MACHINE LEARNING

LAB – WEEK 10

TITLE: Support Vector Machine(SVM)

NAME: Nandana Mathew

SRN: PES2UG23CS913

COURSE: Machine Learning

SECTION: ‘F’

DATE: 11/10/2025

INTRODUCTION:

The purpose of this lab is to get a hands-on experience of how the SVM is implemented using three different kernels: Linear, RBF and Polynomial, on distinct datasets. Through this lab I could infer the different performances of the three types of kernels on the distinct datasets and visualise their decision boundaries.

DATASETS:

1. Moons dataset

2. Banknote Authentication dataset

ANALYSIS:

PART 1

1. The Linear Kernel performance is poor on the Moons dataset.A linear kernel is used to create a straight-line decision boundary between the two class. It is clearly seen that its does not provide a clear distinction between the 2 classes. Some of the points of class 0 falls under class 1 and vice versa. Hence, we can conclude that the dataset cannot be separated using a linear boundary as it is distributed unevenly in wave shape.

2. RBF Kernel: The RBF (Radial Basis Function) kernel creates a very complex, localized, and flexible boundary like a circle or wave which is required for this dataset as it is distributed as a wave form across the boundary.

Polynomial Kernel: The Polynomial kernel creates a curved, polynomial boundary. This boundary is less localized than RBF and is governed by a degree parameter. As seen, it forms a non-linear boundary trying to fit the dataset but generalised. Hence It does not perfectly fit the dataset.

<u>Conclusion</u>: The RBF kernel captures the shape of the data more naturally for highly complex, organic shapes like the Moons dataset because its influence is based on the distance from the support vectors, allowing it to adapt closely to the local structure of the data.

PART 2

1. The Soft Margin (C=0.1) is wider. A smaller C value means there's a higher penalty tolerance for misclassified points. It tries to maximize the margin, even if it means allowing a few errors. This leads to a larger, more general separation corridor.

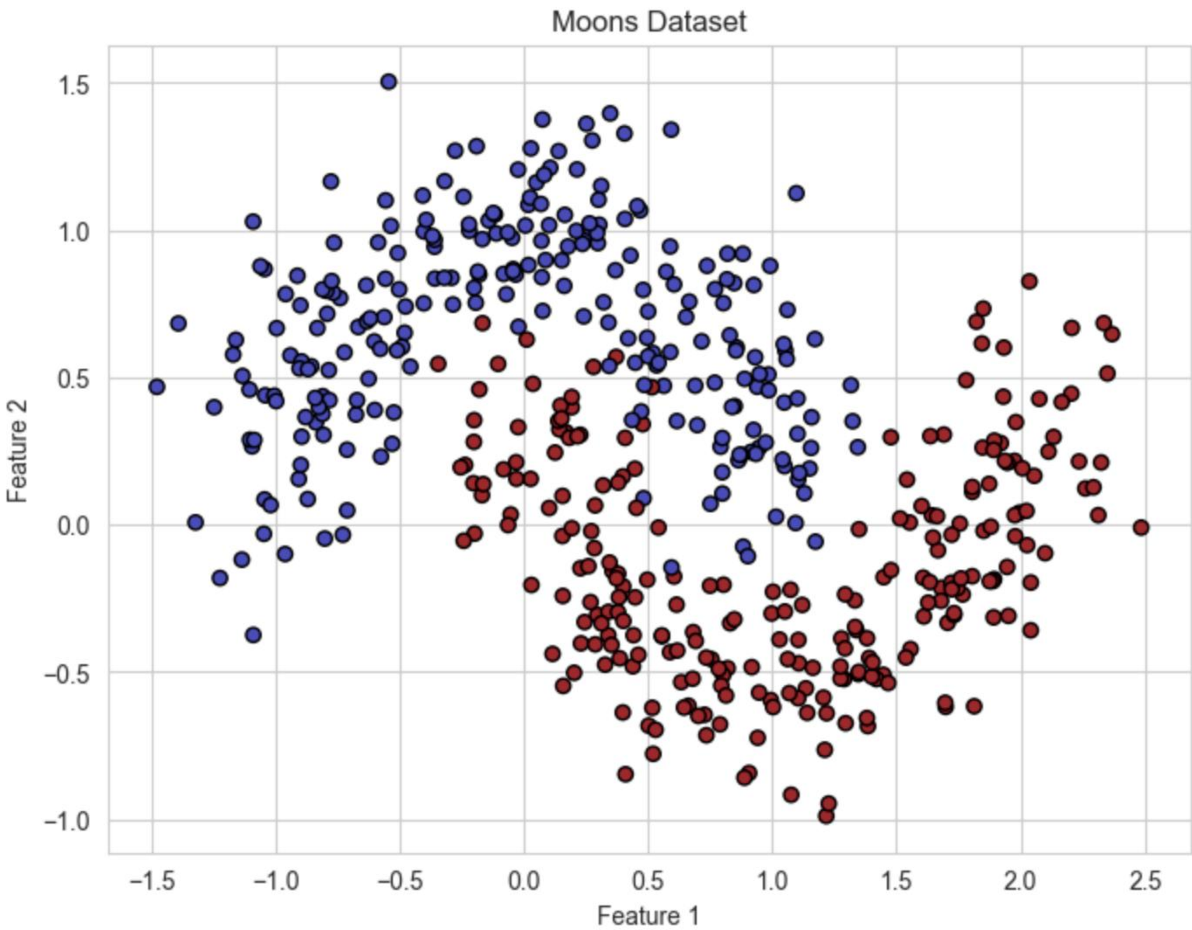
2. Soft margin model main goal is to maximize the margin and minimize the classification error. When C is small, the penalty for making errors is low. The model prioritizes finding a wide margin, which often requires ignoring a few outliers or noisy points. These "mistakes" are necessary to achieve better generalization.

