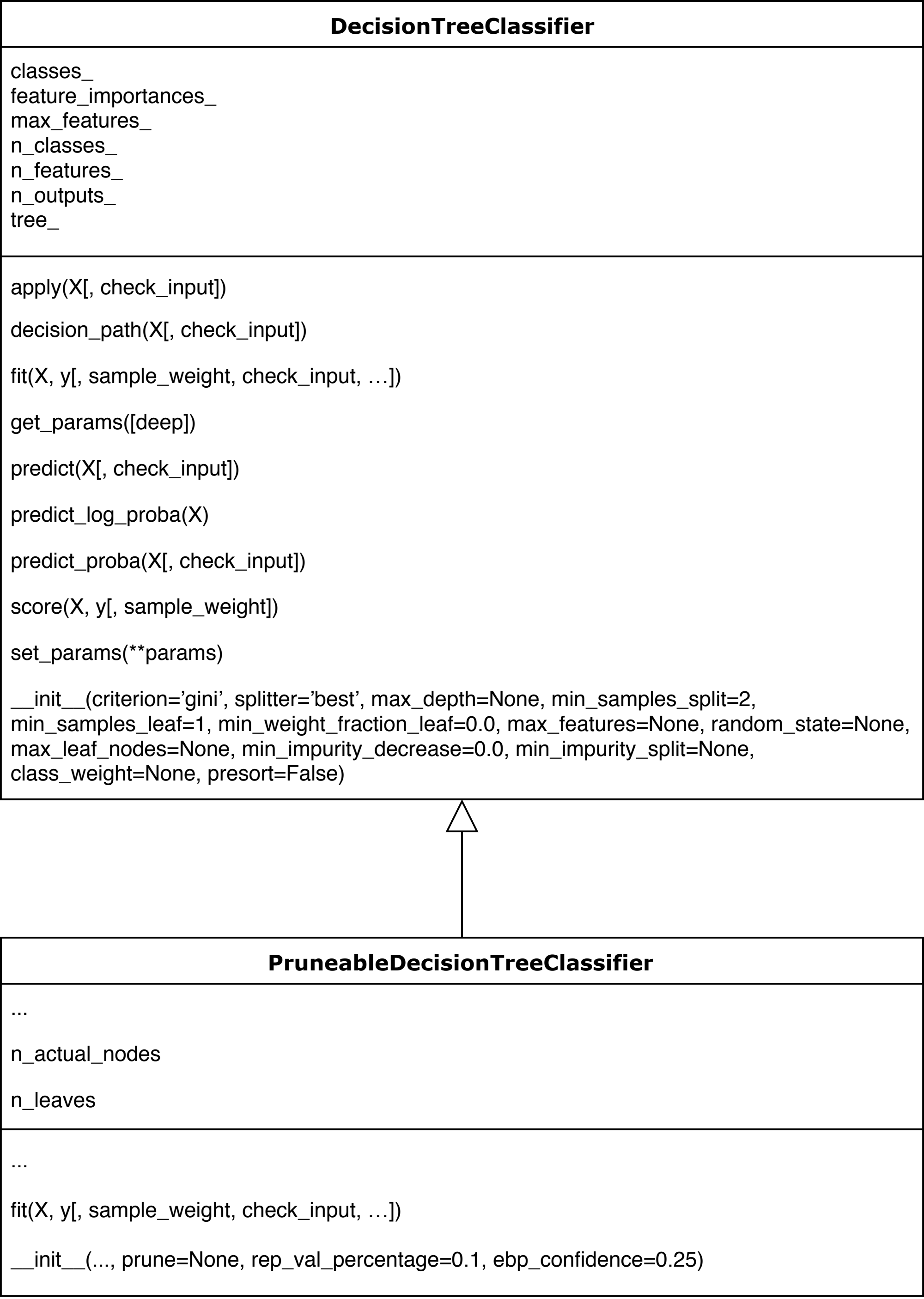

DecisionTreeClassifier

classes_
feature_importances_
max_features_
n_classes_
n_features_
n_outputs_
tree_

apply(X[, check_input])

decision_path(X[, check_input])

fit(X, y[, sample_weight, check_input, …])

get_params([deep])

predict(X[, check_input])

predict_log_proba(X)

predict_proba(X[, check_input])

score(X, y[, sample_weight])

set_params(**params)

__init__(criterion=’gini’, splitter=’best’, max_depth=None, min_samples_split=2,
min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features=None, random_state=None,
max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split=None,
class_weight=None, presort=False)

PruneableDecisionTreeClassifier

...

n_actual_nodes

n_leaves

...

fit(X, y[, sample_weight, check_input, …])

__init__(..., prune=None, rep_val_percentage=0.1, ebp_confidence=0.25)