NAME : SAMEEKSHA KINI

ROLL NO : J029

1)**RIDGE:**

Ridge regression penalizes the model based on the sum of squares of magnitude of the coefficients.

Alpha-Regularization strength; must be a positive float. Regularization improves the conditioning of the problem and reduces the variance of the estimates. Larger values specify stronger regularization.

Fit(X,y)-fits the regression model

Predict(x)-predicting using the linear model.

Score(X,y)-returns the coefficient of determination

CODE:

sklearn.linear_model.Ridge(alpha=1.0, *, fit_intercept=True, normalize=False, copy_X=True, max_iter=None, tol=0.001, solver='auto', random_state=None)

2)**LASSO:**

LASSO regression penalizes the model based on the sum of magnitude of the coefficients.

Fit(X,y)-fit model with coordinate descent

Predict(X)-predict using linear model

Score(X,y)-returns the coefficient of determination of the prediction

CODE:

sklearn.linear_model.Lasso(alpha=1.0, *, fit_intercept=True, normalize=False, precompute=False, copy_X=True, max_iter=1000, tol=0.0001, warm_start=False, positive=False, random_state=None, selection='cyclic')

3) **LINEAR REGRESSION:**

Linear Regression is a model which assumes there's a linear relationship between the input variables X and the single output variable y, and y can be calculated from a linear combination of X.

Fit(X, y)- fit the linear model.

Predict(X)-predict using linear model.

Score(X,y)-returns the coefficient of determination R^2 of the prediction.

CODE:

sklearn.linear_model.LinearRegression(*, fit_intercept=True, normalize=False, copy_X=True, n_jobs=None, positive=False)

4) LOGISTIC REGRESSION:

Logistic regression is used to predict a dependent variable by analyzing the relationship between a given set of independent variables. It predicts the output of a categorical dependent variable.

Fit(X,y)-fit the model according to the given training data

Predict(x)-predict class labels

Score(X,y)-returns mean accuracy on the given test data and label

CODE:

```
sklearn.linear_model.LogisticRegression(penalty='l2', *, dual=False, tol=0.0001, C=1.0, fit_intercept=True, intercept_scaling=1, class_weight=None, random_state=None, solver='lbfgs', max_iter=100, multi_class='auto', verbose=0, warm_start=False, n_jobs=None, l1_ratio=None)
```