3. The Hard margin (C=100) model is overfitting as it tries to classify each and every point in the training set making a narrow margin. Hence it performs well on the training set but for a test data it performs bad as it tries to fit in all the datapoints even the noise and outliers thereby not predicting the pattern in the dataset.

4. I would trust the Soft Margin model for the new dataset. As the model has a wider margin and high tolerance for training errors. It has a more generalised model than the hard margin model. Hard margin model as mentioned before has trained the model even for the noises and outliers. In a real-world data, the dataset would be more noisy hence Soft Margin model is preferred.

SCREENSHOTS:

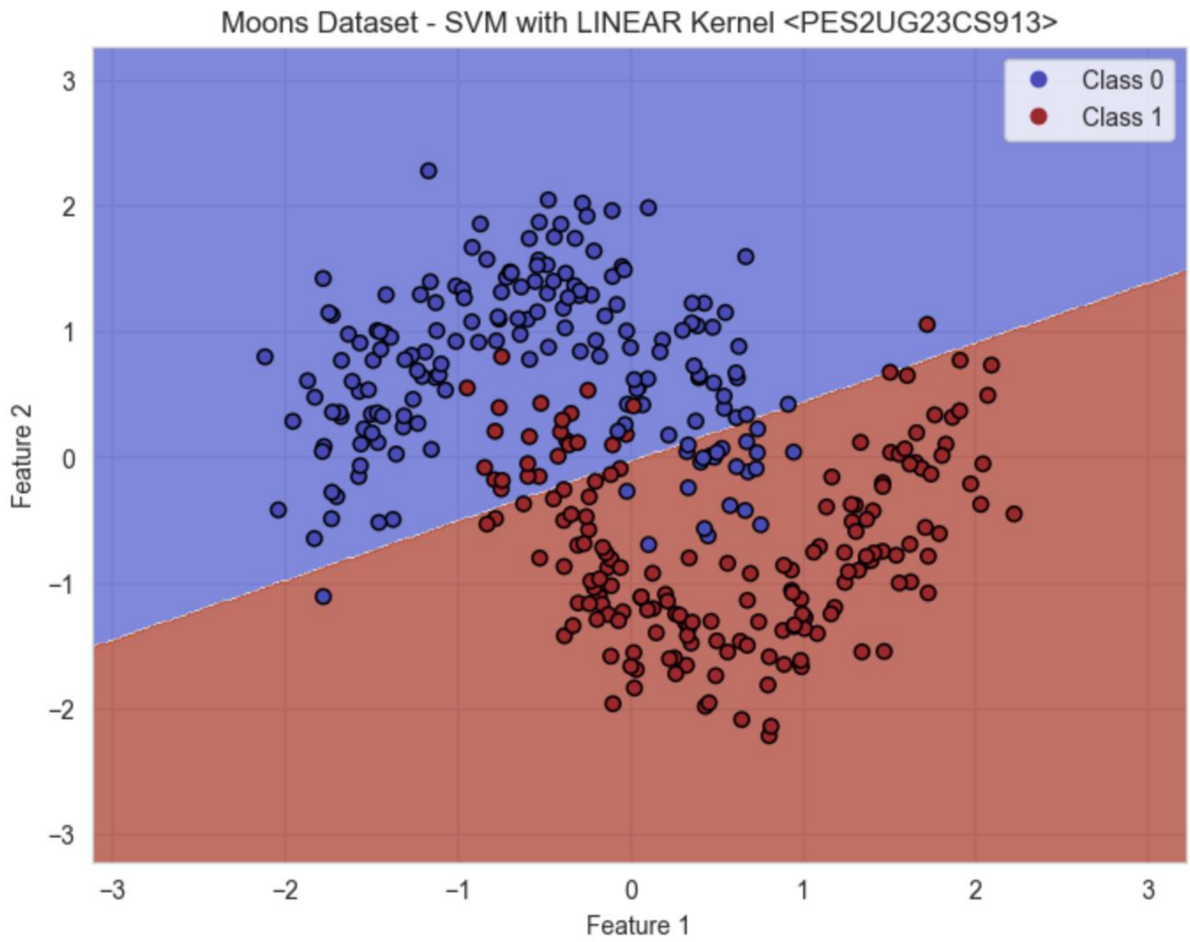
PART 1



1. LINEAR

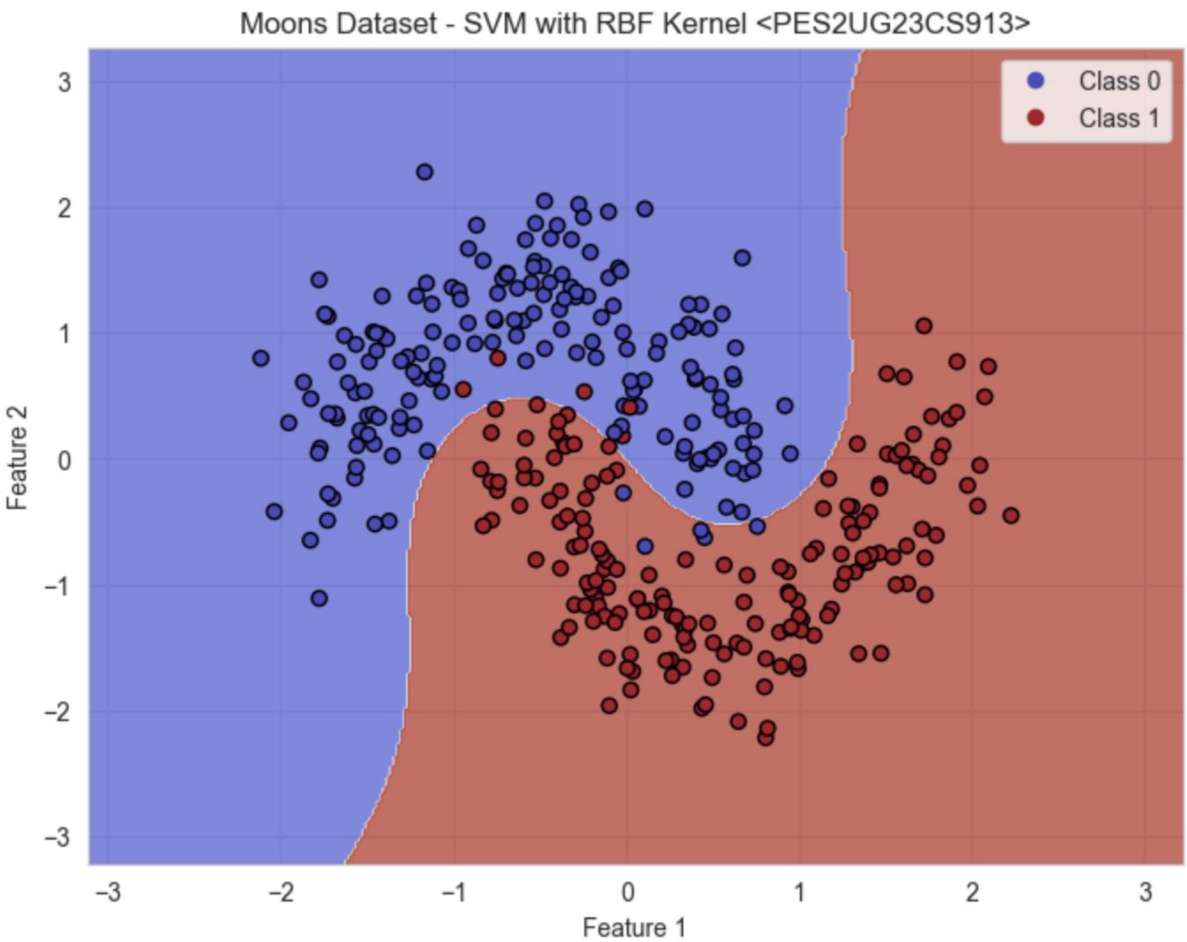
SVM with LINEAR Kernel <PES2UG23CS913>

	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



