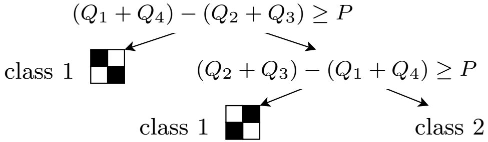
- Only 5 features participate in the routing function of the above leaf.
- Max order of feature interactions is limited by the depth Δ in axis-aligned trees.

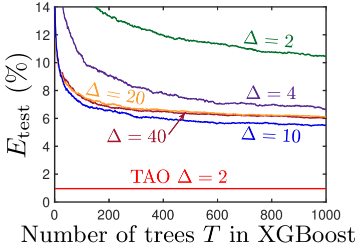


- Each decision node is a function of all the features.
- Their non-linear combination is a much more complex order-D interaction.
- For modeling complex functions, a forest of oblique trees should achieve higher accuracy and require fewer and shallower trees.

Synthetic MNIST binary classification

Imagine splitting 28×28 pixel image into 4 quadrants $\boxed{\begin{smallmatrix}1 & 2\\ 3 & 4\end{smallmatrix}}$.
Let Q_i be the sum of $[0,1]$ pixel intensities in quadrant i, and $P = 30$.





Tree Alternating Optimization (TAO) for GB objective function

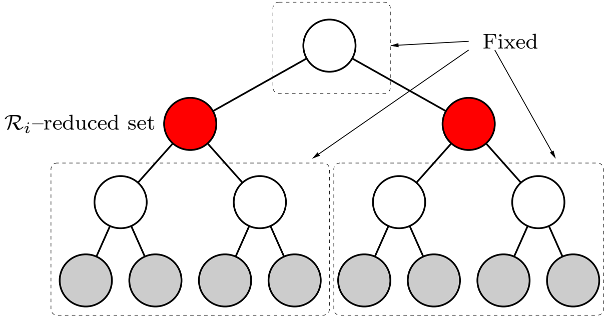
Let $\boldsymbol{\tau}(\mathbf{x};\boldsymbol{\Theta})$ be a binary decision tree of some predetermined structure with parameters $\boldsymbol{\Theta} = \{(\mathbf{w}_i, w_{i0})\}_{i\in\mathcal{D}} \cup \{\boldsymbol{\theta}_i\}_{i\in\mathcal{L}}$, decision nodes in set $\mathcal{D}$ and leaves in set $\mathcal{L}$.

$$\min_{\boldsymbol{\Theta}} \sum_{n=1}^{N} l(\mathbf{g}_n, \mathbf{H}_n, \boldsymbol{\tau}(\mathbf{x}_n;\boldsymbol{\Theta})) + \alpha \sum_{i\in\mathcal{D}} \|\mathbf{w}_i\|_1$$

$$\text{where} \qquad l(\mathbf{g}, \mathbf{H}, \boldsymbol{\gamma}) = \mathbf{g}^T\boldsymbol{\gamma} + \frac{1}{2}\boldsymbol{\gamma}^T\mathbf{H}\boldsymbol{\gamma}.$$

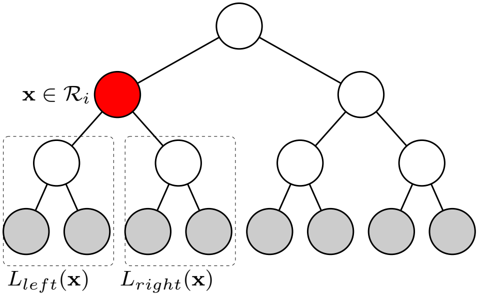
Alternating optimization and separability condition

- Any set of non-descendant nodes of a tree can be optimized independently:



Reduced problem over a decision node

- Evaluate loss induced by left/right subtrees;
- Generate pseudolabel for each instance in reduced set $\mathcal{R}_i$;
- Solve weighted binary classification problem (linear):



Decision node The reduced problem takes the form:

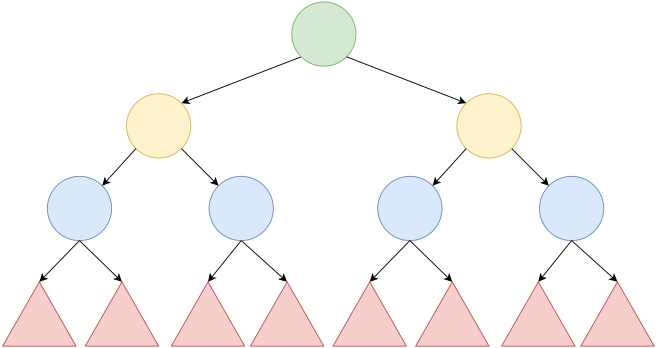
$$\min_{\mathbf{w}_i, w_{i0}} \sum_{n \in \mathcal{R}_i} \bar{L}(\mathbf{g}_n, \mathbf{H}_n, f_i(\mathbf{x}; \mathbf{w}_i, w_{i0})) + \alpha \left\| \mathbf{w}_i \right\|_1. \qquad (1)$$

This problem is NP-hard but can be well approximated with a convex surrogate; we use ℓ_1-regularized logistic regression, and solve it using LIBLINEAR [1].

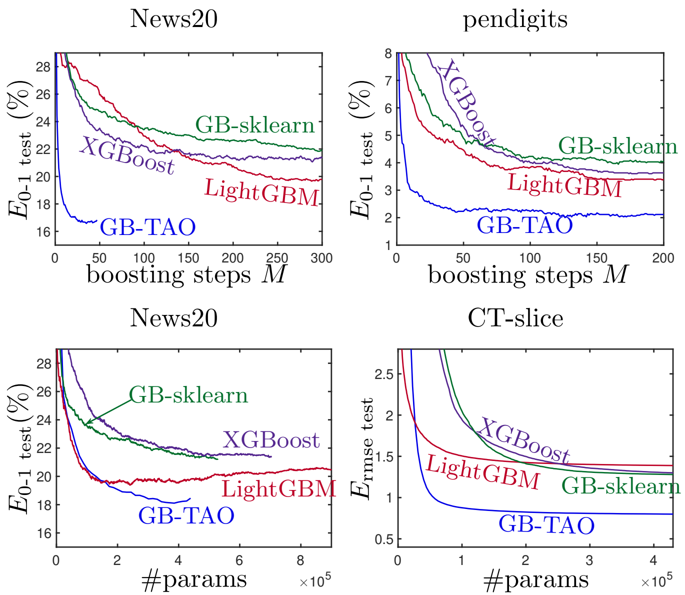
Leaf The reduced problem consists of optimizing the original loss but over the leaf classifier on its reduced set:

$$\min_{\boldsymbol{\theta}_i} \sum_{n \in \mathcal{R}_i} \mathbf{g}_n^T \boldsymbol{\theta}_i + \frac{1}{2} \boldsymbol{\theta}_i^T \mathbf{H}_n \boldsymbol{\theta}_i. \qquad (2)$$

If $\sum_{n \in \mathcal{R}_i} \mathbf{H}_n$ is positive definite, the exact solution is $\boldsymbol{\theta}_i = -\left(\sum_{n \in \mathcal{R}_i} \mathbf{H}_n\right)^{-1} \sum_{n \in \mathcal{R}_i} \mathbf{g}_n$. In practice either θ_i is scalar (e.g. binary classification) or one uses a diagonal approximation to the Hessian.



Experimental results: comparison



Conclusion

- We have motivated the use of a significantly more powerful tree type having hyperplane splits, which are able to learn many-feature interactions effectively.
- Key to this is the ability to optimize the GB loss over such trees, a difficult problem which we address using a variation of tree alternating optimization.
- In raw accuracy, the oblique forests consistently improve over all competitors, sometimes by a surprisingly large margin, using few, shallow trees, often having fewer parameters overall.
- Our work also suggests that exploring other types of trees or loss functions, properly optimized, may result in even better GB forests.
- Work supported by NSF award IIS–2007147

References

[1] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *J. Machine Learning Research*, 9:1871–1874, Aug. 2008.