

ML Lab Week 10: SVM Classifier Lab

Name:
Dinakar Emmanuel

SRN:
PES2UG23CS178

Section:
C

Moons Dataset

1. Inferences about the Linear Kernel's performance.

The Linear Kernel has an accuracy of 0.87. The precision, recall, and f1-score, indicate that it performs reasonably well on both classes, but the visualization shows that the linear kernel draws a straight line decision boundary, which does not effectively separate the non-linearly separable data and thus, is not able to fully capture the pattern of this dataset.

2. Comparison between RBF and Polynomial kernel decision boundaries.

The RBF kernel captures the shape of the data better, effectively separating the two classes along their natural curve. The Polynomial kernel, also creates a non-linear boundary, but has a comparatively more rigid shape and does not smoothly conform to the contours of the data.

Banknote Dataset

1. Which kernel was most effective for this dataset?

The RBF kernel is the most effective with the highest stats for both classes compared to the linear and polynomial kernels. The RBF kernel's decision boundary also shows a good separation of the two classes.

2. Why might the polynomial kernel have underperformed here?

Moons dataset	Banknote dataset
It has a clear non-linear separation that can be effectively captured by the polynomial kernel.	The boundary, though non-linear, is more intricate. The polynomial kernel cannot effectively capture the pattern of this dataset.

Hard vs Soft Margin

1. Which margin is wider?

The characteristic of a soft margin SVM is that it is supposed to produce a wider margin. This allows it to tolerate a few misclassifications. Looking at the plots, it looks like the soft margin SVM produces a narrower band between the classes. Using 'coef_' to get the w vectors and the width using $2/||w||$, we can see that the "Soft Margin" (C=0.1) model produces a wider margin (1.6779). The hard margin model tries to strictly separate the points, including outliers, resulting in a narrower margin (0.7560). The soft margin model is more tolerant to misclassifications.

2. Why does the soft margin allow “mistakes”?

A soft margin SVM model allows some data points to be misclassified or to fall within the margin. This is controlled by the hyperparameter C . A smaller C value (0.1) means the penalty for misclassification is lower, making the model more tolerant to these mistakes.

The primary goal of a Soft Margin SVM is to find a balance between maximizing the margin and minimizing the training errors. By allowing some misclassifications, the soft margin model can find a wider margin that generalizes better, making it more robust to noise and outliers in the dataset.

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

Hard margin is more likely to overfit to the training data because:

- Large C values have high penalties for misclassifications and try to correctly classify every training data point, including outliers and noise.
- The hard margin SVM produces a narrower margin and the boundary is highly influenced by outliers.
- This may result in low training errors but could perform poorly with test data.

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

I trust the soft margin more as it is less prone to overfitting and generalises better. The hard margin is more likely to have more misclassifications than the soft margin with test data. With noisy data, the C value should preferably be a lower value as it promotes generalisation. But the C value has to be tuned to prevent underfitting.

Training results:

Moon dataset

```
SVM with LINEAR Kernel <PES2UG23CS178>
      precision    recall  f1-score   support

         0         0.85      0.89      0.87         75
         1         0.89      0.84      0.86         75

 accuracy          0.87          0.87          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 <PES2UG23CS178>
      precision    recall  f1-score   support

         0         0.95      1.00      0.97         75
         1         1.00      0.95      0.97         75

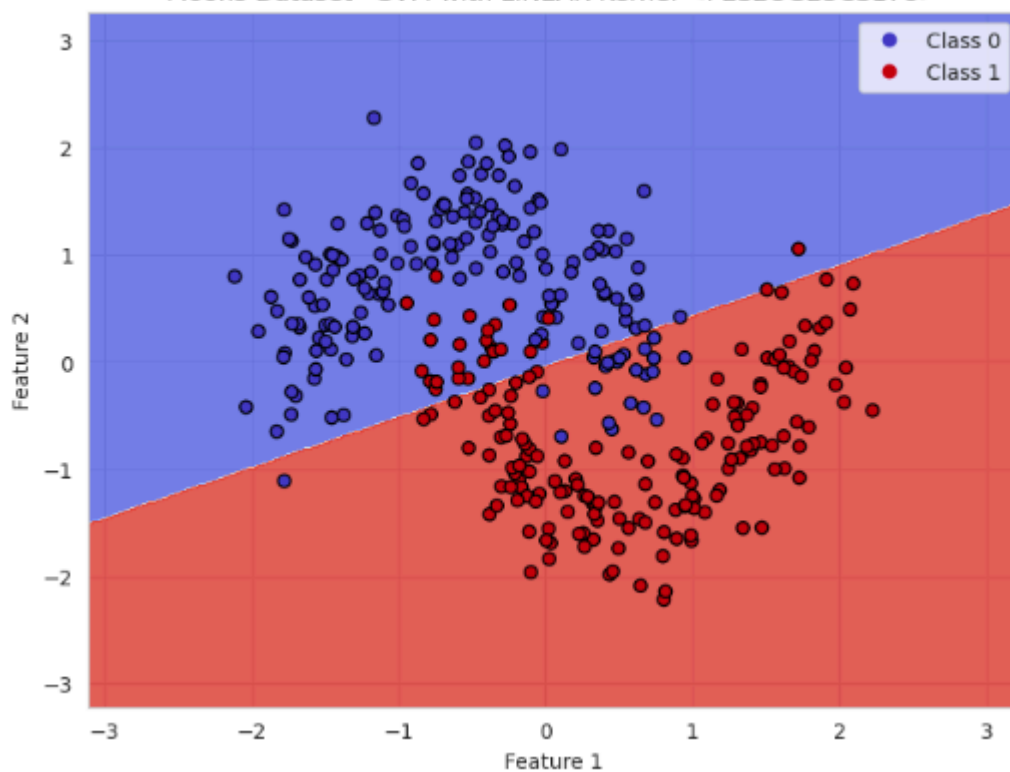
 accuracy          0.97          0.97          0.97        150