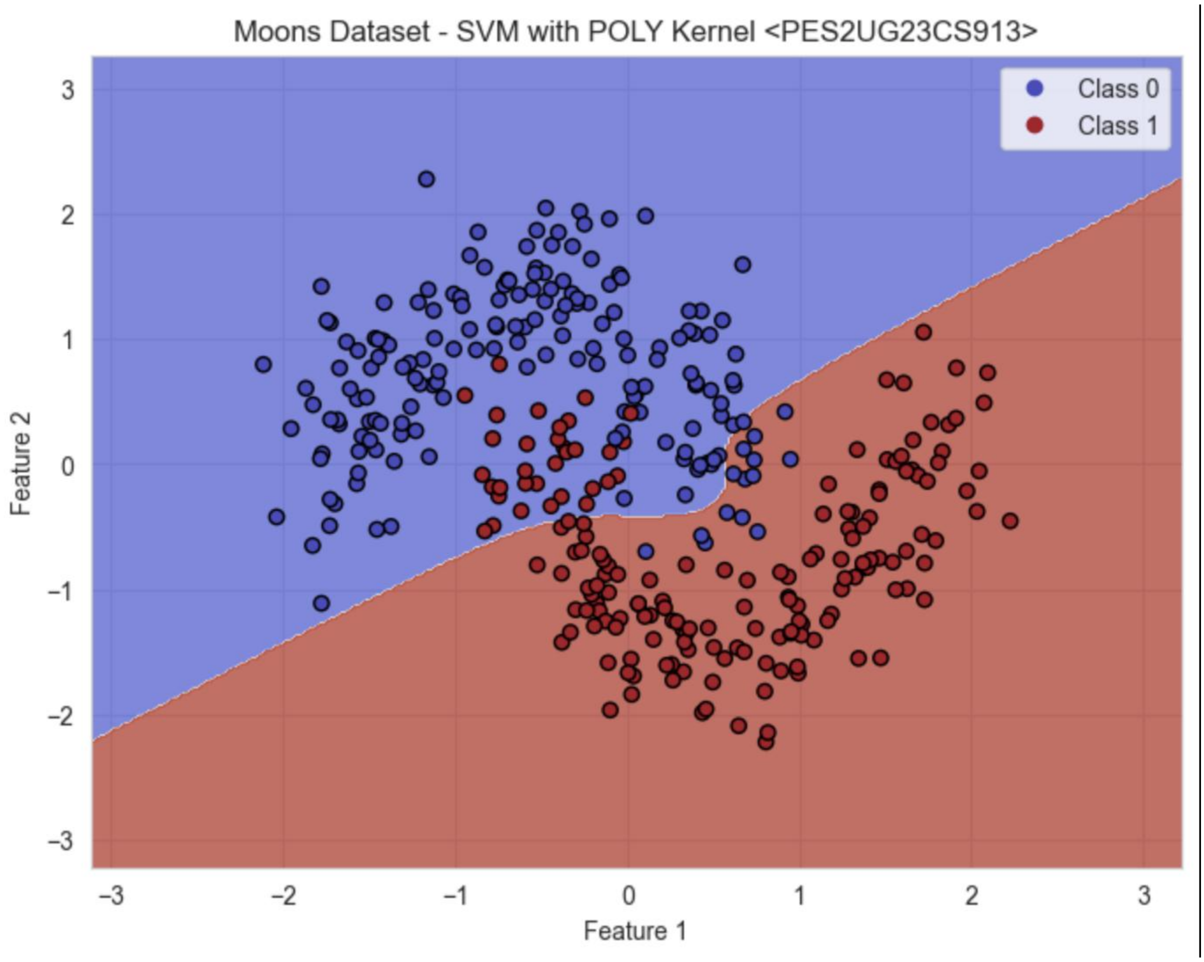
2. RBF

```
SVM with RBF Kernel <PES2UG23CS913>
              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. POLYNOMIAL

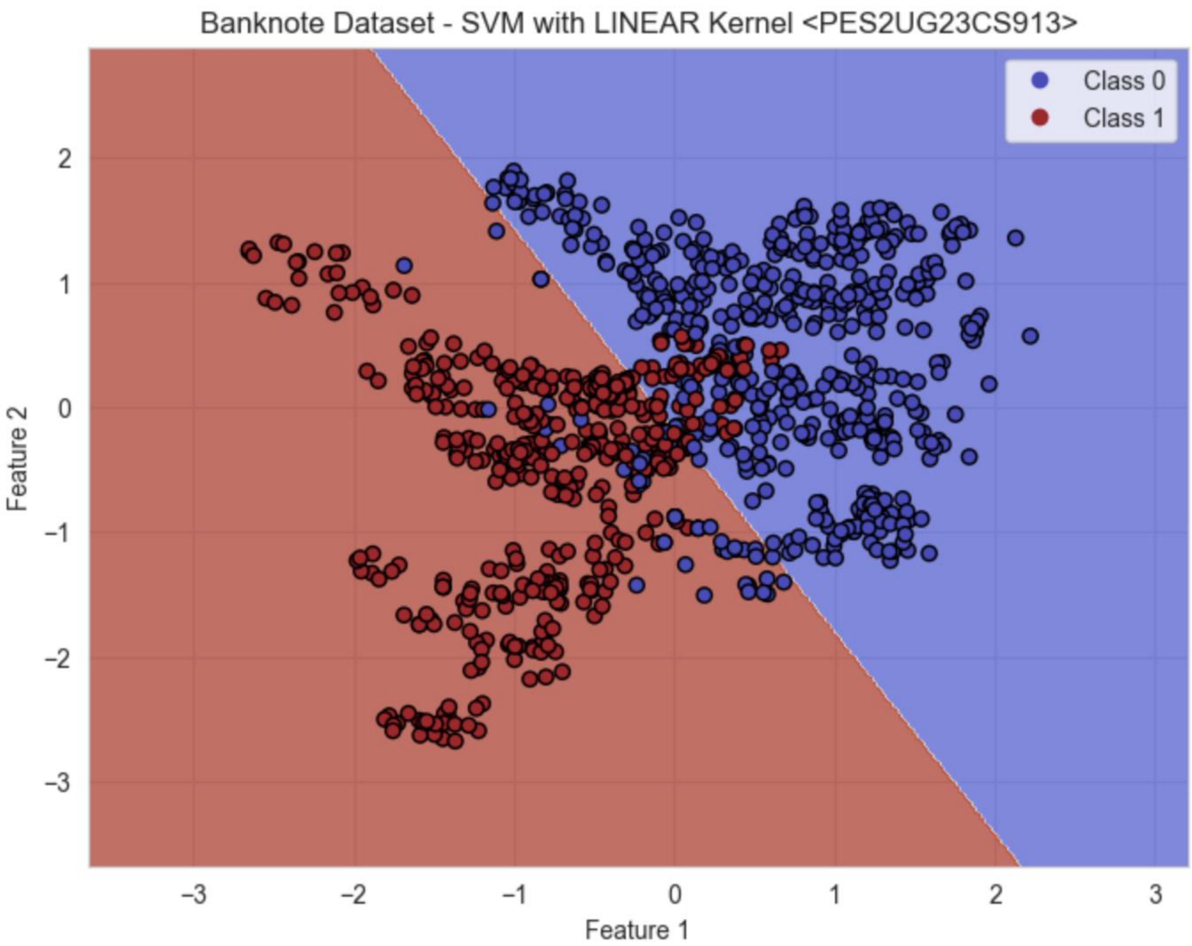
```
SVM with POLY Kernel <PES2UG23CS913>
              precision    recall  f1-score   support

