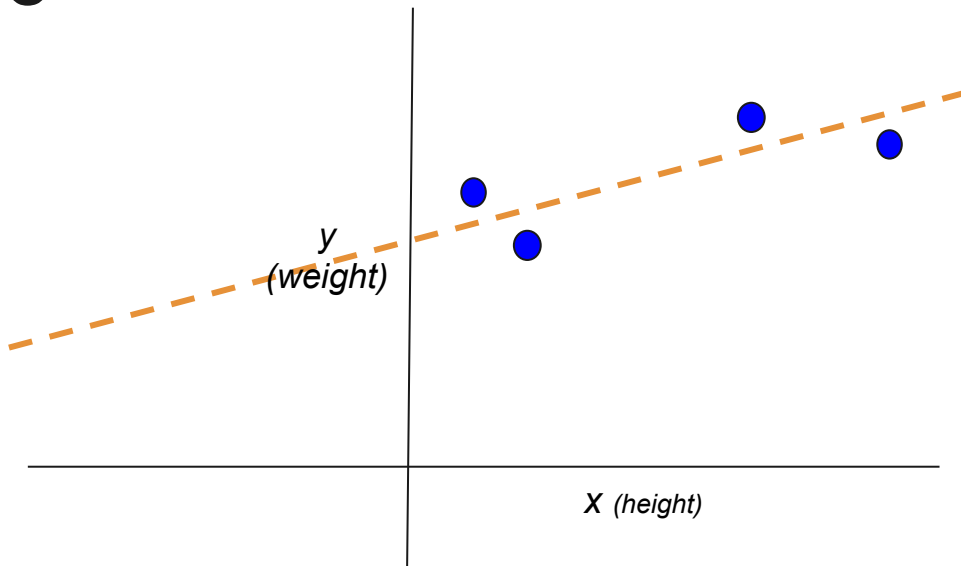




CIS 635 - Knowledge Discovery & Data Mining

HP Optimization

Regularization



So, essentially we are fitting a function; right?

