Analysis of Machine Learning Models for Digit Classification

Yunzhen Wu

Date: Feb. 16, 2025

Management/Research Question

How can we accurately classify handwritten digits using machine learning models, and what methods provide the best performance?

Problem Statement

The objective of this project is to classify handwritten digits using a combination of supervised and unsupervised machine learning models. We explore different modeling techniques, including:

- A Random Forest classifier trained on the full dataset.
- A Random Forest classifier trained on principal components derived from PCA.
- A K-Means clustering model for unsupervised classification.
- An improved version of the PCA-based Random Forest model that corrects an experimental flaw.

Data Description

The dataset consists of labeled handwritten digit images. The training set contains labeled observations, while the test set has unlabeled images for model evaluation. Each image is represented as a flattened array of pixel values.

Data Preprocessing

- Normalized pixel values to [0,1] for better numerical stability.
- Split training data into train/validation sets for hyperparameter tuning.

Model Training

We implement the following models:

- Model 1: A Random Forest classifier trained on the original pixel values.
    - Used all pixel values as features.
    - Applied 5-fold cross-validation.
    - Recorded training time and performance metrics.
- Model 2: A PCA-based Random Forest classifier trained on principal components retaining 95% variance.
    - Performed PCA to reduce dimensionality while retaining 95% variance.
    - Used transformed features to train a Random Forest classifier.
    - Applied hyperparameter tuning via GridSearchCV.
    - Key insight: PCA reduced dimensionality but slightly impacted accuracy.
- Model 3: A K-Means clustering approach to categorize digit images.
    - Applied MiniBatchKMeans clustering to assign digit categories.
    - Evaluated performance using the silhouette score.
    - Adjusted labels using inferred cluster mappings.
- Model 4: A corrected PCA-based Random Forest classifier trained only on PCA-transformed training data (not including test data).
    - Addressed a major flaw in Model 2: PCA was previously applied to combined train and test sets.
    - Recomputed PCA using only training data.

- Applied transformations to test data post-training.
- Results showed improved generalization.

Hyperparameter tuning and cross-validation are applied to optimize model performance. Model 1 underwent only cross-validation without tuning, as it used the full feature set without PCA, making hyperparameter tuning computationally expensive due to the high dimensionality of X. In contrast, Models 2, 3, and 4 incorporated both hyperparameter tuning and cross-validation.

Model Evaluation

- Performance Metrics
  - Accuracy, precision, recall, and F1-score were compared.
  - K-Means clustering results were analyzed using silhouette scores.
  - Hyperparameter tuning results for Models 2 and 4 were tabulated.
  - Training time comparisons were documented.
- Graphs and Tables
  - PCA variance plot demonstrated dimensionality reduction effectiveness.
  - Confusion matrices were generated for each supervised model.

Insights, Conclusions, and Comparisons

- Random Forest (Model 1) performed well without PCA. (Refer to Figure 1: Model 1 Accuracy and Confusion Matrix, showing an accuracy of 96.39%)

