Machine Learning Assignment – 1

Team -12

1601-25-742-018 K. Sai Pranav

1601-25-742-022 E. Vinay

1601-25-742-016 K. Shiva Kumar

1Q. List and briefly explain about various kernel functions of SVM

Ans. The Support Vector Machines find the optimal hyperplane to separate classes by projecting the input data into a high-dimensional feature space with the help of kernel functions. Due to these kernel functions, SVM can handle data that is both linearly and non-linearly separable. They help in mapping data points into a space where separation becomes easier to do by assessing the similarity between them. Some commonly used kernel functions in SVM are as follows:

1. Linear Kernel:

Formula: K(x,y)=xTyK(x, y) = x^T yK(x,y)=xTy

The linear kernel is the most straightforward and fundamental kernel function. It finds its application when data is linearly separable, or, in other words, it means that by drawing a straight line-a so-called hyperplane-the classes can be effectively separated. This kernel is very efficient computationally and works fine for high-dimensional datasets like those in text classification or document categorization.

2. Polynomial Kernel:

Formula: K(x,y)=(xTy+c)dK(x, y) = (x^T y + c)^dK(x,y)=(xTy+c)d

This kernel allows nonlinear separation by converting data into higher polynomial dimension space. Here, c is a constant, and d is the degree of the polynomial. It is useful when interaction between features is of prime significance because it captures complex relationships amongst the features.

3. Radial Basis Function (RBF) / Gaussian Kernel:

Formula: K(x,y)=exp(−γ‖x−y‖2)

The RBF kernel is the most used kernel in SVM. It performs well in complex situations and when the data is nonlinear. This parameter gamma (????γ) gives an idea of how far the effect of a training example goes. When ????γ is low, the boundaries are smooth. When ????γ is high, the model might overfit.

4. Sigmoid Kernel:

Formula: K(x,y)=tanh(αxTy+c)

This kernel works somewhat like the activation function in neural networks. It behaves in some cases like a two-layer neural network. It can sometimes help with classification problems, but in case of inappropriate setting of parameters, it may be unstable.

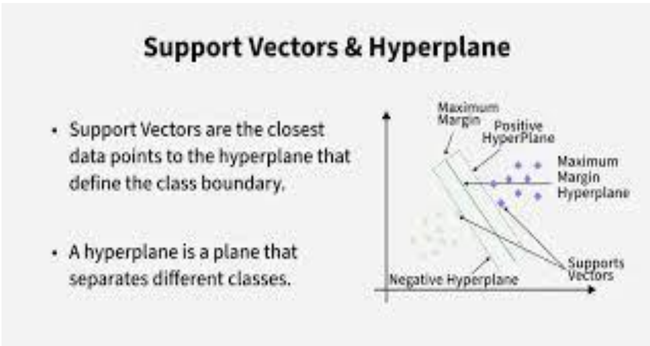
5. Laplacian Kernel:

Formula: $K(x,y)=\exp(-\gamma\|x-y\|)$

It is like the RBF kernel, but it uses L1 distance instead of L2 distance. This makes it less sensitive to noise and outliers. It makes the judgment limits crisper and is better when the data has big differences. 6. Exponential Kernel: Formula: $K(x,y)=\exp(-\gamma\|x-y\|2)$ This kernel is another variety of the RBF kernel and creates smooth boundaries of decisions. It works well for continuous feature data that has to be separated smoothly. 7. Custom Kernel: Standard kernels may not always give the performance that you want. In such cases, users may create their own custom kernel function using their own knowledge in the field or specific similarity metrics. In general, custom kernels are helpful in scientific or image-based tasks. Conclusion: These kernel functions can make SVM much better in the classification of complex data by making non-linear correlations linear in multi-dimensional spaces. The selection of an appropriate kernel depends on the nature of the data set, processing requirements, and the particular problem at hand.

2Q. How does Support Vector Machine work?

Ans. The Support Vector Machine, or SVM, is a type of supervised learning algorithm that is used mainly for classification purposes. The key idea underlying the SVM is to find the clear boundary separating different classes. Instead of just drawing any line or surface, the best possible dividing line, known as a hyperplane in SVM terms, should leave the most space between the classes. This space is called the margin; the greater the margin, the better performance and accuracy in general.



