Machine Learning Laboratory - Week 10

Experiment: Support Vector Machine (SVM) Classifiers

Name: ROHAN SURESH

SRN: PES1UG23AM240

Section: 5th SEM D SECTION

1. Objective

To understand and implement Support Vector Machine (SVM) classifiers using Linear, RBF, and Polynomial kernels on different datasets, visualize their decision boundaries, and analyze the effect of margin parameters.

2. Theory

• Support Vector Machine (SVM): Finds the optimal hyperplane that separates data points of different classes.

• Kernel Trick: Enables SVMs to handle non-linear data by mapping it into higher-dimensional space.

• Linear, RBF, Polynomial Kernels: Define the shape of the decision boundary (straight, curved, or complex).

• Hard vs. Soft Margin: The C parameter controls the balance between maximizing the margin and minimizing misclassification.

3. Experimental Procedure

1. Loaded Moons and Banknote datasets.

2. Scaled data using StandardScaler.

3. Trained SVM models with Linear, RBF, and Polynomial kernels.

4. Compared model performance using classification reports.

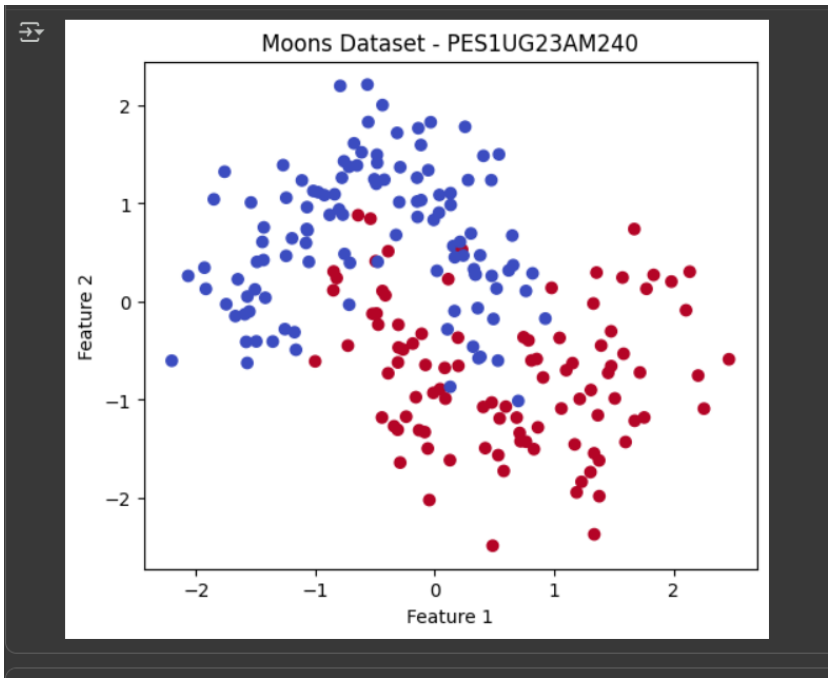
5. Visualized decision boundaries for Moons dataset.

6. Compared Hard vs. Soft margins (C = 0.1 and C = 100).

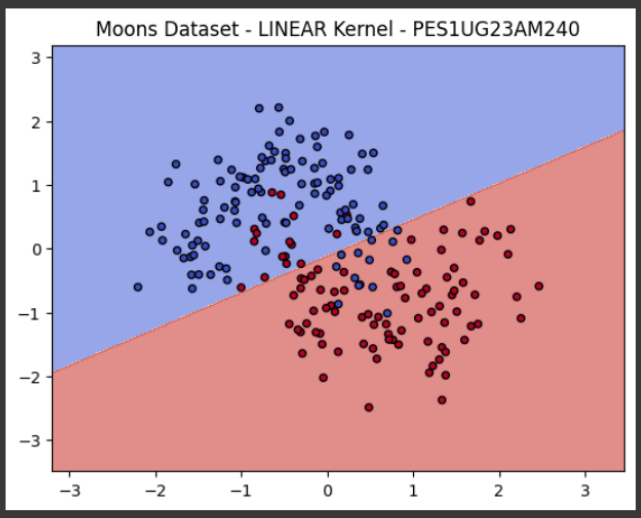
4. Results and Screenshots

Include all 14 screenshots clearly labeled below (insert images in the spaces provided):

