Appendix

A Robust Relabeling Criterion

In Algorithm 1 we give pseudocode for training decision trees using the robust relabeling procedure as a splitting criterion. The algorithm does a depth-first search through all possible decision nodes up to a maximum depth and sets thresholds that locally maximize adversarial accuracy.

The maximum matching step takes worst-case $\mathcal{O}(n^{2.5})$ time. Since this step gets called for every decision node (2^d nodes) and for every possible split value (nm values) the worst-case overall runtime is $\mathcal{O}(2^d mn^{3.5})$. Here n is the number of samples, m the number of features and d the maximum depth of the tree.

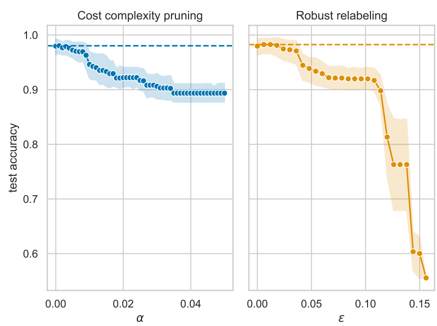
Algorithm 1 Robust decision tree learning with robust relabeling criterion

Input: dataset X (n samples, m features), labels y, tree leaves $\mathcal{T}_L$, maximum depth d

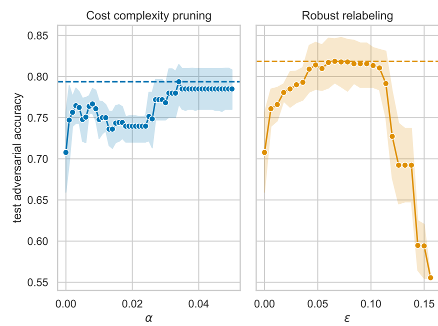
$L \leftarrow \{i \mid y_i = 0\}$
$R \leftarrow \{i \mid y_i = 1\}$
$V_j \leftarrow$ sorted values of feature $j = 1...m$
loss $\leftarrow n$
for $k \in 1...2^d$ **do** ▷ For each decision node up to maximum depth
    best_loss $\leftarrow$ loss
    $\mathcal{T}_L^* \leftarrow \mathcal{T}_L$
    for $j = 1...m,\ v \in V_j$ **do** ▷ Try every split value
        $\mathcal{T}'_L \leftarrow$ ADD_DECISION_NODE$(\mathcal{T}_L, j, v, k)$
        $E' = \{(u, v) \mid u \in L, v \in R, \mathcal{T}_L'^{S(u)} \cap \mathcal{T}_L'^{S(v)} \neq \emptyset\}$
        loss$' \leftarrow |$MAXIMUM_MATCHING$(L, R, E')|$ ▷ Use matching size as loss
        if loss$'$ < best_loss **then**
            $\mathcal{T}_L^* \leftarrow \mathcal{T}'_L$
            best_loss $\leftarrow$ loss$'$
        end if
    end for
    $\mathcal{T}_L \leftarrow \mathcal{T}_L^*$ ▷ Keep the split with lowest loss
end for
$\mathcal{T}_L \leftarrow$ ROBUST_RELABELING$(X, y, \mathcal{T}_L)$ ▷ Relabel to set leaf predictions

B Cost Complexity Pruning vs. Robust Relabeling

In the paper we compared Cost Complexity Pruning and robust relabeling on the Pima-Indians-diabetes dataset. Below we give the same plots for all 10 datasets that we used in the paper.

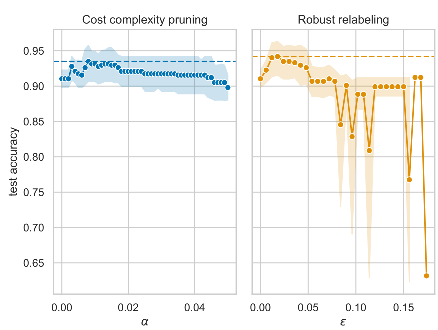


(a) Accuracy

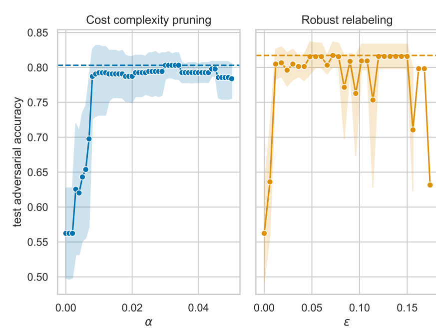


(b) Adversarial accuracy

Fig. 1: 5-fold cross validation on the Banknote-authentication dataset.