      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
```



PART 2

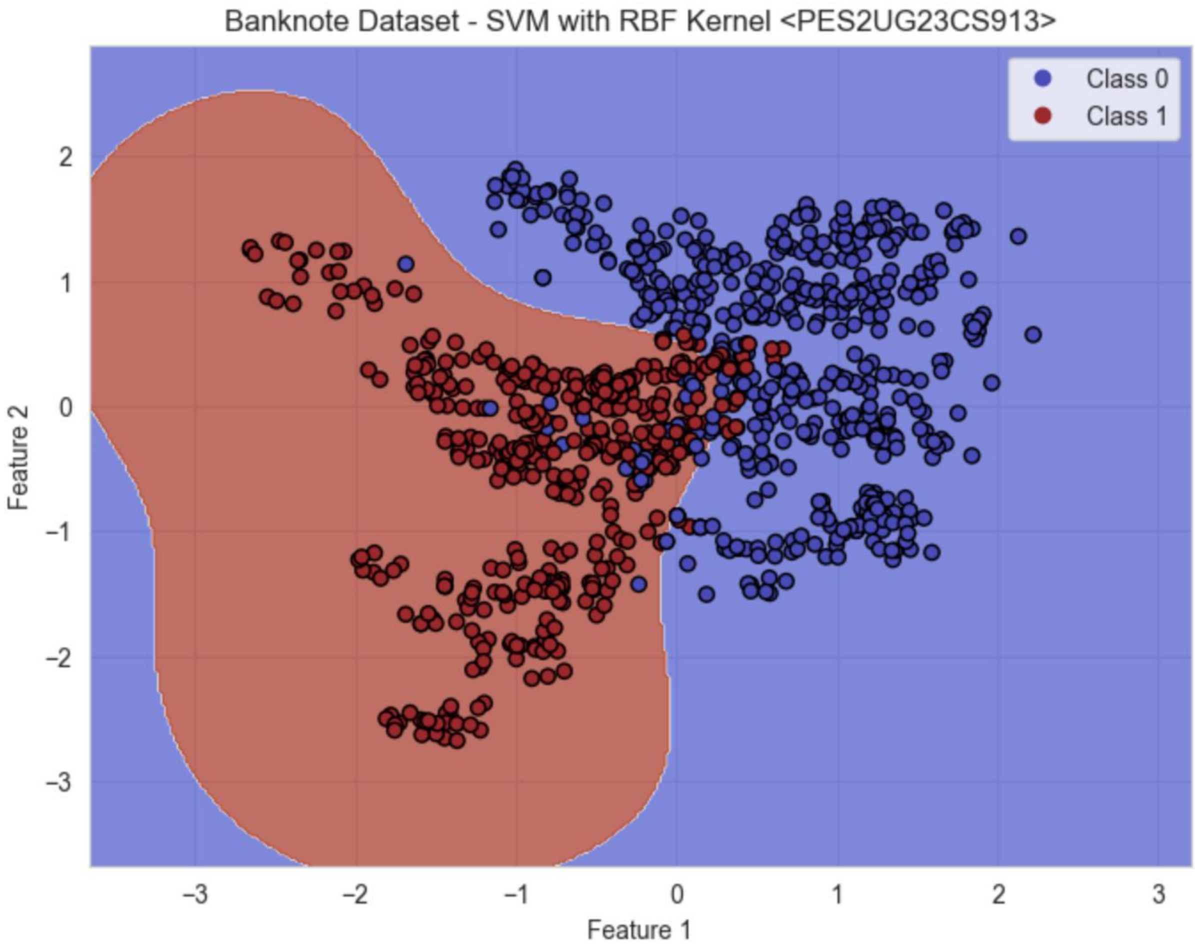
1. LINEAR

```
SVM with LINEAR Kernel <PES2UG23CS913>
              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