Model

$$\hat{y} = \beta_0 + \beta_1 x$$

$$\Theta = \{\beta_0, \beta_1\}$$

Fitting Error

$$\epsilon = |\hat{y} - y|$$

Optimization function

$$E_{\Theta} = \frac{1}{2} \sum_{i=1}^N (\hat{y}_i - y_i)^2$$

$$\Theta^* = \operatorname{argmin}_{\Theta} E\{(x_i, y_i)\}_{i=1, \dots, N}$$

Regularization

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2$$

Model

$$\hat{y} = \beta_0 + \beta_1 x$$

$$\Theta = \{\beta_0, \beta_1\}$$

$$\epsilon = |\hat{y} - y|$$

Optimization function

$$E_{\Theta} = \frac{1}{2} \sum_{i=1}^N (\hat{y}_i - y_i)^2$$

$$\Theta^* = \operatorname{argmin}_{\Theta} E\{(x_i, y_i)\}_{i=1, \dots, N}$$

Regularization

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2$$

Essentially, the same formulation

Generally **ML** vs **Math** conventions

Model

$$\hat{y} = \beta_0 + \beta_1 x$$

$$\Theta = \{\beta_0, \beta_1\}$$

$$\epsilon = |\hat{y} - y|$$

Optimization function

$$E_{\Theta} = \frac{1}{2} \sum_{i=1}^N (\hat{y}_i - y_i)^2$$

$$\Theta^* = \operatorname{argmin}_{\Theta} E\{(x_i, y_i)\}_{i=1, \dots, N}$$

Regularization

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2$$

x : scalar
 \mathbf{x}, \mathbf{x} : vector
 \mathbf{X} : Matrix

Model

$$\hat{y} = \beta_0 + \beta_1 x$$

$$\Theta = \{\beta_0, \beta_1\}$$

Essentially, the same formulation

Generally ML vs Math conventions

$$\epsilon = |\hat{y} - y|$$

Optimization function

$$E_{\Theta} = \frac{1}{2} \sum_{i=1}^N (\hat{y}_i - y_i)^2$$

$$\Theta^* = \operatorname{argmin}_{\Theta} E\{(x_i, y_i)\}_{i=1, \dots, N}$$

Regularization

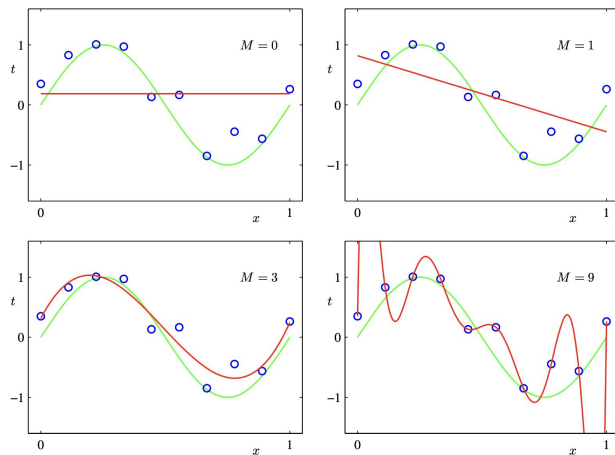


Table 1.1 Table of the coefficients w^* for polynomials of various order. Observe how the typical magnitude of the coefficients increases dramatically as the order of the polynomial increases.

	$M = 0$	$M = 1$	$M = 6$	$M = 9$
w_0^*	0.19	0.82	0.31	0.35
w_1^*		-1.27	7.99	232.37
w_2^*			-25.43	-5321.83
w_3^*			17.37	48568.31
w_4^*				-231639.30
w_5^*				640042.26
w_6^*				-1061800.52
w_7^*				1042400.18
w_8^*				-557682.99
w_9^*				125201.43



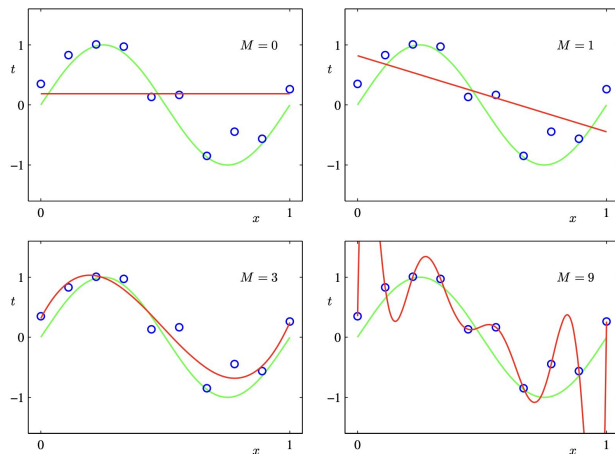
Regularization

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2$$

Regularizer

$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

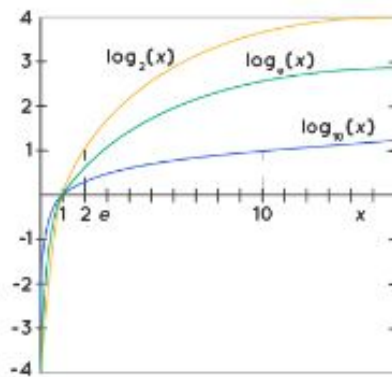
Regularization



$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

Table 1.2 Table of the coefficients w^* for $M = 9$ polynomials with various values for the regularization parameter λ . Note that $\ln \lambda = -\infty$ corresponds to a model with no regularization, i.e., to the graph at the bottom right in Figure 1.4. We see that, as the value of λ increases, the typical magnitude of the coefficients gets smaller.

