Machine Learning
LAB 10

SVM

Medha Raj
PES2UG23CS334
Section : F

PART 1

1. Inferences about the Linear Kernel's Performance

The Linear Kernel SVM achieved an accuracy of approximately 0.87 on the Moons dataset. This performance indicates that while the model was able to capture some of the structure in the data, it could not fully separate the two classes due to the dataset's non-linear nature. The Moons dataset consists of two interleaving semicircular clusters, which cannot be divided by a straight line. Since a linear kernel can only create a linear decision boundary, the classifier misclassifies points located along the curved sections where the two classes overlap. As a result, the decision boundary produced by the linear kernel appears as a roughly straight line cutting across the data, leaving misclassified samples near the edges of each moon shape. This shows that the linear kernel is effective only for linearly separable problems and is not suitable for datasets with complex, curved boundaries like the Moons dataset.

2. Comparison between RBF and Polynomial Kernel Decision Boundaries

When comparing the RBF (Radial Basis Function) and Polynomial kernels for the Moons dataset, both produce non-linear decision boundaries, but their behaviors differ significantly in flexibility and smoothness. The RBF kernel achieved the highest accuracy of 0.97, clearly outperforming the polynomial kernel's 0.89. The RBF kernel generates a smooth, adaptive boundary that tightly follows the curved shape of the moons, effectively capturing the underlying structure of the data. In contrast, the Polynomial kernel tends to produce more rigid and sometimes oscillating boundaries, depending on the polynomial degree. This can lead to slight overfitting or underfitting — either creating unnecessary complexity in certain regions or failing to follow the curvature perfectly. The visual difference is that the RBF decision region looks smooth and continuous, while the Polynomial one may appear jagged or uneven. Overall, the RBF kernel demonstrates superior generalization and flexibility, making it more suitable for non-linear datasets like the Moons dataset.

PART2

1. Which kernel was most effective for this dataset?

The RBF kernel again proved to be the most effective for the Banknote dataset, achieving an accuracy of 0.93, compared to 0.88 for the linear kernel and 0.84 for the polynomial kernel. Although the Banknote dataset is closer to linearly separable, the RBF kernel's flexibility helped capture subtle variations and non-linear patterns in the feature space that the linear kernel could not. This improvement in accuracy suggests that even small nonlinearities in real-world data can be effectively modeled using RBF.

2. Why might the Polynomial kernel have underperformed here?

For the Banknote dataset, the polynomial kernel likely introduced unnecessary complexity in an otherwise mostly linear classification problem. When the decision boundary can already be represented adequately by a linear model, a polynomial transformation may distort feature relationships and create higher-dimensional patterns that don't reflect true class boundaries. This can result in poorer generalization and slightly lower accuracy, as seen with the 0.84 performance.

PART 3

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

The soft margin is wider than the hard margin. This is because the soft margin SVM introduces a relaxation parameter that allows some points to fall within the margin or even be misclassified, in order to achieve better generalization. By allowing this flexibility, the decision boundary is less constrained by outliers, resulting in a wider and smoother margin compared to the hard margin SVM, which forces all data points to be classified correctly and tightly hugs the closest support vectors.

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

The soft margin model allows “mistakes” — i.e., misclassifications — to improve generalization. In real-world datasets, perfect separation is rare due to noise, outliers, or overlapping features. The soft margin SVM introduces a penalty parameter (C) that controls the trade-off between maximizing the margin and minimizing classification errors. By allowing some errors, the model avoids overfitting to noisy data and produces a boundary that works better for unseen samples.

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

The hard margin model is more likely to overfit because it enforces strict separation with no tolerance for errors. In datasets with noise or overlapping classes, a hard margin will create a boundary that perfectly fits training data, including outliers, leading to poor generalization. In contrast, the soft margin model introduces flexibility through the penalty term, preventing the classifier from being overly influenced by individual noisy points.

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

The soft margin model is more trustworthy for new data because it strikes a balance between fitting the training data and maintaining generalization performance. It acknowledges the inherent imperfection in real-world datasets and avoids overfitting by allowing small, controlled violations of the margin. Therefore, the soft margin SVM is expected to perform better when classifying unseen examples, making it the preferred choice for practical applications.