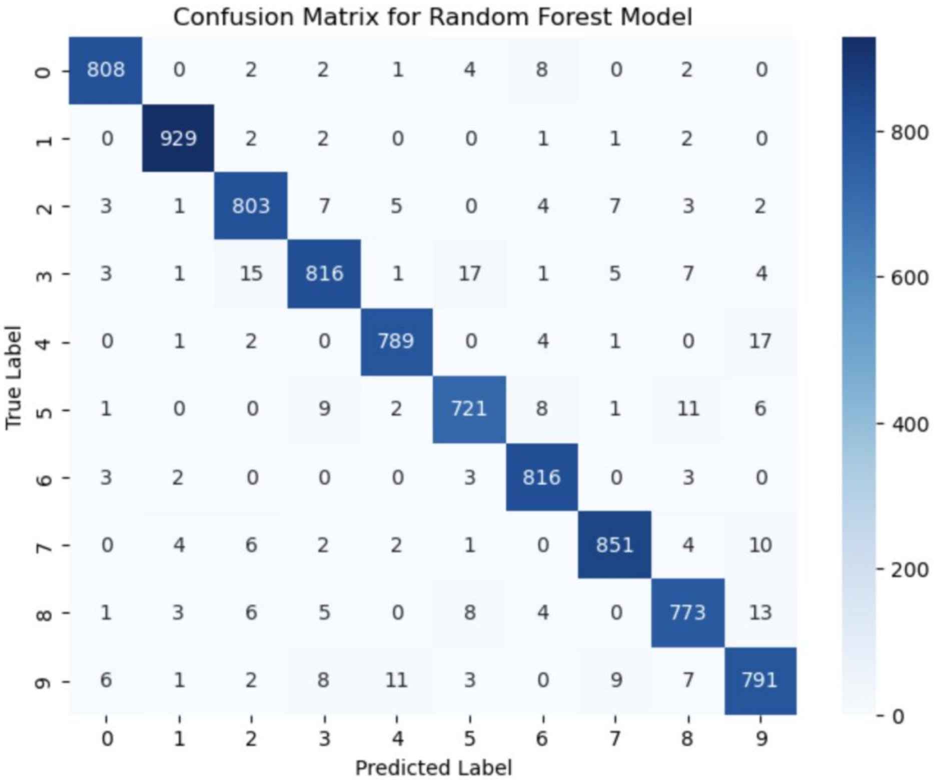


Figure 1: Model 1 Accuracy and Confusion Matrix

- After PCA, principal components are significantly reduced, from 784 to 154. (Refer to Figure 3: PCA Variance Plot)

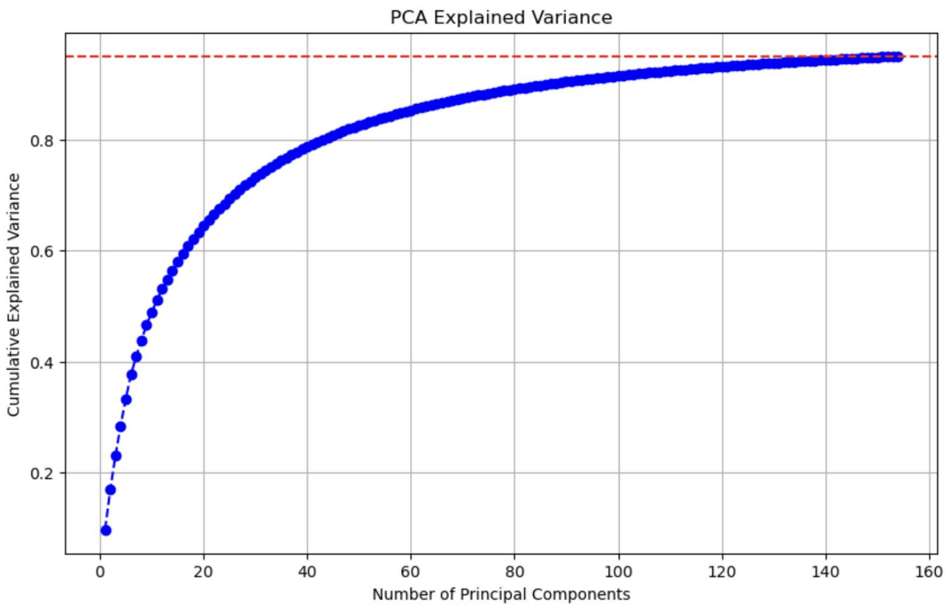


Figure 2: PCA Variance Plot

- PCA (Model 2) reduced dimensionality but slightly affected accuracy. (Refer to Figure 3: Model 2 Accuracy and Confusion Matrix, where accuracy dropped to 94.74%)