	$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
w_0^*	0.35	0.35	0.13
w_1^*	232.37	4.74	-0.05
w_2^*	-5321.83	-0.77	-0.06
w_3^*	48568.31	-31.97	-0.05
w_4^*	-231639.30	-3.89	-0.03
w_5^*	640042.26	55.28	-0.02
w_6^*	-1061800.52	41.32	-0.01
w_7^*	1042400.18	-45.95	-0.00
w_8^*	-557682.99	-91.53	0.00
w_9^*	125201.43	72.68	0.01






Hyperparameter Optimization



Regularization

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2$$



Regularizer

$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$



LR Hyperparameter

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2$$

Regularizer

$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

λ (lambda)
is the
Hyperparameter a LR
model

LR Hyperparameter

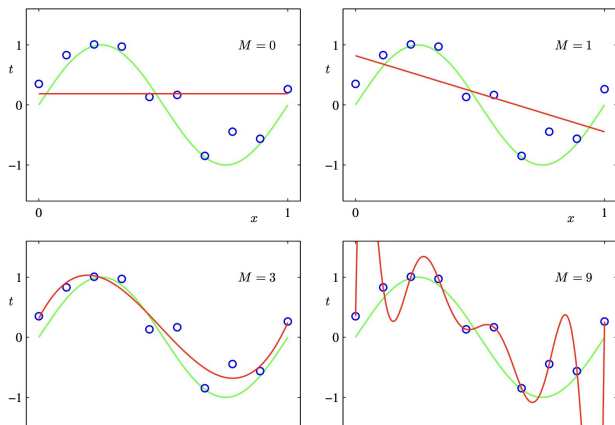
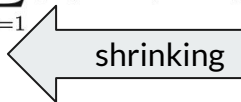


Table 1.1 Table of the coefficients \mathbf{w}^* for polynomials of various order. Observe how the typical magnitude of the coefficients increases dramatically as the order of the polynomial increases.

	$M = 0$	$M = 1$	$M = 6$	$M = 9$
w_0^*	0.19	0.82	0.31	0.35
w_1^*		-1.27	7.99	232.37
w_2^*			-25.43	-5321.83
w_3^*			17.37	48568.31
w_4^*				-231639.30
w_5^*				640042.26
w_6^*				-1061800.52
w_7^*				1042400.18
w_8^*				-557682.99
w_9^*				125201.43

$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$



LR Hyperparameter

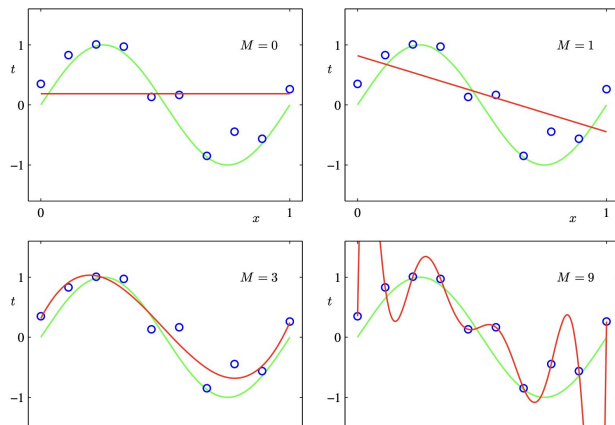


Table 1.1 Table of the coefficients \mathbf{w}^* for polynomials of various order. Observe how the typical magnitude of the coefficients increases dramatically as the order of the polynomial increases.

	$M = 0$	$M = 1$	$M = 6$	$M = 9$
w_0^*	0.19	0.82	0.31	0.35
w_1^*		-1.27	7.99	232.37
w_2^*			-25.43	-5321.83
w_3^*			17.37	48568.31
w_4^*				-231639.30
w_5^*				640042.26
w_6^*				-1061800.52
w_7^*				1042400.18
w_8^*				-557682.99
w_9^*				125201.43

$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

shrinking

→

growing

LR Hyperparameter

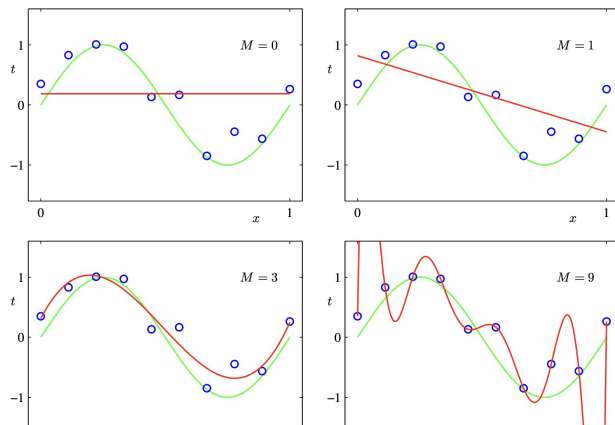


Table 1.1 Table of the coefficients w^* for polynomials of various order. Observe how the typical magnitude of the coefficients increases dramatically as the order of the polynomial increases.

