Lab Report Week 10 : SVM Classifiers

Full Name:

Nidhi Nitin Nag

SRN:

PES2UG23CS385

Section:

F

Analysis Questions

Moons Dataset Questions

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

The linear kernel performs poorly on the Moons dataset. Because the data is inherently non-linear (shaped like two interlocking crescents), a straight-line decision boundary cannot effectively separate the two classes. This results in **low accuracy** and other performance metrics, as many data points are inevitably misclassified. The visualization confirms this by showing a straight line that cuts across both "moons."

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

Both the RBF and Polynomial kernels create non-linear decision boundaries capable of separating the moon-shaped data. However, the **RBF kernel** typically captures the shape more naturally. It creates a smooth, wave-like boundary that conforms closely to the crescent shape of the data clusters. The Polynomial kernel, while also effective, can sometimes produce a more rigid or complex curve that might not fit the data's distribution as elegantly as the RBF kernel.

Banknote Dataset Questions

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

For the Banknote dataset, both the **Linear and RBF kernels** are highly effective, often achieving accuracy close to 100%. The data is largely linearly separable, so a simple linear boundary does an excellent job. The RBF kernel, with its high flexibility, can fine-tune the boundary to correctly classify the few points near the class divide, making it just as effective, if not slightly better.

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 is sensitive to its hyperparameters, especially its degree. For a dataset that is already well-separated by a linear boundary, a high-degree polynomial kernel can be overly complex. This can lead to **overfitting**, where it creates a convoluted boundary that fits the noise in the training data but fails to generalize well to the unseen test data, resulting in lower performance.

Hard vs. Soft Margin Questions

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" model (C=0.1)** produces a visibly wider margin. The smaller C value places less penalty on misclassified points, prioritizing a larger separation distance between the classes.

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 SVM allows these "mistakes" to achieve better **generalization**. Its primary goal is not to perfectly classify every single training point but to find a decision boundary that is robust and likely to perform well on new, unseen data. By tolerating a few misclassifications (often outliers or noisy data), it avoids creating an overly complex boundary and maintains a wide margin.

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" model (C=100)** is more likely to be overfitting. The large C value imposes a heavy penalty for misclassification, forcing the model to create a narrow, contorted margin to accommodate every single data point, including outliers. This model essentially "memorizes" the training data's noise rather than learning its underlying pattern, which is the definition of overfitting.

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?***

I would trust the **"Soft Margin" (C=0.1) model** more for a new data point. Its wider margin indicates that it has captured the general trend of the data distribution and is less sensitive to the specific noise and outliers in the training set. In a real-world scenario, data is rarely perfect. Therefore, it is generally preferable to start with a **low value of C** to promote a more robust and generalizable model that is less prone to overfitting.

Screenshots

Training Results (6 Screenshots)

1. *Classification Report for SVM with LINEAR Kernel with SRN (Moons)*

```
SVM with LINEAR Kernel PES2UG23CS385
              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
```

