Classification Trees and Random Forests for TKI Resistance Data

Muhammad Ali[1]

Abstract

In this report, we apply classification trees and random forests to a TKI resistance data set. We demonstrate how to build a single decision tree using a Gini-impurity split criterion, and then extend it to random forests. Out-of-bag (OOB) permutation-based variable importance is computed and compared to a non-random ensemble. Furthermore, we investigate 3-feature combinations using both data-based and structure-based approaches.

Classification Trees, Random Forests, OOB Importance

[1] *ma76193@student.uni-lj.si*

Introduction

Decision trees are popular for their interpretability and simplicity [1], yet a single tree tends to overfit, resulting in high variance. Ensemble methods—such as bagging and random forests—reduce this variance by averaging multiple trees built on bootstrap samples [2]. In this project, we analyze TKI resistance data using both a single decision tree (with `min_samples`=2) and a random forest. We further compute out-of-bag (OOB) permutation variable importance and extend this to evaluate 3-feature combinations using both data-based and structure-based approaches.

Methods

Data. The dataset consists of FTIR spectral samples from TKI-resistant cell lines. The training data is in `tki-train` and the testing data in `tki-test.tab`. Labels *Bcr-abl* and *Wild type* are mapped to 1 and 0, respectively.

Decision Tree. A `Tree` class is implemented to recursively split the training data based on the Gini impurity criterion. Splitting stops when a node is pure or when the sample count falls below `min_samples`=2.

Random Forest. A `RandomForest` class builds an ensemble of $n = 100$ trees using bootstrap samples. At each split, only a random subset of $\sqrt{d}$ features (where d is the total number of features) is considered. In-bag indices are stored with each tree so that out-of-bag (OOB) predictions can be made for variable importance evaluation.

Permutation-based Importance. For each feature, we compute the baseline OOB error and then permute the feature's values only in the OOB samples of the trees that used that feature. The increase in error is recorded as the feature's importance.

3-Feature Combinations. Two approaches are used:

1. **Data-based:** Rank features by single-feature importance, select the top k, and evaluate all 3-feature combinations on a random subset.
2. **Structure-based:** Traverse each tree to record which features are used in splits, and count the frequency of 3-feature combinations.

The best triple from each approach is then used to build a single decision tree for performance evaluation.

Results

Single Decision Tree. A single decision tree (with `samples=2`) achieves 0% training error and approximately 23% test error (see Table 1).

Table 1. Single Decision Tree Performance (min_samples=2).

	Training Error	Test Error
Error	0.00%	23.33%
SE	0.00%	5.46%

Random Forest. A random forest with 100 trees yields 0% training error and about 18% test error (Table 2), indicating a reduction in variance compared to the single tree.

Table 2. Random Forest Performance ($n = 100$).

	Training Error	Test Error
Error	0.00%	18.33%
SE	0.00%	5.00%

Permutation-based Importance. Figure 1 displays the top 20 features by absolute OOB importance for both the random forest and the non-random ensemble. In this version, feature names from the dataset legend are used.

3-Feature Combinations. The best triple selected by the data-based approach yields approximately 36.67% test error, while the structure-based approach yields about 25% test error. Figure 2 compares these two approaches.

Discussion

The single decision tree overfits the training data (0% error) but generalizes poorly (23% test error). The random forest, by averaging multiple trees, achieves lower test error (18%), demonstrating the benefit of bootstrap aggregation. OOB-based permutation importance identifies a small set of key features; most features exhibit negligible importance. Furthermore, the structure-based 3-feature combination outperforms the data-based approach, suggesting that interactions among features are important for TKI resistance classification.

Acknowledgments

We thank the course staff for providing the TKI data and test scripts, and for their valuable feedback.

References

[1] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, 2nd edition, 2009.

