ELL409 - Support Vector Machines and Ensembling

Priyal Jain 2021MT60949

November 2024

Contents

1 Introduction

This report presents an analysis of Support Vector Machine (SVM) and Ensemble models applied to a subset of the MNIST dataset. The evaluation includes various configurations for SVMs (hard margin and soft margin with linear and RBF kernels) and Ensemble methods (Random Forest and AdaBoost). The objective is to compare their performance based on training and validation metrics, visualize support vectors, and examine misclassifications.

2 Data Preprocessing

2.1 Data Filtering

The MNIST dataset was filtered to include digits based on the last digit of the entry number (4), resulting in the selection of digits:

- Entry number digit: 9
- Target digits: [8, 9, 0, 1]

2.2 Scaling and Dimensionality Reduction

- Scaling: Standard scaling was applied to normalize the feature values.
- PCA: Principal Component Analysis was employed where specified to reduce data dimensionality.

3 Support Vector Machine (SVM) Models Analysis

3.1 Hard Margin Linear SVM

3.1.1 Optimal Hyperparameters for Best Validation F1-Score

- C: 1×10^9 (simulating infinity)
- Kernel: Linear
- Gamma: Not applicable
- Use PCA: None

3.1.2 Training and Validation Results

- Number of Support Vectors: 110
- Training Accuracy: 1.0000
- Training F1-Score: 1.0000
- Misclassified Instances (Training): 0
- Validation Accuracy: 0.9942
- Best Validation F1-Score: **0.9923**
- Misclassified Instances (Validation): 28

3.1.3 Visualizations

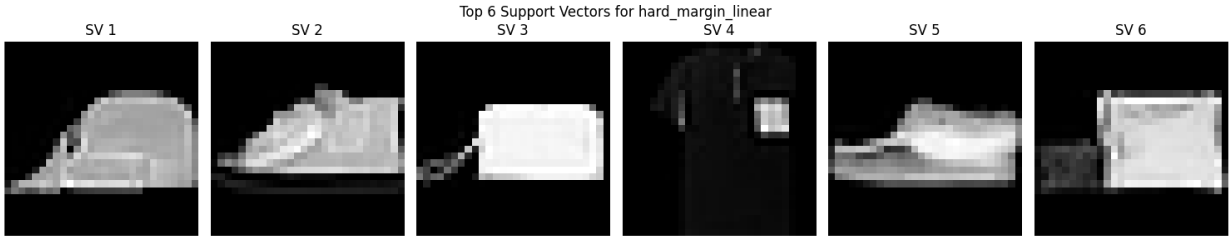

Figure 1: Top 6 Support Vectors for Hard Margin Linear SVM

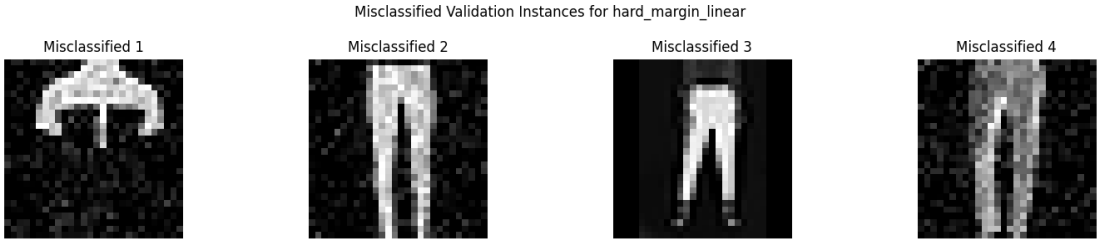

Figure 2: Misclassified Validation Instances for Hard Margin Linear SVM

3.2 Hard Margin RBF SVM

3.2.1 Optimal Hyperparameters for Best Validation F1-Score

- C: 1×10^9 (simulating a hard margin)
- Kernel: RBF
- Gamma: 0.01 (selected from 0.01, 0.1, 1, and 10)
- Use PCA: 10 components

3.2.2 Training and Validation Results

- Number of Support Vectors: 686
- Training Accuracy: 1.0000
- Training F1-Score: 1.0000
- Misclassified Instances (Training): 0
- Validation Accuracy: 0.9888
- Best Validation F1-Score: **0.9848**
- Misclassified Instances (Validation): 54

