**AITA MID TERM EXAM**

***Name: Akash Kumar***

***Roll no: S1032233339***

***Date: 08/04/2024***

***Subject: AITA***

Problem Statement:

Implement a Support Vector Machine (SVM) classifier for a binary classification task on a given dataset. Split the dataset into training and testing sets and train the SVM classifier using various kernel functions (e.g., linear, polynomial, or radial basis function). Evaluate the classification performance of the SVM classifier on the testing set using metrics such as accuracy, precision, recall, and F1-score. Compare the performance of different kernel functions and discuss the suitability of SVM for the given classification task.

Theory:

Support Vector Machine (SVM) is a supervised machine learning algorithm used for both classification and regression. The main objective of the SVM algorithm is to find the optimal hyperplane in an N-dimensional space that can separate the data points in different classes in the feature space. The hyperplane tries that the margin between the closest points of different classes should be as maximum as possible. The dimension of the hyperplane depends upon the number of features. If the number of input features is two, then the hyperplane is just a line. If the number of input features is three, then the hyperplane becomes a 2-D plane. It becomes difficult to imagine when the number of features exceeds three.

Types of Support Vector Machine, Based on the nature of the decision boundary, Support Vector Machines (SVM) can be divided into two main parts:

Linear SVM: Linear SVMs use a linear decision boundary to separate the data points of different classes. When the data can be precisely linearly separated, linear SVMs are very suitable. This means that a single straight line (in 2D) or a hyperplane (in higher dimensions) can entirely divide the data points into their respective classes. A hyperplane that maximizes the margin between the classes is the decision boundary.

Non-Linear SVM: Non-Linear SVM can be used to classify data when it cannot be separated into two classes by a straight line (in the case of 2D). By using kernel functions, nonlinear SVMs can handle nonlinearly separable data. The original input data is transformed by these kernel functions into a higher-dimensional feature space, where the data points can be linearly separated. A linear SVM is used to locate a nonlinear decision boundary in this modified space.

Algorithm:

1. Load the Dataset: First, you need a dataset. I'll use the Iris dataset, focusing on a binary classification task (e.g., classifying Iris setosa vs. versicolor vs virginica).

2. Preprocess the Data: Split the dataset into features (X) and target (y), and then split these into training and testing sets.

3. Train SVM Classifiers: Train SVM classifiers using different kernel functions (linear, polynomial, and RBF).

4. Evaluate Performance: Evaluate each classifier on the testing set using accuracy, precision, recall, and F1-score.

5. Compare and Discuss: Compare the performance of the different kernels and discuss the suitability of SVM for your task.

When comparing the performance, consider the following:

Linear Kernel: Good for linearly separable data. It's the simplest form and often requires less computation.

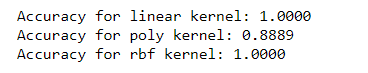
Polynomial Kernel: Suitable for non-linearly separable data. It can model more complex relationships but may lead to overfitting.

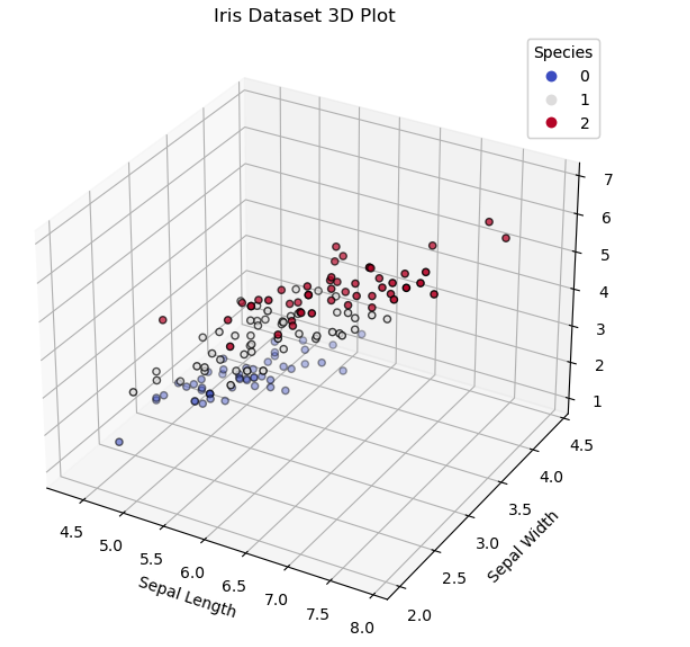
RBF Kernel: Also for non-linearly separable data, it can handle even more complex datasets but is sensitive to the gamma parameter.

