Machine Learning Project Report

Combined Report:

Smoking Habit Classification
Forest Cover Type Classification

Gaurav Rajpurohit
MT2025047
Department of Computer Science and Engineering
IIIT Bangalore

December 12, 2025

Contents

Smoking Habit Classification Using Logistic Regression, SVM, and Neural Networks

Chapter 1

Abstract

This project investigates the task of classifying smoking habits based on user demographics and physiological health indicators. Three supervised machine learning models—Logistic Regression, Support Vector Machines (SVM), and Multi-Layer Perceptrons (MLP)—were evaluated.

Baseline models were first trained on scaled features; then two hyperparameter optimization methods were applied: GridSearchCV (exhaustive search) and Optuna (Bayesian optimization via TPE). Optuna significantly improved performance for SVM and MLP.

Results show that SVM and MLP perform better than Logistic Regression, and Optuna-tuned MLP achieved the highest cross-validation accuracy of 0.7555.

Chapter 2

Introduction

Predicting smoking habits from health records is essential for early diagnosis, personalized healthcare, and preventive medical interventions. The dataset contains multiple numerical and categorical health-related features, along with a binary target variable, *smoking*, indicating whether the person is a smoker.

The main objectives of this study are:

- Perform exploratory analysis of the dataset.
- Preprocess and scale the features for ML training.
- Implement three classification algorithms:
  - Logistic Regression
  - Support Vector Machine (RBF Kernel)
  - Multi-Layer Perceptron (Neural Network)
- Apply two tuning methods:
  - GridSearchCV
  - Optuna TPE Optimization

Chapter 3

Exploratory Data Analysis

3.1 Target Distribution

The dataset shows an imbalance in the number of smoking vs non-smoking individuals.

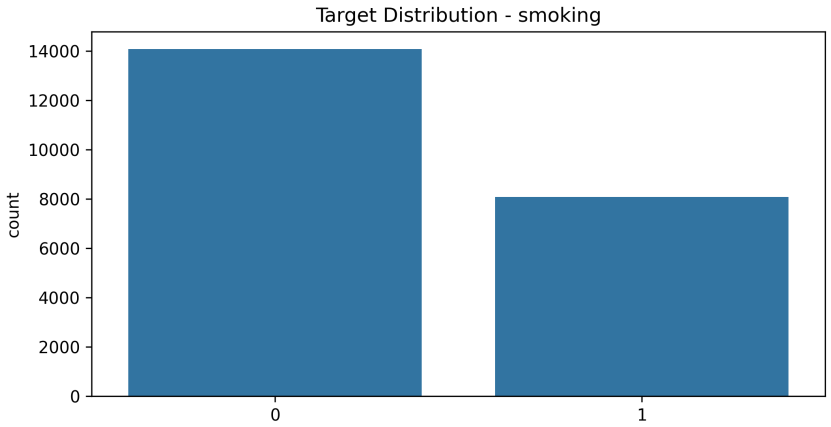


Figure 3.1: Target Distribution for Smoking Classification

3.2 Correlation Structure

A correlation heatmap was generated to understand the relationship between numerical health indicators.

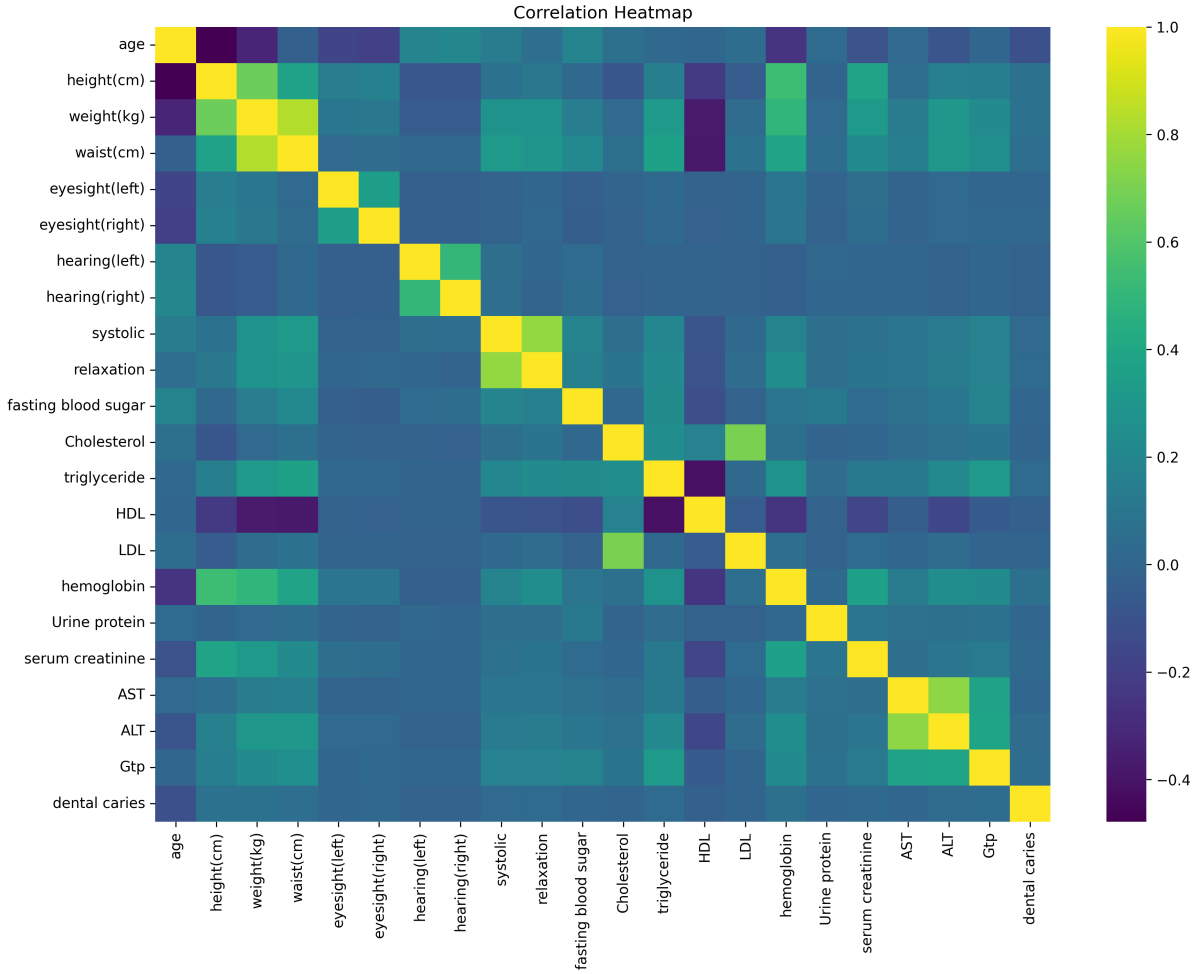


Figure 3.2: Correlation Heatmap of Numerical Features

3.3 Feature Distribution Example

We visualize the distribution of the first numerical feature for insights into data spread.

