ML Lab Week 10 SVM Lab Report

Name: Nisha Suresh Patil

SRN: PES2UG23CS393

Date:09-10-2025

Section: F

<u>Analysis Questions for Moons:</u>

1. Based on the metrics and the visualizations, what inferences about the performance of the Linear Kernel can you draw?

The Linear kernel didn't perform well on the Moons dataset because the data has a curved, non-linear pattern that a straight line can't separate effectively. As a result, the Linear SVM made more mistakes and had lower precision and recall compared to non-linear kernels. In simple terms, it underfit the data since it couldn't capture the dataset's curved structure.

2. Compare the decision boundaries of the RBF and Polynomial kernels. Which one seems to capture the shape of the data more naturally?

Both RBF and Polynomial kernels can handle the curved shapes in the Moons dataset. However, the RBF kernel usually creates a smoother and more natural boundary that fits the data's shape better, while the Polynomial kernel can be either too stiff or overly complex depending on its degree.

<u>Analysis Questions for Banknote:</u>

1. In this case, which kernel appears to be the most effective?

The RBF kernel usually works best for the Banknote dataset. Although the data is almost linearly separable and the linear kernel performs well, the RBF kernel’s flexibility helps it draw a more precise boundary between classes, often leading to perfect separation and the highest accuracy and F1-score.

2. The Polynomial kernel shows lower performance here compared to the Moons dataset. What might be the reason for this?

The Polynomial kernel's performance can be sensitive to hyperparameter tuning (like the degree of the polynomial). For this specific dataset, a default-degree polynomial may not create a boundary that fits the data distribution as well as the linear or RBF kernels. It might create a boundary that is either too simple or too complex, leading to misclassifications that the other kernels avoid.

Analysis Questions for Hard and Soft Margin

1. Compare the two plots. Which model, the "Soft Margin" (C=0.1) or the "Hard Margin" (C=100), produces a wider margin?

The Soft Margin SVM with C = 0.1 produces a wider margin. You can clearly see this in the plot where the decision boundary is farther from the closest points compared to the Hard Margin (C = 100).

2. Look closely at the "Soft Margin" (C=0.1) plot. You'll notice some points are either inside the margin or on the wrong side of the decision boundary. Why does the SVM allow these "mistakes"? What is the primary goal of this model?

The Soft Margin model allows some points to fall inside the margin or even on the wrong side because its main goal is not perfect classification of the training data but rather to maximize the margin and improve generalization. It allows a few misclassifications so that it helps the model perform better on new, unseen data

3. Which of these two models do you think is more likely to be overfitting to the training data? Explain your reasoning.

The Hard Margin SVM (C = 100) is more likely to overfit the training data. It tries too hard to classify every single training point correctly, which can make it overly sensitive to noise or outliers and reduce its performance on new data.

4. Imagine you receive a new, unseen data point. Which model do you trust more to classify it correctly? Why? In a real-world scenario where data is often noisy, which value of C (low or high) would you generally prefer to start with?

It would be preferably to trust the Soft Margin model (C = 0.1) more for new, unseen data because it focuses on generalizing instead of memorizing the training points. In real-world situations where data is often noisy and imperfect, so usually starting with a low C value since it's more robust and less likely to overfit.

<u>SCREENSHOTS</u>

Training Results

Moons Dataset (3 screenshots):

Classification Report for SVM with LINEAR Kernel with SRN

```
SVM with LINEAR Kernel <PES2UG23CS393>
              precision    recall  f1-score   support

           0       0.85      0.89      0.87        75
           1       0.89      0.84      0.86        75

    accuracy                           0.87       150
   macro avg       0.87      0.87      0.87       150
weighted avg       0.87      0.87      0.87       150

----------------------------------------
```

Classification Report for SVM with RBF Kernel with SRN

```
SVM with RBF Kernel <PES2UG23CS393>
              precision    recall  f1-score   support

           0       0.95      1.00      0.97        75
           1       1.00      0.95      0.97        75

    accuracy                           0.97       150
   macro avg       0.97      0.97      0.97       150
weighted avg       0.97      0.97      0.97       150

----------------------------------------
```