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



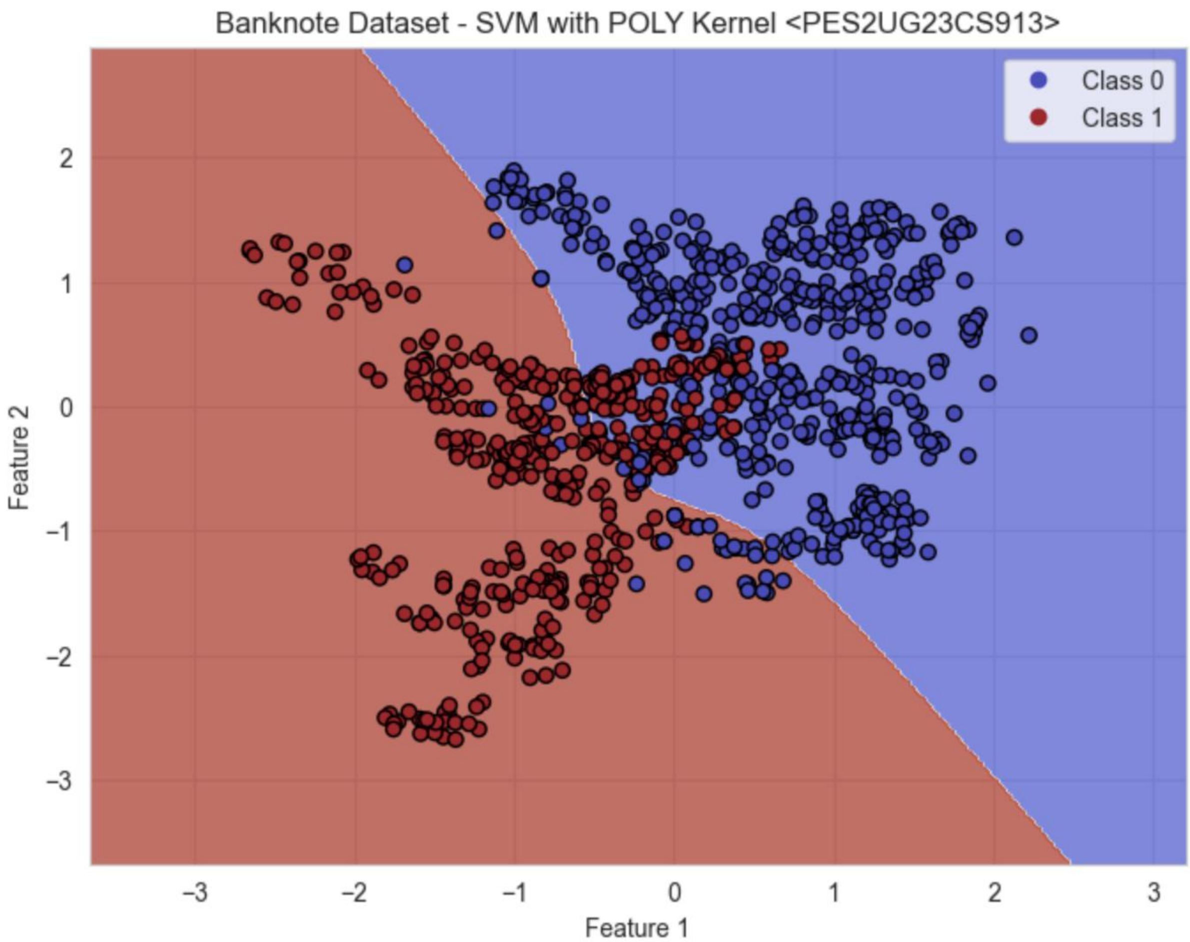
2. RBF

```
SVM with RBF Kernel <PES2UG23CS913>
              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
```