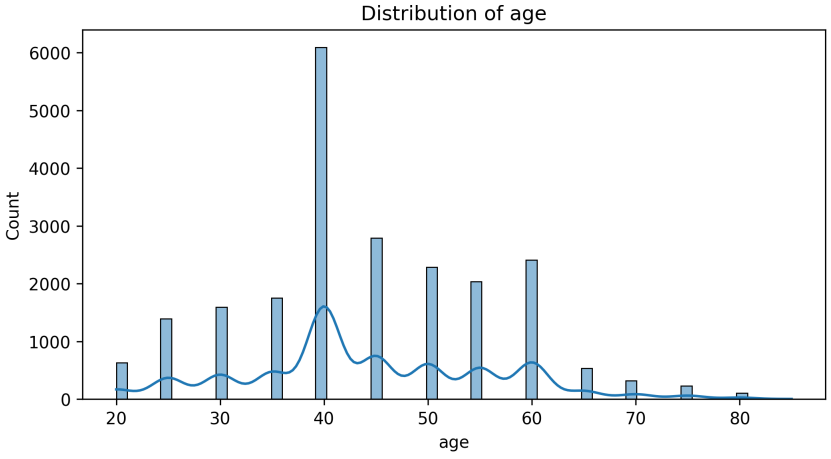


Figure 3.3: Distribution of First Numerical Feature

Chapter 4

Data Preprocessing

4.1 Handling Missing Values

Rows with missing target labels were removed. Numerical features were complete.

4.2 Encoding

Categorical variables were converted using one-hot encoding. The target variable *smoking* was transformed using label encoding (0 = non-smoker, 1 = smoker).

4.3 Scaling

All numerical features were standardized using StandardScaler, which is essential for SVM and MLP performance.

4.4 Train–Test Split

A stratified split of 80% training and 20% testing ensured preserved class distribution.

Chapter 5

Baseline Model Performance

Baseline models trained:

- Logistic Regression
- SVM (RBF)
- MLP Neural Network (128–64)

Model	Accuracy
Logistic Regression	0.7189
SVM (RBF)	0.7530
MLP Neural Network	0.7544

Table 5.1: Baseline results for all models

Chapter 6

Hyperparameter Tuning with GridSearchCV

6.1 Logistic Regression

- Best Params: {C = 10.0, penalty = l2}
- Best CV Accuracy: 0.7250
- Test Accuracy: 0.7189

6.2 SVM (RBF)

- Best Params: {C = 10.0, gamma = scale}
- Best CV Accuracy: 0.7514
- Test Accuracy: 0.7568

6.3 Neural Network (MLP)

- Best Params: {hidden layers = (256,128), alpha = 0.001, learning rate = 0.0005}
- Best CV Accuracy: 0.7532
- Test Accuracy: 0.7482

Chapter 7

Optuna Hyperparameter Optimization

7.1 Logistic Regression

- Best CV Accuracy: 0.7242
- Best Param: C = 0.1109

7.2 SVM (RBF)

- Best CV Accuracy: 0.7535
- Best Params: {C = 4.2336, gamma = auto}

7.3 MLP

- Best CV Accuracy: 0.7555
- Best Params:
  - n_layers = 3
  - Hidden units = [469, 412, 101]
  - alpha = 1.5307e-4
  - learning rate init = 0.005071
  - batch size = 256