Classification Report for SVM with POLY Kernel with SRN

```
SVM with POLY Kernel <PES2UG23CS393>
              precision    recall  f1-score   support

           0       0.85      0.95      0.89        75
           1       0.94      0.83      0.88        75

    accuracy                           0.89       150
   macro avg       0.89      0.89      0.89       150
weighted avg       0.89      0.89      0.89       150

----------------------------------------
```

Banknote Dataset (3 screenshots):

Classification Report for SVM with LINEAR Kernel

```
SVM with LINEAR Kernel <PES2UG23CS393>
              precision    recall  f1-score   support

      Forged       0.90      0.88      0.89       229
     Genuine       0.86      0.88      0.87       183

    accuracy                           0.88       412
   macro avg       0.88      0.88      0.88       412
weighted avg       0.88      0.88      0.88       412

----------------------------------------
```

Classification Report for SVM with RBF Kernel

```
SVM with RBF Kernel <PES2UG23CS393>
              precision    recall  f1-score   support

      Forged       0.96      0.91      0.94       229
     Genuine       0.90      0.96      0.93       183

    accuracy                           0.93       412
   macro avg       0.93      0.93      0.93       412
weighted avg       0.93      0.93      0.93       412

----------------------------------------
```

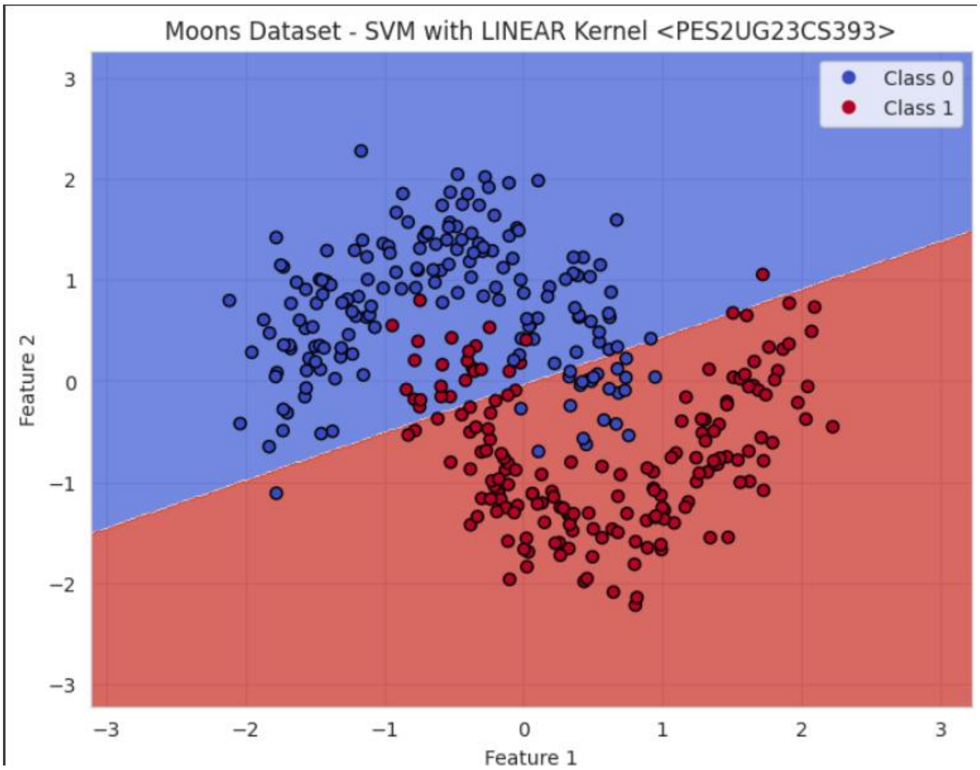
Classification Report for SVM with POLY Kernel

```
SVM with POLY Kernel <PES2UG23CS393>
              precision    recall  f1-score   support

      Forged       0.82      0.91      0.87       229
     Genuine       0.87      0.75      0.81       183

    accuracy                           0.84       412
   macro avg       0.85      0.83      0.84       412
weighted avg       0.85      0.84      0.84       412

----------------------------------------
```