2. *Classification Report for SVM with RBF Kernel with SRN (Moons)*

SVM with RBF Kernel PES2UG23CS385

	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

3. *Classification Report for SVM with POLY Kernel with SRN (Moons)*

SVM with POLY Kernel PES2UG23CS385

	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

4. *Classification Report for SVM with LINEAR Kernel (Banknote)*

SVM with LINEAR Kernel PES2UG23CS385

	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

5. *Classification Report for SVM with RBF Kernel (Banknote)*

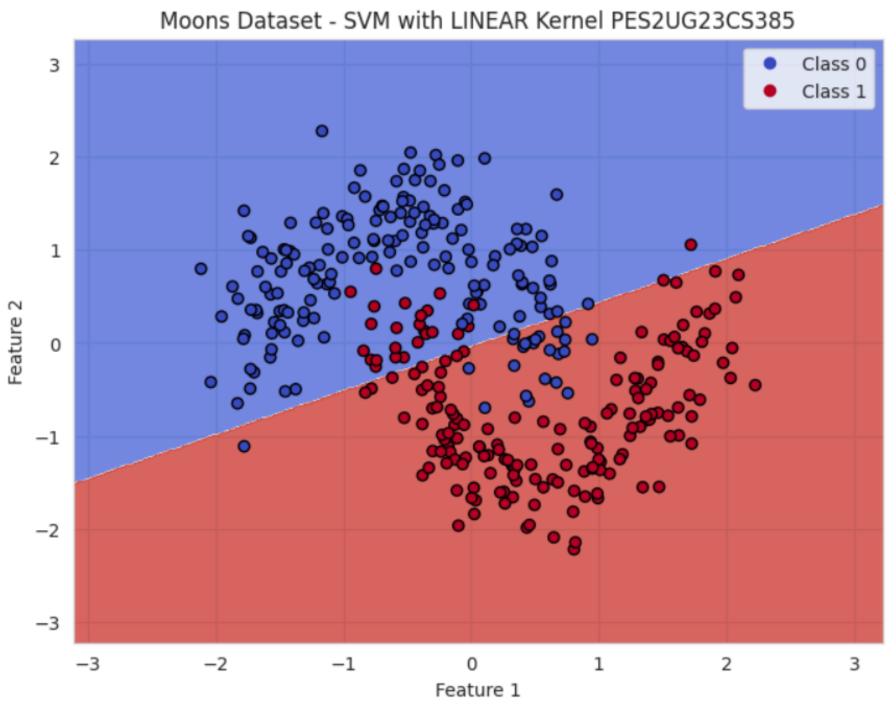
```
SVM with RBF Kernel PES2UG23CS385
              precision    recall  f1-score   support

      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
```