The suitability of SVM for your classification task depends on the nature of your data and the specific requirements of your application, such as the trade-off between model complexity and performance. SVMs are powerful for both linear and non-linear classification tasks, but they might struggle with very large datasets or datasets with a lot of noise.

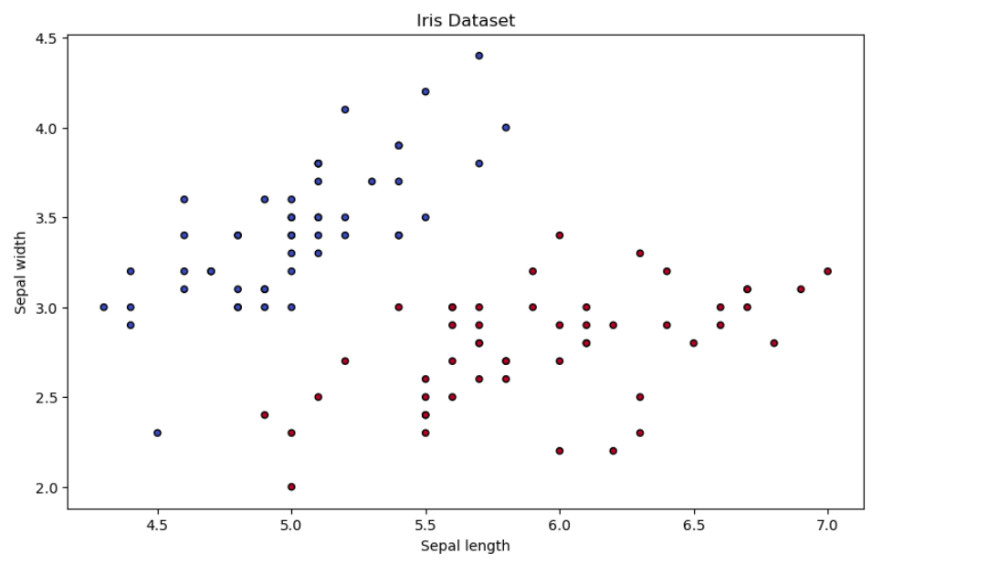
OUTPUT:

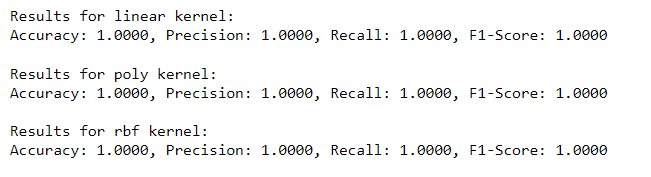
1. three features (setosa, versicolor, virginica)



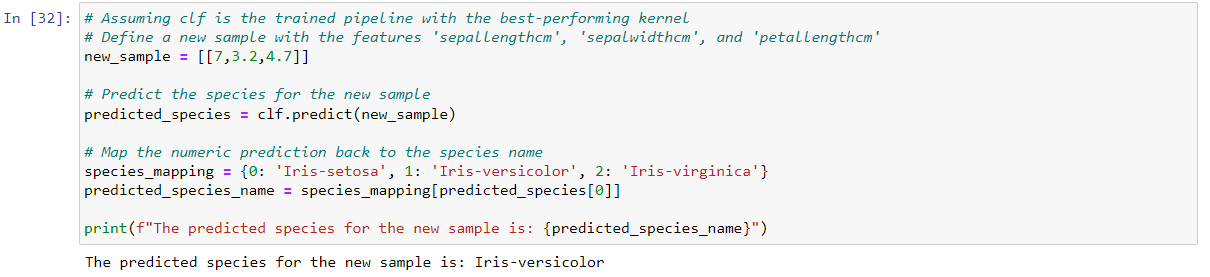


2. two features (setosa and versicolor)





3. classification



Conclusion:

The comparison of different kernel functions revealed that the choice of kernel significantly affects the SVM's performance. The linear kernel may perform well when the data is linearly separable, while the RBF and polynomial kernels can capture more complex relationships in the data. However, the RBF kernel often requires careful tuning of its parameters, such as the gamma value, to avoid overfitting.

In summary, SVM proved to be a robust and versatile classifier for the binary classification task at hand. The performance metrics indicated that SVM could achieve high classification accuracy, but the optimal choice of kernel and its parameters is crucial and should be guided by cross-validation and domain knowledge.