2025

4Geeks Academy: data science cohort 12

DAY 21: RANDOM FORESTS

TODO

RANDOM FORESTS

Model details, applications and hyperparameters

DECISION TREE PROJECT

Submit Decision Tree Project Tutorial (Decision Tree Algo. module) if you haven't done so already

RANDOM FOREST PROJECT

Work on Random Forest Project Tutorial (Random Forest Algo. module), plant to finish before class Wednesday

TOPICS

01 RANDOM FOREST MODEL

02 APPLICATIONS

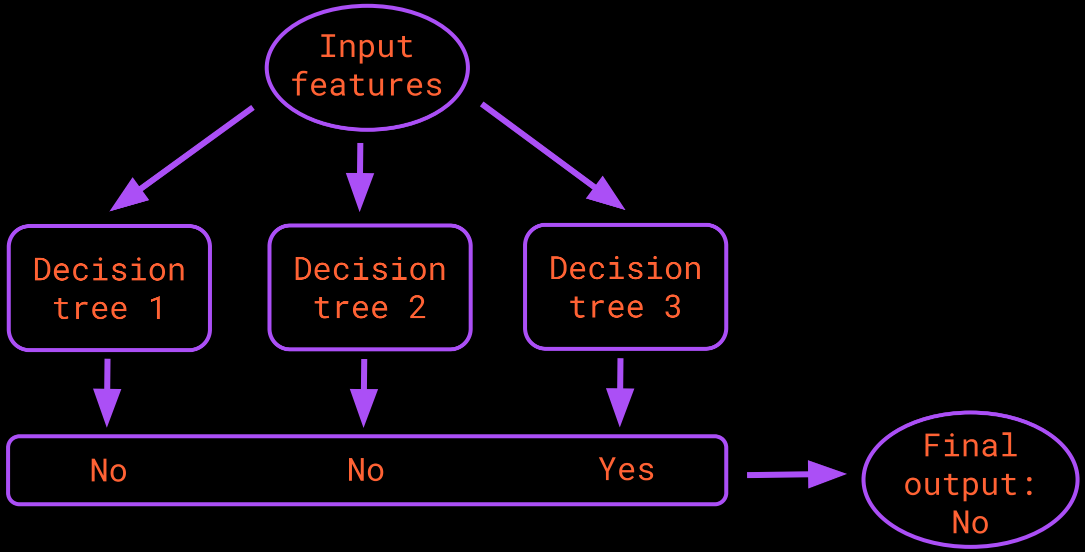
03 HYPERPARAMETERS

RANDOM FOREST MODEL

WHAT Ensemble of decision trees, uses majority voting or averaging to make predictions

WHY Less prone to overfitting than simple decision trees

HOW



APPLICATIONS

TYPES

- **Scikit-learn** `RandomForestClassifier()`: for classification problems
- **Scikit-learn** `RandomForestRegressor()`: for regression problems

PROS

- Less prone to overfitting than single decision tree
- Generally performs better than single decision tree
- Individual trees can be parallelized (fast)
- Retains advantages of decision trees vs linear models

CONS

- Regression trees don’t extrapolate outside of training label range
- Sensitive to imbalanced classes
- Can be computationally expensive for large datasets

HYPERPARAMETERS

ENSEMBLE

- `n_estimators`: number of individual trees to build
- `bootstrap`: True/False - use a randomly sampled subset of the data for each tree
- `oob_score`: 'out-of-bag' True/False - calculate generalization error from out-of-sample bootstrap data

TREE

- `max_depth`: how many splits deep will the tree go?
- `min_samples_split`: minimum sample remaining in a leaf to keep splitting
- `max_features`: maximum features to consider for splitting at each node
- `max_leaf_nodes`: maximum number of leaf nodes to create
- `min_impurity_decrease`: minimum gain in score to split

Optimizing these parameters can still help with overfitting!