Moons Dataset - Linear Kernel

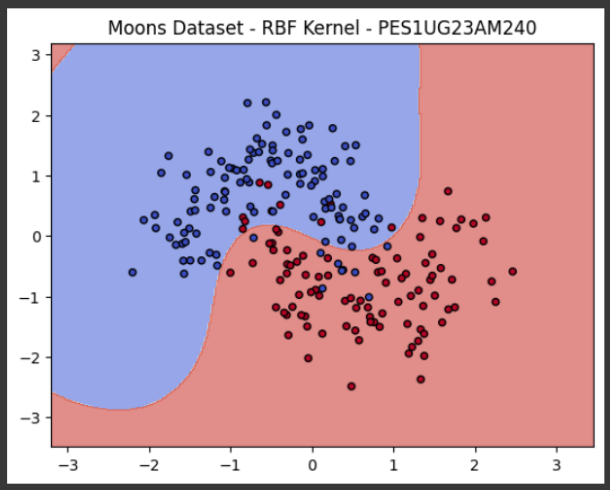



```
Training SVM with LINEAR kernel (PES1UG23AM240)...
Classification Report for SVM with LINEAR Kernel (PES1UG23AM240):
              precision    recall  f1-score   support

           0       0.85      0.87      0.86        38
           1       0.90      0.88      0.89        52

    accuracy                           0.88        90
   macro avg       0.87      0.88      0.88        90
weighted avg       0.88      0.88      0.88        90
```



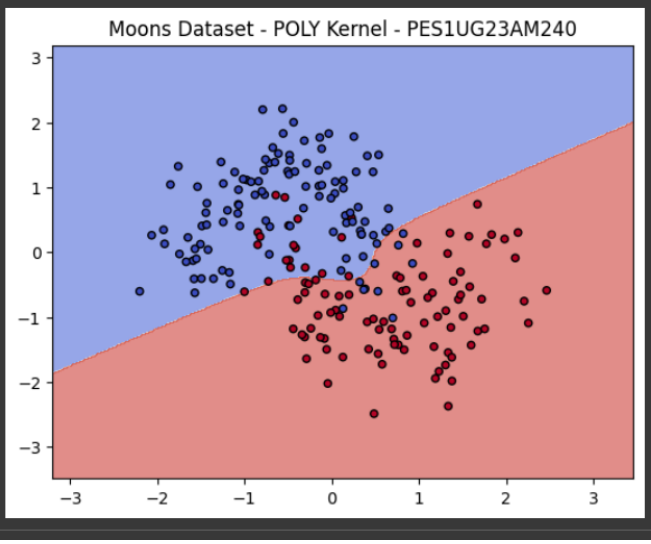
Moons Dataset - RBF Kernel

```
Training SVM with RBF kernel (PES1UG23AM240)...
Classification Report for SVM with RBF Kernel (PES1UG23AM240):
              precision    recall  f1-score   support

           0       0.89      0.87      0.88        38
           1       0.91      0.92      0.91        52

    accuracy                           0.90        90
   macro avg       0.90      0.90      0.90        90
weighted avg       0.90      0.90      0.90        90
```



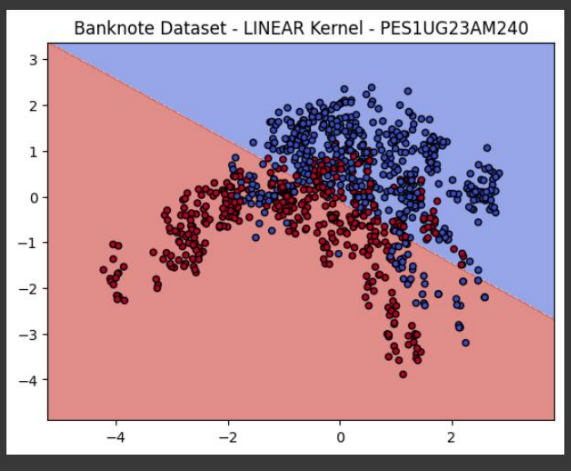
Moons Dataset - Polynomial Kernel

```
Training SVM with POLY kernel (PES1UG23AM240)...
Classification Report for SVM with POLY Kernel (PES1UG23AM240):
              precision    recall  f1-score   support

           0       0.87      0.87      0.87        38
           1       0.90      0.90      0.90        52

    accuracy                           0.89        90
   macro avg       0.89      0.89      0.89        90
weighted avg       0.89      0.89      0.89        90
```



