ML Lab Week 10: SVM Lab

Name	Diya D Bhat
SRN	PES2UG23CS183
Section	C
Date	10/10/25
Course name	Machine Learning

1. Introduction

The main purpose of this lab is to learn how the Support Vector Machine (SVM) algorithm works for classifying data. SVM helps find the best possible line (or hyperplane) that separates two classes of data with the largest margin. In this experiment, we use different types of SVM kernels such as Linear, RBF, and Polynomial, to see how each kernel performs on different kinds of datasets. We also compare Soft Margin (C = 0.1) and Hard Margin (C = 100) SVMs to understand how the value of C affects the model's flexibility and accuracy.

2. Datasets used

2.2.1 Moons Dataset:
This is a synthetic dataset with two interlocking half-moon shapes. It is non-linearly separable, which means a straight line cannot easily separate the two classes.
It helps us test how well non-linear kernels like RBF and Polynomial perform.

2.2.2 Banknote Authentication Dataset:
This is a real-world dataset used to check if a banknote is genuine or forged.
The dataset has numerical features such as variance, skewness, kurtosis, and entropy of wavelet-transformed images of the notes.
Since the data is almost linearly separable, it is good for testing the Linear kernel performance.

3. Training Results & Decision Boundary Visualizations

3.1 Moons Dataset

3.1.1 Linear Kernel

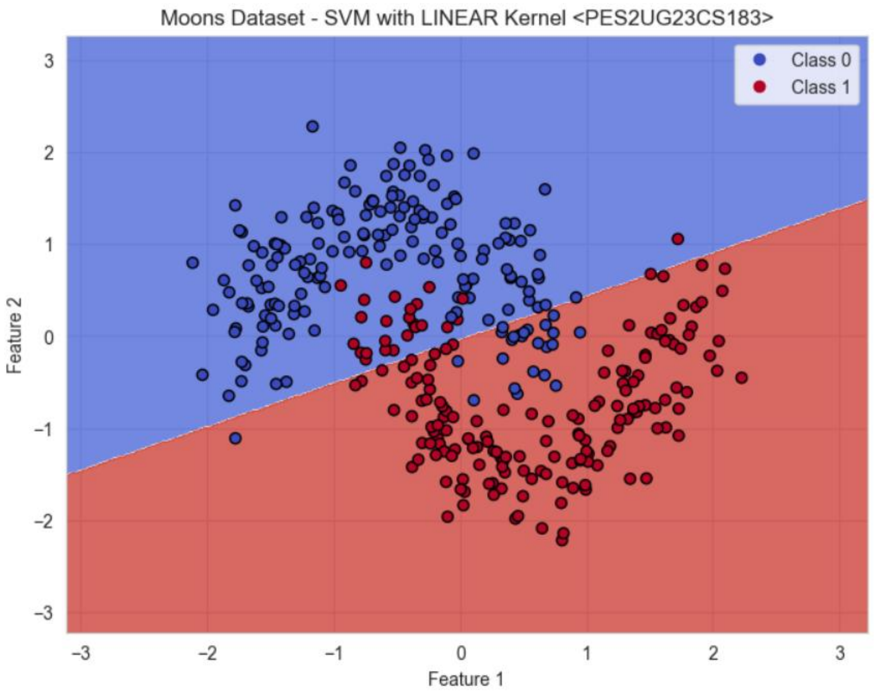
Screenshot of classification report

```
SVM with LINEAR Kernel <PES2UG23CS183>
              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
```

Decision boundary plot



3.1.2 RBF Kernel

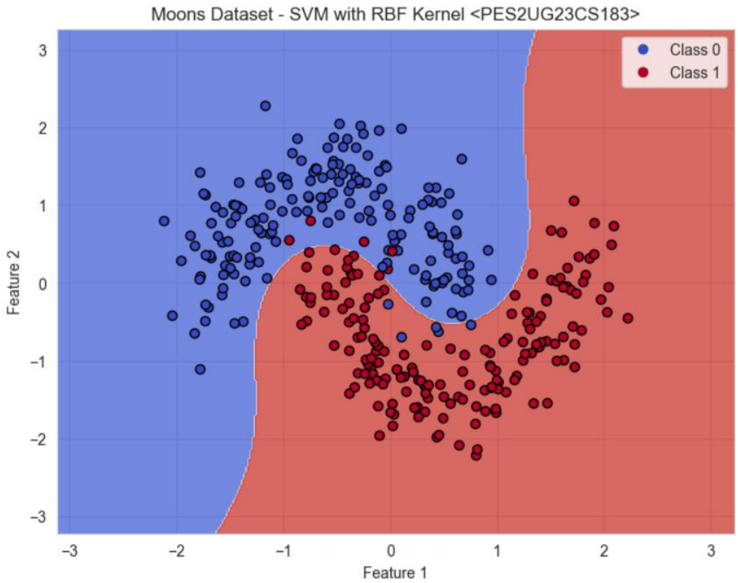
Screenshot of classification report

```
SVM with RBF Kernel <PES2UG23CS183>
              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
```

