

Training Day-9 Report:

get_depth

Return the depth of the decision tree.

The depth of a tree is the maximum distance between the root and any leaf.

Returns:

self.tree_.max_depth*int*

The maximum depth of the tree.

get_metadata_routing()

Get metadata routing of this object.

Returns:

routing*MetadataRequest*

A [MetadataRequest](#) encapsulating routing information.

get_n_leaves()

Return the number of leaves of the decision tree.

Returns:

self.tree_.n_leaves*int*

Number of leaves.

get_params(deep=True)

Get parameters for this estimator.

Parameters:

deep*bool, default=True*

If True, will return the parameters for this estimator and contained subobjects that are estimators.

Returns:

params*dict*

Parameter names mapped to their values.

predict(X, check_input=True)

Predict class or regression value for X.

For a classification model, the predicted class for each sample in X is returned.

For a regression model, the predicted value based on X is returned.

Parameters:

X*{array-like, sparse matrix} of shape (n_samples, n_features)*

The input samples. Internally, it will be converted to `dtype=np.float32` and if a sparse matrix is provided to a sparse `csr_matrix`.

check_inputbool, default=True

Allow to bypass several input checking. Don't use this parameter unless you know what you're doing.

Returns:

yarray-like of shape (n_samples,) or (n_samples, n_outputs)

The predicted classes, or the predict values.

predict_log_proba(X)

Predict class log-probabilities of the input samples X.

Parameters:

X{array-like, sparse matrix} of shape (n_samples, n_features)

The input samples. Internally, it will be converted to `dtype=np.float32` and if a sparse matrix is provided to a sparse `csr_matrix`.

Returns:

probandarray of shape (n_samples, n_classes) or list of n_outputs such arrays if n_outputs > 1

The class log-probabilities of the input samples. The order of the classes corresponds to that in the attribute [classes](#).

predict_proba(X, check_input=True)

Predict class probabilities of the input samples X.

The predicted class probability is the fraction of samples of the same class in a leaf.

Parameters:

X{array-like, sparse matrix} of shape (n_samples, n_features)

The input samples. Internally, it will be converted to `dtype=np.float32` and if a sparse matrix is provided to a sparse `csr_matrix`.

check_inputbool, default=True

Allow to bypass several input checking. Don't use this parameter unless you know what you're doing.

Returns:

probandarray of shape (n_samples, n_classes) or list of n_outputs such arrays if n_outputs > 1

The class probabilities of the input samples. The order of the classes corresponds to that in the attribute [classes](#).

score(X, y, sample_weight=None)

Return the mean accuracy on the given test data and labels.

In multi-label classification, this is the subset accuracy which is a harsh metric since you require for each sample that each label set be correctly predicted.

Parameters:

Xarray-like of shape (n_samples, n_features)

Test samples.

yarray-like of shape (n_sa