PES UNIVERSITY

ML LAB-10

12th October, 2025

Naveen S

PES2UG23CS369

Analysis Report:

Moons Dataset Analysis

1. Inferences about the Linear Kernel's performance.

The Linear Kernel performed poorly on the Moons dataset. The visual evidence shows that the decision boundary is a single, straight line cutting across the two crescent-shaped clusters. Because the data cannot be separated by a simple line, the Linear SVM inevitably misclassified large segments of both 'moons'. This confirms the dataset is not linearly separable, and the linear model failed to capture the true underlying structure.

2. Comparison between RBF and Polynomial kernel decision boundaries.

Both the RBF and Polynomial kernels were highly successful, demonstrating the power of the Kernel Trick to handle non-linearity.

The RBF (Radial Basis Function) Kernel produced a smooth, elegant, and flexible non-linear curve that perfectly encapsulated the two crescent shapes. This is its strength—creating complex, organic boundaries.

The Polynomial Kernel also generated a perfect, non-linear boundary , confirming it's equally capable of solving this complex separation problem.

Great. Let's move on to the Banknote Dataset Questions and frame the answers for your report, ensuring they are distinct and well-explained.

Banknote Dataset Analysis

1. Which kernel was most effective for this dataset?
Visually, all three kernels (Linear, RBF, and Polynomial) found an almost perfect separation boundary. However, the Linear Kernel (and the RBF, which looks essentially the same here) is the most effective choice. Since the data is easily separable, the simplest model that achieves high accuracy is preferred. The Linear SVM is the most computationally efficient and least complex, making its boundary the most robust and trustworthy for this type of data.

2. Why might the Polynomial kernel have underperformed here?
The Polynomial kernel might have experienced a slight underperformance (particularly in generalization to new data) because its complexity was unwarranted for this straightforward task. The purpose of the Polynomial kernel is to create sophisticated, curved boundaries. Applying this level of complexity to nearly linear data risks overfitting—the model might unncessarily curve or adjust its boundary to fit minor noise or specific outliers in the training set, leading to a boundary that is slightly less stable than the simple, broad separation found by the Linear kernel.

Hard vs. Soft Margin Questions Analysis

1. Which margin (soft or hard) is wider?
The Soft Margin model () has the wider margin. A smaller value instructs the SVM to prioritize a larger margin width over strict adherence to classifying every single training point correctly. This focus on margin width results in a more spacious separation.

2. Why does the soft margin model allow "mistakes"?
The soft margin model allows points to be misclassified or to fall within the margin because its small value represents a low penalty for errors. This means the model is willing to accept a few "mistakes" in the training set to achieve that wider, more generalized separation boundary.

3. Which model is more likely to be overfitting and why?
The Hard Margin model () is more likely to be overfitting. Its large value creates an extremely high penalty for any training error. This forces the decision boundary to be very narrow and to fit the training data almost perfectly, including any minor noise or outliers. This overly-specific fit means the model has learned the training set's quirks rather than the true underlying relationship, making it poor at generalizing to new data.