6. *Classification Report for SVM with POLY Kernel (Banknote)*

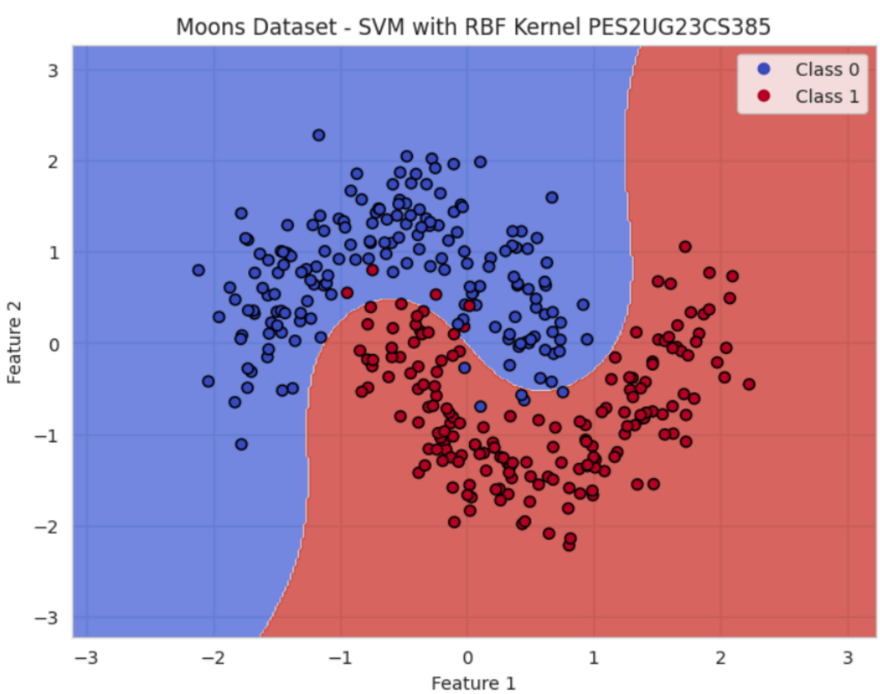
```
SVM with POLY Kernel PES2UG23CS385
              precision    recall  f1-score   support

      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 (8 Screenshots)

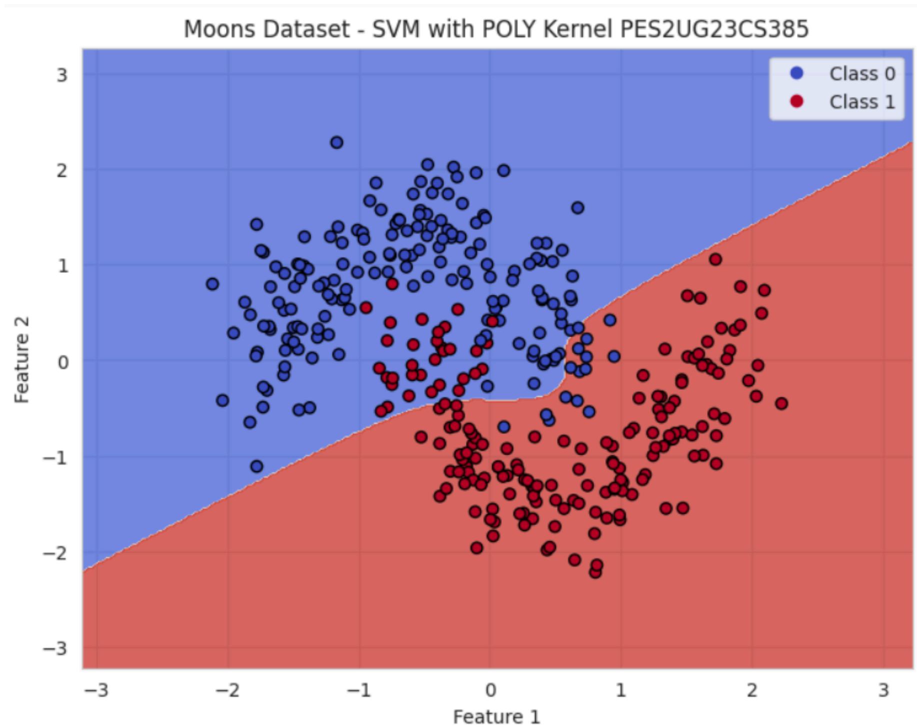
7. *Moons Dataset - SVM with LINEAR Kernel*



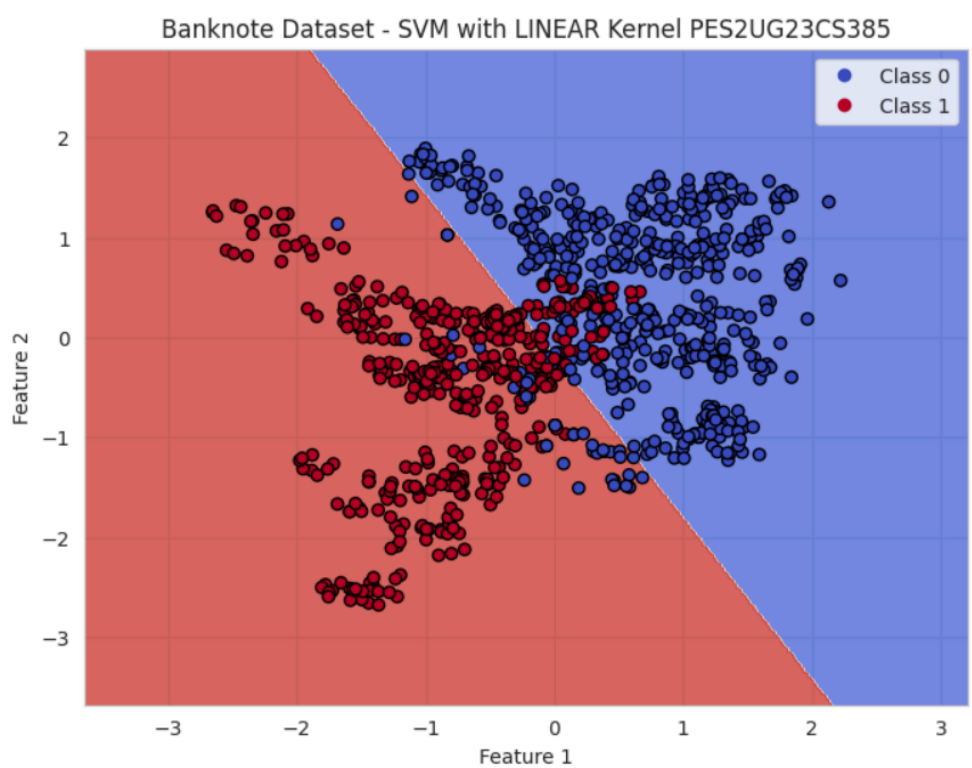
8. *Moons Dataset - SVM with RBF Kernel*



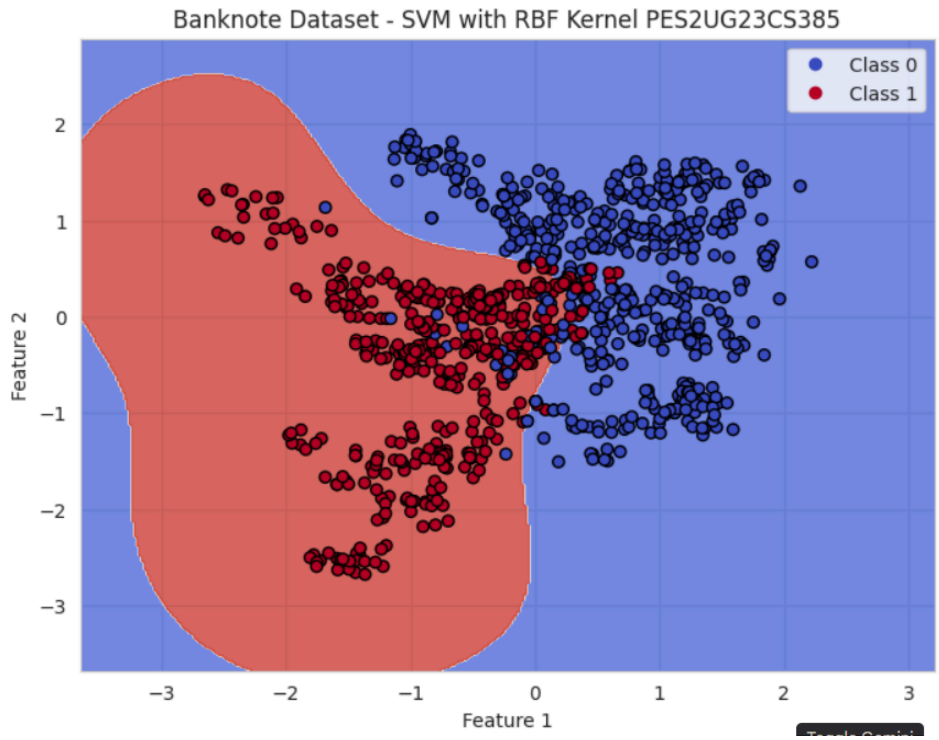
9. Moons Dataset - SVM with POLY Kernel