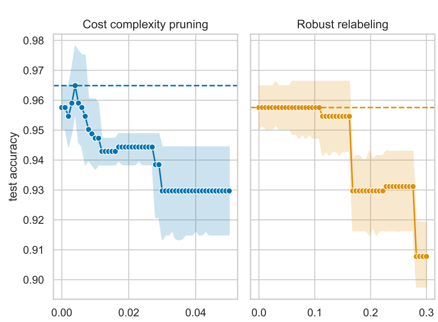


(a) Accuracy

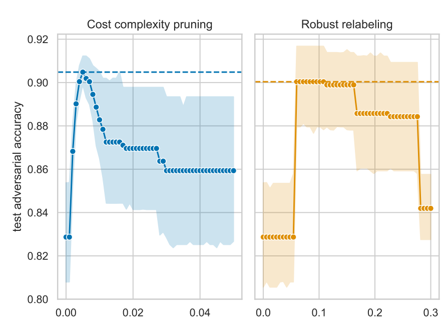


(b) Adversarial accuracy

Fig. 2: 5-fold cross validation on the Breast-cancer-diagnostic dataset.

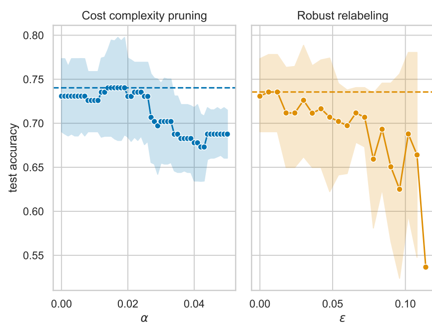


(a) Accuracy

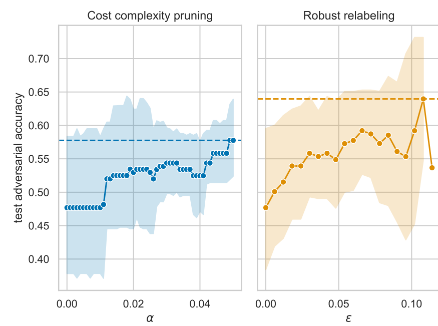


(b) Adversarial accuracy

Fig. 3: 5-fold cross validation on the Breast-cancer dataset.

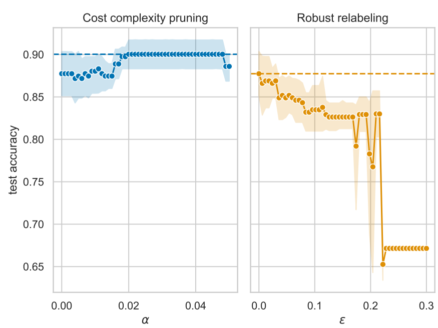


(a) Accuracy

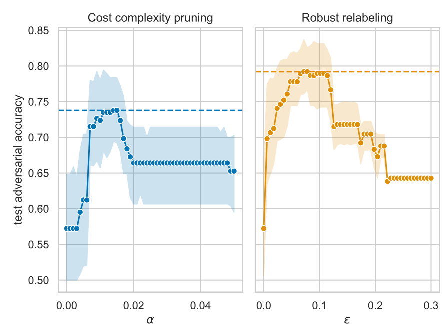


(b) Adversarial accuracy

Fig. 4: 5-fold cross validation on the Connectionist-bench-sonar dataset.

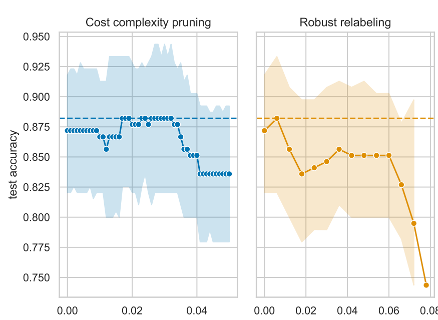


(a) Accuracy

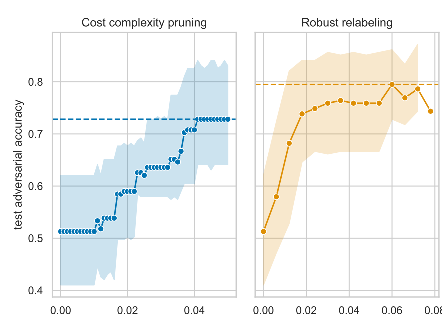


(b) Adversarial accuracy

Fig. 5: 5-fold cross validation on the Ionosphere dataset.