SCREENSHOTS

Moon Dataset

1. Classification Report for SVM with LINEAR Kernel with SRN

```
SVM with LINEAR Kernel PES2UG23CS334
              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

```
SVM with RBF Kernel PES2UG23CS334
              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

```
SVM with POLY Kernel PES2UG23CS334
              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

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

Banknote Dataset

1. Classification Report for SVM with LINEAR Kernel

```
SVM with LINEAR Kernel PES2UG23CS334
              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. Classification Report for SVM with RBF Kernel

```
SVM with RBF Kernel PES2UG23CS334
              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
```

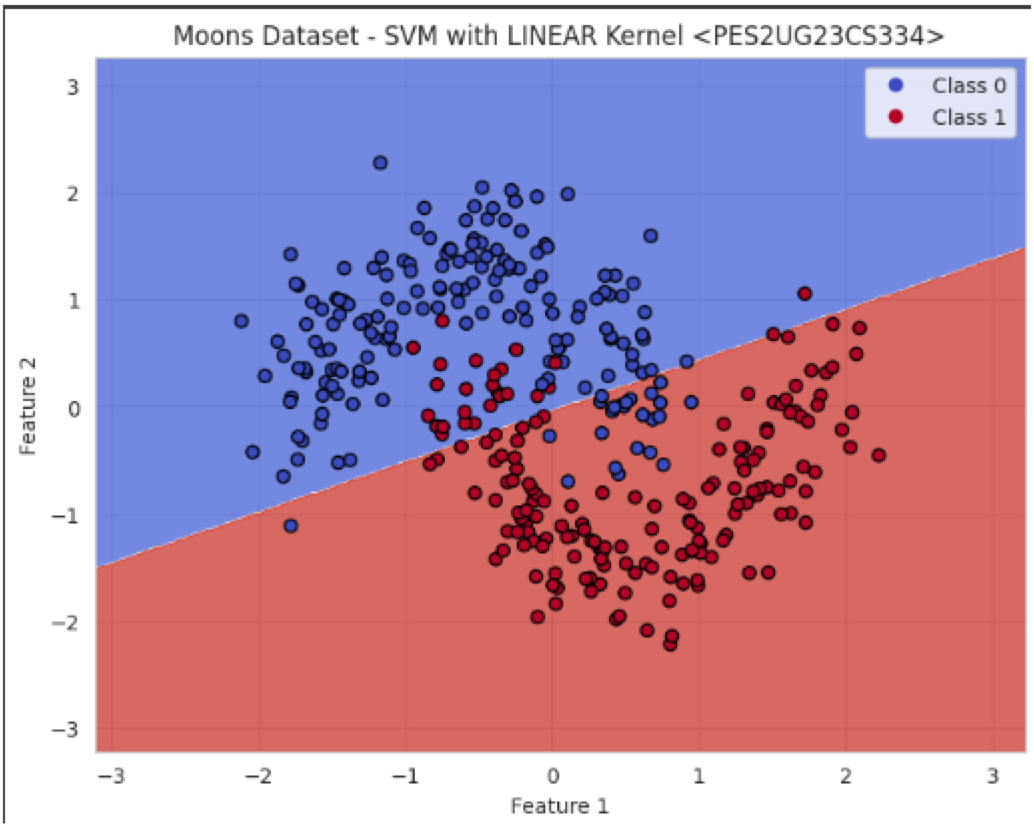
3. Classification Report for SVM with POLY Kernel

```
SVM with POLY Kernel PES2UG23CS334
              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
```

