Soham Shah J059

Support Vector Classification

API:

class `sklearn.svm.`**SVC**(**, C=1.0, kernel='rbf', degree=3, gamma='scale', coef0=0.0, shrinking=True, probability=False, tol=0.001, cache_size=200, class_weight=None, verbose=False, max_iter=- 1, decision_function_shape='ovr', break_ties=False, random_state=None*)

C- the regularization parameter

kernel- will decide the type of kernel to use for the algo. The default one we use is 'radial basis function kernel.'

Class weight – Setting the parameter C of some class 'I'. By default it will give all the classes the value of 1.

Decision_function_state- can return two types of function shapes(either ovr or ovo).

SVM working

SVM is one of the classic ML algorithm used mainly for classification approach. It constructs a hyperplane that will separate two classes and gets the optimal hyperplane which will minimize the error. The core idea is to find a maximum marginal hyperplane(MMH) that best divides the dataset.

It's a linear classifier at it's core but unlike Logistic regression that will use the concept of best fit line, Support Vector Machines uses the concept of Margins to come up with predictions.

The idea behind the SVM classifier is to come up with a hyper-lane in an N-dimensional space that divides the data points belonging to different classes.

However, this hyper-pane is chosen based on margin as the hyperplane providing the maximum margin between the two classes is considered.

These margins are calculated using data points known as Support Vectors. Support Vectors are those data points that are near to the hyper-plane and help in orienting it.

Advantages:

- Works well with unstructured or semi-structured data.
- The kernel trick is the real strength of SVM
- Scales well for higher dimension

Disadvantages:

- Choosing a useful kernel function is difficult
- The hyperparameters aren’t easy to fine tune
- Long training time for larger datasets