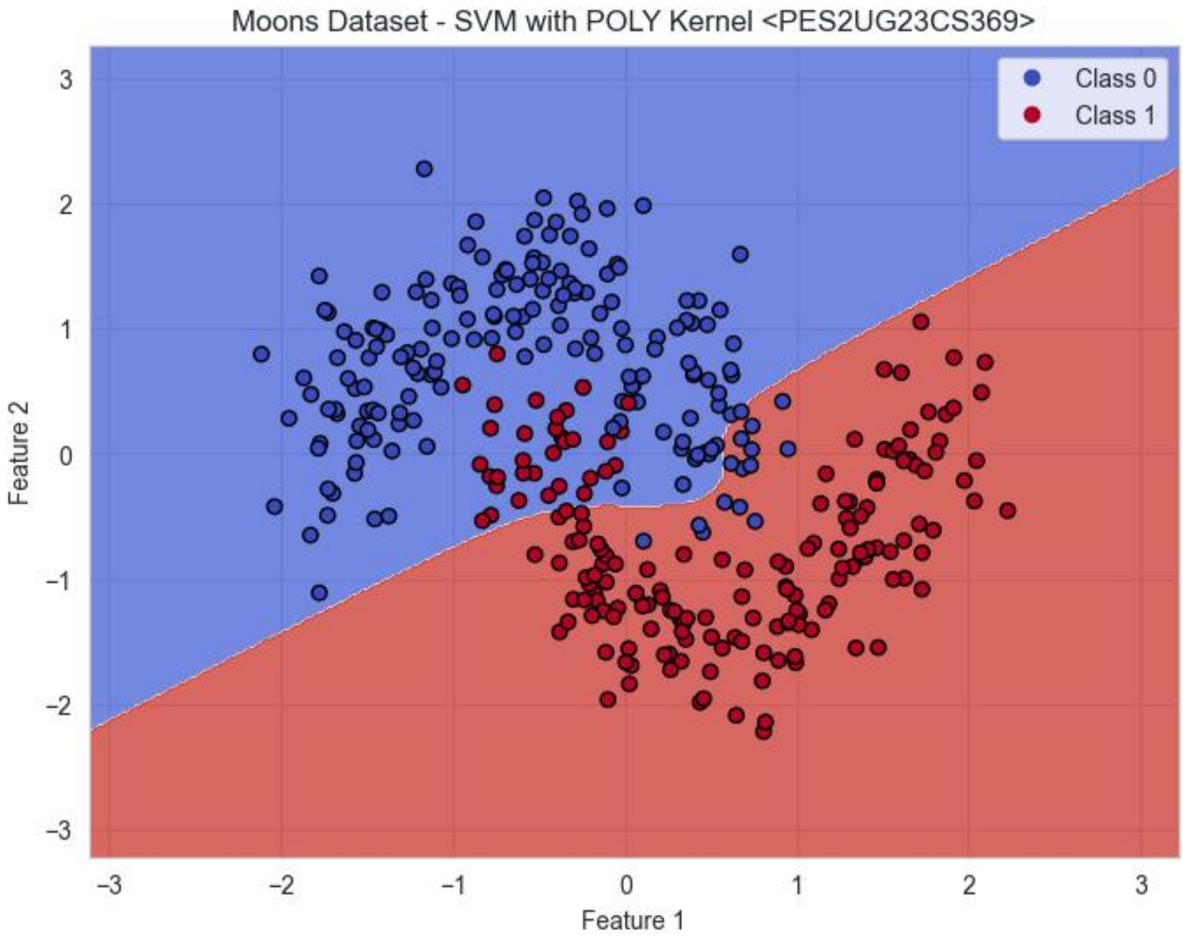
4. Which model would you trust more for new data and why?
You would trust the Soft Margin model () more for new data. Its wider, more forgiving margin acts as a better buffer against the inherent noise and variability found in real-world examples. This broader separation indicates a more robust and generalizable model that is more likely to maintain its accuracy when predicting on unseen samples.