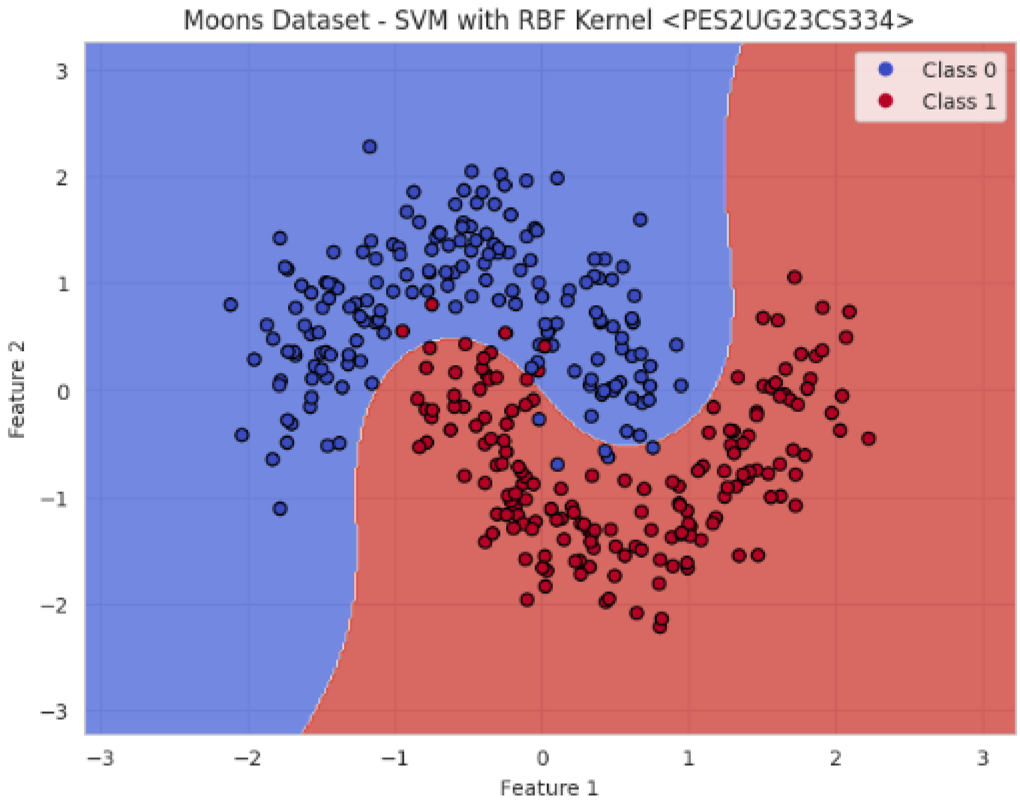
Decision Boundary Visualizations:

Moons Dataset:

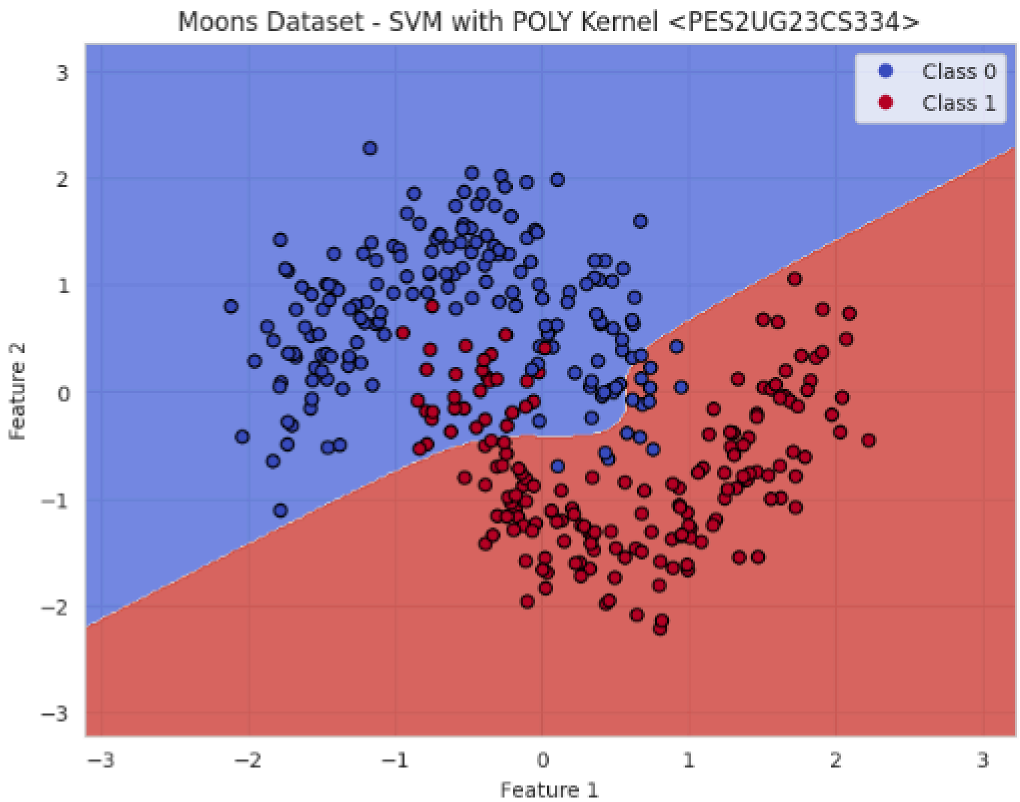
1. Moons Dataset - SVM with LINEAR Kernel