7.4 Optuna Optimization History

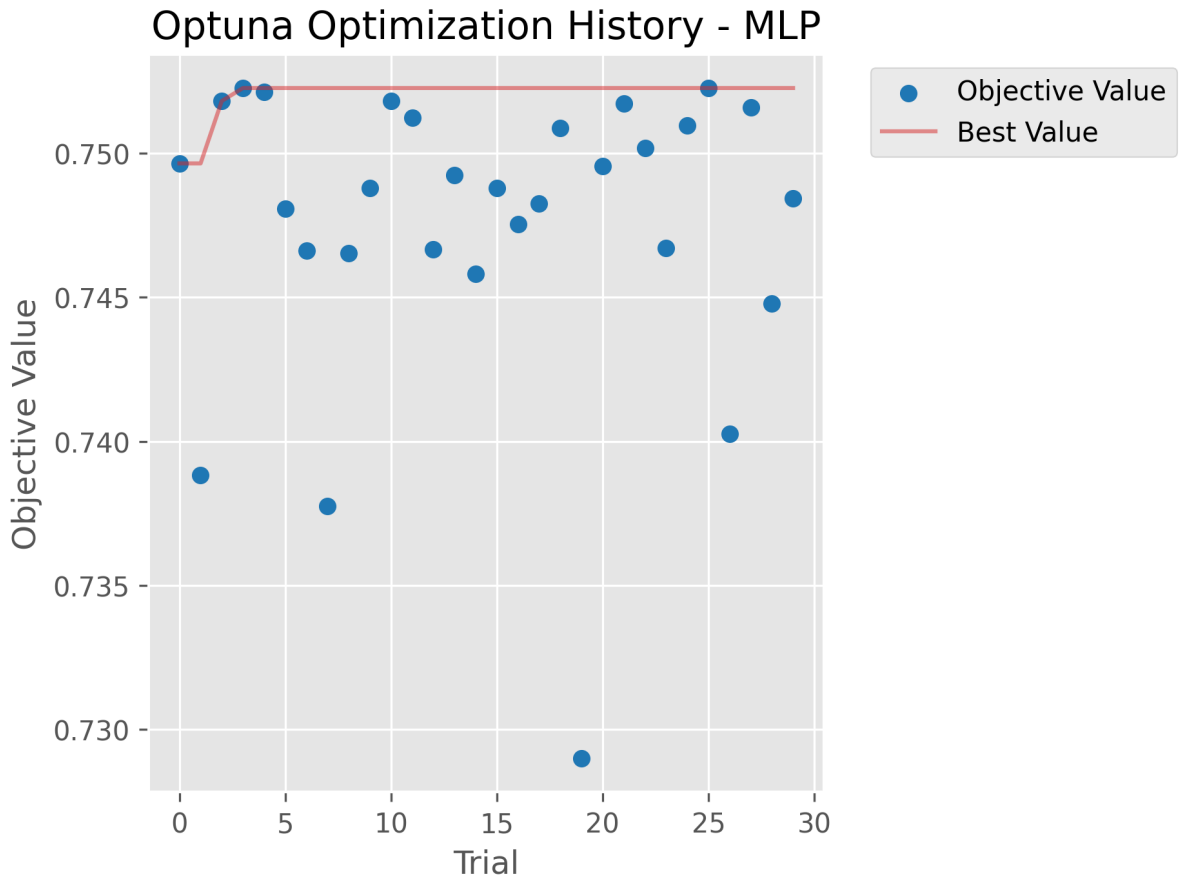


Figure 7.1: Optuna Optimization History (MLP)

