SVR API

Aryaman Gautam

J001

The method of Support Vector Classification can be extended to solve regression problems. This method is called Support Vector Regression.

In Support Vector Regression, the straight line that is required to fit the data is referred to as hyperplane.

The model produced by support vector classification (as described above) depends only on a subset of the training data, because the cost function for building the model does not care about training points that lie beyond the margin. Analogously, the model produced by Support Vector Regression depends only on a subset of the training data, because the cost function ignores samples whose prediction is close to their target.

There are three different implementations of Support Vector Regression: SVR, NuSVR and LinearSVR. LinearSVR provides a faster implementation than SVR but only considers the linear kernel, while NuSVR implements a slightly different formulation than SVR and LinearSVR. See Implementation details for further details.

As with classification classes, the fit method will take as argument vectors X, y, only that in this case y is expected to have floating point values instead of integer values.

class sklearn.svm.SVR(*, kernel='rbf', degree=3, gamma='scale', coef0=0.0, tol=0.001, C=1.0, epsilon=0.1, shrinking=True, cache_size=200, verbose=False, max_iter=- 1)

Parameters

- kernel{‘linear’, ‘poly’, ‘rbf’, ‘sigmoid’, ‘precomputed’}, default=’rbf’

Specifies the kernel type to be used in the algorithm. If none is given, ‘rbf’ will be used. It is used to precompute the kernel matrix.

- degree : nt, default=3 : Degree of the polynomial kernel function (‘poly’).

- gamma{‘scale’, ‘auto’} or float, default=’scale’ : if ‘auto’, uses 1 / n_features.
- coef0 : float, default=0.0 It is only significant in ‘poly’ and ‘sigmoid’.
- tol : float, default=1e-3

Tolerance for stopping criterion.

- C : float, default=1.0

Regularization parameter. The strength of the regularization is inversely proportional to C. Must be strictly positive. The penalty is a squared l2 penalty.

- epsilon float, default=0.1

  It specifies the epsilon-tube within which no penalty is associated in the training loss function with points predicted within a distance epsilon from the actual value.

- shrinking bool, default=True

Whether to use the shrinking heuristic.

- cache_size float, default=200

Specify the size of the kernel cache (in MB).

- verbose bool, default=False

Enable verbose output. May not work properly in a multithreaded context.

- max_iter int, default=-1

Hard limit on iterations within solver, or -1 for no limit.

Attributes :

- class_weight_	ndarray of shape (n_classes,)
- coef_	ndarray of shape (1, n_features)
- dual_coef_	ndarray of shape (1, n_SV)
- fit_status_	int
- intercept_	ndarray of shape (1,)
- n_support_	ndarray of shape (n_classes,), dtype=int32
- shape_fit_	tuple of int of shape (n_dimensions_of_X,)
- support_	ndarray of shape (n_SV,)
- support_vectors_	ndarray of shape (n_SV, n_features)