When plotting data points in a feature space, SVM attempts to create a boundary that separates categories as cleanly as possible. In 2D, this looks like a line; in 3D, it is a plane, and in higher dimensions, it is a hyperplane. The algorithm ensures this hyperplane is placed in a manner to increase the distance from the closest points of both classes.

- The data points that lie nearest to the decision boundary are called support vectors. These points are very important because:
  - They determine the position and the angle of the hyperplane.
  - These features have the most influence on the model.
  - Removing any of them would change the decision boundary.

All the other points are less influential, thus making SVM efficient and avoiding overfitting.

- Real-world data is rarely perfectly separable. To handle such cases, SVM uses a soft margin approach that: Allows some points to be on the wrong side of the boundary, Introduces: Slack variables, A penalty parameter (C)

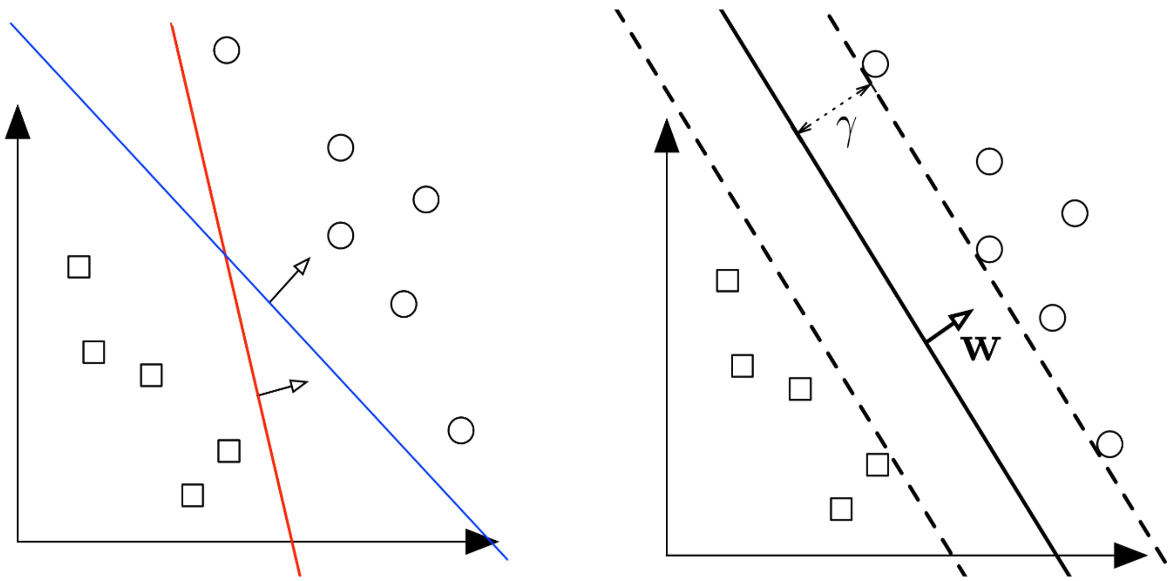
The balance between a wider margin and reducing the classification errors depends on the value of C.

Not every data set can be separated with a straight line. For this, the kernel trick is adopted by SVM. The data is transformed into a higher dimensionality space such that a linear separation becomes possible. Common kernel functions are: Linear ,Polynomial, Radial Basis Function (RBF), Sigmoid

In other words, the Support Vector Machine finds the best possible boundary that separates the data into distinct classes with the maximum possible margin. The closest points to this boundary, called support vectors, define this boundary. The introduction of soft margins and kernel functions also allows SVM to handle noisy and nonlinear data, thus making it one of the most versatile and powerful algorithms in machine learning.

3Q. What is dual problem for SVM?