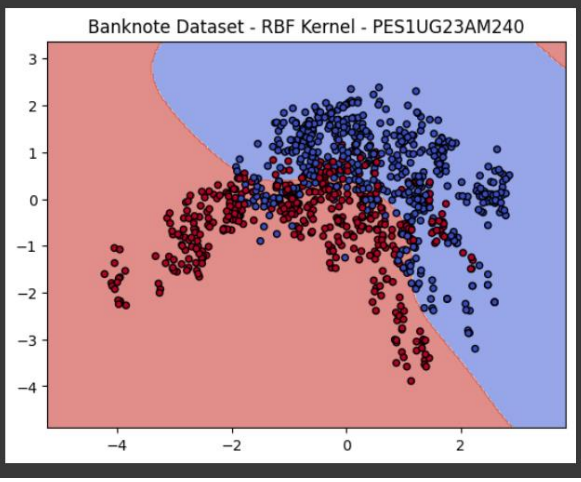
Banknote Dataset - Linear Kernel

```
Training SVM with LINEAR kernel (PES1UG23AM240)...
Classification Report for SVM with LINEAR Kernel (PES1UG23AM240):
              precision    recall  f1-score   support

           0       1.00      0.98      0.99       229
           1       0.97      0.99      0.98       183

    accuracy                           0.99       412
   macro avg       0.98      0.99      0.99       412
weighted avg       0.99      0.99      0.99       412
```



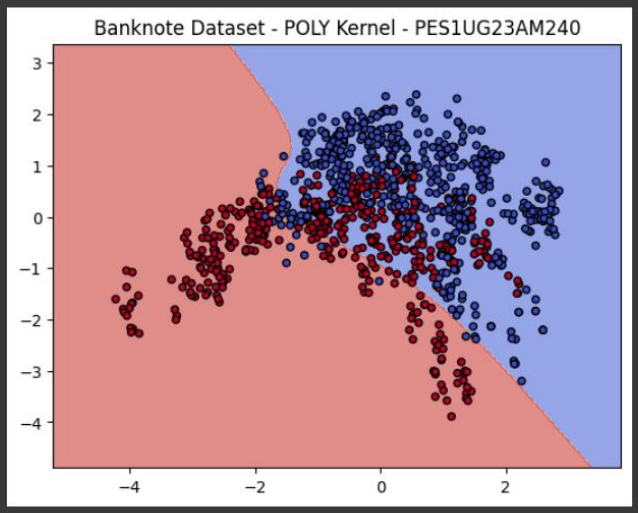
Banknote Dataset - RBF Kernel

```
Training SVM with RBF kernel (PES1UG23AM240)...
Classification Report for SVM with RBF Kernel (PES1UG23AM240):
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       229
           1       1.00      1.00      1.00       183

    accuracy                           1.00       412
   macro avg       1.00      1.00      1.00       412
weighted avg       1.00      1.00      1.00       412
```

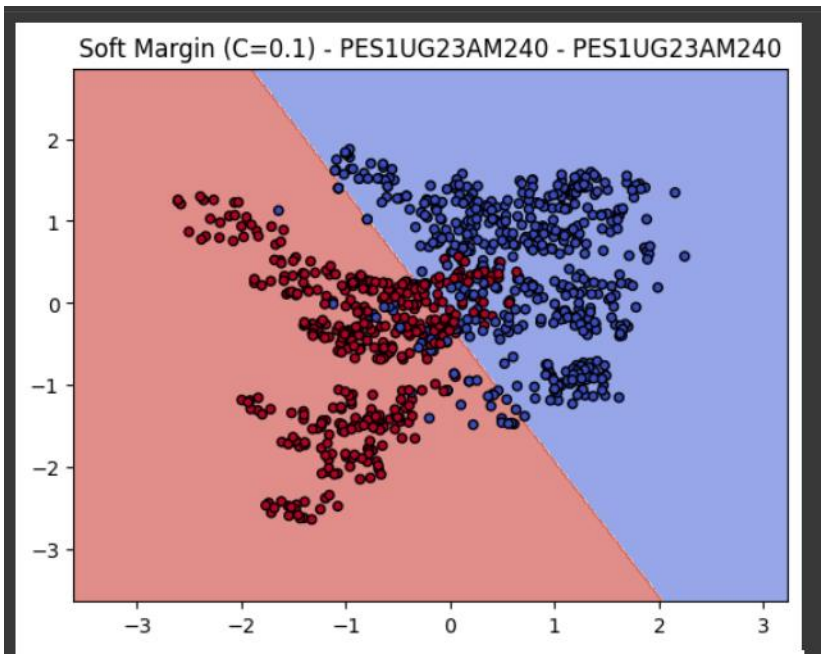


Banknote Dataset - Polynomial Kernel

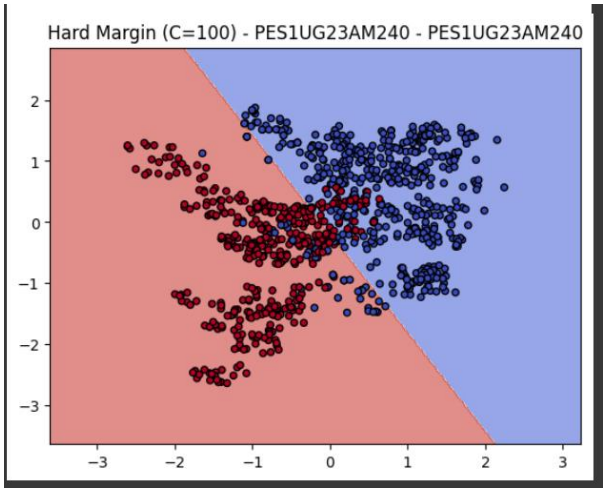
```
Training SVM with POLY kernel (PES1UG23AM240)...
Classification Report for SVM with POLY Kernel (PES1UG23AM240):
              precision    recall  f1-score   support

           0       1.00      0.98      0.99       229
           1       0.98      1.00      0.99       183

    accuracy                           0.99       412
   macro avg       0.99      0.99      0.99       412
weighted avg       0.99      0.99      0.99       412
```



Soft Margin (C=0.1) - Decision Boundary



Hard Margin (C=100) - Decision Boundary



5. Analysis Questions

Moons Dataset Questions

1. Inferences about the Linear Kernel's performance.

The Linear Kernel performs poorly on the moons dataset because the data is not linearly separable. It fails to capture the curved boundaries of the two moon-shaped clusters, leading to many misclassifications.

2. Comparison between RBF and Polynomial kernel decision boundaries.

The RBF kernel produces a smooth, non-linear boundary that closely follows the shape of the data, resulting in better classification. The Polynomial kernel can also handle non-linear patterns but may produce more complex, less smooth boundaries, potentially overfitting or underfitting depending on the degree chosen. Overall, RBF usually fits this dataset more effectively.

Banknote Dataset Questions

1. Which kernel was most effective for this dataset?

The RBF kernel is generally the most effective for the banknote dataset because it can

handle complex, non-linear relationships between features while maintaining good generalization.

2. Why might the Polynomial kernel have underperformed here?

The Polynomial kernel may underperform due to overfitting or inability to capture the subtle variations in feature relationships. If the degree of the polynomial is too high, it may fit the training data too closely, while a low degree may be too simple to capture the patterns.

Hard vs. Soft Margin Questions

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

The Soft Margin SVM (C = 0.1) produces a wider margin. A smaller C allows the model to tolerate misclassifications, resulting in a smoother and broader decision boundary. The Hard Margin SVM (C = 100) forces perfect classification and creates a narrower margin.

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

The Soft Margin SVM allows some points to violate the margin or be misclassified because its primary goal is maximizing the margin rather than perfectly fitting every training sample. This improves generalization and reduces overfitting, especially on noisy data. The parameter C controls this trade-off between margin width and training accuracy.

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

The Hard Margin SVM (C = 100) is more likely to overfit because it forces the decision boundary to perfectly separate the training data. This makes the model sensitive to noise and small fluctuations in the data, resulting in poor generalization to unseen data. The Soft Margin model is more flexible and robust.

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

The Soft Margin SVM (C = 0.1) is more trustworthy for classifying new, unseen data. Real world data is often noisy, so a model that allows a few training errors with a wider

margin is more robust and generalizes better. Starting with a lower C value is generally preferred in practice.

6. Inferences

• Linear kernel works best for linearly separable data (e.g., Banknote dataset).

• RBF performs best for non-linear datasets (e.g., Moons dataset).

• Polynomial kernels can be flexible but may overfit.

• Soft margin (lower C) allows generalization, while hard margin (high C) can overfit.

7. Conclusion

SVMs are powerful classification models capable of handling both linear and non-linear data. Kernel selection and the margin parameter C play crucial roles in determining model accuracy and generalization performance.

8. References

1. Scikit-learn Documentation: https://scikit-learn.org/stable/modules/svm.html

2. Course Lab Manual and Lecture Notes