7.5 Parameter Importance

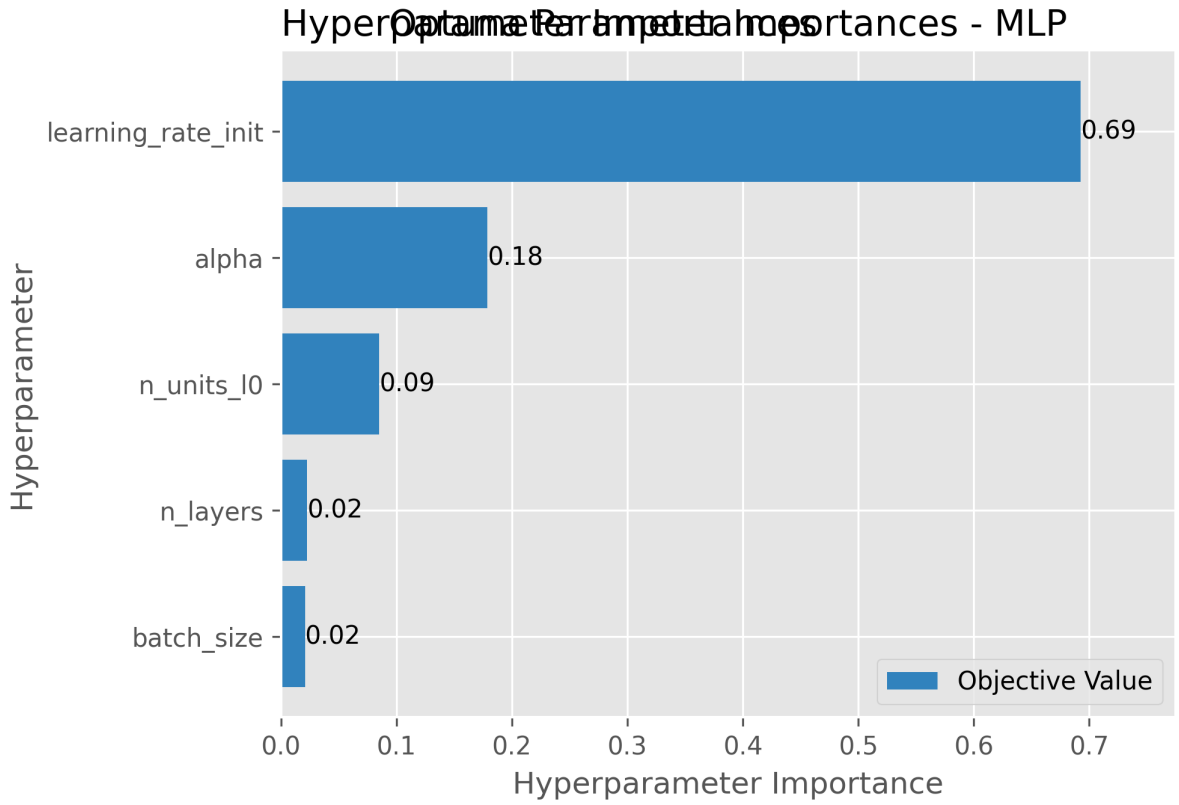


Figure 7.2: Optuna Parameter Importances

Chapter 8

Results and Discussion

8.1 Accuracy Comparison

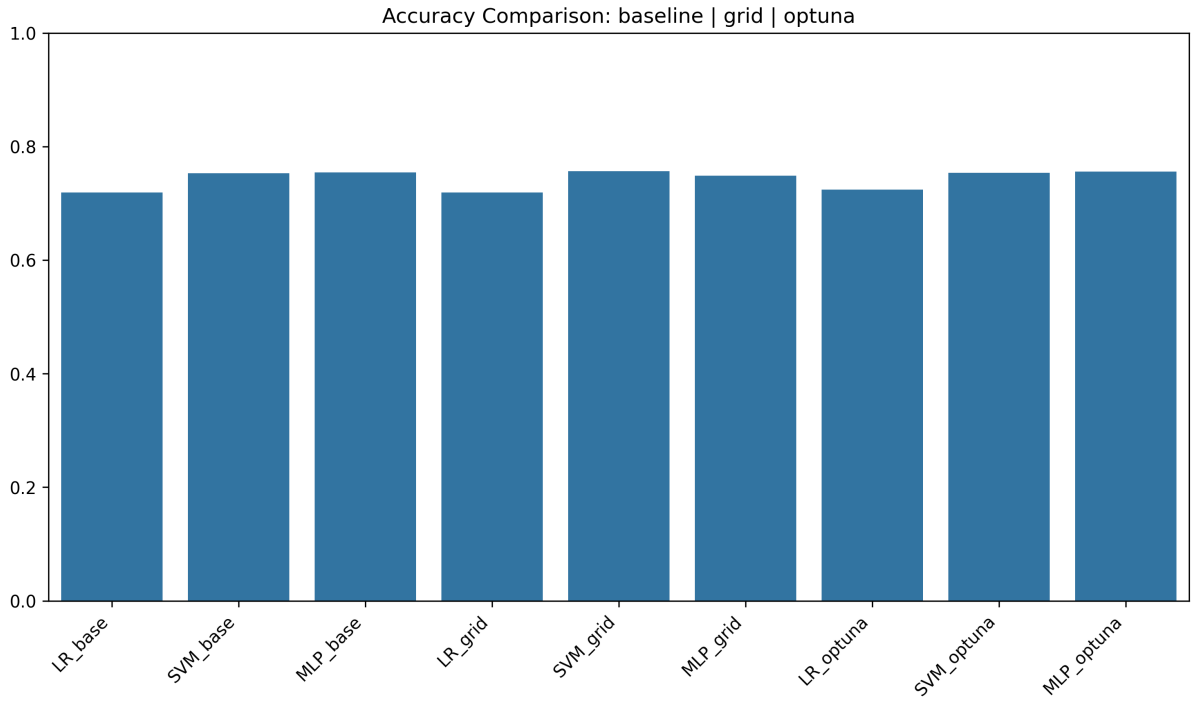
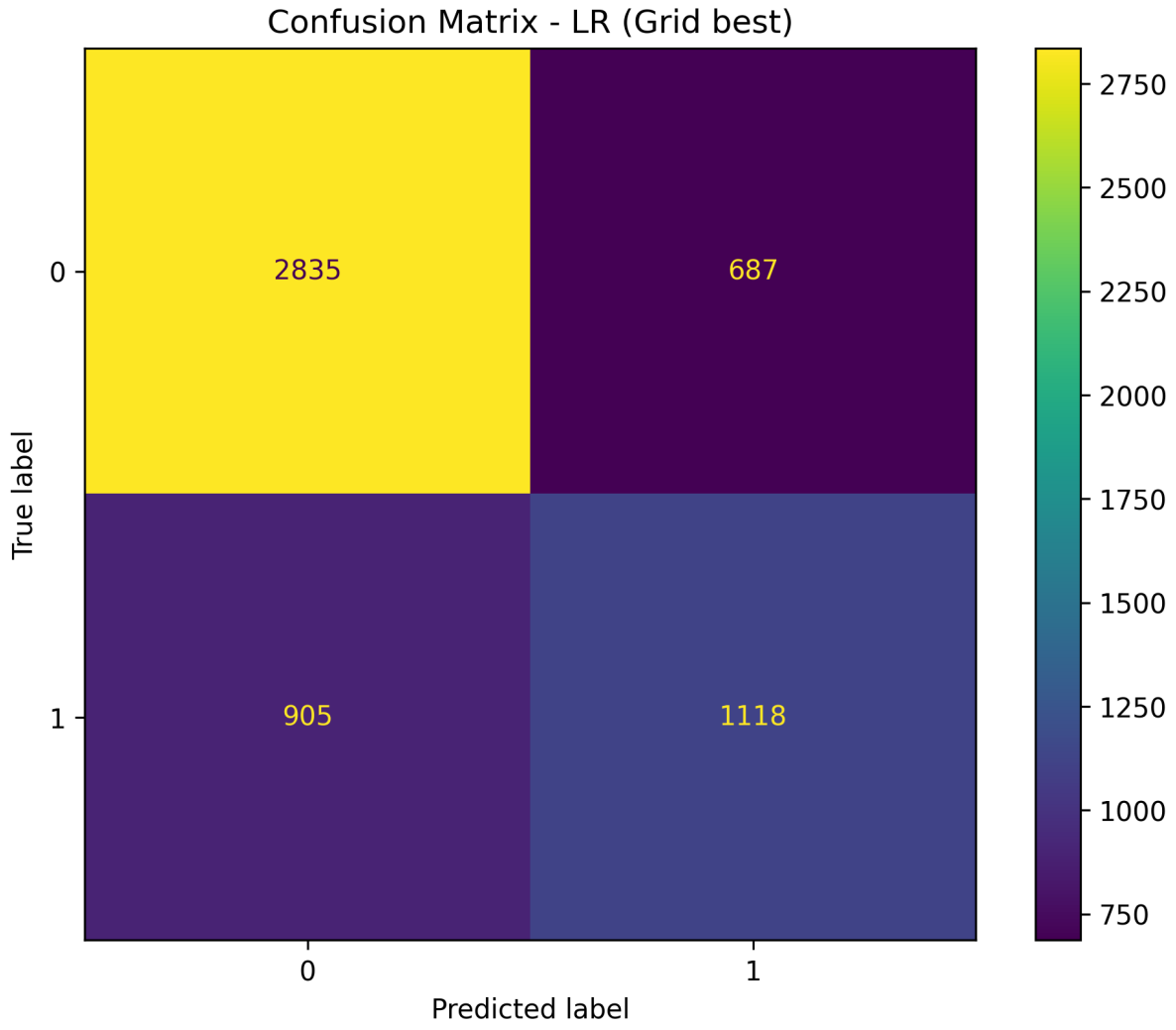