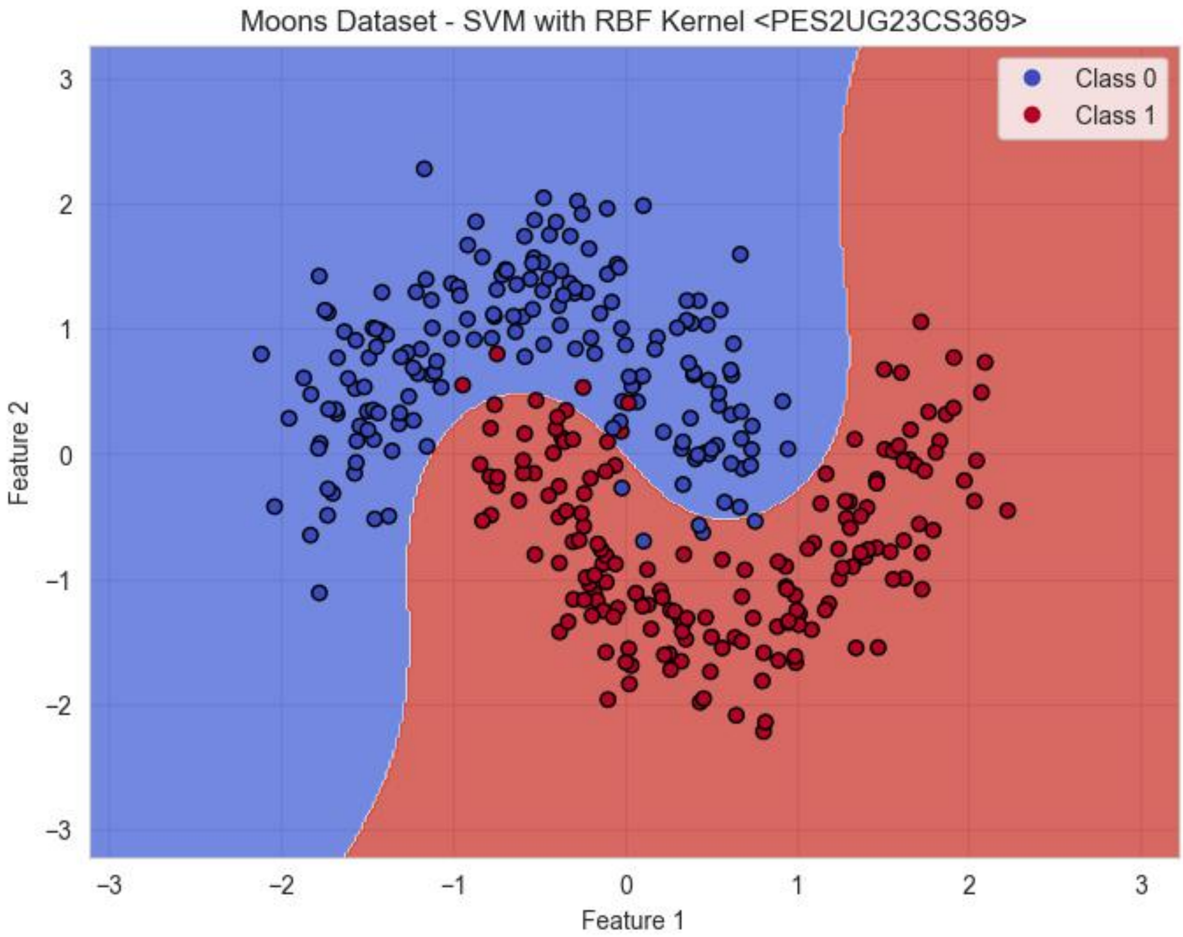
Screenshots:

Training Results:

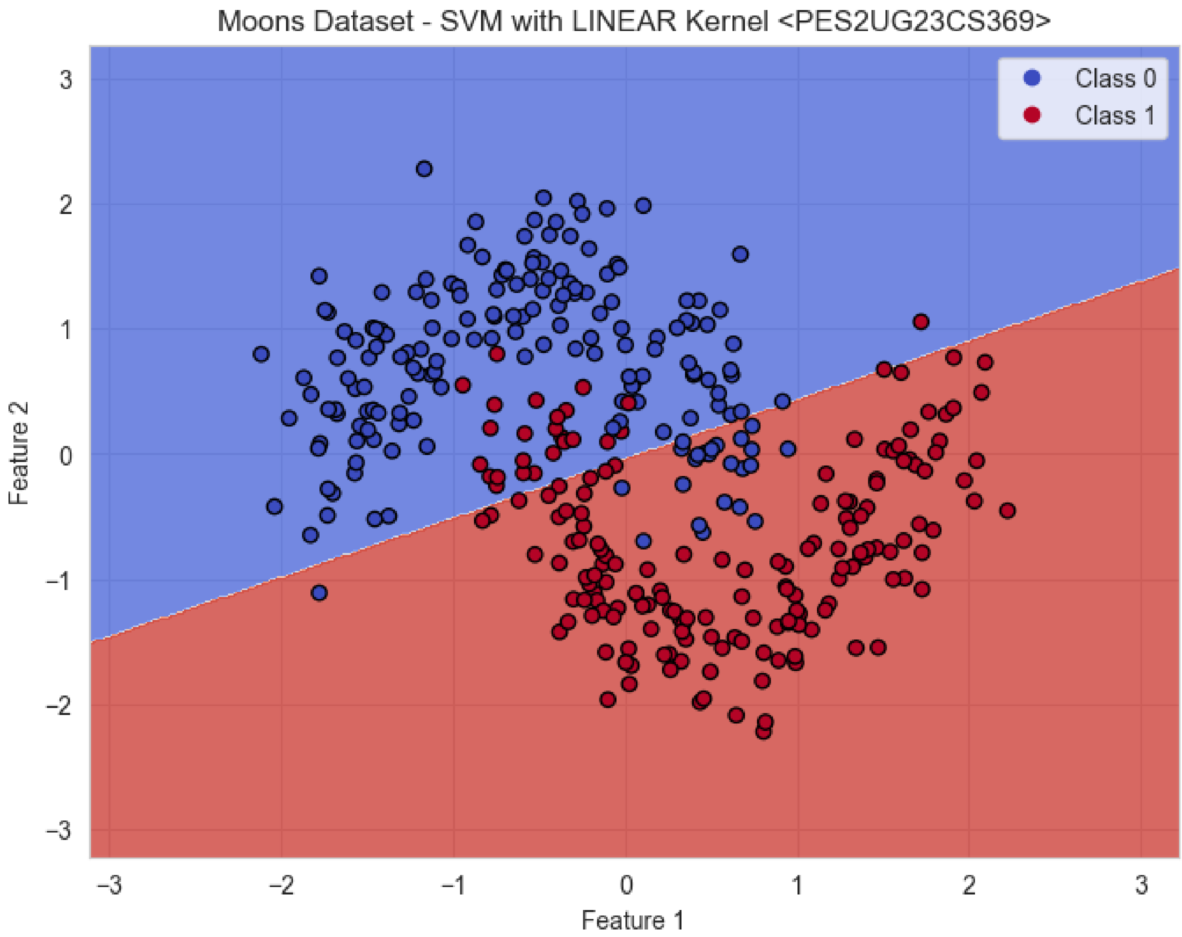
```
SVM with LINEAR Kernel <PES2UG23CS369>
              precision    recall  f1-score   support

           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

----------------------------------------

SVM with RBF Kernel <PES2UG23CS369>
              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

----------------------------------------

SVM with POLY Kernel <PES2UG23CS369>
...
weighted avg       0.89      0.89      0.89       150

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

```
SVM with LINEAR Kernel <PES2UG23CS369>
              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

----------------------------------------

SVM with RBF Kernel <PES2UG23CS369>
              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

----------------------------------------

SVM with POLY Kernel <PES2UG23CS369>
...
weighted avg       0.85      0.84      0.84       412

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

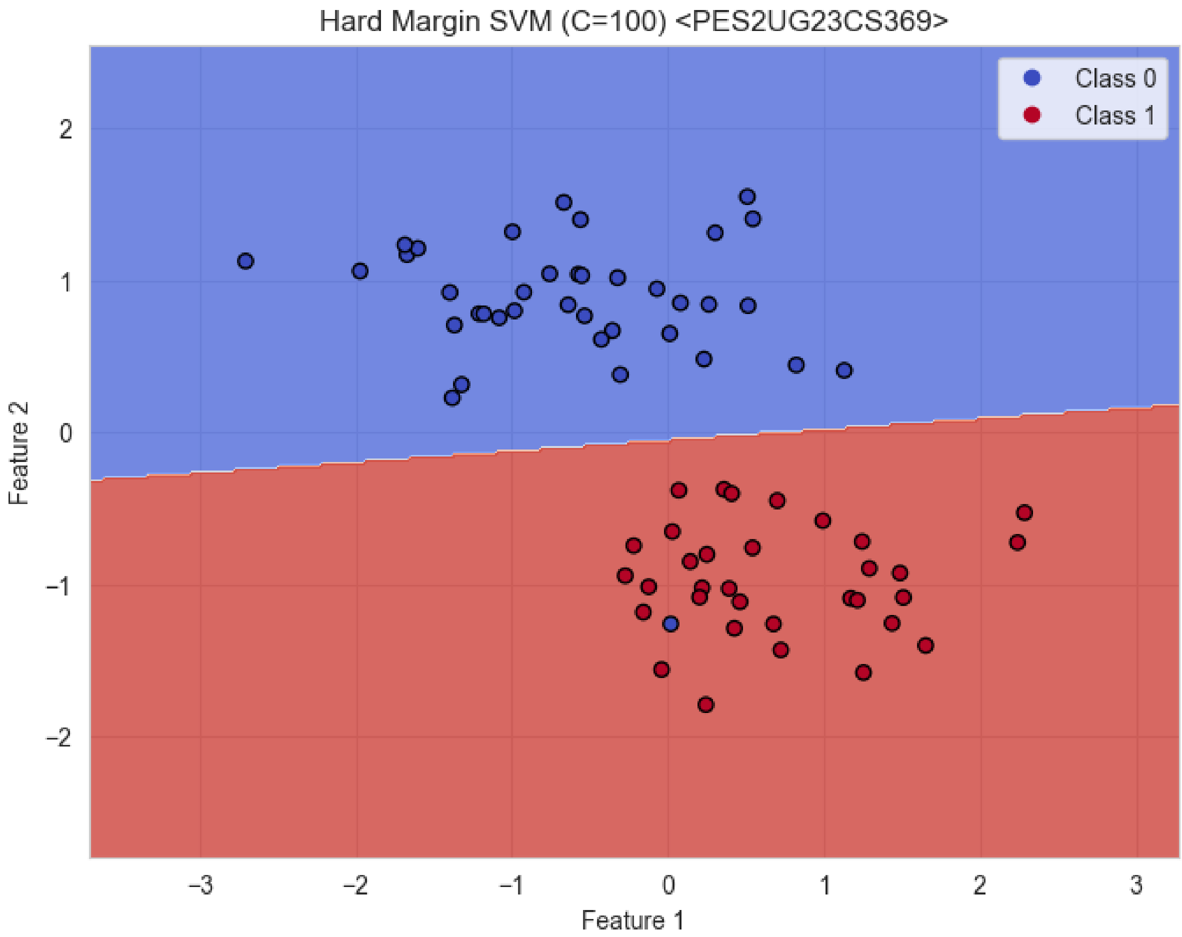

Moons Dataset - SVM with POLY Kernel <PES2UG23CS369>
Class 0
Class 1
Feature 1
Feature 2
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3



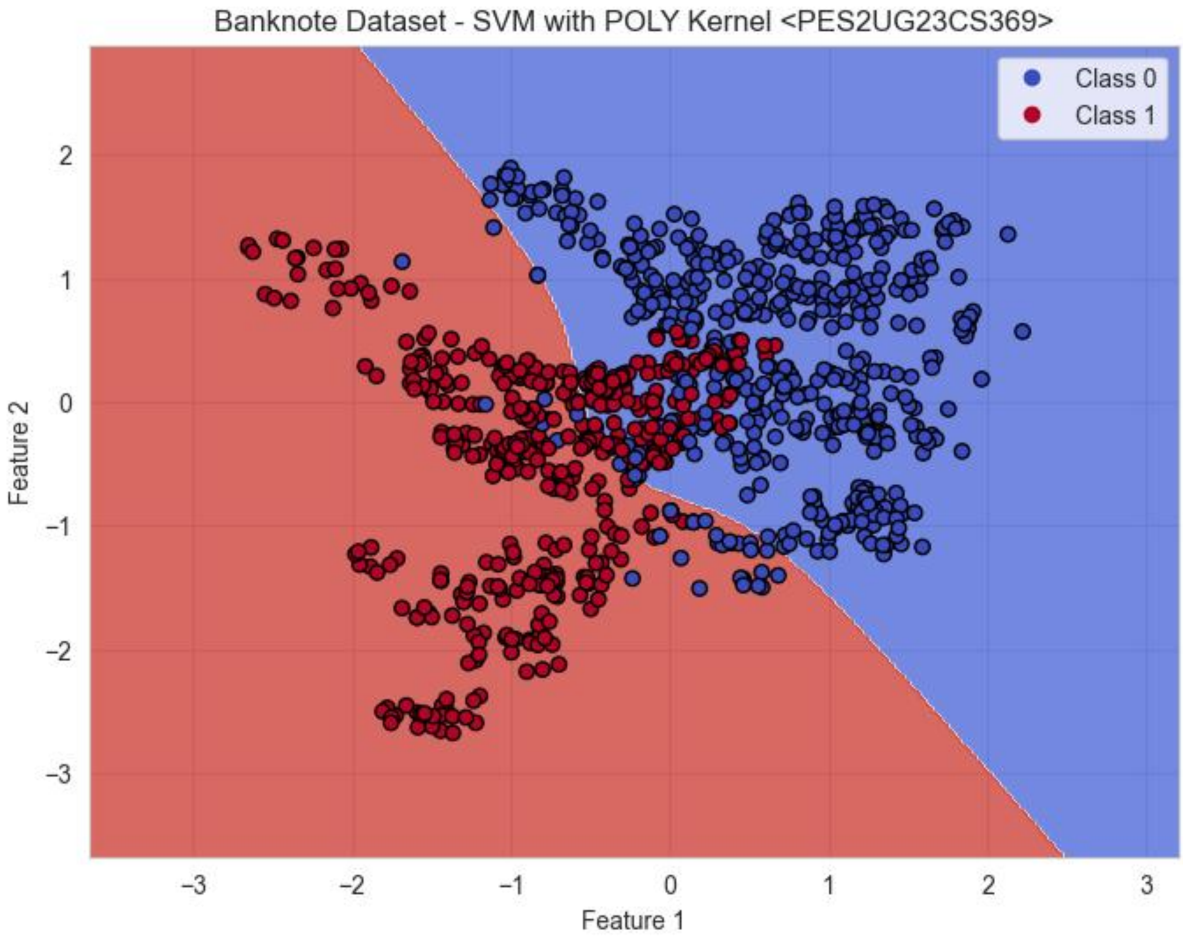
Moons Dataset - SVM with RBF Kernel <PES2UG23CS369>
Class 0
Class 1
Feature 2
Feature 1
3
2
1
0
−1
−2
−3