Decision boundary plot



3.1.3 Polynomial Kernel

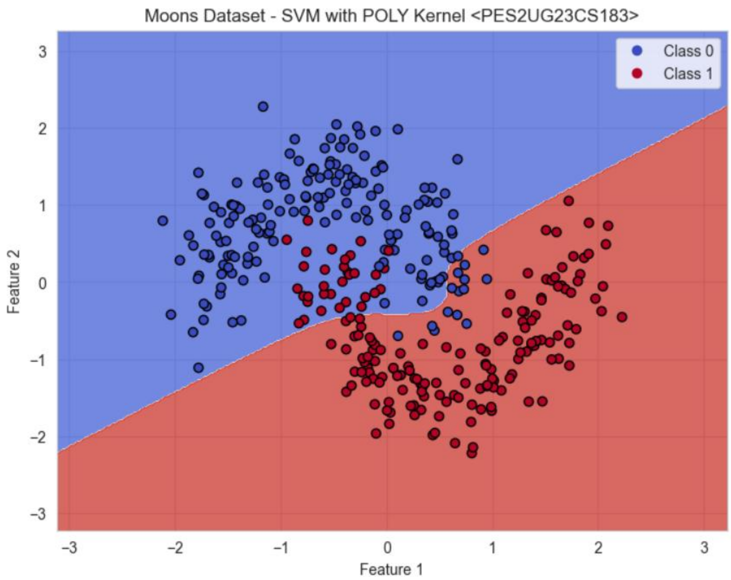
Screenshot of classification report

```
SVM with POLY Kernel <PES2UG23CS183>
              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
```