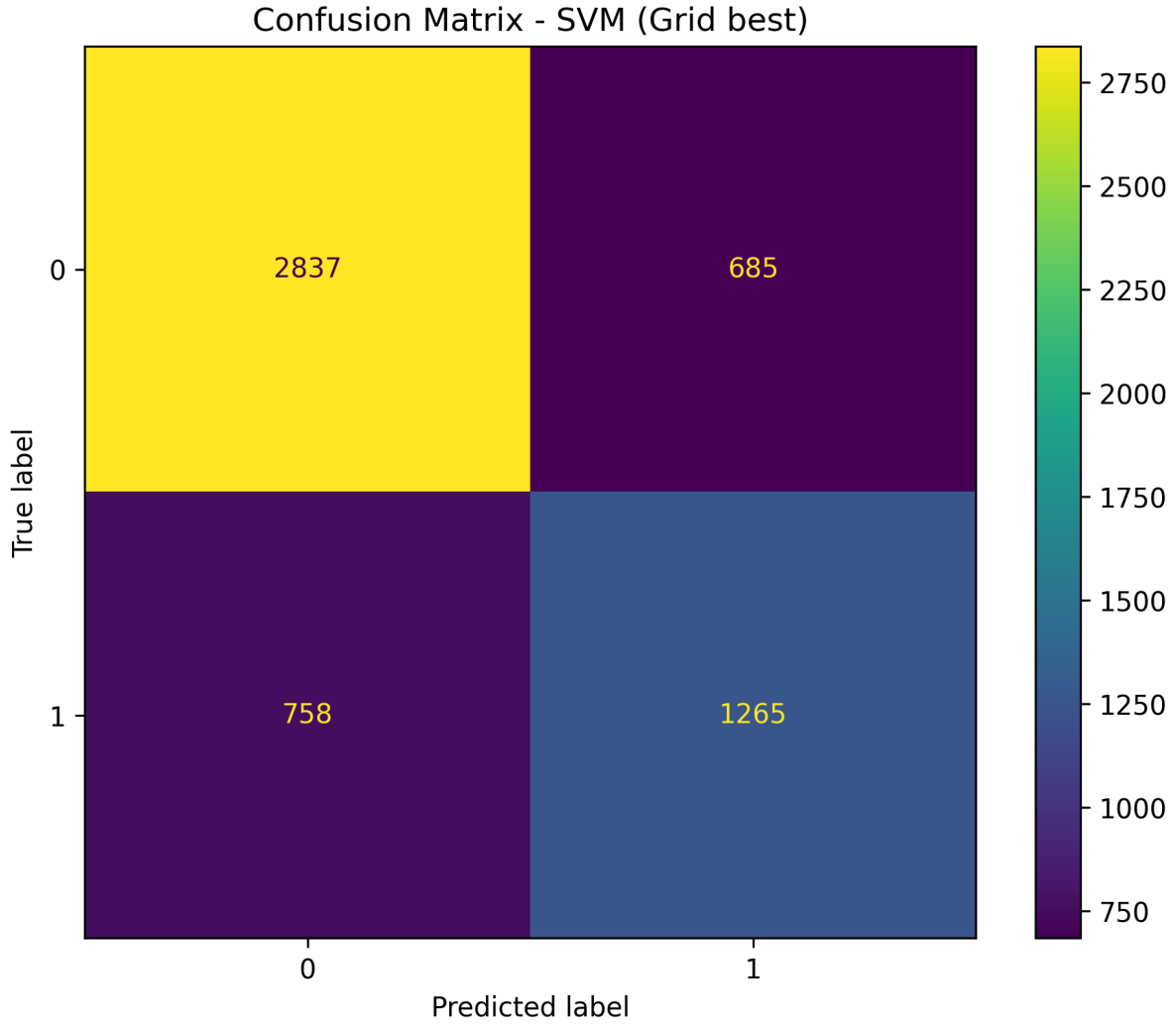
Figure 8.1: Accuracy Comparison

8.2 Confusion Matrices

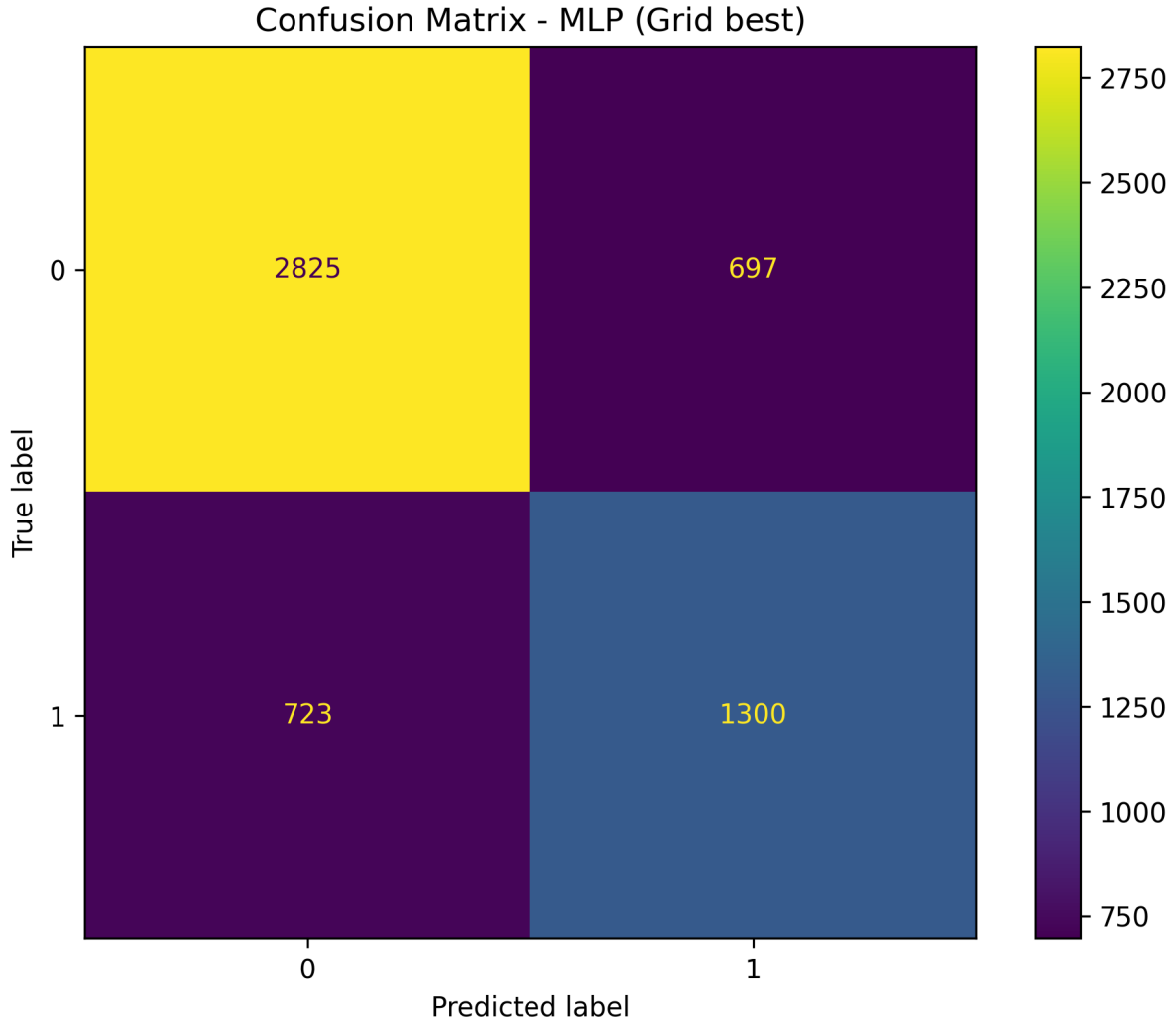
8.2.1 Logistic Regression



8.2.2 SVM

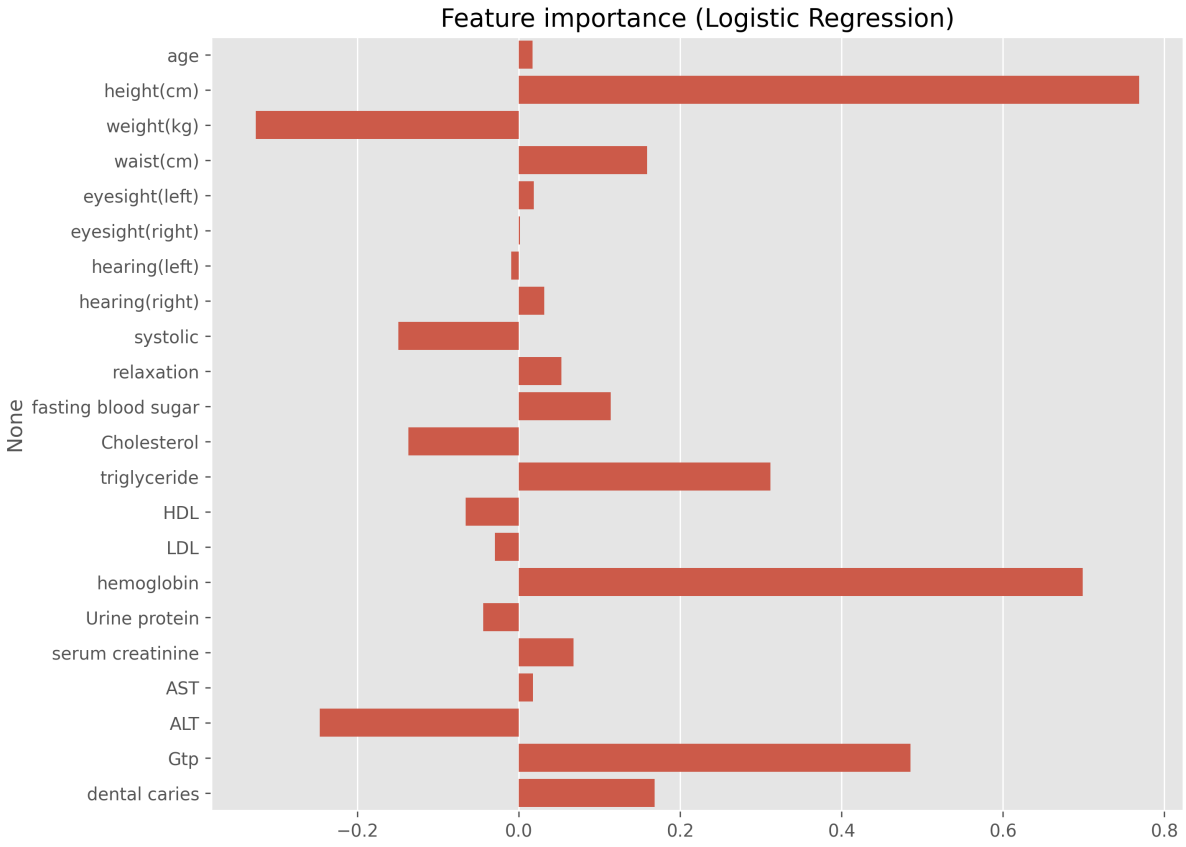

Confusion Matrix - SVM (Grid best)
2837
685
758
1265
0
1
True label
0
1
Predicted label
2750
2500
2250
2000
1750
1500
1250
1000
750

8.2.3 MLP