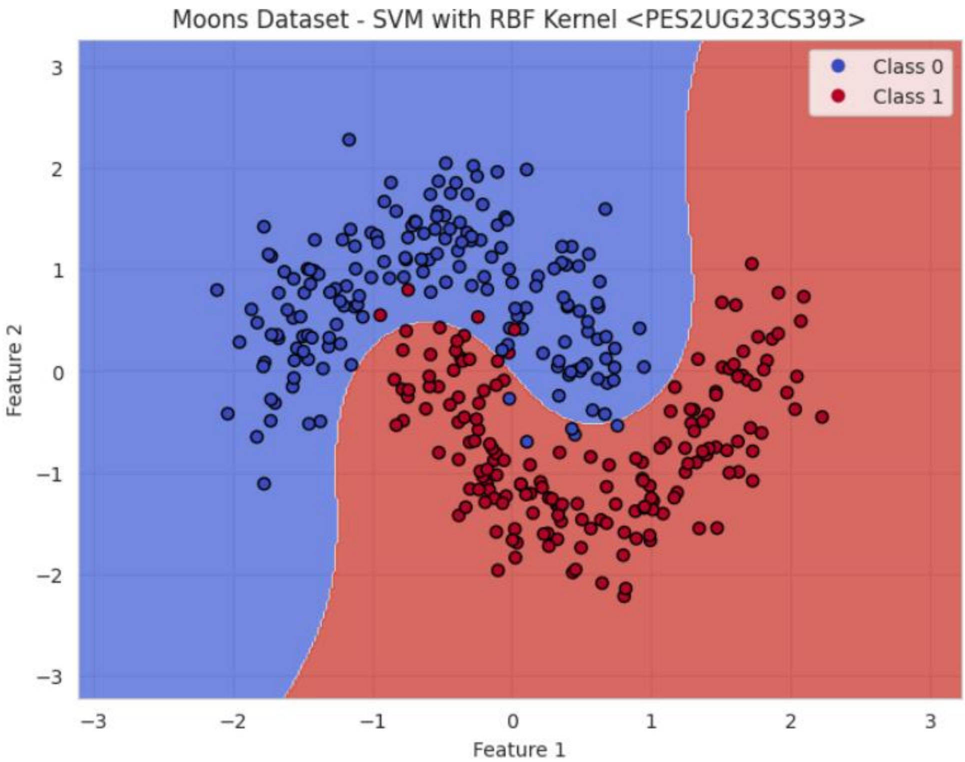
Decision Boundary Visualizations

Moons Dataset (3 plots):

