SCIKIT-LEARN API SUMMARY

AVANI NARVEKAR J042

Linear Regression

LinearRegression fits a linear model with coefficients w = (w1, …, wp) to minimize the residual sum of squares between the observed targets in the dataset, and the targets predicted by the linear approximation.

PARAMETERS:
- fit_intercept: bool, default=True
- normalize: bool, default=False
- copy_X: bool, default=True
- n_jobs: int, default=None
- positive: bool, default=False

ATTRIBUTES:
- coef_: array of shape (n_features,) or (n_targets, n_features)
- rank_int
- singular_: array of shape (min(X, y),)
- intercept_: float or array of shape (n_targets,)

Ridge

Linear least squares with l2 regularization.
Minimizes the objective function:

$$||y - Xw||^2_2 + alpha * ||w||^2_2$$

This model solves a regression model where the loss function is the linear least-squares function and regularization is given by the l2-norm. Also known as Ridge Regression or Tikhonov regularization. This estimator has built-in support for multi-variate regression (i.e., when y is a 2d-array of shape (n_samples, n_targets)).

PARAMETERS:
- alpha{float, ndarray of shape (n_targets,)}, default=1.0
- fit_intercept: bool, default=True
- normalize: bool, default=False
- copy_X: bool, default=True
- max_iter: int, default=None
- tol: float, default=1e-3
- solver:{'auto', 'svd', 'cholesky', 'lsqr', 'sparse_cg', 'sag', 'saga'}, default='auto'

- random_state:int, RandomState instance, default=None

ATTRIBUTES:
- coef_:ndarray of shape (n_features,) or (n_targets, n_features)
- intercept_:float or ndarray of shape (n_targets,)
- n_iter_:None or ndarray of shape (n_targets,)

Lasso

Linear Model trained with L1 prior as regularizer (aka the Lasso)

The optimization objective for Lasso is:

```
(1 / (2 * n_samples)) * ||y - Xw||^2_2 + alpha * ||w||_1
```

PARAMETERS:
- alphafloat, default=1.0
- fit_interceptbool, default=True
- normalizebool, default=False
- precomputebool or array-like of shape (n_features, n_features), default=False
- copy_Xbool, default=True
- max_iterint, default=1000
- tolfloat, default=1e-4
- warm_startbool, default=False
- positivebool, default=False
- random_stateint, RandomState instance, default=None
- selection{'cyclic', 'random'}, default='cyclic'

ATTRIBUTES:
- coef_:ndarray of shape (n_features,) or (n_targets, n_features)
- dual_gap_:float or ndarray of shape (n_targets,)
- sparse_coef_:sparse matrix of shape (n_features, 1) or (n_targets, n_features)
- intercept_:float or ndarray of shape (n_targets,)
- n_iter_:int or list of int

Logistic Regression:

Logistic Regression (aka logit, MaxEnt) classifier.

In the multiclass case, the training algorithm uses the one-vs-rest (OvR) scheme if the 'multi_class' option is set to 'ovr', and uses the cross-entropy loss if the 'multi_class' option is set to 'multinomial'. (Currently the 'multinomial' option is supported only by the 'lbfgs', 'sag', 'saga' and 'newton-cg' solvers.)

PARAMETERS:
- Penalty : {'l1', 'l2', 'elasticnet', 'none'}, default='l2'
- dual : bool, default=False
- tol : float, default=1e-4
- C : float, default=1.0
- fit_intercept : bool, default=True
- intercept_scaling : float, default=1
- class_weightdict or 'balanced', default=None
- random_state : int, RandomState instance, default=None
- solver{'newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga'},
- default='lbfgs'
- max_iter : int, default=100
- multi_class{'auto', 'ovr', 'multinomial'}, default='auto'
- verbose : int, default=0
- warm_start : bool, default=False
- n_jobs : int, default=None
- l1_ratio : float, default=None

ATTRIBUTES:
- classes_ndarray of shape (n_classes,)
- coef_ndarray of shape (1, n_features) or (n_classes, n_features)
- intercept_ndarray of shape (1,) or (n_classes,)
- n_iter_ndarray of shape (n_classes,) or (1,)