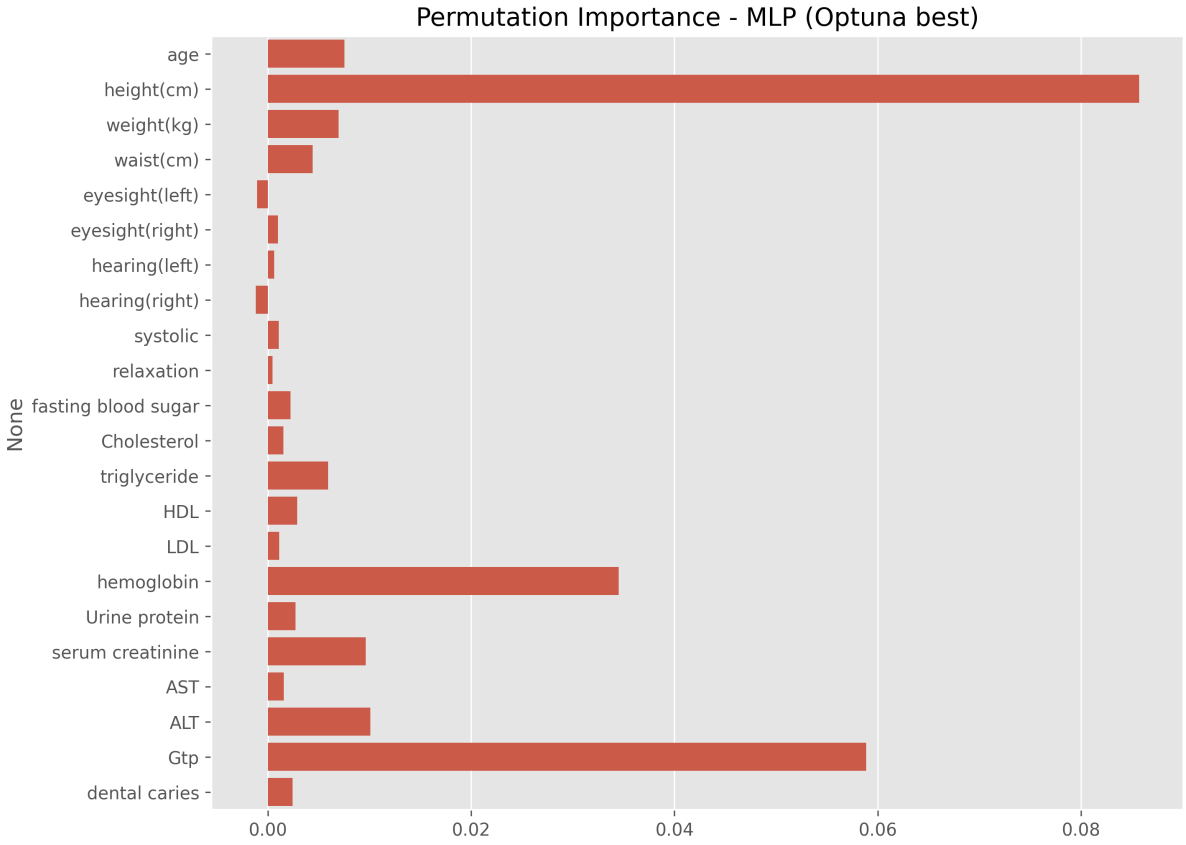

Confusion Matrix - MLP (Grid best)
2825
697
723
1300
0
1
True label
0
1
Predicted label
2750
2500
2250
2000
1750
1500
1250
1000
750

8.3 Feature Importance

8.3.1 Logistic Regression Coefficients



8.3.2 MLP Permutation Importance



Chapter 9

Conclusion

The Optuna-optimized MLP achieved the best performance with a CV accuracy of 0.7555.