 macro avg         0.97      0.97      0.97        150
 weighted avg      0.97      0.97      0.97        150
```

```
-----
SVM with POLY Kernel <PES2UG23CS178>
      precision    recall  f1-score   support

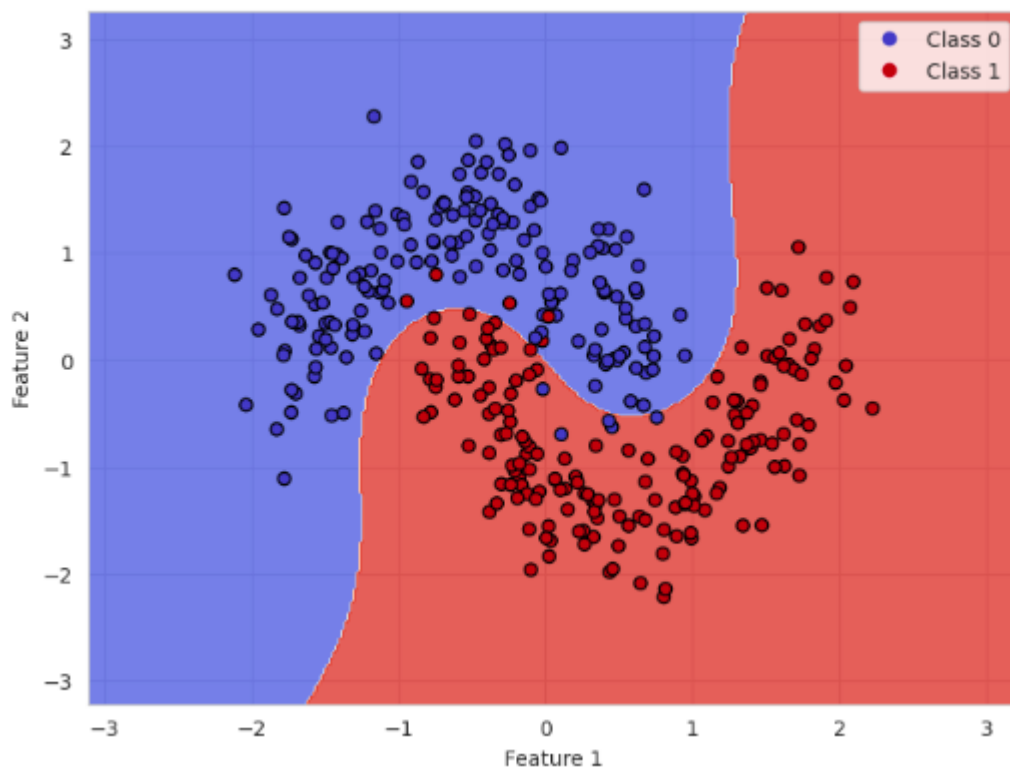
         0         0.85      0.95      0.89         75
         1         0.94      0.83      0.88         75

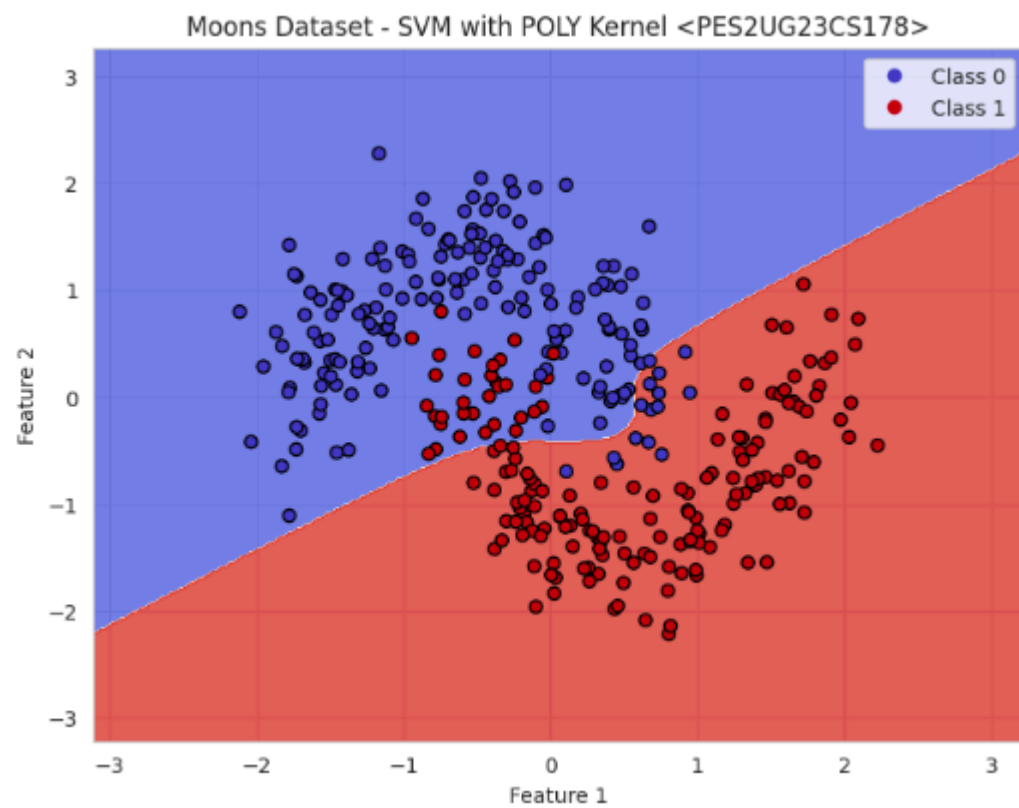
 accuracy          0.89          0.89          0.89        150
 macro avg         0.89      0.89      0.89        150
 weighted avg      0.89      0.89      0.89        150
```