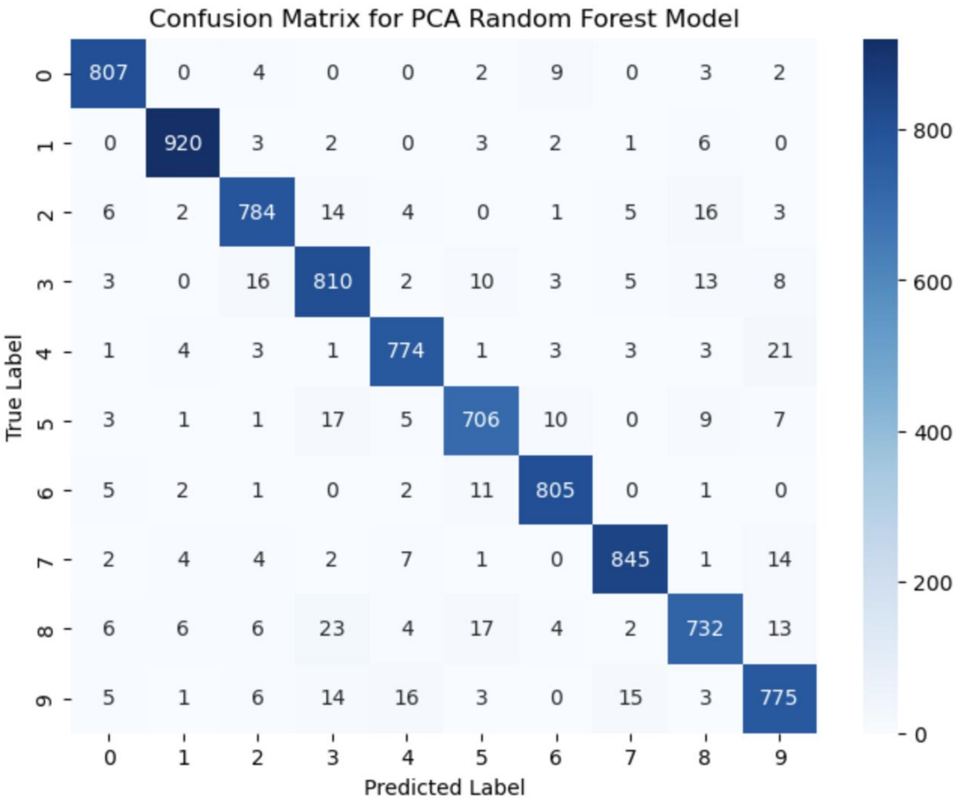


Figure 3: Model 2 Accuracy and Confusion Matrix

- K-Means (Model 3)'s Silhouette Score is very low (0.0666), which means the 10 clusters are highly overlapped, and n_clusters=10 is not effective enough. Thus, the performance of Classification Accuracy is only 55.02%. (Refer to Figure 4: Model 3 Accuracy and Confusion Matrix)