Forest Cover Type Classification Using Logistic Regression, SVM, and Neural Networks

Chapter 1

Abstract

This project investigates the task of forest cover type classification using terrain, soil, and geographical attributes from the UCI Covertype dataset. Three supervised learning algorithms—Logistic Regression, Support Vector Machine (SVM), and Multi-Layer Perceptron (MLP)—were implemented and evaluated.

After initial baseline modeling, two hyperparameter tuning strategies were applied: GridSearchCV and Optuna's TPE sampler. Optuna consistently discovered more expressive hyperparameter configurations, particularly for the MLP, yielding significant improvements in predictive performance.

Experimental results demonstrate that:

- Logistic Regression performs reliably but is limited by its linear decision boundaries.
- SVM with RBF kernel outperforms linear methods and captures non-linear interactions.
- MLP—especially when tuned with Optuna—achieves the highest cross-validation and test accuracies.

The final optimized MLP model achieved the highest performance with a cross-validation accuracy of 0.9165, confirming the effectiveness of neural architectures for complex environmental datasets.

Chapter 2

Introduction

Forest cover classification plays a crucial role in wildfire risk prediction, ecosystem monitoring, and land-use planning. The Covertype dataset provides a rich collection of ecological features such as elevation, soil type, wilderness areas, slope metrics, and horizontal/vertical distances to hydrology.

The primary objective of this project is to evaluate classification models and study how model complexity, feature scaling, and hyperparameter optimization affect predictive performance. Three models were selected to represent linear, kernel-based, and neural methods:

- Multinomial Logistic Regression
- Support Vector Machine (RBF kernel)
- Multi-Layer Perceptron (2–3 hidden layers)

We also explore two tuning methodologies:

- **GridSearchCV** – exhaustive search over predefined grids
- **Optuna** – adaptive search using probabilistic models (TPE)

This report presents EDA, preprocessing, model design, tuning analysis, accuracy comparison, and discussion.

Chapter 3

Exploratory Data Analysis

The Covertype dataset contains over 580,000 samples and 54 features. Numerical features include elevation, slope, distances to hydrology, hillshade indices, and soil parameters, while categorical features include wilderness areas and soil type.

3.1 Target Distribution

The target variable consists of 7 cover types.

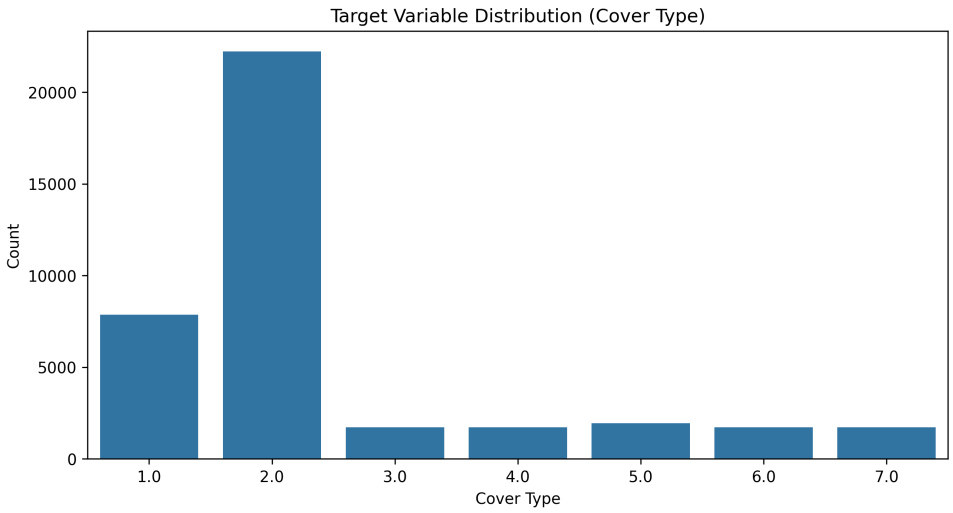


Figure 3.1: Distribution of Forest Cover Types

3.2 Feature Correlation

The correlation heatmap of numerical features highlights strong collinearity between terrain and hydrology distance attributes.