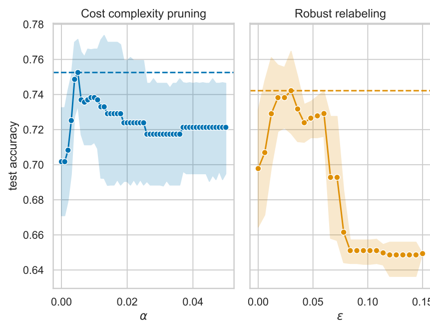


(a) Accuracy

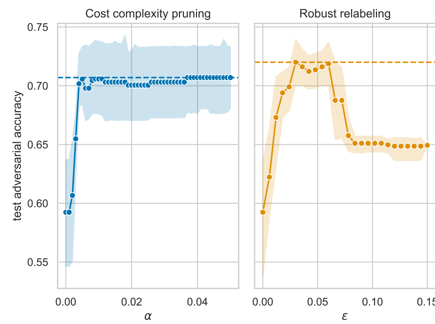


(b) Adversarial accuracy

Fig. 6: 5-fold cross validation on the Parkinsons dataset.

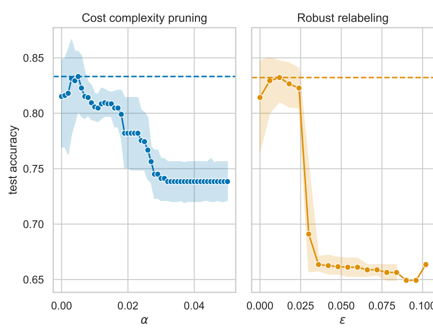


(a) Accuracy

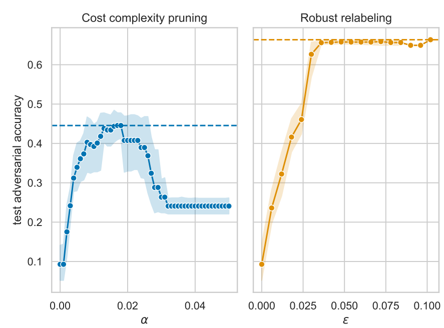


(b) Adversarial accuracy

Fig. 7: 5-fold cross validation on the Pima-Indians-diabetes dataset.

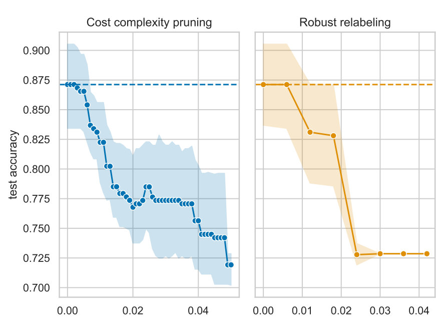


(a) Accuracy

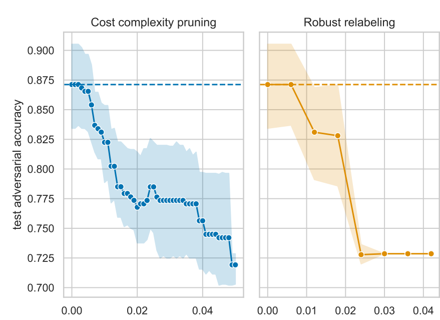


(b) Adversarial accuracy

Fig. 8: 5-fold cross validation on the Qsar-biodegradation dataset.

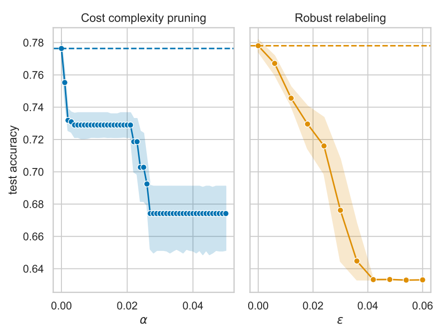


(a) Accuracy

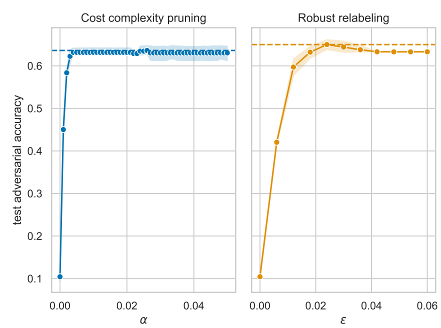


(b) Adversarial accuracy

Fig. 9: 5-fold cross validation on the Spectf-heart dataset.



(a) Accuracy



(b) Adversarial accuracy

Fig. 10: 5-fold cross validation on the Wine dataset.