Moons Dataset - SVM with LINEAR Kernel <PES2UG23CS178>



Moons Dataset - SVM with RBF Kernel <PES2UG23CS178>





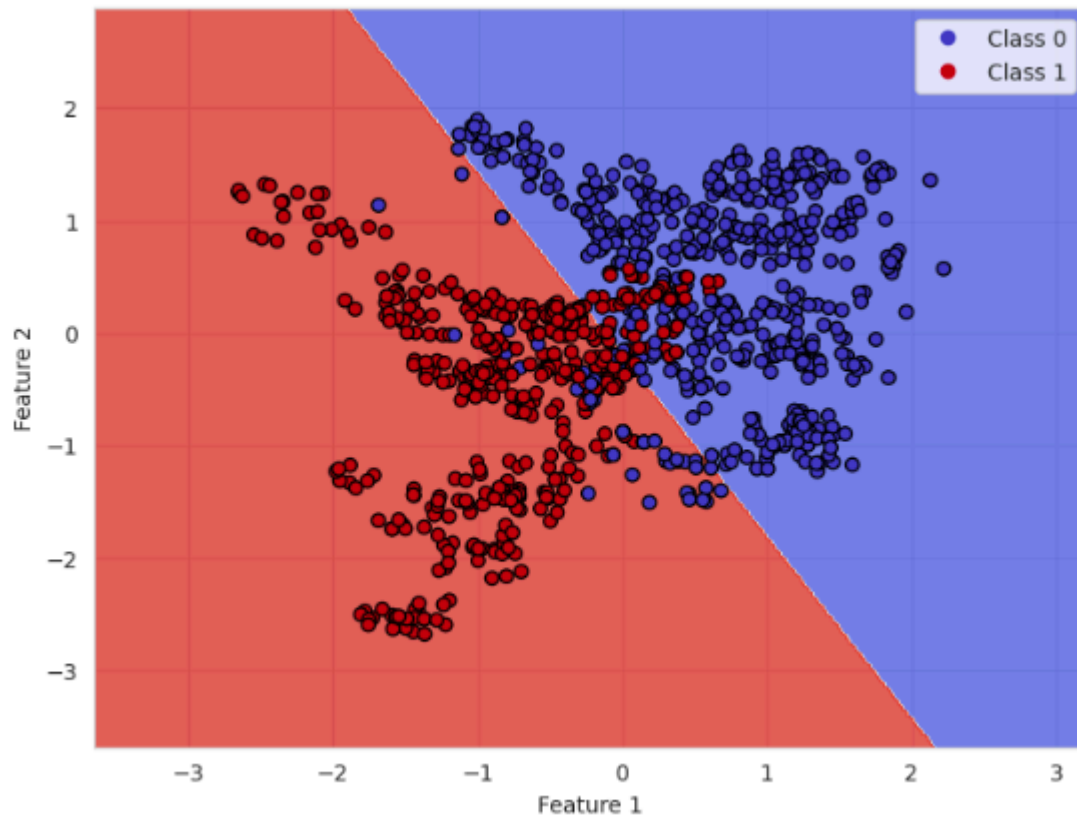
Banknote dataset:

SVM with LINEAR Kernel <PES2UG23CS178>				
	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

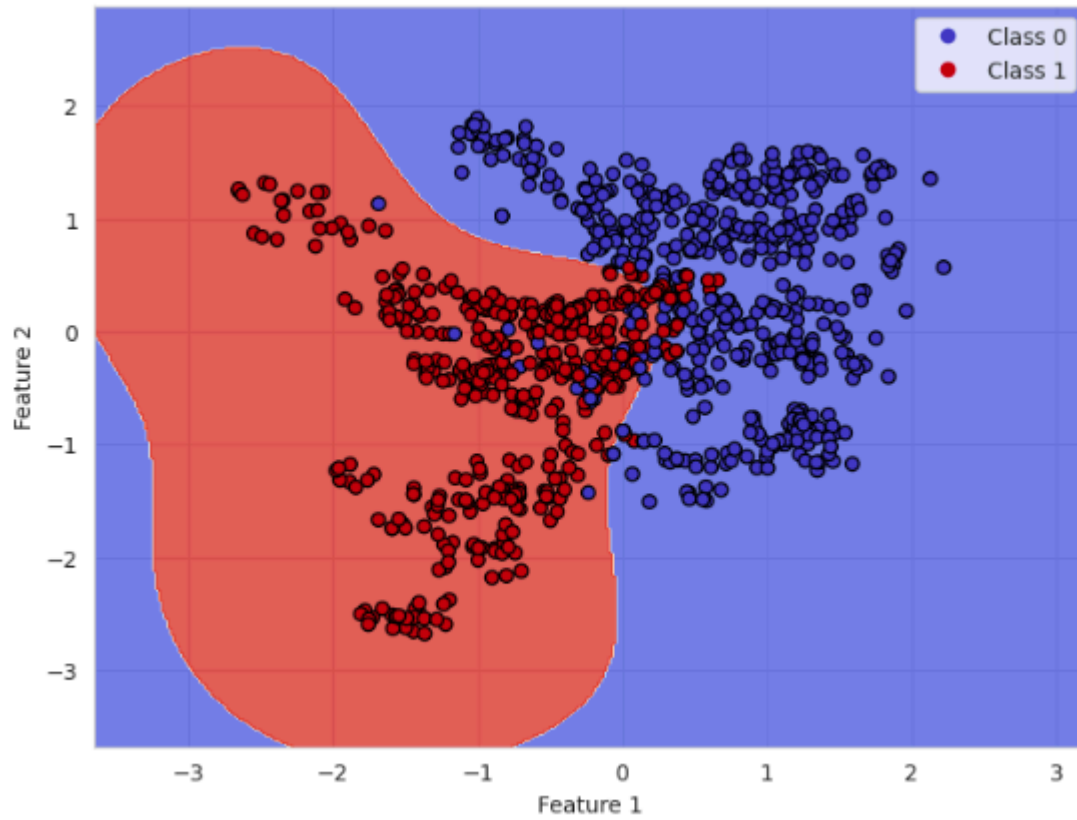
SVM with RBF Kernel <PES2UG23CS178>				
	precision	recall	f1-score	support
Forged	0.96	0.91	0.94	229
Genuine	0.90	0.96	0.93	183
accuracy			0.93	412
macro avg	0.93	0.93	0.93	412
weighted avg	0.93	0.93	0.93	412