3.2.3 Visualizations

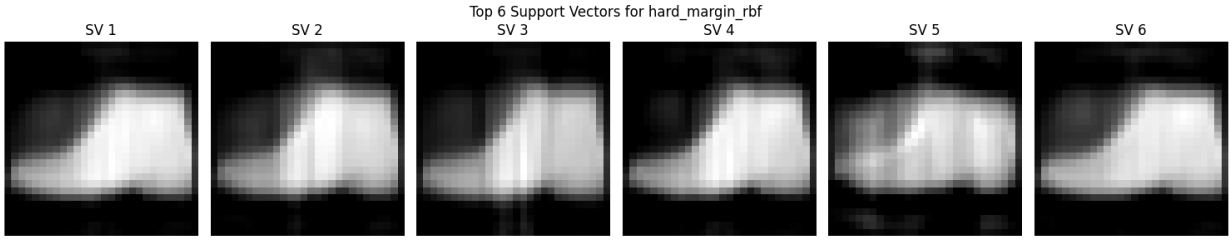

Figure 3: Top 6 Support Vectors for Hard Margin RBF SVM

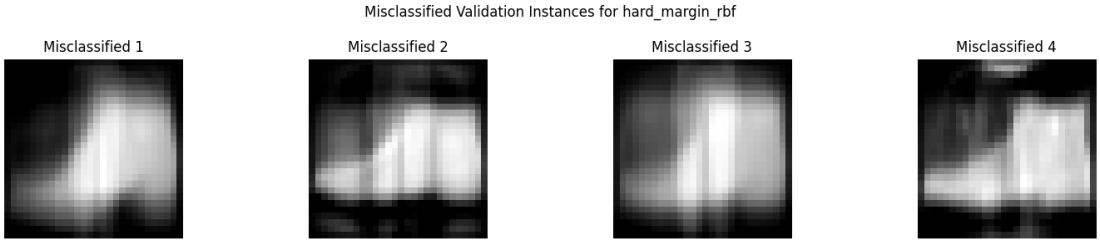

Figure 4: Misclassified Validation Instances for Hard Margin RBF SVM

3.3 Soft Margin Linear SVM

3.3.1 Optimal Hyperparameters for Best Validation F1-Score

- C: 0.1 (selected from 0.1, 1, and 10)
- Kernel: Linear
- Gamma: Not applicable
- Use PCA: 10 components

3.3.2 Training and Validation Results

- Number of Support Vectors: 55
- Training Accuracy: 0.9985
- Training F1-Score: 0.9980
- Misclassified Instances (Training): 15
- Validation Accuracy: 0.9904
- Best Validation F1-Score: **0.9871**
- Misclassified Instances (Validation): 46

3.3.3 Visualizations

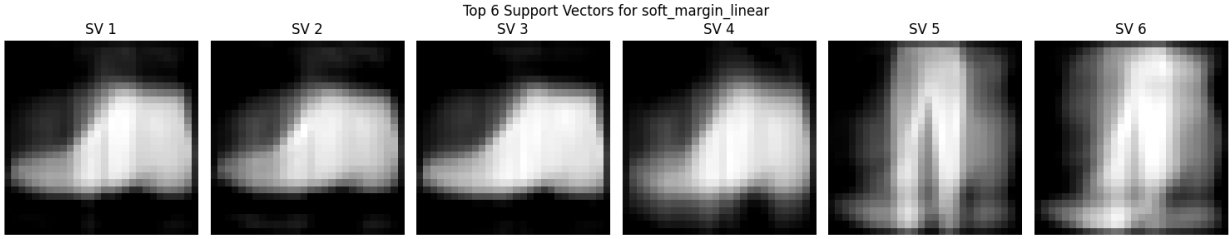

Figure 5: Top 6 Support Vectors for Soft Margin Linear SVM

Figure 6: Misclassified Training Instances for Soft Margin Linear SVM

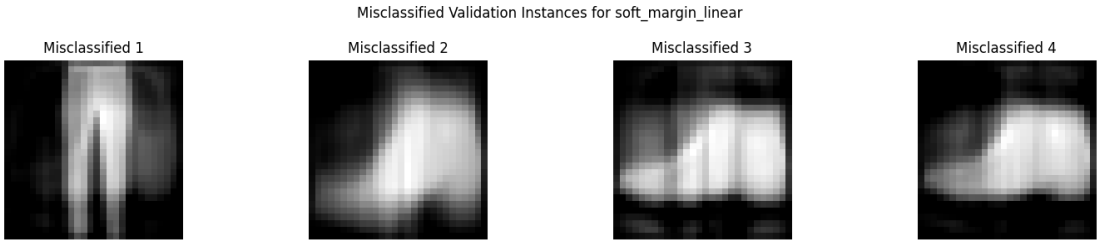

Figure 7: Misclassified Validation Instances for Soft Margin Linear SVM

3.4 Soft Margin RBF SVM

3.4.1 Optimal Hyperparameters for Best Validation F1-Score

- C: 0.1 (selected from 0.1, 1, and 10)
- Kernel: RBF
- Gamma: 0.01 (selected from 0.01, 0.1, 1, and 10)
- Use PCA: 10 components