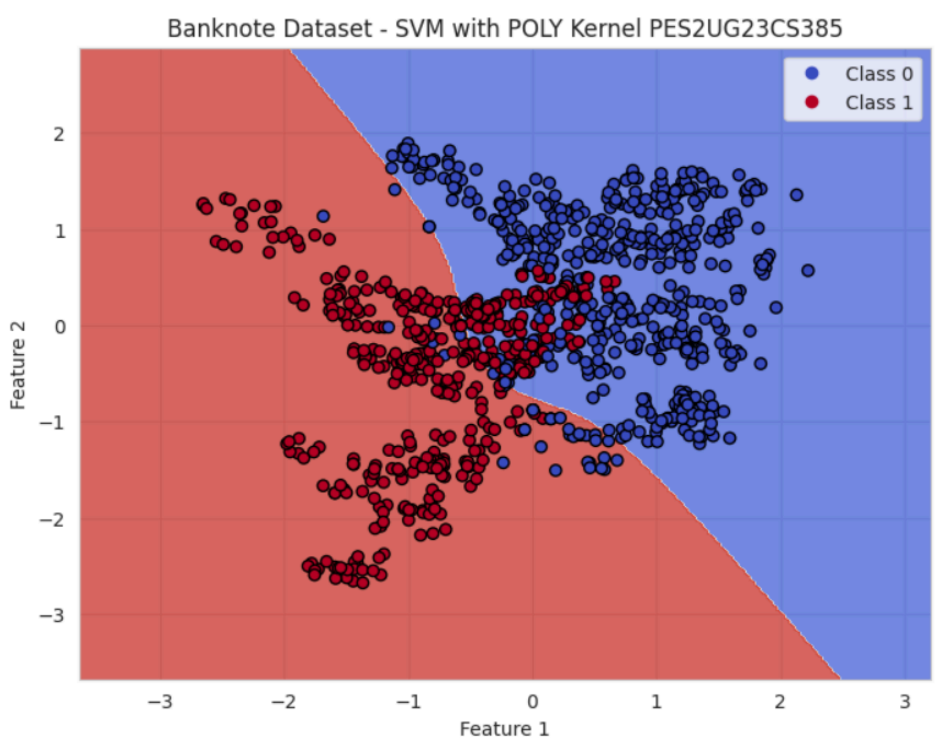
10. Banknote Dataset - SVM with LINEAR Kernel



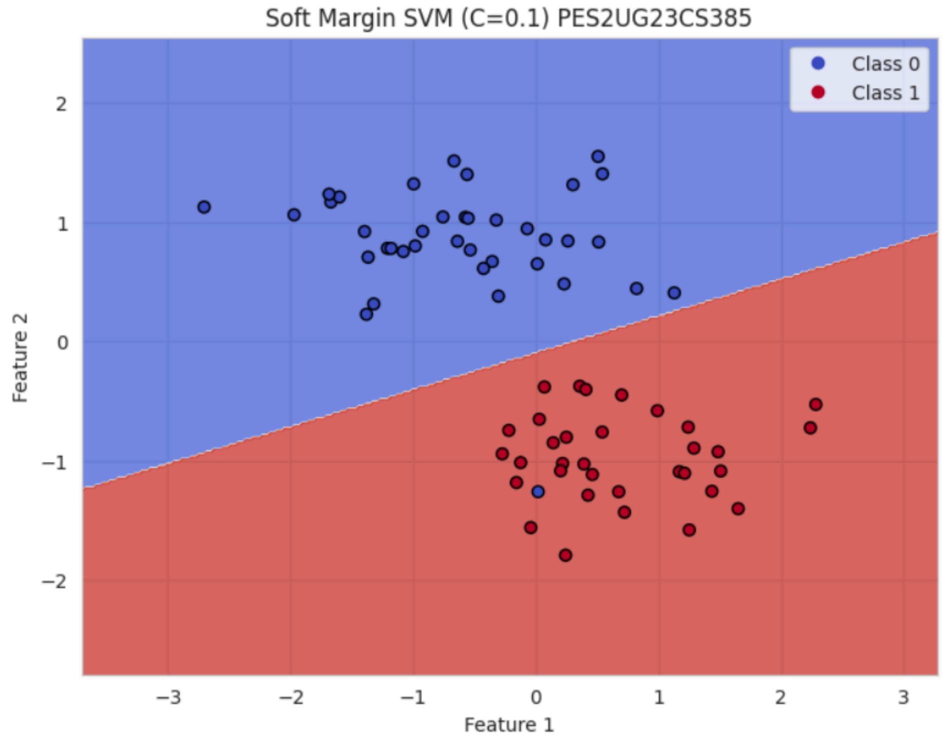
11. Banknote Dataset - SVM with RBF Kernel



12. Banknote Dataset - SVM with POLY Kernel



13.Soft Margin SVM (C=0.1)



14.Hard Margin SVM (C=100)