Support Vector Machine (SVM) Epoch IIT Hyderabad

Chakka Surya Saketh
AI22BTECH11005

Introduction

SVM is a supervised machine learning algorithm that is used for both classification and regression problems. SVM is used for both linear separable data and non-linear separable data. For non-linear data, kernel functions are used.
SVM algorithm finds the best hyperplane that goes in the middle of the two classes with a maximum margin on both sides. To address non-linearly separable data, SVM leverages the concept of kernels, which transform the data into a higher-dimensional space, making it easier to find a separating hyperplane.

Important terms

1) Hyperplane: Hyperplanes are decision boundaries that classify data. A hyperplane is an n-1 dimensional subspace in an n-dimensional space.
2) Support Vectors: The data points closest to the hyperplane in both the classes are known as support vectors.If a data point which is a support vector is removed, then the position of the hyperplane is changed. If a data point that is not a support vector is removed, it has no effect on the model.
3) Margin: The distance between the hyperplane and the support vector is called the margin
4) Hard margin: Hard margin means none of the data points from both the classes falls within the margin. The hard margin SVM is suitable for linearly separable data, where there exists a clear linear boundary that separates the classes. While hard margin SVMs offer a straightforward solution, they are sensitive to outliers and noisy data.
5) Soft margin: soft margin SVM allows for some degree of misclassification to handle non-linearly separable data or noisy datasets. A smaller value of penalty parameter(C) allows for a wider margin and permits more misclassifications, leading to a more robust model that is less sensitive to outliers.

Implementation

The basic principle behind SVM is we want to draw a hyperplane with a maximum margin that separated two classes. We use a linear equation.

$$g(X) = w^T X + b$$

where w = weight vector perpendicular to the hyperplane and b = position of hyperplane in the d-dimensional space
For every feature vector, we have to calculate the linear function in such a way that

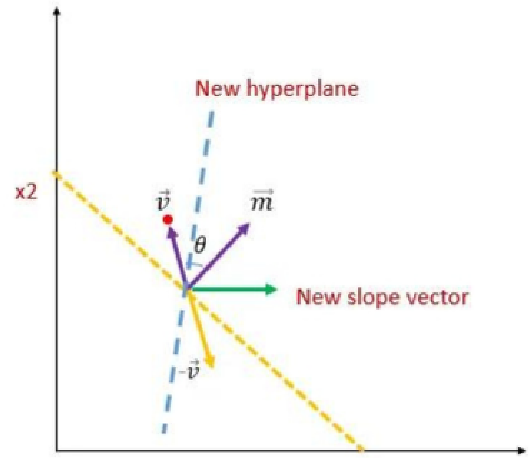
If X lies on hyperplane , $g(X_i)=w^TX_i+b = 0$
If X is the support vector belongs to Class C1, then $g(X_i)=w^TX_i+b = 1$
If X is the support vector belongs to Class C2,then $g(X_i)=w^TX_i+b = -1$

If X is the feature vector belongs to Class C1 , then $g(X_i)=w^TX_i+b > 1$

If X is the feature vector belongs to Class C2 , then $g(X_i)=w^TX_i+b < -1$

Finding the best hyperplane:

During the training phase, it will start with some random hyperplane and check whether there is an error. If a point is wrongly classified then it find the correct value of m and b that give zero training error. Suppose in the random hyperplane drawn, we get one error point - To push the datapoint below the line, we have to increase the angle between the slope vector and the data vector.



If we subtract both vectors (**m** a vector perpendicular to the hyperplane and **b**, from origin of slope vector to error point.), the angle between them will be increased. Similarly, SVM will calculate the new slope for all error points and find the hyperplane which splits both classes.
After finding all possible hyperplane which separates the two classes, we will calculate the margin for all hyperplanes and choose the plane with the highest margin.

Non linear SVM or Kernels

Non-linear SVM addresses limitations by using kernels to map data into a higher-dimensional space, making it linearly separable. Common kernels include Polynomial Kernel and Radial Basis Function (RBF) Kernel. The Polynomial Kernel transforms data into a polynomial feature space, while the RBF Kernel uses distance in the transformed space. With kernels, SVM handles non-linearly separable data, expanding its applicability to real-world problems.

Conclusion

SVM requires tuning of hyperparameters, such as the regularization parameter (C) and kernel parameters (e.g., degree for polynomial kernel or gamma for RBF kernel), to optimize model performance. Proper tuning helps prevent overfitting and enhances generalization to unseen data. SVM's ability to handle both linear and non-linear problems, combined with the flexibility provided by kernels, makes it a preferred choice for complex real-world datasets. Understanding SVM's hyperparameters and the selection of appropriate kernels is essential for building accurate predictive models.