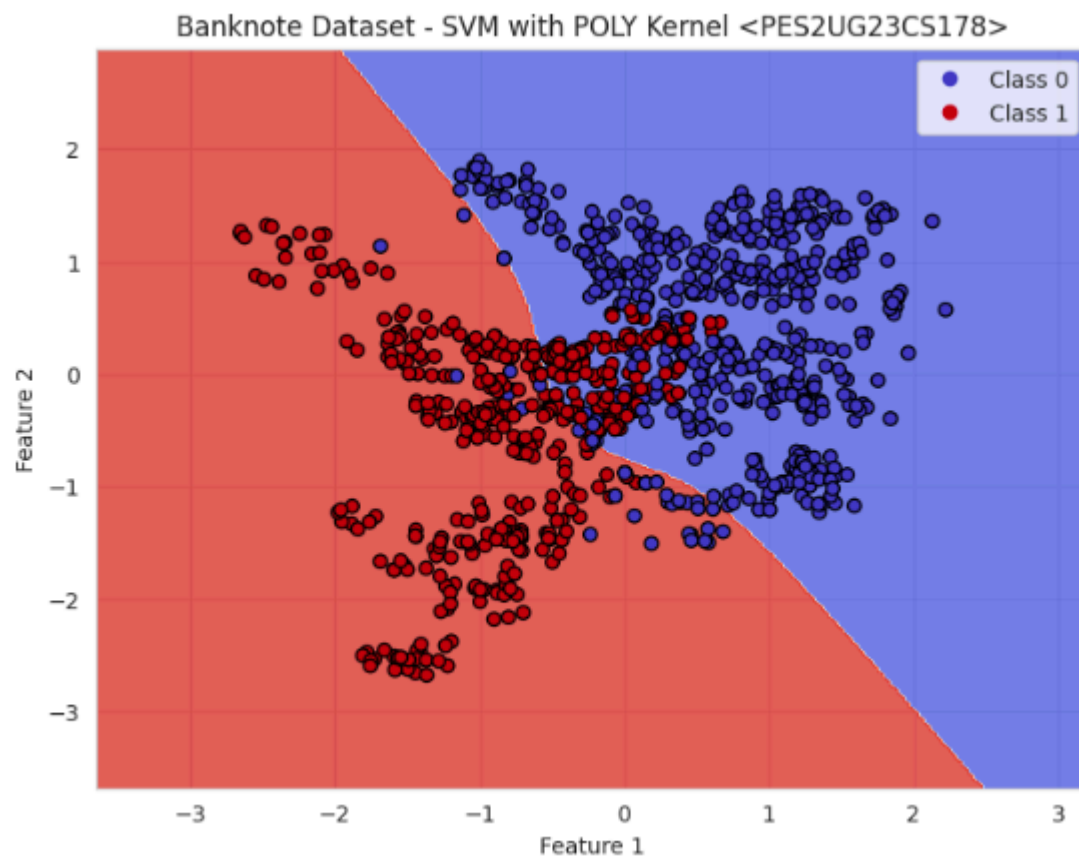
SVM with POLY Kernel <PES2UG23CS178>				
	precision	recall	f1-score	support
Forged	0.82	0.91	0.87	229
Genuine	0.87	0.75	0.81	183
accuracy			0.84	412
macro avg	0.85	0.83	0.84	412
weighted avg	0.85	0.84	0.84	412

Banknote Dataset - SVM with LINEAR Kernel <PES2UG23CS178>



Banknote Dataset - SVM with RBF Kernel <PES2UG23CS178>





Margin Analysis:

