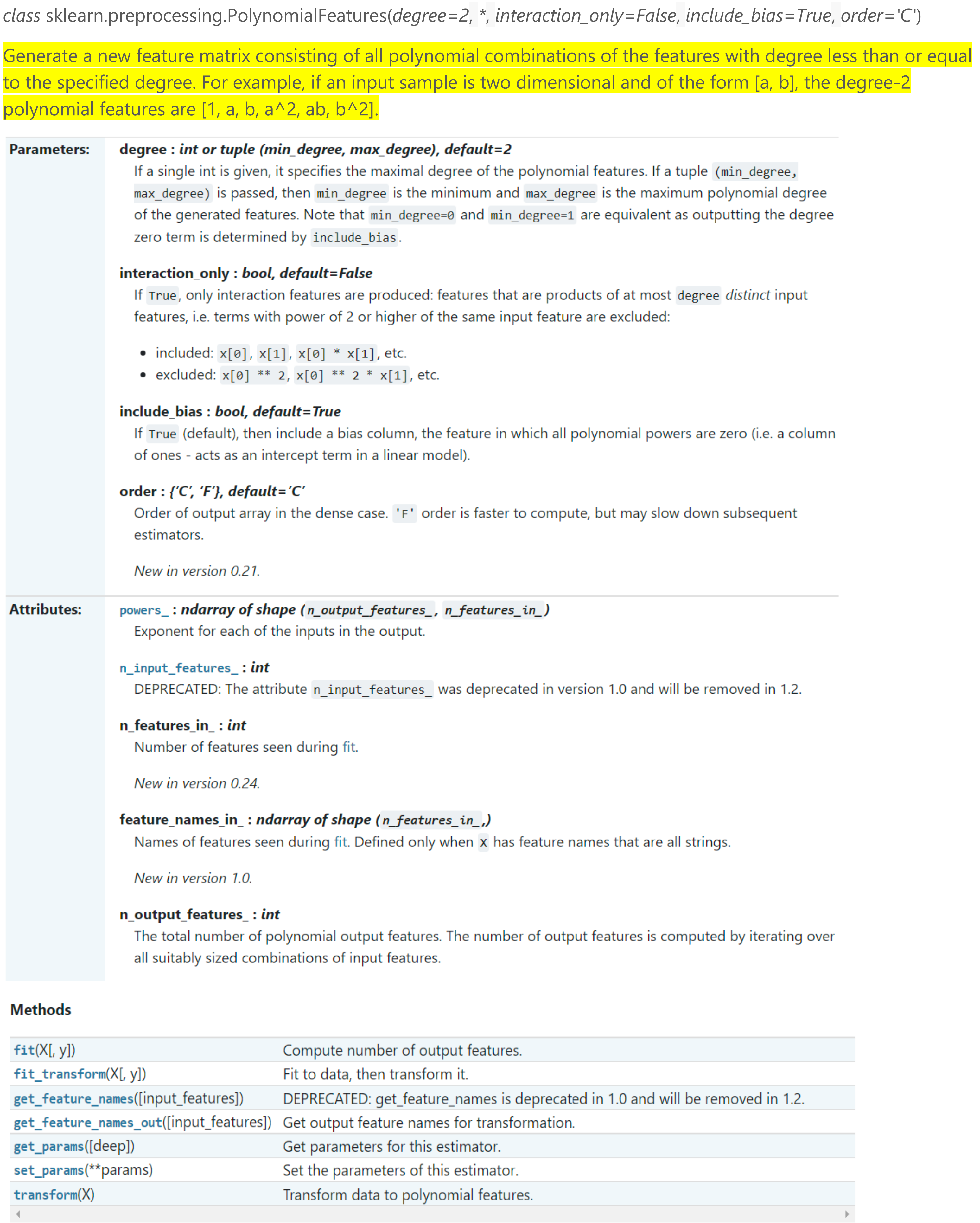
sklearn.preprocessing.PolynomialFeatures



class sklearn.preprocessing.PolynomialFeatures(*degree=2*, ***, *interaction_only=False*, *include_bias=True*, *order='C'*)

Generate a new feature matrix consisting of all polynomial combinations of the features with degree less than or equal to the specified degree. For example, if an input sample is two dimensional and of the form [a, b], the degree-2 polynomial features are [1, a, b, a^2, ab, b^2].

Parameters:

degree : *int or tuple (min_degree, max_degree), default=2*

If a single int is given, it specifies the maximal degree of the polynomial features. If a tuple `(min_degree, max_degree)` is passed, then `min_degree` is the minimum and `max_degree` is the maximum polynomial degree of the generated features. Note that `min_degree=0` and `min_degree=1` are equivalent as outputting the degree zero term is determined by `include_bias`.

interaction_only : *bool, default=False*

If `True`, only interaction features are produced: features that are products of at most `degree` *distinct* input features, i.e. terms with power of 2 or higher of the same input feature are excluded:

- included: `x[0]`, `x[1]`, `x[0] * x[1]`, etc.
- excluded: `x[0] ** 2`, `x[0] ** 2 * x[1]`, etc.

include_bias : *bool, default=True*

If `True` (default), then include a bias column, the feature in which all polynomial powers are zero (i.e. a column of ones - acts as an intercept term in a linear model).

order : *{'C', 'F'}, default='C'*

Order of output array in the dense case. `'F'` order is faster to compute, but may slow down subsequent estimators.

New in version 0.21.

Attributes:

`powers_` **: *ndarray of shape (`n_output_features_`, `n_features_in_`)***

Exponent for each of the inputs in the output.

`n_input_features_` **: *int***

DEPRECATED: The attribute `n_input_features_` was deprecated in version 1.0 and will be removed in 1.2.

n_features_in_ : *int*

Number of features seen during fit.

New in version 0.24.

feature_names_in_ : *ndarray of shape (`n_features_in_`,)*

Names of features seen during fit. Defined only when `X` has feature names that are all strings.

New in version 1.0.

n_output_features_ : *int*

The total number of polynomial output features. The number of output features is computed by iterating over all suitably sized combinations of input features.

Methods

`fit`(X[, y])	Compute number of output features.
`fit_transform`(X[, y])	Fit to data, then transform it.
`get_feature_names`([input_features])	DEPRECATED: get_feature_names is deprecated in 1.0 and will be removed in 1.2.
`get_feature_names_out`([input_features])	Get output feature names for transformation.
`get_params`([deep])	Get parameters for this estimator.
`set_params`(**params)	Set the parameters of this estimator.
`transform`(X)	Transform data to polynomial features.