

Centre for ARTIFICIAL INTELLIGENCE
TKM COLLEGE OF ENGINEERING, Kollam
Semester II
Advanced Computing Lab

TUTORIAL QUESTION: SET 2 (25.05.21)

Instructions

1. Visit <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>
 2. Prepare & add solutions next to questions.
 3. Save your file name like: ebin_lab_tutorial_2.docx & attach back in ETlab.
 4. Be ready to attend Viva (SVM & Following questions) in the next lab hour.
-

1. Study and present relevance of various parameters available in sklearn - SVC function.

1) C: float, default=1.0

Regularization parameter-It is used to find the strength of the regularization, it is inversely proportional to C.

2) kernel: {'linear', 'poly', 'rbf', 'sigmoid', 'precomputed'}, default='rbf'

It specifies the kernel type to be used in the algorithm. It may be 'linear', 'poly', 'rbf', 'sigmoid', 'precomputed' or a callable. 'rbf' is the default type. If a callable is given it is used to pre-compute the kernel matrix from data matrices with dimension (n_samples, n_samples).

3) degree: int, default=3

It is used to find the degree of polynomial kernel and not used for all other kernels.

4) gamma: {'scale', 'auto'} or float, default='scale'

It is the Kernel coefficient for 'rbf', 'poly' and 'sigmoid'.

- if gamma='scale' (default) is passed then it uses $1 / (n_features * X.var())$ as value of gamma,
- if 'auto', uses $1 / n_features$.

The default value of gamma changed from 'auto' to 'scale'.

5) coef0: float, default=0.0

It is an independent term in kernel function, and only significant in 'poly' and 'sigmoid' kernels.

6) shrinking: bool, default=True

It is used to shorten the training time when the number of iterations is large.

7) probability: bool, default=False

When probability is set to True, class membership probability estimates are enabled.

8) **cache size: float, default=200**

It specifies the size of the kernel cache (in MB).

9) **class_weight: dict or 'balanced', default=None**

It sets the parameter C of class i to class_weight[i]*C for SVC. If weight is not given, all classes are supposed to have weight one.

10) **verbose: bool, default=False**

It is used to enable verbose output. It takes the advantage of a per-process runtime setting in libsvm that, if enabled, may not work properly in a multithreaded context.

11) **max_iter: int, default=-1**

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

12) **decision_function_shape: {'ovo', 'ovr'}, default='ovr'**

It is not used in binary classification, and it is used whether to return a one-vs-rest ('ovr') decision function of shape (n_samples, n_classes) as all other classifiers, or the original one-vs-one ('ovo') decision function of libsvm which has shape (n_samples, n_classes * (n_classes - 1) / 2). However, one-vs-one ('ovo') is always used as multi-class strategy.

Default decision_function_shape is 'ovr'.

13) **break_ties: bool, default=False**

If break_ties='true', decision_function_shape='ovr', and number of classes > 2, predict will break ties according to the confidence values of decision_function; otherwise the first class among the tied classes is returned. It comes at a relatively high computational cost compared to a simple predict.

14) **random_state: int, RandomState instance or None, default=None**

It controls the pseudo random number generation for shuffling the data for probability estimates. Only Pass an int for reproducible output across multiple function calls, ignored when probability is False.

2. Study and prepare a NOTE (Intro/Definition -2/3 lines, Over-coming any problems? – 1/2 lines, Advantages, Disadvantages, Applications, Equation if any) separately on LINEAR, POLY, RBF, SIGMOID kernels.

The kernel functions are used to map the original dataset (linear/nonlinear) into a higher dimensional space with view to making it linear dataset. The linear, polynomial, RBF or Gaussian and sigmoid kernel are simply different in case of making the hyper plane decision boundary between the classes.

Linear Kernel:

Linear Kernel is used when the data is Linearly separable, that is, it can be separated using a single Line. It mostly used for text classification.

Its advantages are:

- Linear kernel is faster
- Less parameters to optimize

Disadvantages are

- It is not suitable for large data sets.
- It does not perform very well when the data set has more noise i.e. target classes are overlapping.
- In cases where the number of features for each data point exceeds the number of training data samples, it becomes underperform.

Polynomial Kernel:

It represents the similarity of vectors in training set of data in a feature space over polynomials of the original variables used in kernel

It is popular in image processing.

$$K(x, y) = \tanh(\gamma \cdot x^T y + r)^d, \gamma > 0$$

RBF Kernel:

It is a general-purpose kernel; used when there is no prior knowledge about the data.

The RBF kernel function for two points X_1 and X_2 computes the similarity or how close they are to each other. This kernel can be mathematically represented as follows:

$$K(x, y) = e^{-(\gamma ||x - y||^2)}$$

for

$$\gamma > 0$$

Sometimes parametrized using:

$$\gamma = 1/2\sigma^2$$

Usually linear and polynomial kernels are less time consuming and provides less accuracy than the rbf or Gaussian kernels.

Sigmoid Kernel:

This function is equivalent to a two-layer, perceptron model of neural network, which is used as activation function for artificial neurons.

$$K(x, y) = \tanh(\gamma \cdot x^T y + r)$$

.