Decision boundary plot



3.2 Banknote Authentication Dataset

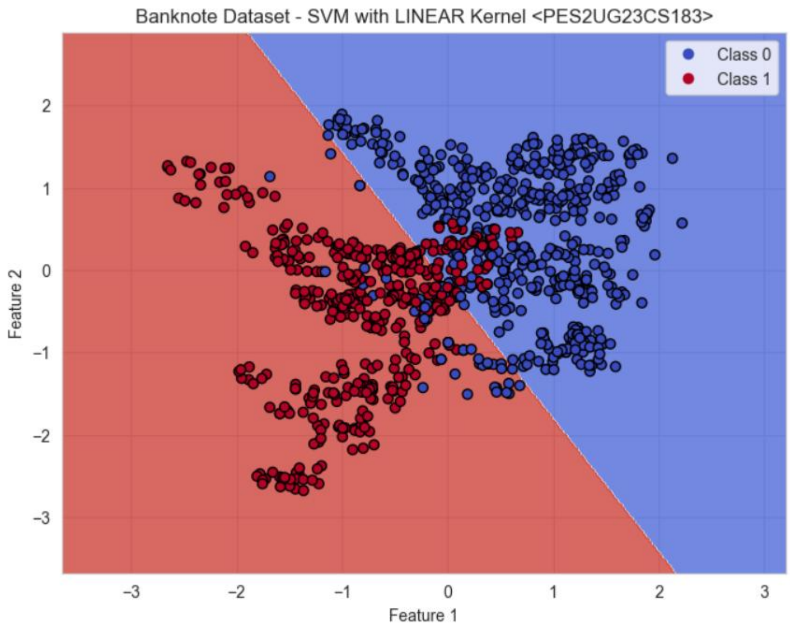
3.2.1 Linear Kernel

Screenshot of classification report

SVM with LINEAR Kernel <PES2UG23CS183>

	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

Decision boundary plot



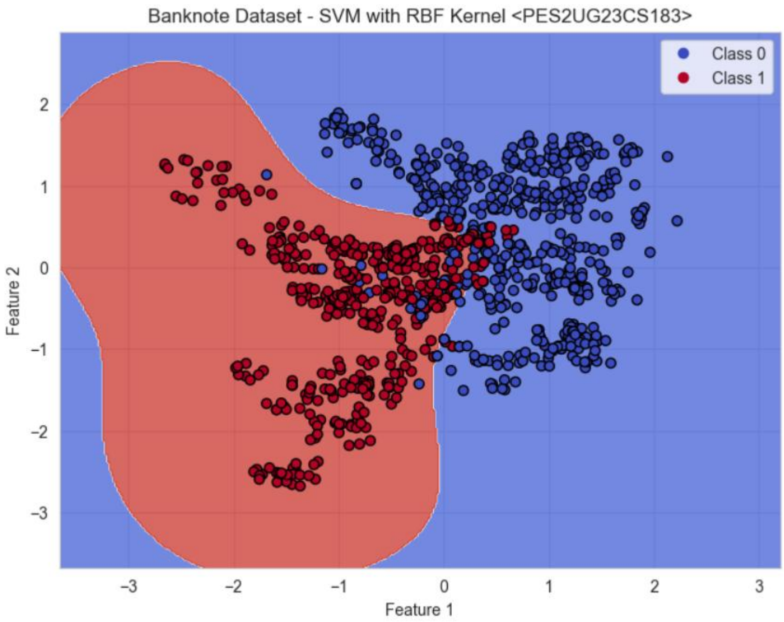
3.2.2 RBF Kernel

Screenshot of classification report

SVM with RBF Kernel <PES2UG23CS183>

	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

Decision boundary plot



3.2.3 Polynomial Kernel

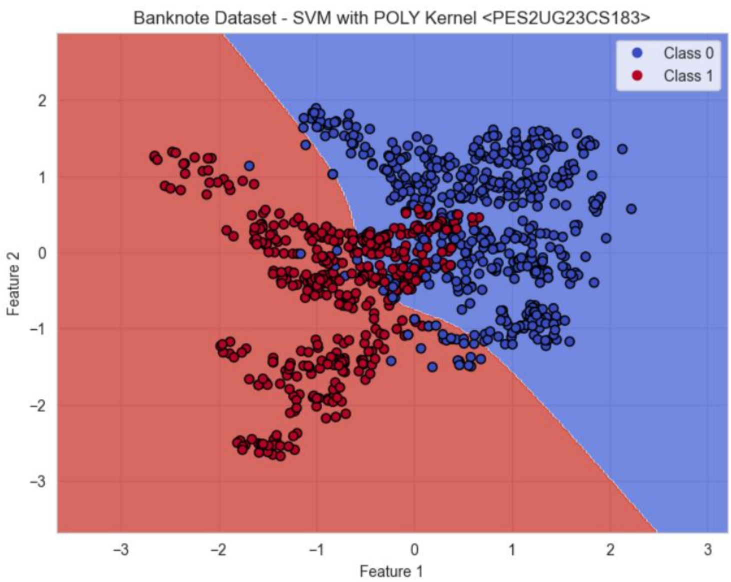
Screenshot of classification report

```
SVM with POLY Kernel <PES2UG23CS183>
              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
```