	$M = 0$	$M = 1$	$M = 6$	$M = 9$
w_0^*	0.19	0.82	0.31	0.35
w_1^*		-1.27	7.99	232.37
w_2^*			-25.43	-5321.83
w_3^*			17.37	48568.31
w_4^*				-231639.30
w_5^*				640042.26
w_6^*				-1061800.52
w_7^*				1042400.18
w_8^*				-557682.99
w_9^*				125201.43

λ (lambda)
will help to counter
back
(a balance)

$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

shrinking

growing

LR Hyperparameter

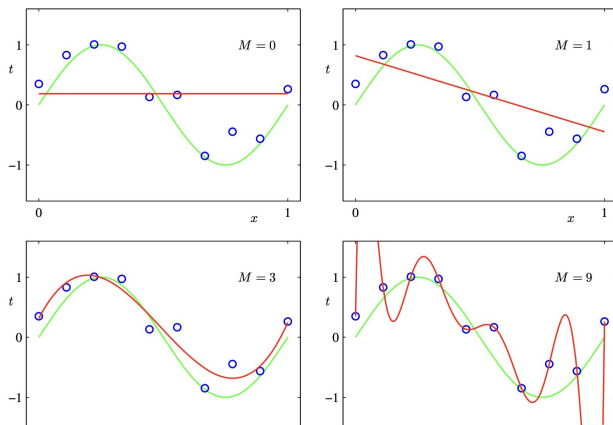


Table 1.1 Table of the coefficients \mathbf{w}^* for polynomials of various order. Observe how the typical magnitude of the coefficients increases dramatically as the order of the polynomial increases.

	$M = 0$	$M = 1$	$M = 6$	$M = 9$
w_0^*	0.19	0.82	0.31	0.35
w_1^*		-1.27	7.99	232.37
w_2^*			-25.43	-5321.83
w_3^*			17.37	48568.31
w_4^*				-231639.30
w_5^*				640042.26
w_6^*				-1061800.52
w_7^*				1042400.18
w_8^*				-557682.99
w_9^*				125201.43

λ (lambda)
will help to counter
back
(a balance)

$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

shrinking

growing

HP Opt is to find the best λ (lambda) on a validation set (cross validation)



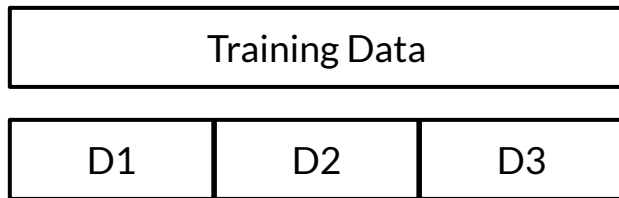
K-fold-cross validation

Training Data

*Find λ (lambda) that gives the best
k-fold-validation score*



K-fold-cross validation



3-fold-cv

*Find λ (lambda) that gives the best
k-fold-validation score*



K-fold-cross validation



3-fold-cv



Train



Validate

*Find λ (lambda) that gives the best
k-fold-validation score*



K-fold-cross validation



3-fold-cv

Train

Validate

*Find λ (lambda) that gives the best
k-fold-validation score*

K-fold-cross validation

Training Data

D1	D2	D3
----	----	----



3-fold-cv

Train

Validate

Find λ (lambda) that gives the best k -fold-validation score

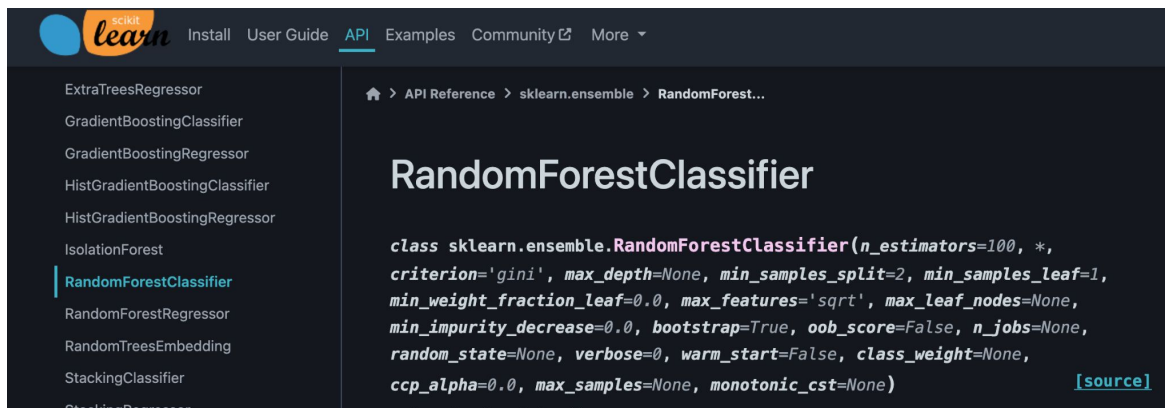


HPs of other models

- RF
- SVM
- NNs

HPs of other models

- RF (sklearn)



The screenshot shows the sklearn website's API reference for the `RandomForestClassifier` class. The left sidebar lists various sklearn models, with `RandomForestClassifier` highlighted. The main content area displays the class name and its constructor signature with default parameters.

sklearn

Install User Guide API Examples Community More

API Reference > sklearn.ensemble > RandomForest...

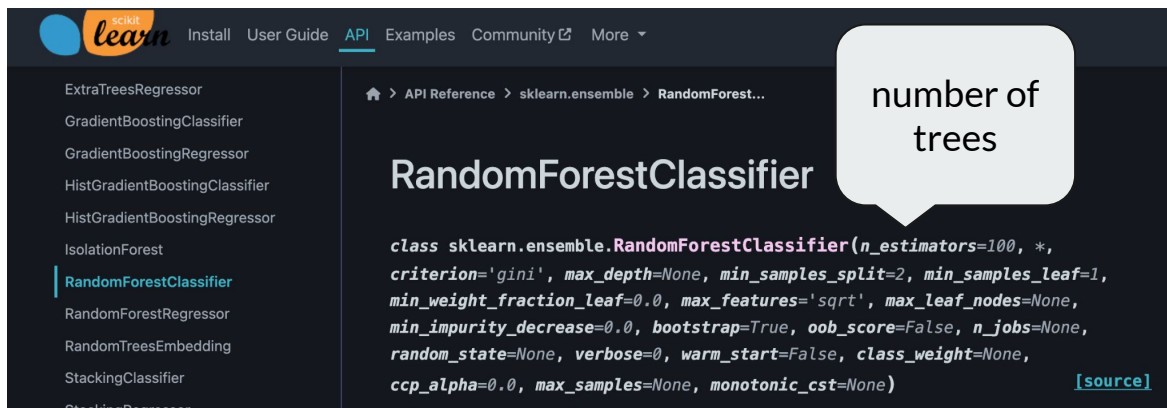
RandomForestClassifier

```
class sklearn.ensemble.RandomForestClassifier(n_estimators=100, *,
        criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1,
        min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None,
        min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None,
        random_state=None, verbose=0, warm_start=False, class_weight=None,
        ccp_alpha=0.0, max_samples=None, monotonic_cst=None)
```

[\[source\]](#)

HPs of other models

- RF (sklearn)



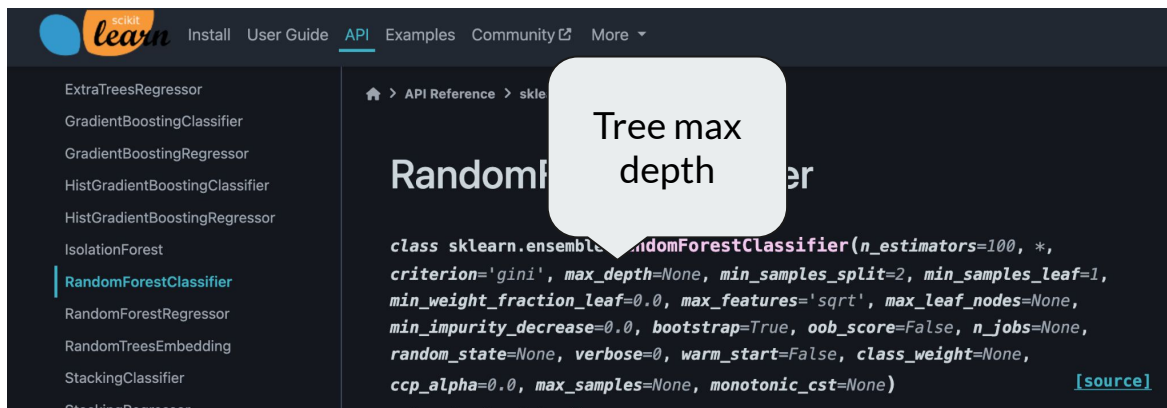
The screenshot shows the sklearn API reference page for `RandomForestClassifier`. The page has a dark theme. On the left is a sidebar with a list of sklearn models, including `ExtraTreesRegressor`, `GradientBoostingClassifier`, `GradientBoostingRegressor`, `HistGradientBoostingClassifier`, `HistGradientBoostingRegressor`, `IsolationForest`, `RandomForestClassifier` (highlighted with a blue bar), `RandomForestRegressor`, `RandomTreesEmbedding`, `StackingClassifier`, and `StackingRegressor`. The main content area shows the title `RandomForestClassifier` and its class signature: `class sklearn.ensemble.RandomForestClassifier(n_estimators=100, *, criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False, class_weight=None, ccp_alpha=0.0, max_samples=None, monotonic_cst=None)`. A blue link labeled `[source]` is at the bottom right. A white speech bubble with the text "number of trees" points to the `n_estimators` parameter in the class signature. The top navigation bar includes links for `Install`, `User Guide`, `API` (active), `Examples`, `Community`, and `More`.

```
class sklearn.ensemble.RandomForestClassifier(n_estimators=100, *,
criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1,
min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None,
min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None,
random_state=None, verbose=0, warm_start=False, class_weight=None,
ccp_alpha=0.0, max_samples=None, monotonic_cst=None)
```

number of trees

HPs of other models

- RF (sklearn)



The screenshot shows the sklearn API reference page for the `RandomForestClassifier` class. The left sidebar lists various sklearn models, with `RandomForestClassifier` highlighted. The main content area displays the class signature and its parameters. A callout bubble points to the `max_depth` parameter in the signature.

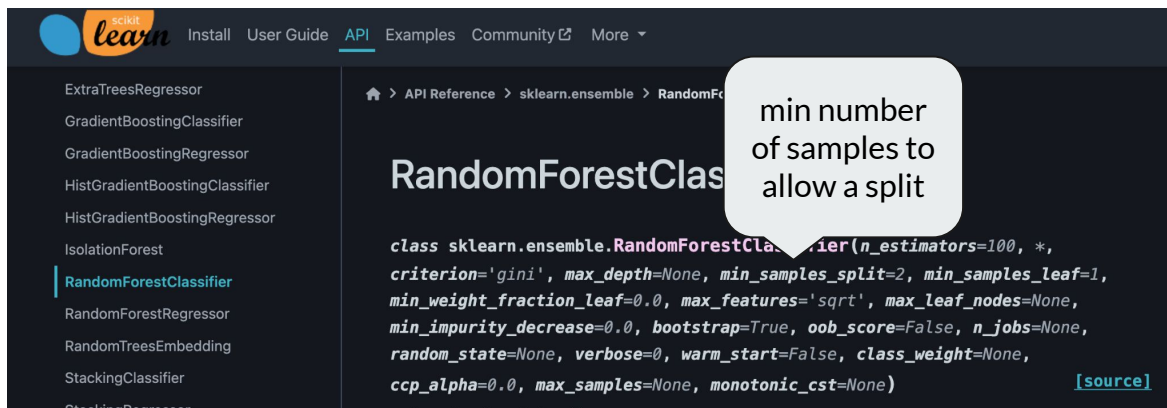
Tree max depth

```
class sklearn.ensemble.RandomForestClassifier(n_estimators=100, *,
        criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1,
        min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None,
        min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None,
        random_state=None, verbose=0, warm_start=False, class_weight=None,
        ccp_alpha=0.0, max_samples=None, monotonic_cst=None)
```