2. Moons Dataset - SVM with RBF Kernel

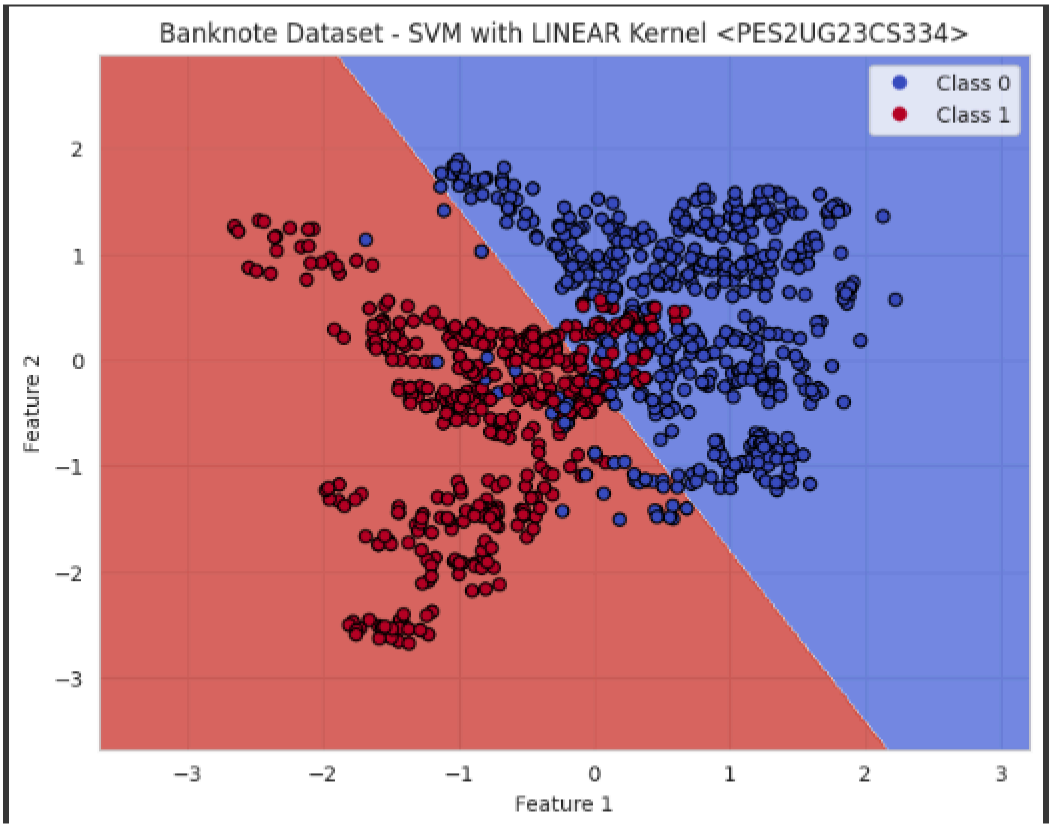


2. Moons Dataset - SVM with RBF Kernel