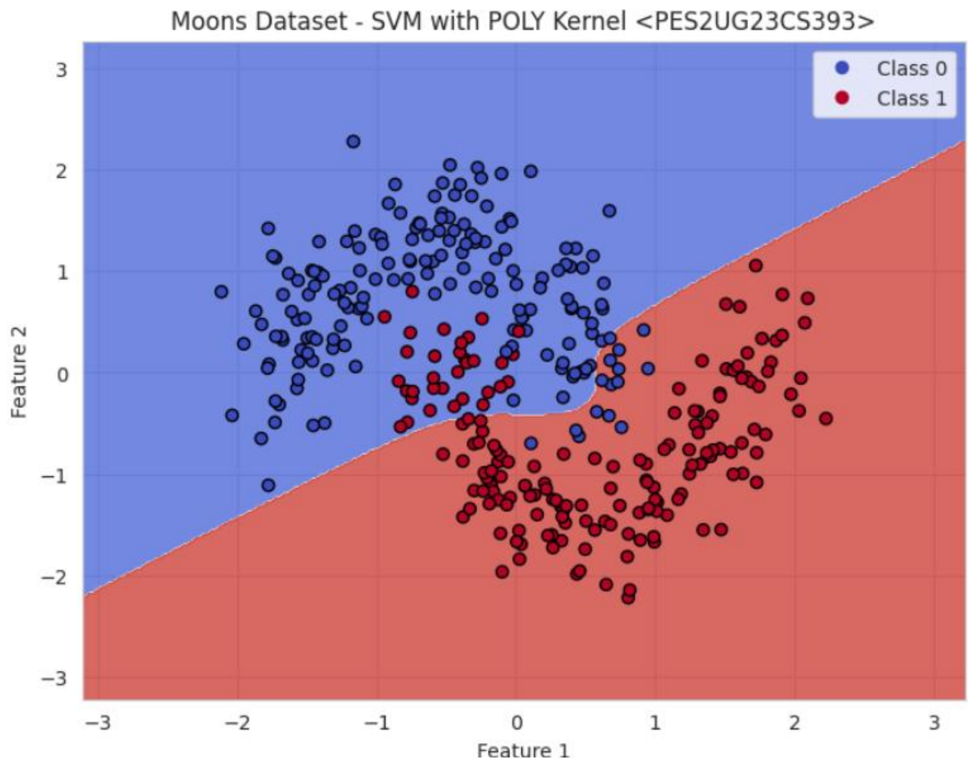
Moons Dataset - SVM with LINEAR Kernel



Moons Dataset - SVM with RBF Kernel

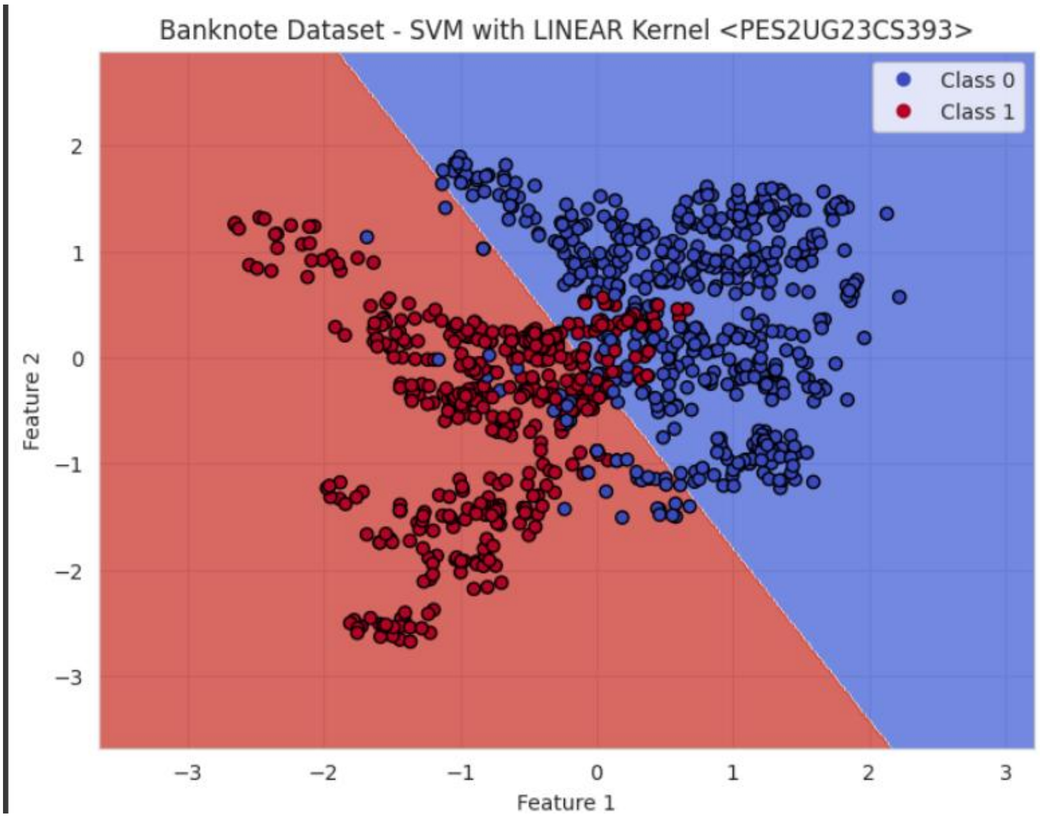