[\[source\]](#)

HPs of other models

- RF (sklearn)



The screenshot shows the scikit-learn API reference page for the `RandomForestClassifier` class. The left sidebar lists various models, with `RandomForestClassifier` highlighted. The main content area displays the class name and its constructor signature. A callout bubble points to the `min_samples_split` parameter in the signature.

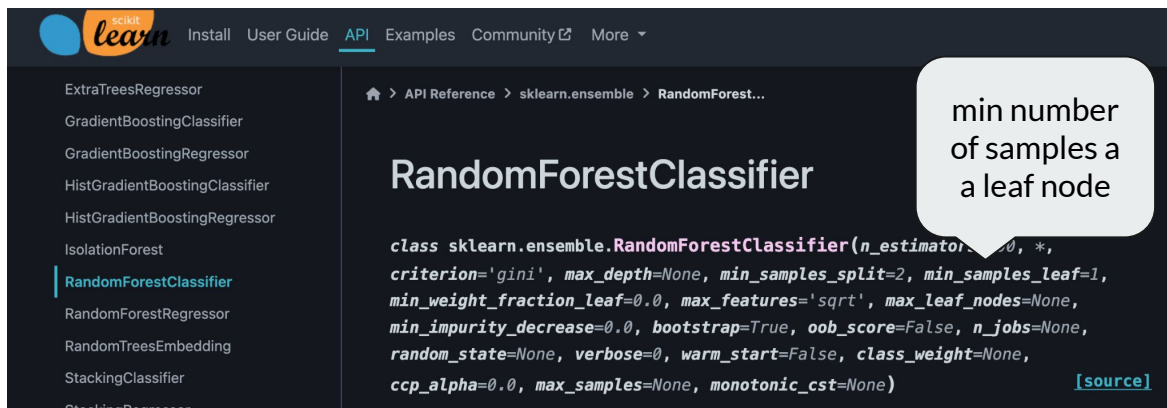
min number of samples to allow a split

```
class sklearn.ensemble.RandomForestClassifier(n_estimators=100, *,
        criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1,
        min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None,
        min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None,
        random_state=None, verbose=0, warm_start=False, class_weight=None,
        ccp_alpha=0.0, max_samples=None, monotonic_cst=None)
```

[\[source\]](#)

HPs of other models

- RF (sklearn)



The screenshot shows the sklearn API reference page for the `RandomForestClassifier` class. The page has a dark theme with a sidebar on the left containing a list of sklearn classes. The main content area displays the class name and its constructor signature. A callout box highlights the `min_samples_leaf` parameter.

min number of samples a leaf node

```
class sklearn.ensemble.RandomForestClassifier(n_estimators=100, *,
        criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1,
        min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None,
        min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None,
        random_state=None, verbose=0, warm_start=False, class_weight=None,
        ccp_alpha=0.0, max_samples=None, monotonic_cst=None)
```

[\[source\]](#)

HPs of other models

- RF (sklearn)



The screenshot shows the sklearn API reference page for the `RandomForestClassifier` class. The left sidebar lists various sklearn models, with `RandomForestClassifier` highlighted. The main content area shows the class signature and its parameters. A white callout box with the text "Others" is positioned over the class name and its parameters.

Others

```
class sklearn.ensemble.RandomForestClassifier(n_estimators=100, *,
criterion='gini', min_samples_split=2, min_samples_leaf=1,
min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None,
min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None,
random_state=None, verbose=0, warm_start=False, class_weight=None,
ccp_alpha=0.0, max_samples=None, monotonic_cst=None)
```

[\[source\]](#)



Notebook presentation

[Medical Insurance Cost Prediction](#)