3.4.2 Training and Validation Results

- Number of Support Vectors: 1146
- Training Accuracy: 0.9981
- Training F1-Score: 0.9975
- Misclassified Instances (Training): 19

- Validation Accuracy: 0.9792
- Best Validation F1-Score: **0.9714**
- Misclassified Instances (Validation): 100

3.4.3 Visualizations

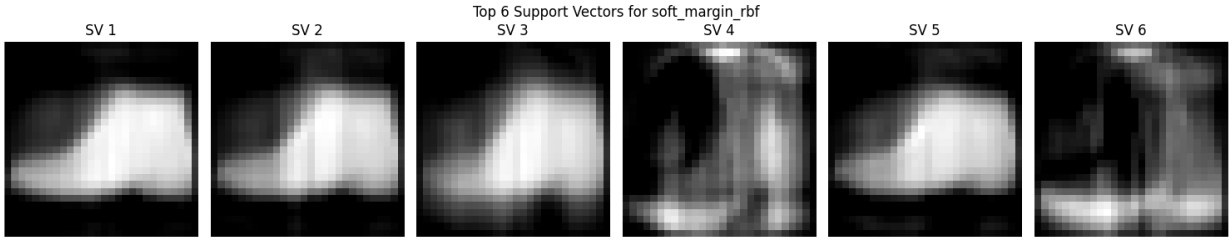

Figure 8: Top 6 Support Vectors for Soft Margin RBF SVM

Figure 9: Misclassified Training Instances for Soft Margin RBF SVM

Figure 10: Misclassified Validation Instances for Soft Margin RBF SVM

4 Ensemble Models Analysis

4.1 Random Forest Classifier

4.1.1 Optimal Hyperparameters for Best Validation F1-Score

- Number of Trees: 20 (selected from 5, 10, and 20)
- Max Depth: 10 (selected from 5, 10, and 20)
- Min Samples Split: 5 (selected from 2, 3, and 5)
- Max Features: None (selected from 'sqrt', 'log2', and None)
- Use PCA: 50 components

4.1.2 Training and Validation Results

- Training Accuracy: 0.9998
- Training F1-Score: 0.9997
- Misclassified Instances (Training): 2
- Validation Accuracy: 0.98
- Best Validation F1-Score: **0.9727**
- Misclassified Instances (Validation): 96

4.1.3 Visualizations

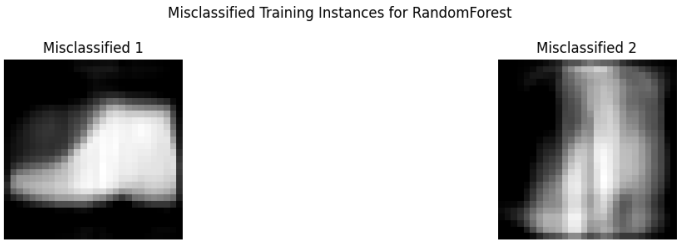

Figure 11: Misclassified Training Instances for Random Forest

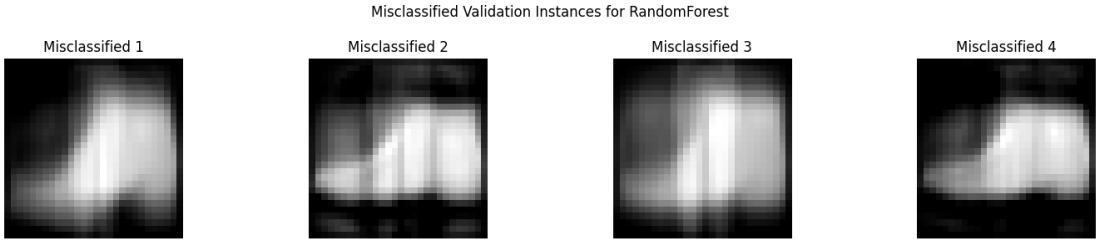

Figure 12: Misclassified Validation Instances for Random Forest

4.2 AdaBoost Classifier

4.2.1 Optimal Hyperparameters for Best Validation F1-Score

- Number of Trees: 50
- Use PCA: 50 components

4.2.2 Training and Validation Results

- Training Accuracy: 0.9992
- Training F1-Score: 0.9989
- Misclassified Instances (Training): 8
- Validation Accuracy: 0.9892
- Best Validation F1-Score: **0.9854**
- Misclassified Instances (Validation): 52