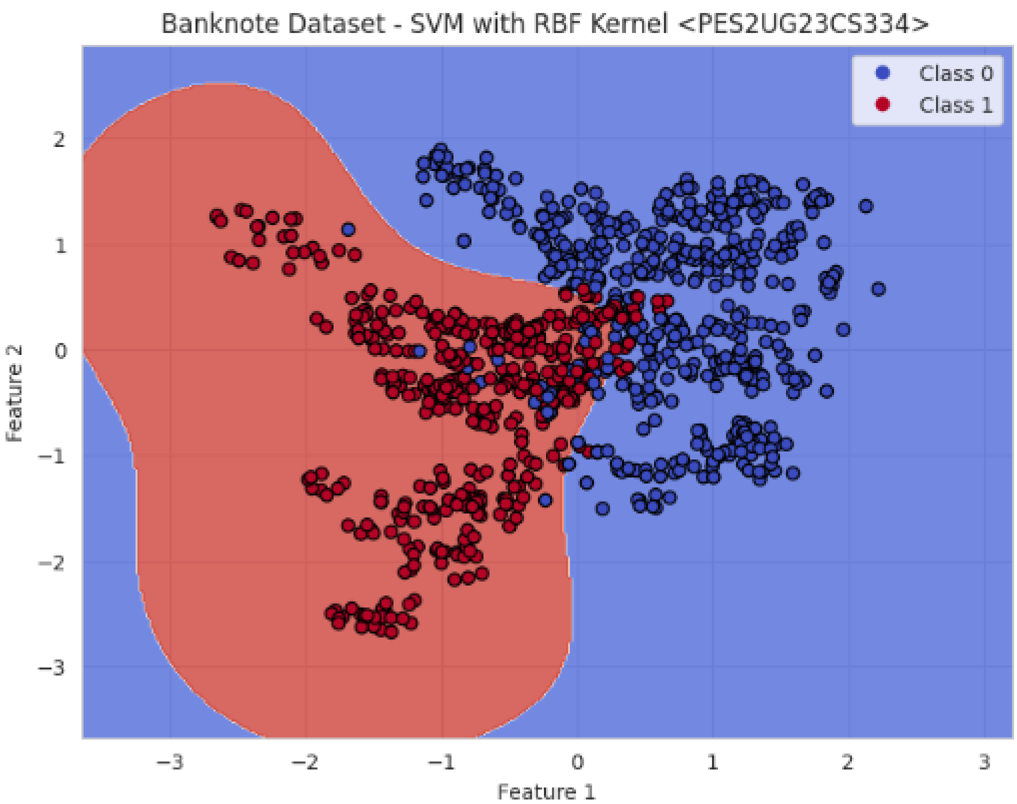


Banknote Dataset:

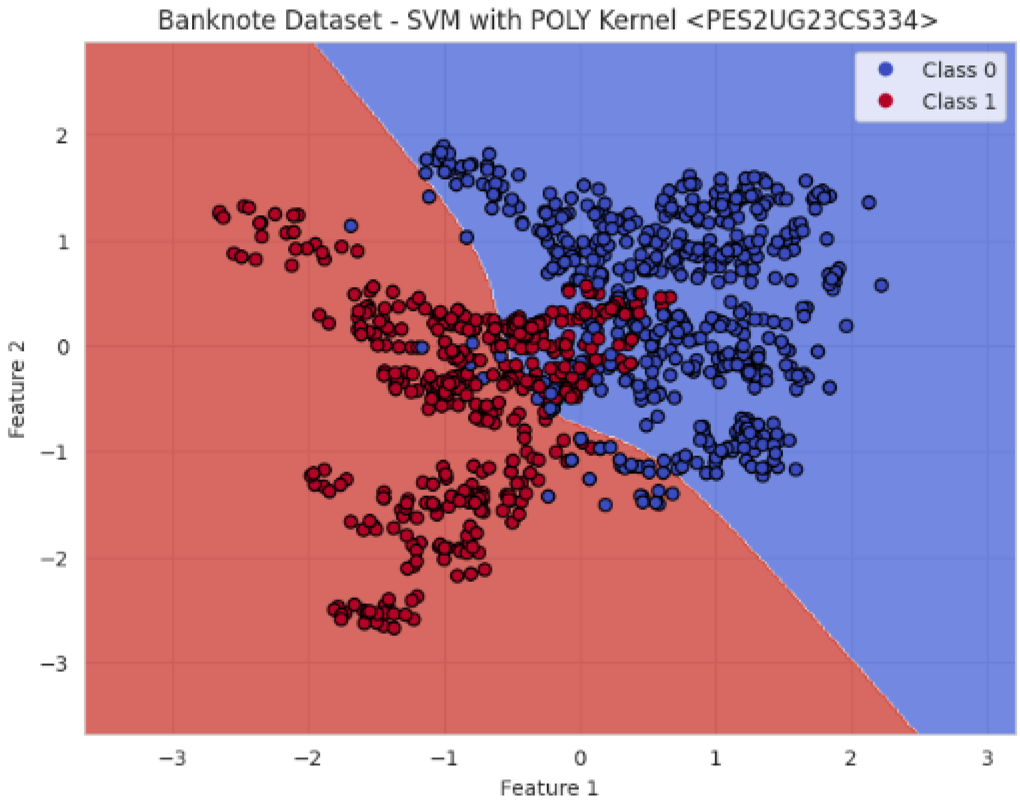
1. Banknote Dataset - SVM with LINEAR Kernel