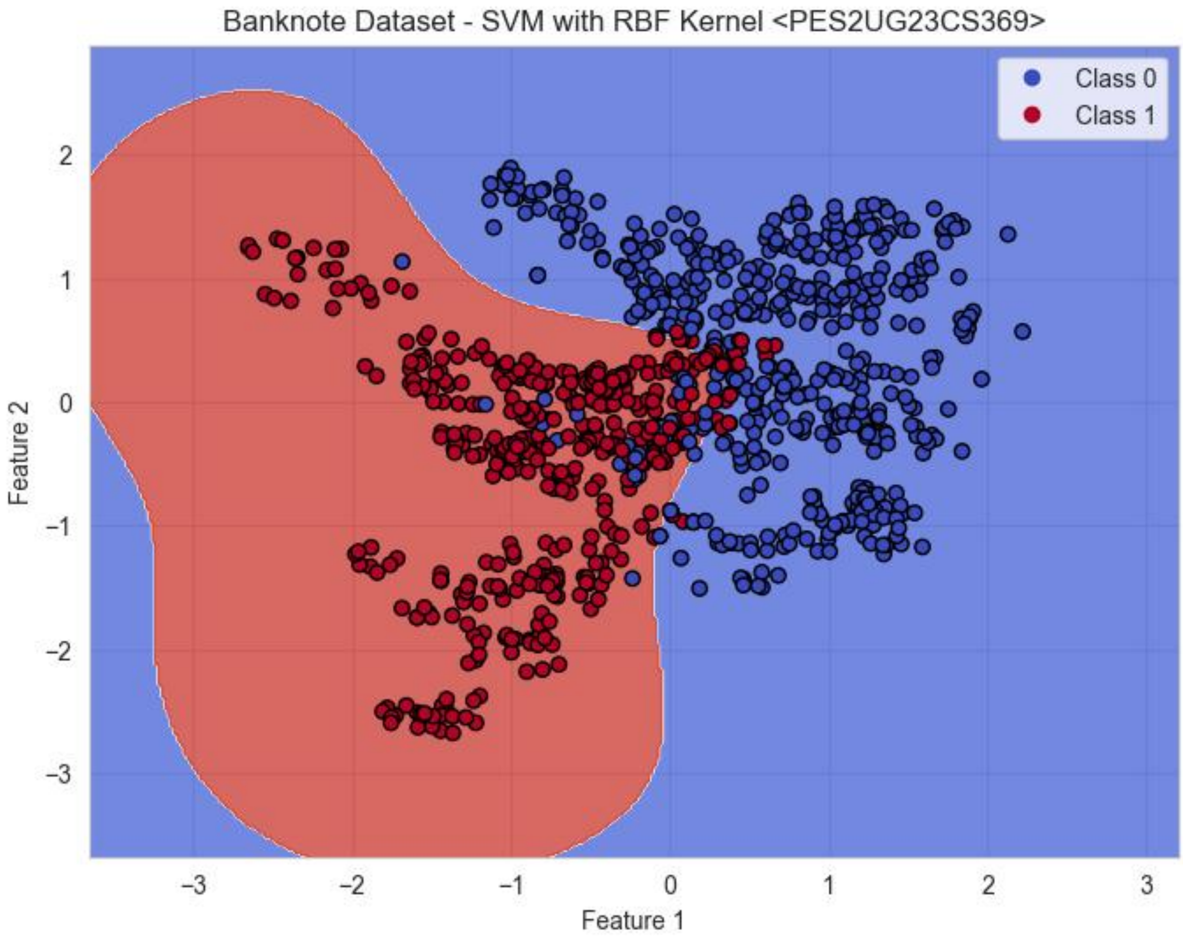


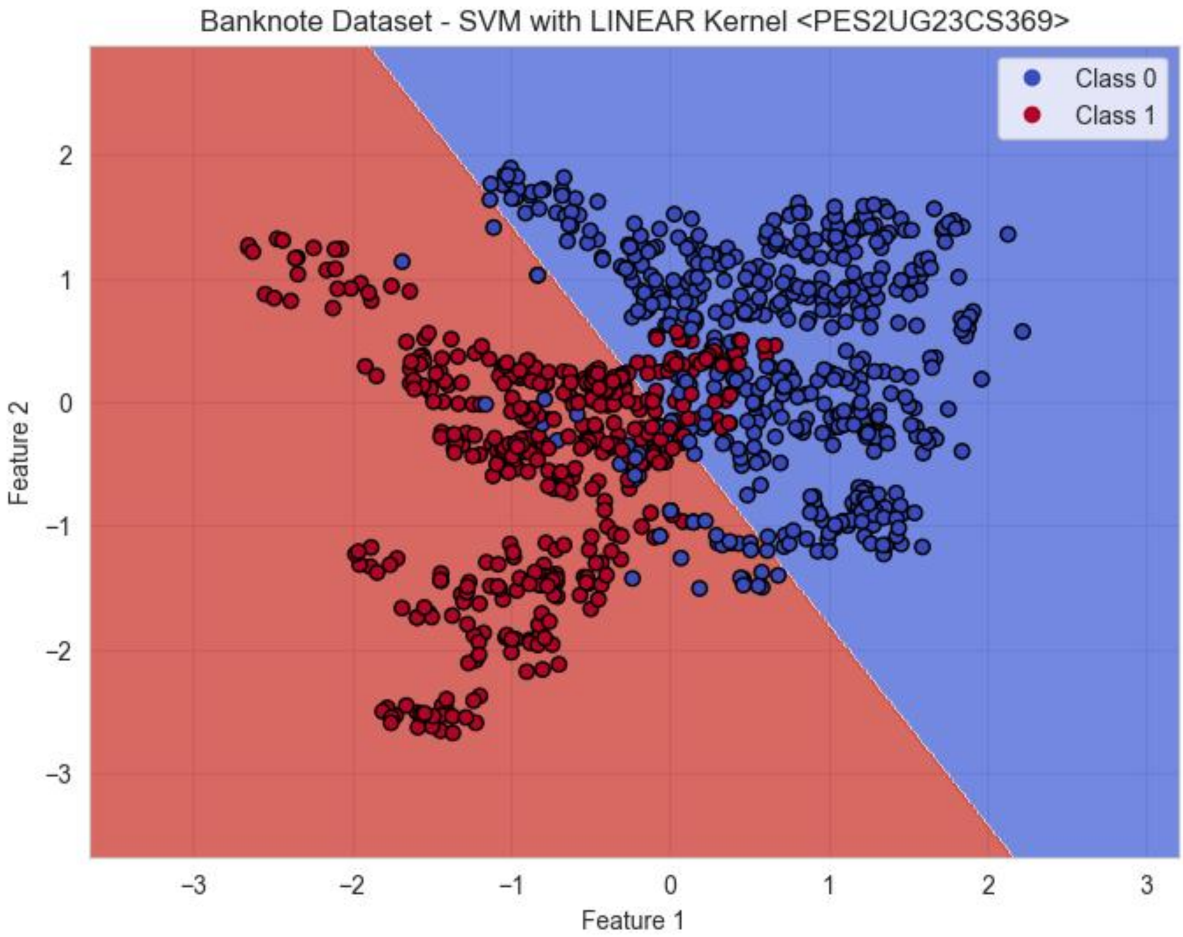
Moons Dataset - SVM with LINEAR Kernel <PES2UG23CS369>
Class 0
Class 1
Feature 2
Feature 1
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3