4.2.3 Visualizations

Figure 13: Misclassified Training Instances for AdaBoost

Figure 14: Misclassified Validation Instances for AdaBoost

5 Discussion

- **Performance Comparison**:
  - SVM Models:
    - Hard Margin Linear SVM achieved the highest validation F1-Score of **0.9923**.
    - Soft Margin Linear SVM followed with an F1-Score of **0.9871**.
    - Hard Margin RBF SVM and Soft Margin RBF SVM scored **0.9848** and **0.9714**, respectively.
    - Linear kernel with a large margin performs best, likely due to near-linear separability of the data.
  - Ensemble Methods:
    - AdaBoost achieved a validation F1-Score of **0.9854**, outperforming Random Forest's **0.9727**.
    - Both ensembles performed slightly behind the top SVM models.
    - Random Forest showed potential overfitting with high training accuracy (**0.9998**) but lower validation performance.
- **Model Complexity and Support Vectors**:
  - Support Vectors in SVM:
    - Hard Margin RBF SVM used **686** support vectors, indicating a complex decision boundary.
    - Hard Margin Linear SVM used **110** support vectors, reflecting a simpler, linear separation.
    - More support vectors increase computational costs and may signify complex data relationships.
  - Ensemble Models:
    - Random Forest combines multiple trees to handle non-linear patterns but may overfit if not regularized.

    * AdaBoost focuses on correcting previous errors, leading to robust performance but can amplify noise with inadequate preprocessing.

- **Misclassification Analysis**:
  - Soft Margin SVMs showed better generalization with fewer misclassifications compared to hard margin counterparts.
  - Among ensembles, AdaBoost had fewer misclassifications than Random Forest, consistent with their F1-Scores.
  - Misclassified instances often indicate overlapping classes or ambiguous features, suggesting potential for further feature engineering or data augmentation.
- **Support Vectors Analysis**:
  - The number and distribution of support vectors provide insights into the learned decision boundaries.
  - Hard Margin SVMs require more support vectors, especially with non-linear kernels like RBF.
  - Analyzing support vectors helps understand model complexity and potential overfitting; excessive support vectors may indicate a highly tailored boundary to the training data.

6 Conclusion

- **Key Findings**:
  - Hard Margin Linear SVM emerged as the top performer with the highest validation F1-Score of **0.9923**, effective in scenarios with near-linear separability.
  - AdaBoost showed strong performance among ensemble methods, achieving a validation F1-Score of **0.9854**, balancing complexity and accuracy.
  - Random Forest exhibited signs of overfitting, indicated by high training accuracy (**0.9998**) but lower validation performance (**0.9727** F1-Score).
- **Model Complexity vs. Performance**:
  - SVMs with non-linear kernels (RBF) require more support vectors, increasing computational complexity without significant performance gains for this dataset subset.
  - Ensemble Methods manage complexity through aggregation and boosting but must be carefully tuned to avoid overfitting.

7 Updated `config.py`

Based on the analysis, the following configurations should be updated in `config.py`:

```
# config.py
ENTRY_NUMBER_LAST_DIGIT = 9 # change with yours
ENTRY_NUMBER = '2021MT60949'

PRE_PROCESSING_CONFIG ={
    "hard_margin_linear" : {
        "use_pca" : None,
    },

    "hard_margin_rbf" : {
        "use_pca" : 10,
```

```
    },

    "soft_margin_linear" : {
        "use_pca" : 10,
    },

    "soft_margin_rbf" : {
        "use_pca" : 10,
    },

    "AdaBoost" : {
        "use_pca" : 50,
    },

    "RandomForest" : {
        "use_pca" : 50,
    }
}

SVM_CONFIG = {
    "hard_margin_linear": {
        "C": 1e9,  # A large value to simulate hard margin
        "kernel": 'linear',
        "val_score": 0.9923,  # Replace with your expected validation score
    },
    "soft_margin_linear": {
        "C": 0.1,
        "kernel": 'linear',
        "val_score": 0.9871,  # Replace with your expected validation score
    },
    "hard_margin_rbf": {
        "C": 1e9,  # A large value to simulate hard margin
        "kernel": 'rbf',
        "gamma": 0.01,  # Adjust gamma based on your data
        "val_score": 0.9848,  # Replace with your expected validation score
    },
    "soft_margin_rbf": {
        "C": 0.1,
        "kernel": 'rbf',
        "gamma": 0.01,  # Adjust gamma based on your data
        "val_score": 0.9714,  # Replace with your expected validation score
    }
}
ENSEMBLING_CONFIG = {
    'AdaBoost':{
        'num_trees' : 50,
        "val_score" : 0.9854,
    },

    'RandomForest': {
        'num_trees': 20,
        'max_depth': 10,
        'min_samples_split': 5,
        'max_features': None,
```

```
        "val_score": 0.9727,  # Replace with your validation score after tuning
    }

}
```

8 References

- Reading 1 on SVM
- Reading 2 on SVM
- Reading 3 on Ensembling
- Reading 4 on Ensembling
- Reading 5 on Ensembling
- Reading 6 on PCA)