2. Banknote Dataset - SVM with RBF Kernel

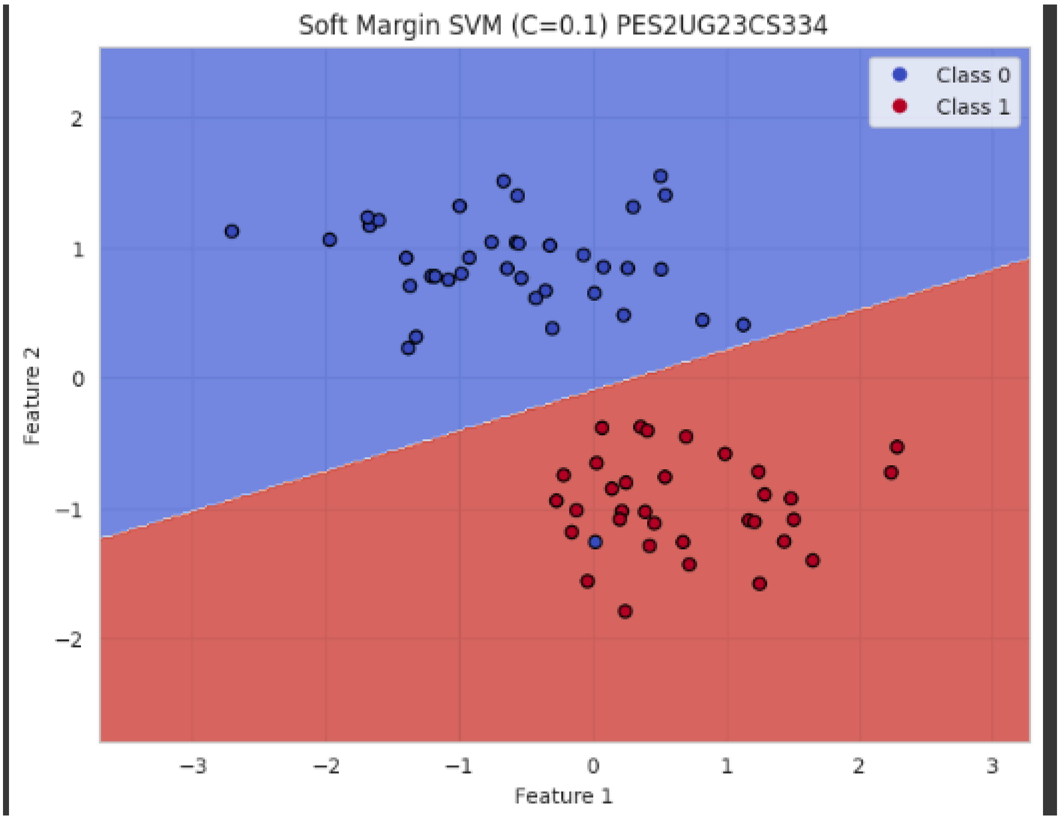


3. Banknote Dataset - SVM with POLY Kernel



Margin Analysis:

1. Soft Margin SVM (C=0.1)



2. Hard Margin SVM (C=100)