

Custom RandomForestClassifier Implementation

The provided Python code implements a custom version of scikit-learn's RandomForestClassifier with two key enhancements:

1. Weighted Predictions Based on Out-of-Bag (OOB) Loss:

- During the fitting process (fit method), for each tree in the random forest, the algorithm calculates the OOB loss using the mean squared error between the actual and predicted values for the OOB samples.
- It then converts this OOB loss into a weight for each tree by using an exponential function: $\exp(-\text{OOB_loss})$. This weight signifies the importance of each tree in making predictions.
- In the predict method, these weights are used to combine the predictions from all trees. Trees with lower OOB loss (and hence higher weight) have more influence on the final prediction.

2. Access to In-Bag and Out-of-Bag Data for Each Tree:

- The algorithm keeps track of which samples are used for training each tree (in-bag) and which are not (OOB). This is done by generating bootstrap indices (sampled with replacement) for in-bag samples and determining the OOB samples as those not included in the bootstrap sample.
- After the model is trained, you can access the in-bag and OOB samples for each tree. This is useful for custom analyses or understanding how each tree in the forest is interacting with the data.

These enhancements make the CustomRandomForestClassifier more flexible, allowing for weighted predictions based on tree performance (OOB loss) and providing insights into the data used for training each individual tree.