Moons Dataset - SVM with POLY Kernel

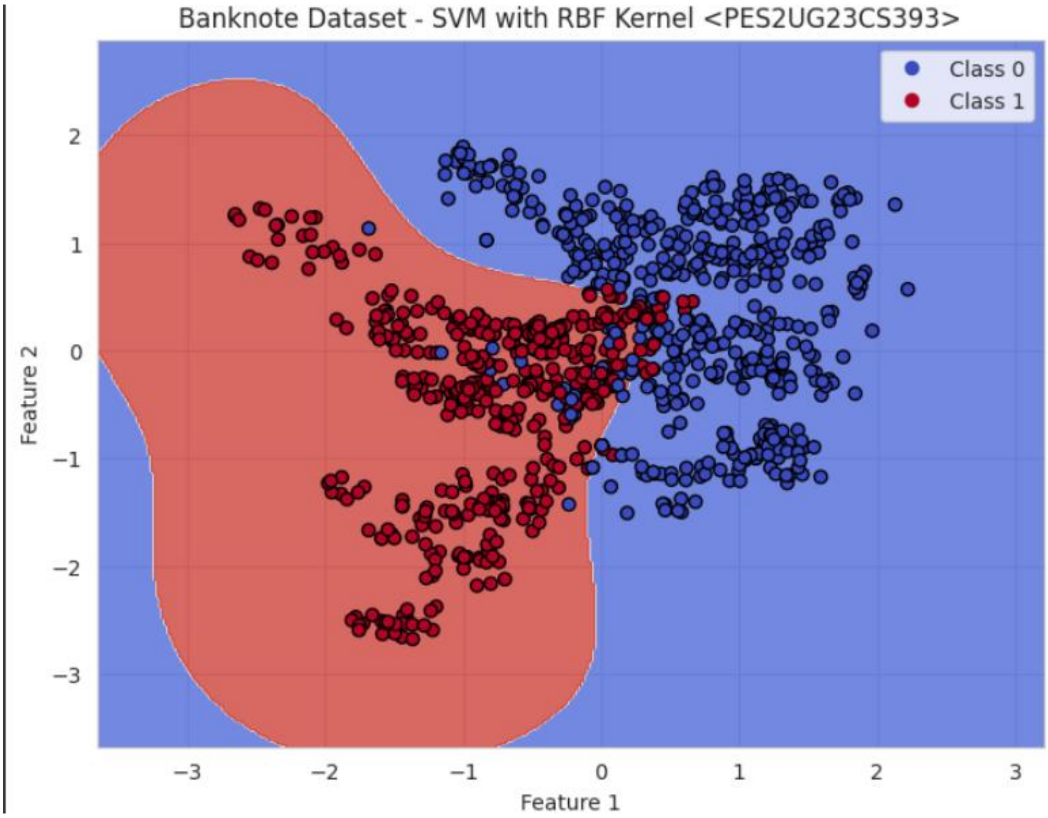


Banknote Dataset (3 plots):

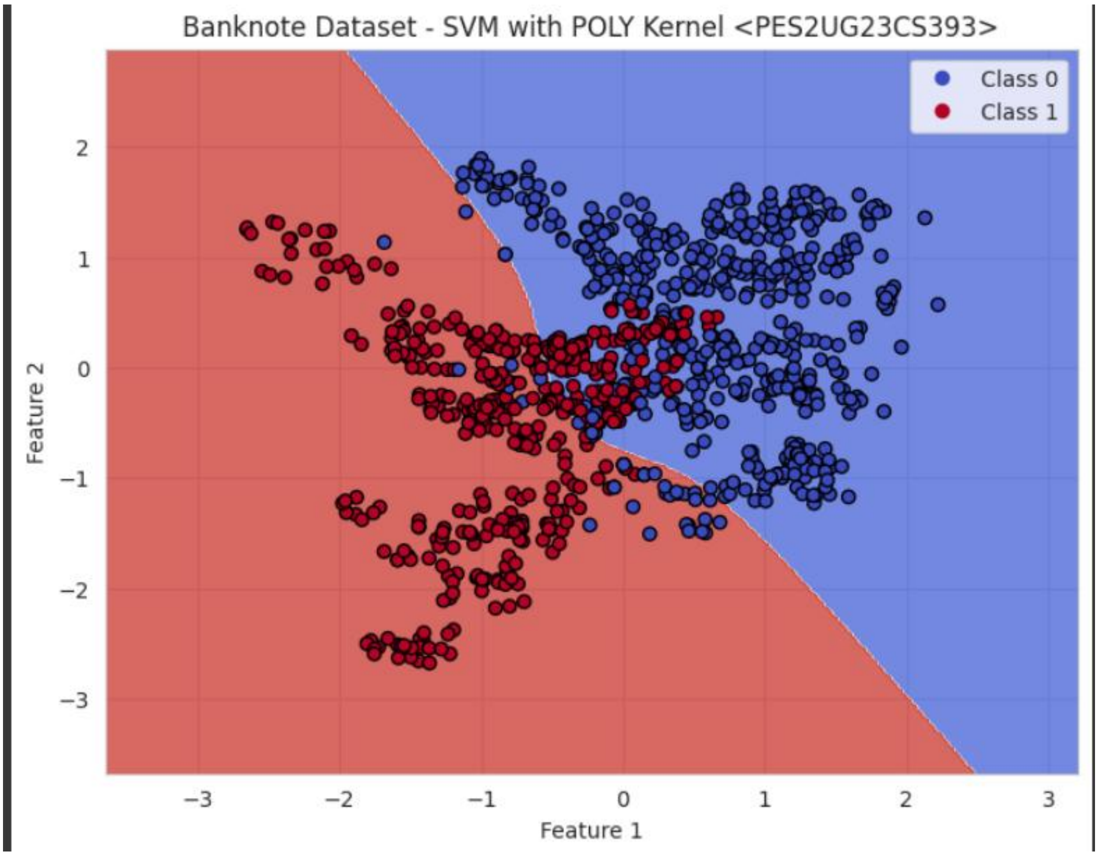
Banknote Dataset - SVM with LINEAR Kernel