[2] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

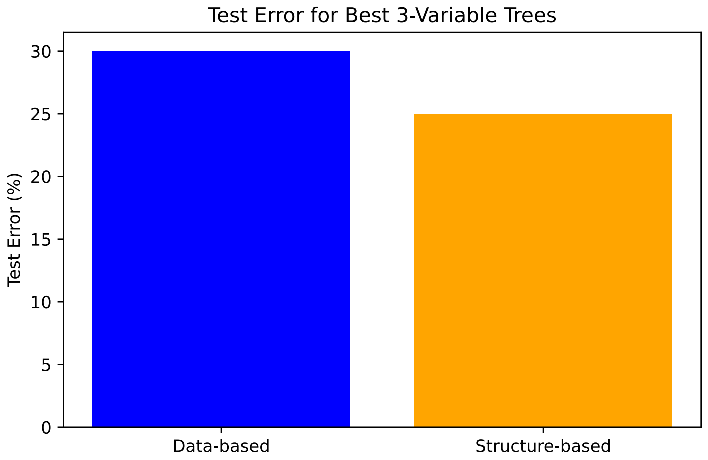

Figure 2. Test Error for Best 3-Variable Trees. Comparison between data-based and structure-based approaches.

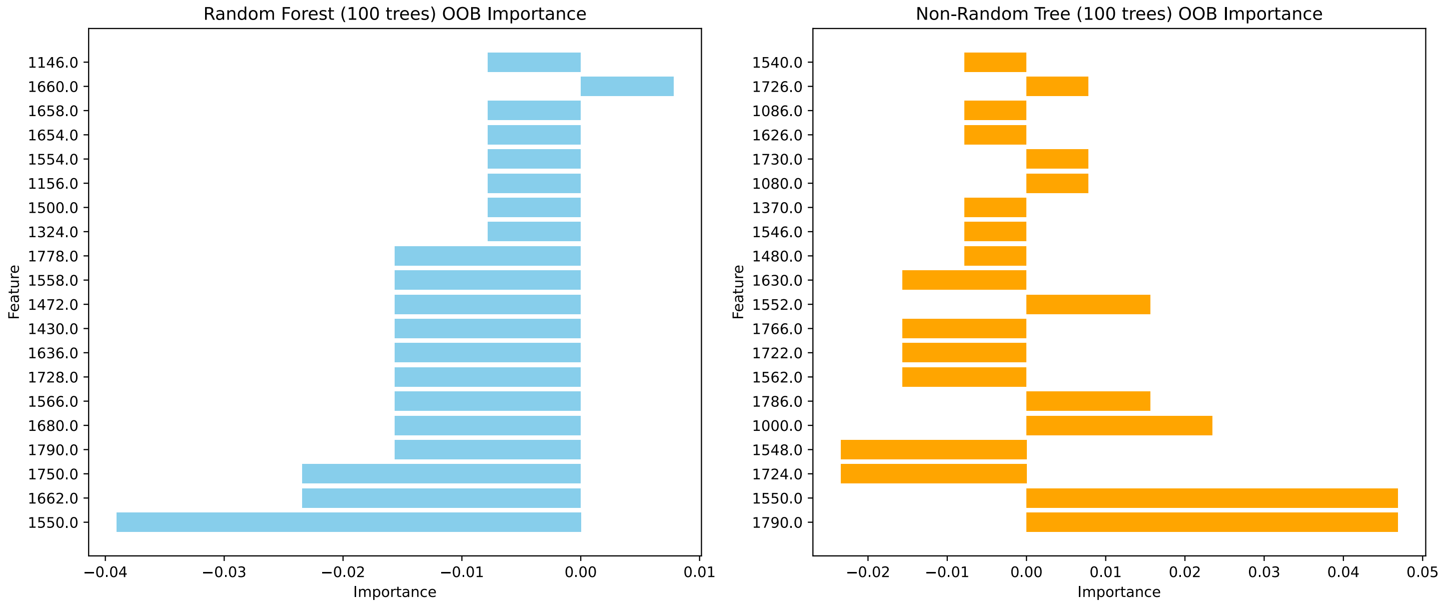

Figure 1. OOB Variable Importance. Top 20 features for Random Forest (left) and Non-Random Ensemble (right).