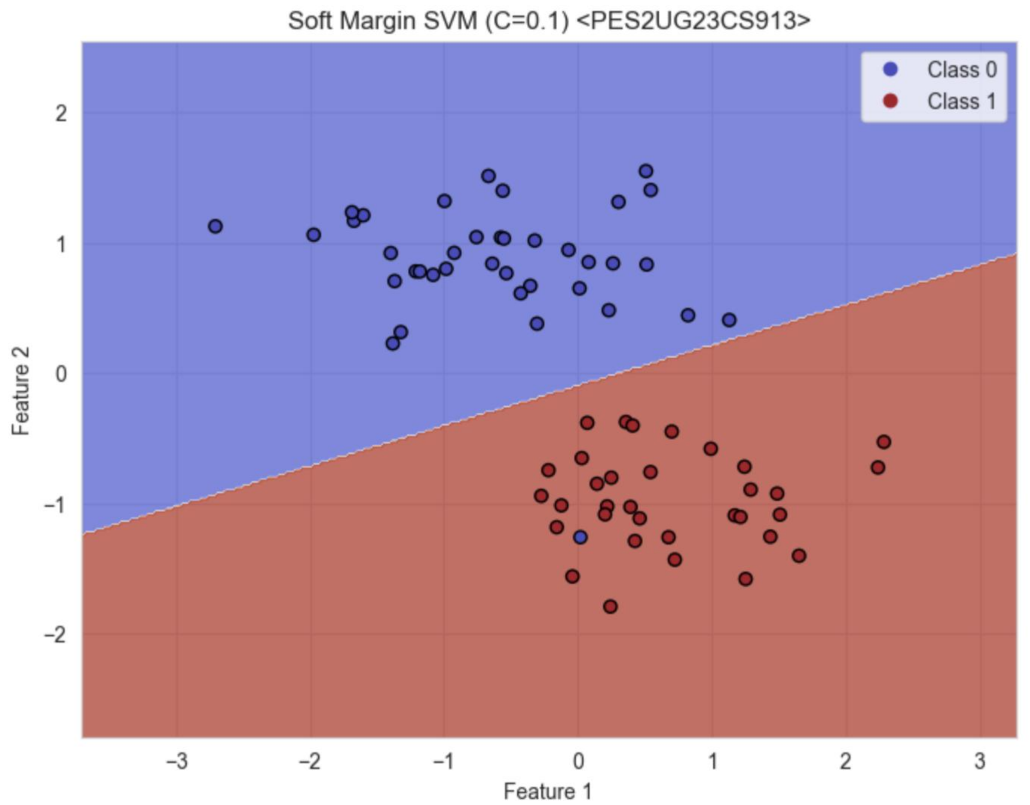


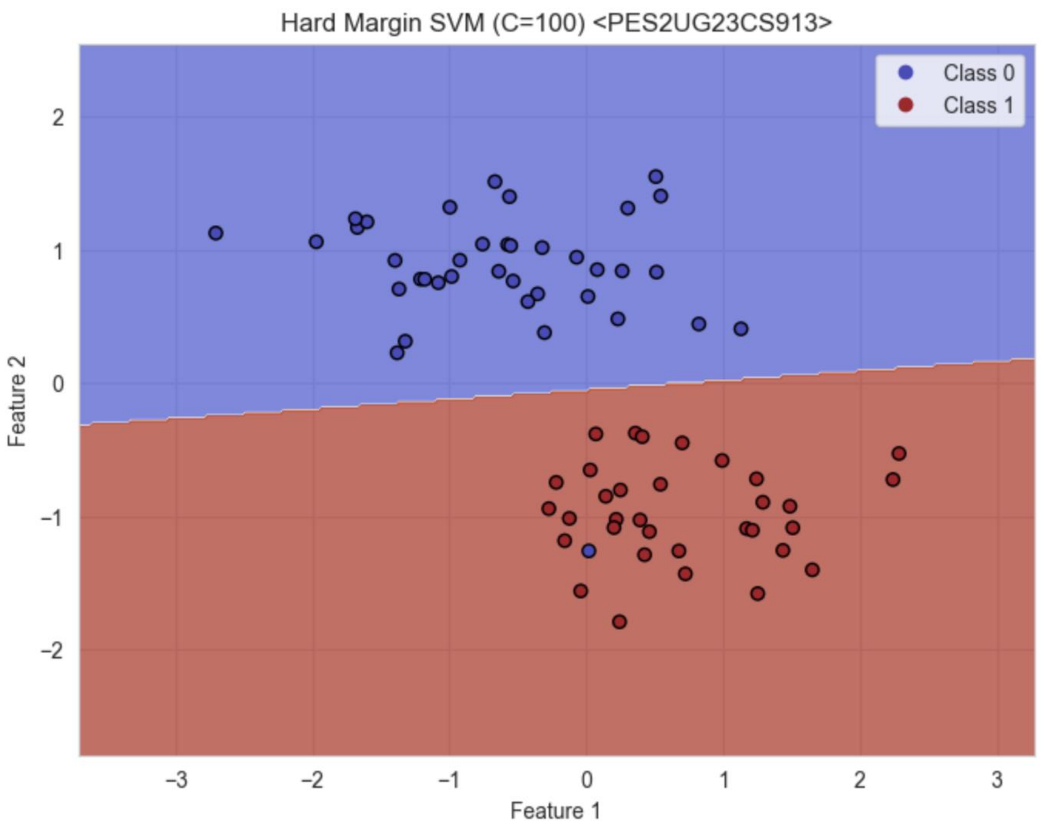
3. POLYNOMIAL

```
SVM with POLY Kernel <PES2UG23CS913>
              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
```



PART 3


Soft Margin SVM (C=0.1) <PES2UG23CS913>
Class 0
Class 1
Feature 2
Feature 1
2
1
0
−1
−2
−3
−2
−1
0
1
2
3


Hard Margin SVM (C=100) <PES2UG23CS913>
Class 0
Class 1
Feature 2
Feature 1
2
1
0
−1
−2
−3
−2
−1
0
1
2
3