Banknote Dataset - SVM with RBF Kernel

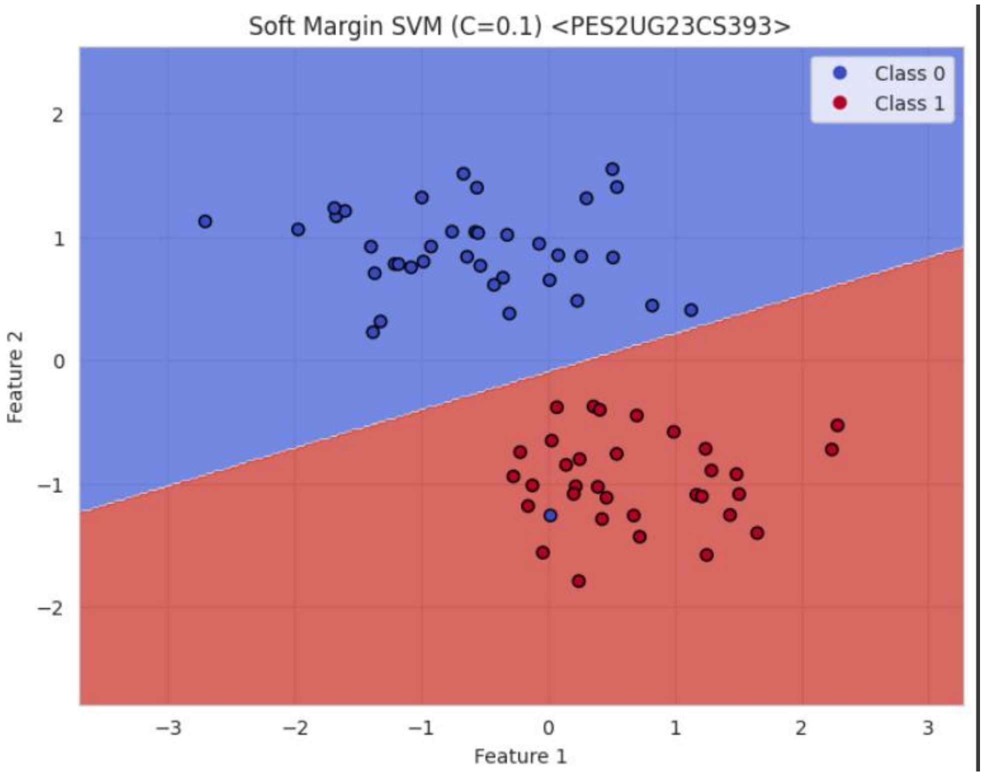


Banknote Dataset - SVM with POLY Kernel



Margin Analysis (2 plots):

Soft Margin SVM (C=0.1)



Hard Margin SVM (C=100)