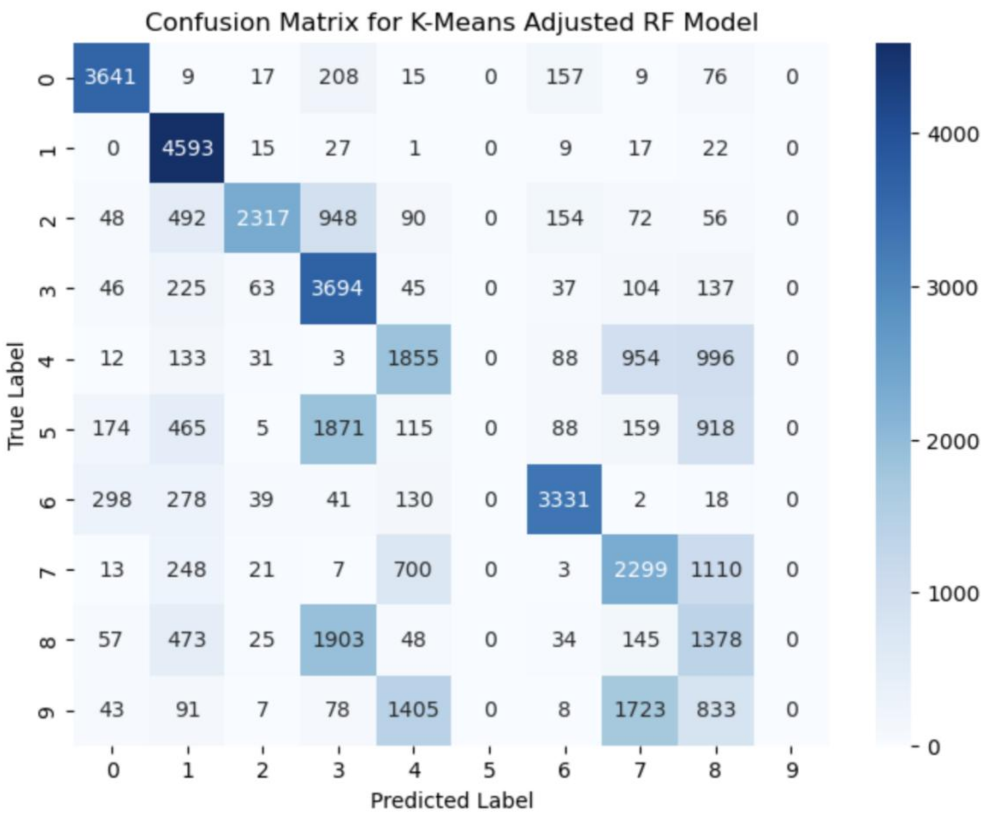
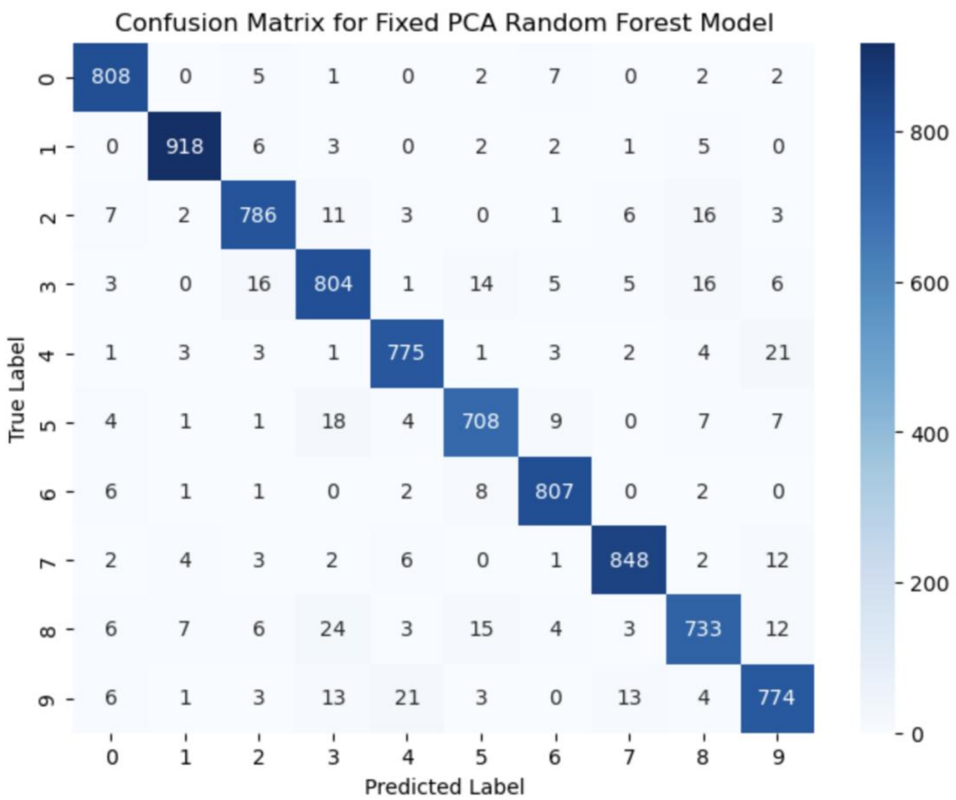


Figure 4: Model 3 Accuracy and Confusion Matrix

- Corrected PCA approach (Model 4) led to the same number of Principal Component (154), but better generalization and improved performance. (Refer to Figure 5: Model 3 Accuracy and Confusion Matrix, showing 0.03% improvement over Model 2)



Final Recommendations

- Random Forest remains a strong baseline model.
- Dimensionality reduction via PCA should be carefully applied to avoid data leakage.
- Unsupervised learning techniques like K-Means should be used in combination with supervised models for feature extraction.
- Hyperparameter tuning is essential for optimizing performance.