Ensemble Tree Verification using Propositional Logic

1 Clique Formulation as Weighted Max-SAT

We aim to formulate a weighted Max-SAT problem which, given a point x with predicted output y and $Ball(x, \epsilon)$ represents the task of searching over a random forest to find a point x' such that:

$\vee_{x'} x' \in Ball(x, \epsilon) \wedge f(x') \neq y$

Let $b = \{b_1, ..., b_T\}$, where $b \in \mathcal{B} = \mathcal{B}_1 \times ... \times \mathcal{B}_T$ where b_i indexes the leaf nodes in tree i. Furthermore, let $Box(b_i)$ define the bounding box in the coordinate space represented by leaf node b_i and $Val(b_i)$ denote the value output by that leaf.

To craft our statement we need two types of weighted clauses:

1. $\phi_{ij}(b_i, b_j)$, Clauses with weight 1 to check whether leaf nodes from two different trees i and j overlap: $Box(b_i) \cap Box(b_j)$
2. $\phi_i(b_i)$, Clauses with weight $Val(b_i)$ to check whether a leaf node from a tree i over laps with the ball: $Box(b_i) \cap Ball(x, \epsilon)$

We can then combine these two clauses to write out

$\wedge_{ij} \phi_{ij} \wedge_i \phi_i$

and with the associated weights and run a weight MaxSAT over the expression. The solver should return a configuration values for the variables b.

2 Boolean encoding

The b variables in the problem of the previous section take on values beyond zero and one. In this section we describe a formulation using the variables described in the table below to rewrite equation 1.

To code the equivalent CNF model of the ensemble tree, we extract all thresholds from the nodes of the tree. For each threshold (Figure 1) we create one column in data which is True if the value of the corresponding feature x_{n_i} is in the right path and is False otherwise. Therefore, $\mathbf{X} \subseteq \{0, 1\}^N$, where N is the number of unique feature threshold pairs.

Using the binary data, let $\mathcal{F} : \mathbf{X}^n \rightarrow \mathcal{R}^2$ be the decision function of the tree based classifier. For each path combination p in a tree ensemble $\mathbf{X}_p \subseteq \mathbf{X}^n$ that leads to traversing p, the corresponding output is $y_p \in \mathcal{R}^2$. To create the SAT clauses we generate the equivalence classes as pairs of $(\mathbf{X}_p, y_p)$, where $y_{p,i}$ if the decision associated with the leaf p_i of tree i and we use the leaf node value $Val(b_{p,i})$ as the MaxSAT weights.

To check the adversarial robustness of the tree based model with SAT, the CNF of the ensemble contains the following components:

1. Declaring Boolean variables for each binary feature $x_{f,th_{f,i}}$ defined for feature f and threshold i associated with that feature $th_{f,i}$; and each decision box $b_{p,i}$.
2. Define the boundaries of the decision boxes, e.g., $b_{1,1} \leftrightarrow (\neg x_{2,th_{2,1}} \wedge x_{5,th_{5,3}})$ means that the boundaries of $b_{1,1}$ are defined by $x_{2,th_{2,1}}$ and $x_{5,th_{5,3}}$. Since $x_{2,th_{2,1}}$ is negated, the

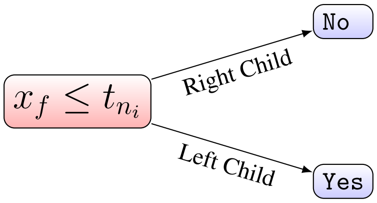

Figure 1: i^{th} node of a tree with threshold t_{n_i}.

values of this box should on the left branch (smaller than or equal to $th_{2,1}$. And the other boundary contains the feature x_5 and should be on the right branch (larger than $th_{5,3}$).

3. Each tree must have only one decision. Therefore, we force each tree to have exactly one True decision, e.g., for a tree with three decision boxes $b_i, i \in \{1, 2, 3\}$ we have $(b_1 \vee b_2 \vee b_3) \wedge (b_1 \rightarrow \neg(b2 \vee b_3)) \wedge (b_2 \rightarrow \neg(b_1 \vee b_3)) \wedge (b_3 \rightarrow \neg(b_1 \vee b2))$.
4. The search should be limited to the boxes that overlap with $Ball(x, \epsilon)$. Therefore, all the decision boxes that do not have an overlap with the ball are negated.
5. To choose the best combination of the leaves, we use the value of the leaves as the weights of the soft assertions; and only choose the leaves that have an overlap with $Ball(x, \epsilon)$. MaxSAT therefore finds the satisfying clauses with the highest score.
6. Since we want the decision result to be changed, the current leaves should not be chosen. To enforce this constraint the decision boxes associated with the input sample are negated.

Assuming that the trees are of equal size, the number of path combinations in the tree ensemble with T trees of depth d is $2^{d \times T}$. However, in practice, decisions made by the individual trees are influenced by a subset of features shared amongst several trees within the same ensemble, and thus several path combinations are infeasible and may be discarded from analysis.