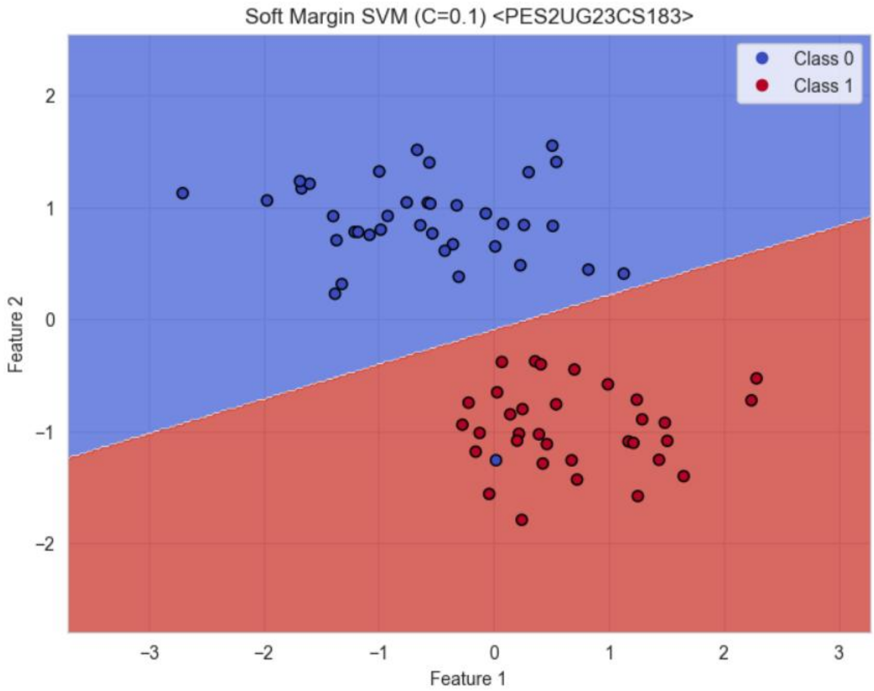
Decision boundary plot



3.3 Margin Analysis

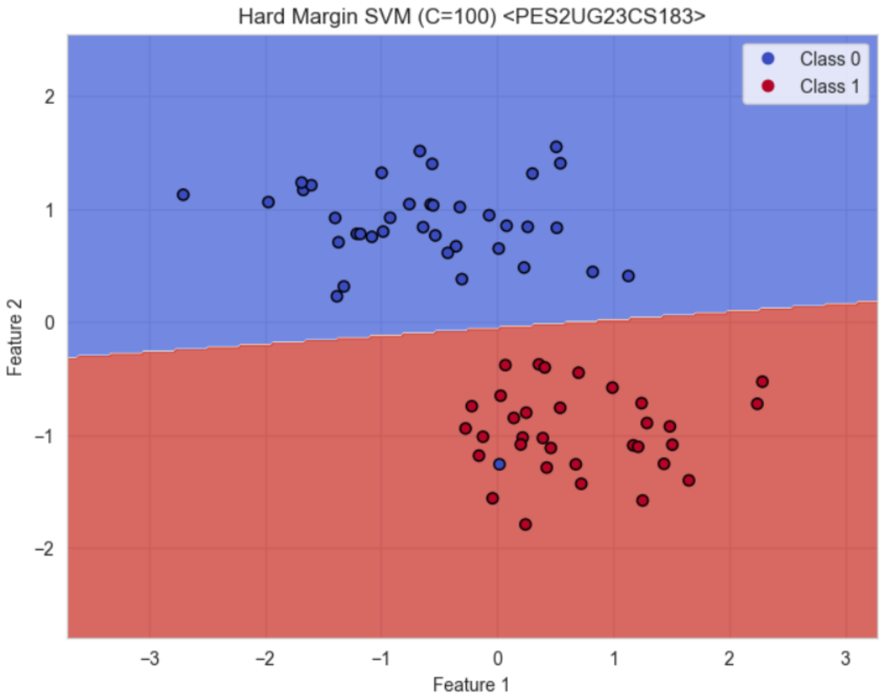
3.3.1 Soft Margin (C=0.1)

Screenshot of classification report and plot



3.3.2 Hard Margin (C=100)

Screenshot of classification report and plot.



5. Analysis and Discussion

5.1 Moons Dataset Analysis

1. Inferences about the Linear Kernel's performance.

The linear kernel performed poorly on the Moons dataset as its straight-line decision boundary cannot capture the nonlinear crescent shapes. This resulted in high misclassification and low accuracy compared to nonlinear kernels.

2. Comparison between RBF and Polynomial kernel decision boundaries.

The RBF kernel formed a smooth, curved boundary that naturally matched the moon shapes, whereas the polynomial kernel generated an irregular, wavy boundary. The RBF kernel achieved better separation with fewer misclassified points.

5.2 Banknote Dataset Analysis

1. Which kernel was most effective for this dataset?

The linear kernel proved to be the most effective, achieving high accuracy with a clear and simple decision boundary. The data seems to be linearly separable, so the use of more complex kernels is not needed.

2. Why might the Polynomial kernel have underperformed here?

The polynomial kernel added unnecessary complexity to linearly separable data, creating overly flexible boundaries that may fit noise. This reduced generalization performance compared to the simpler linear approach.

5.3 Hard vs Soft Margin Analysis

1. Which margin (soft or hard) is wider?

The soft margin (C=0.1) is wider, allowing more data points within the margin bands.

2. Why does the soft margin model allow "mistakes"?

Soft margin tolerates misclassification to maximize margin width and improve generalization. It prioritizes finding a robust boundary over perfect training accuracy.

3. Which model is more likely to be overfitting and why?

The hard margin model (C=100) is more likely to overfit as it creates narrow margins to perfectly classify training data. This makes it sensitive to noise and outliers.

4. Which model would you trust more for new data and why?

The soft margin model is more trustworthy for new data due to its wider margin and better generalization capability. It avoids overfitting to training noise and performs more reliably on unseen samples.

5. Conclusion

From this lab, we learned how the Support Vector Machine (SVM) algorithm separates data using different kernels and margin settings. We saw that the RBF kernel worked best for the Moons dataset because it could handle curved, non-linear data, while the Linear kernel performed best for the Banknote dataset, which was mostly linearly separable.

We also observed that the Soft Margin (C = 0.1) created a wider margin and gave better generalization, while the Hard Margin (C = 100) tried to classify every point correctly but could overfit. Overall, this experiment helped us understand how choosing the right kernel and C value affects the accuracy and flexibility of an SVM model.