

Robustness Verification of Tree-based Models

Hongge Chen* (MIT), Huan Zhang* (UCLA), Si Si (Google), Yang Li (Google),
Duane Boning (MIT), and Cho-Jui Hsieh (UCLA) (*Equal Contribution)



Source code (XGBoost compatible!): <https://github.com/chenhongge/treeVerification>

Introduction

Robustness Verification problem:

$$f^* = \min_{\|\delta\|_\infty \leq \epsilon} f(x + \delta)$$

We compute a **lower bound** of f^* and improve it iteratively.

- We verify the robustness for **tree based models** (include GBDT, random forest, etc)
- Cast as a **max-clique enumeration problem** on a **multi-partite graph with bounded boxicity**.
- up to **100X faster** than exact verification, **small gap** to f^*

Verify your **XGBoost** model today!

<https://github.com/chenhongge/treeVerification>

Single Tree Verification

Insight: decision tree nodes partition the feature space using boxes, whose boundaries can be tracked. The partitions can be generated in **linear time**.

Exact verification of a single tree is easy!

But how to verify a tree ensemble?

- **Naive:** consider the worst case of each tree, and add worst case together (loose bound, but very fast)
- **Ours:** consider multiple trees together using graph theory (much tighter)

References: [1] Cheng, Minhao, et al. "Query-efficient hard-label black-box attack: An optimization-based approach." ICLR 2019
[2] Kantchelian, Alex, J. D. Tygar, and Anthony Joseph. "Evasion and hardening of tree ensemble classifiers." ICML 2016.
[3] Chen, Hongge, et al. "Robust Decision Trees Against Adversarial Examples." ICML 2019

Tree Ensemble Verification

If $f(x) > 0$, we are safe as long as the largest sum of leaf values in all reachable regions is > 0 .

Theorem: The graph connecting all reachable leaves of K trees with d features is a **K-paritite graph** with **boxicity d**

Theorem: Each K -clique of the graph represents a reachable output for x after perturbation

3-partite graph 3-cliques Find 2-cliques=check connectivity

Tree 1 Tree 2 Tree 3 Tree 1 Tree 2 Tree 3 Tree 1 Tree 2 Tree 3

convert 2-cliques to vertices possible due to boxicity Find 2-cliques=check connectivity Each vertex=a 3-clique in original graph possible due to boxicity

Tree 1+2 Tree 3 Tree 1+2 Tree 3 Tree 1+2+3

Efficient Multi-level Verification

Finding K -cliques on all K trees can be expensive. We can group K trees to M groups and find (K/M) -cliques inside each group, and use naive bounds between groups

Tree (1) Tree (2) Tree (3) Tree (4)

Leaf nodes: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Merge (1) and (2) Merge (3) and (4)

Run single-level algorithm to get level 1 bound

Merge (1) (2) and (3) (4)

Run single-level algorithm to get level 2 bound

Final (exact) solution

Experiments

Cheng's Attack [1]/MILP [2] Ours/MILP [2]

breast-cancer covtype diabetes Fashion-MNIST HIGGS ijcnn1 MNIST webspam MNIST 2 vs. 6

The average ℓ_∞ distance of Cheng's attack [1] and our verification method. The distance is normalized with distance by MILP [2]. Numbers close to 1 indicate a tighter bound.

Cheng's Attack [1]/MILP [2] Ours/MILP [2]

breast-cancer covtype diabetes Fashion-MNIST HIGGS ijcnn1 MNIST webspam MNIST 2 vs. 6

The average running time of Cheng's attack and our verification method. The running time is normalized with MILP's running time.

Robustness bounds obtained with different number of nodes in cliques and different number of levels for searching on a 20-tree GBDT model with the diabetes dataset

Running time of MILP and our method on GBDT models with different number of trees on the ijcnn1 dataset.

Verified error ($\epsilon=0.3$)

Minimax adversarial training (Madry et al. 2017) arxiv.org/abs/1706.06083 Not verifiable (~10% attack error)

Convex relaxations (Wong et al. 2018) arxiv.org/abs/1805.12514 43.10%

IBP (Gowal et al. 2018) arxiv.org/abs/1805.12514 8.05%

CROWN-IBP (Zhang et al. 2019) www.huan-zhang.com/crown-ibp.pdf 6.68%

Maximin robust GBDT (this work) 34.48%

Trees are more verifiable than neural networks

Unlike the minimax based adversarial training on deep training, [3] uses a similar maximin robust optimization formulation but can be verified. Compared to DNNs, tree based models are more verifiable (by MILP based exact verification [2] and this work) as tight and fast verification methods are available