Ans. A Support Vector Machine (SVM) is an example of a supervised machine learning model and can also be used on occasion for regression tasks. The goal of an SVM is to find the maximum margin separating hyperplane between two classes. Instead of looking to create separate areas for data items, as other algorithms do, an SVM attempts to create the largest area of the greatest separation possible between two classes of items. Support vectors are considered by SVM when determining the best separation boundaries of training data. These points represent the most critical pieces of information in the training data because they determine where the "optimal" decision boundary lies. A training data set contains feature vectors x_i and their corresponding class labels y_i, where $y_i \in \{-1, +1\}$, and SVM assumes that there exists a linear separation boundary in some feature space. The hyperplane separating two classes is mathematically described as: $w^T * x + b = 0$, w being the "weight vector" that indicates the direction of the hyperplane, and b being the shift of the hyperplane.



To guarantee correct classification, SVM enforces the constraint on the data such that every sample in the data satisfies the condition

$y_i(w^T x_i + b) \geq 1$ for all i

This ensures that all training points are correctly separated by the hyperplane. The margin is equal to $2 / ||w||$, so to maximize the margin we need to minimize the norm of the weight vector. Thus, the primal optimization problem of SVM becomes:

Minimize $1/2 \, ||w||^2$

subject to $y_i(w^T x_i + b) \geq 1$,, for all i.

This is a convex optimization problem, but solving it directly becomes difficult if the data is high-dimensional or if kernel functions are needed. Because of this, the SVM is transformed into another form called the dual problem, which has numerous advantages. This dual formulation uses the so-called kernel trick to perform the non-linear classification task, works efficiently for high-dimensional data, and depends upon dot products rather than direct features themselves.

We use Lagrange multipliers to derive the dual problem. For every constraint, we introduce a multiplier $\alpha_i \geq 0$ and form the Lagrangian:

$L(w, b, \alpha) = 1/2 \, ||w||^2 - \Sigma \, \alpha_i \, [y_i(w^T x_i + b) - 1]$

Now we differentiate the Lagrangian, with respect to w and b, and set them equal to zero in order to find the optimal solution:

$\partial L/\partial w = 0 \rightarrow w = \Sigma \, \alpha_i \, y_i \, x_i$

$\partial L/\partial b = 0 \rightarrow \Sigma \, \alpha_i \, y_i = 0$

Substituting these back into the Lagrangian gives the dual problem of SVM:

maximize $W(\alpha) = \Sigma \, \alpha_i - 1/2 \, \Sigma \, \Sigma \, \alpha_i \, \alpha_j \, y_i \, y_j \, x_i^T x_j$

subject to $\alpha_i \geq 0$ and $\sum \alpha_i \, y_i = 0$

Once we find the values of α_i, we then construct the final classifier. The decision function now becomes: $f(x) = \text{sign}\,(\Sigma \, \alpha_i \, y_i \, x_i^T x + b)$ Only data points with $\alpha_i > 0$ affect the final classifier, hence these constitute the support vectors. That is why SVM is efficient and memory-friendly, because it does not depend on all training samples but only on the most critical ones. The dual formulation is extremely important because, thanks to it, SVM can use kernel functions which allow the transformation of input data into higher dimensions where linear separation becomes possible. In other words, the dual problem of SVM is a mathematical transform of the primal optimization problem using Lagrange multipliers and plays the most important role in enabling SVM to efficiently handle complex high-dimensional, nonlinear classification tasks.

4Q. What are the advantages and disadvantages of SVM?

Ans.

Advantages	Disadvantages
Works well in high-dimensional feature spaces.	Not suitable for very large datasets due to high training time
Effective when number of features is greater than number of samples.	Selecting the right kernel is difficult and highly problem-dependent.
Uses only support vectors, making memory usage efficient.	Performance drops when data is noisy or overlapping.
Kernel trick enables non-linear classification.	Does not work well when classes are not clearly separable.
Good generalization ability and avoids overfitting.	Hard to interpret the final model compared to decision trees.
Can handle both linear and non-linear decision boundaries.	Training becomes slow when dataset is large.
Works well for binary classification problems.	Requires parameter tuning (C and gamma), which can be complex.
Has clear theoretical foundation and geometric interpretation.	Not ideal for multi-class classification without modification.

References-

https://www.geeksforgeeks.org/machine-learning/support-vector-machine-algorithm/

https://www.geeksforgeeks.org/machine-learning/separating-hyperplanes-in-svm/