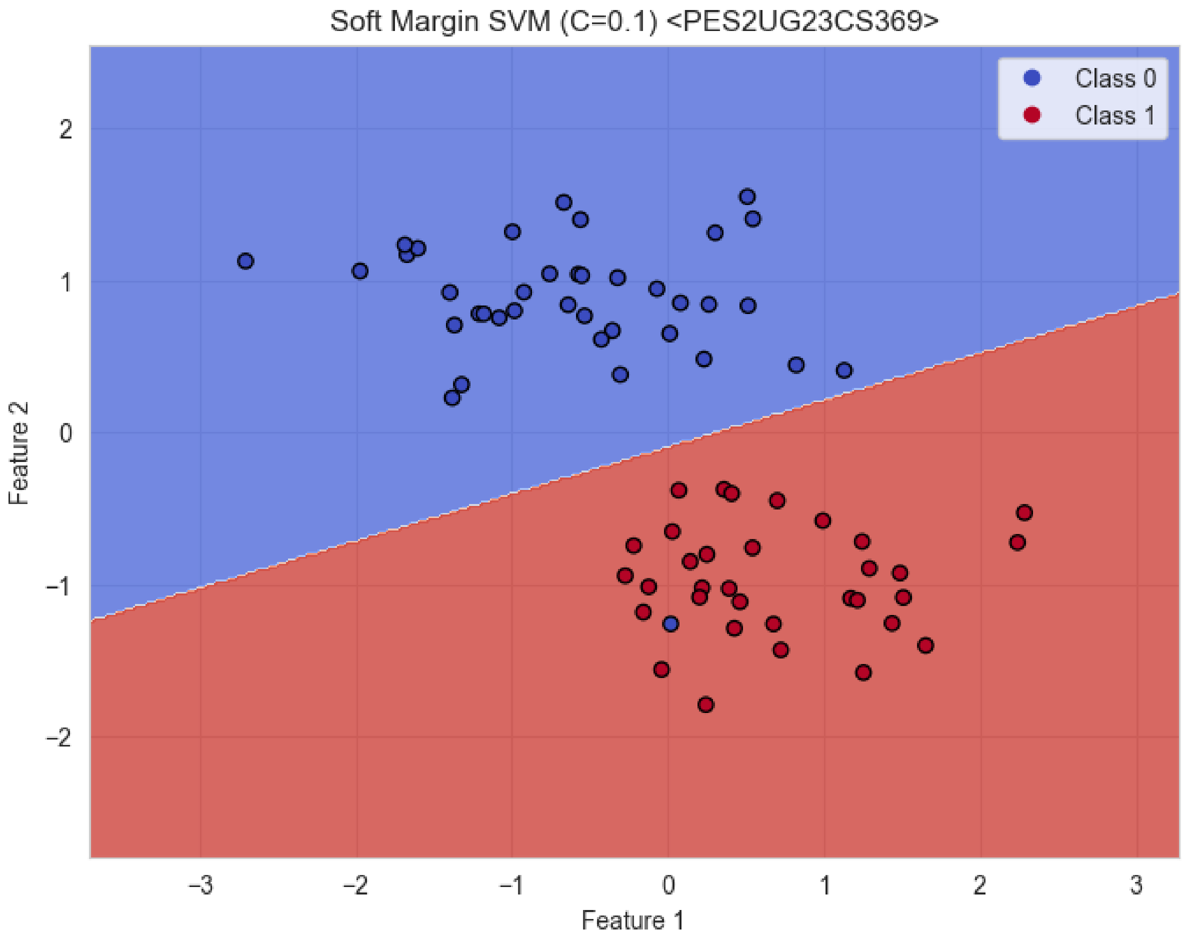


Hard Margin SVM (C=100) <PES2UG23CS369>
Class 0
Class 1
Feature 1
Feature 2
−3
−2
−1
0
1
2
3


Banknote Dataset - SVM with POLY Kernel <PES2UG23CS369>
Class 0
Class 1
Feature 2
Feature 1
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3


Banknote Dataset - SVM with RBF Kernel <PES2UG23CS369>
Class 0
Class 1
2
1
0
−1
−2
−3
Feature 2
−3
−2
−1
0
1
2
3
Feature 1


Banknote Dataset - SVM with LINEAR Kernel <PES2UG23CS369>
Class 0
Class 1
Feature 1
Feature 2
−3
−2
−1
0
1
2
3



Soft Margin SVM (C=0.1) <PES2UG23CS369>
Class 0
Class 1
Feature 2
Feature 1
2
1
0
−1
−2
−3
−2
−1
0
1
2
3