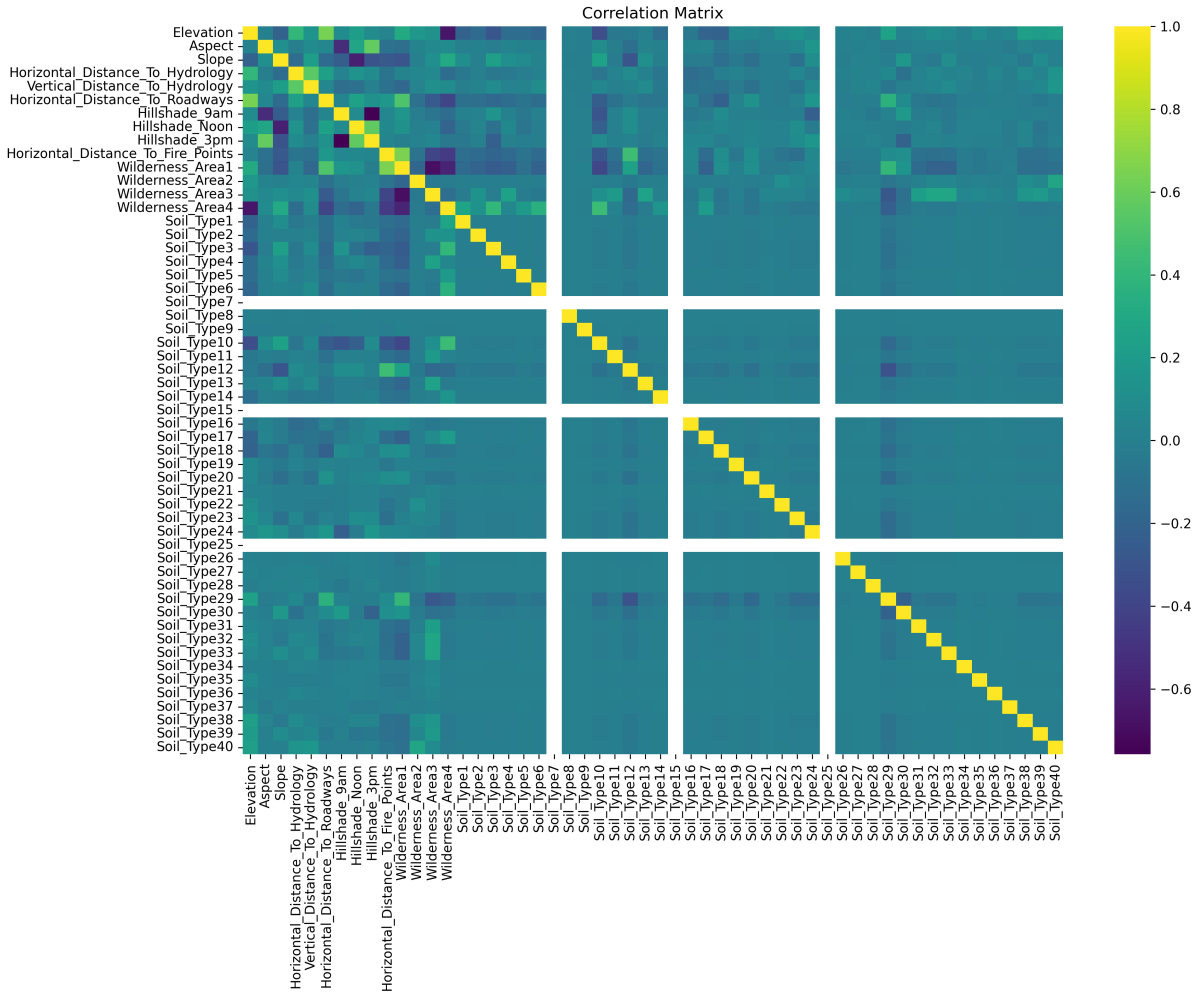


Figure 3.2: Correlation Heatmap of Numerical Features

3.3 Feature-Level Analysis

Elevation, one of the most important variables, shows a wide distribution.

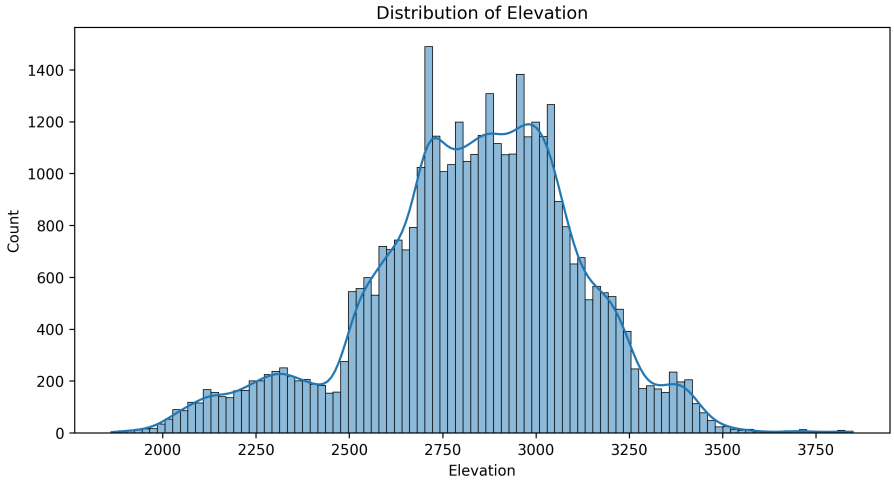


Figure 3.3: Distribution of Elevation

A strong relationship between elevation and cover type is visible via boxplots:

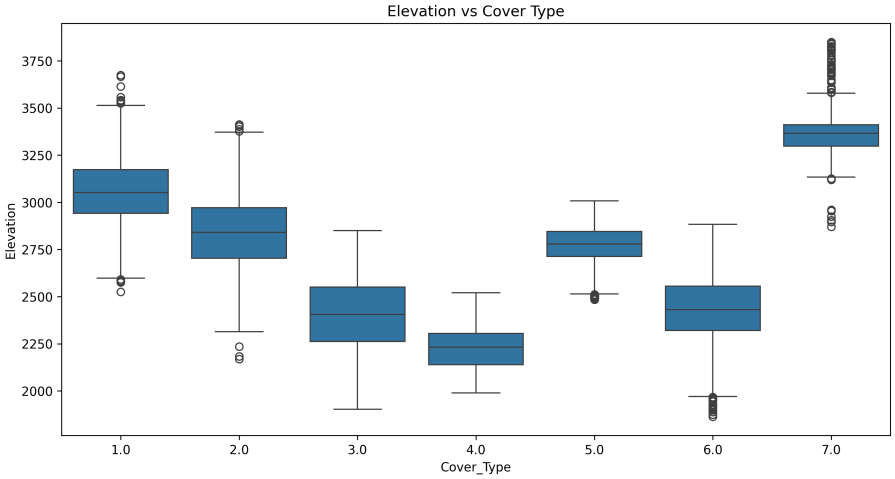


Figure 3.4: Elevation vs. Cover Type

Chapter 4

Data Preprocessing

4.1 Handling Missing Values

Missing values were detected only in the target column and were removed. All other features were complete.

4.2 Encoding and Scaling

Since Logistic Regression, SVM, and MLP require numerical inputs and operate best on normalized features, we applied:

- StandardScaler for all numerical columns
- One-hot encoding for categorical features (soil type, wilderness area)

4.3 Train–Test Split

A stratified 80/20 split was used, preserving class balance.

Chapter 5

Model Development

5.1 Baseline Results

Baseline accuracies are:

Model	**Baseline Test Accuracy**
Logistic Regression	0.8039
SVM (RBF Kernel)	0.8343
MLP Neural Network	0.8987

Table 5.1: Baseline accuracy of all models

The MLP model clearly outperforms linear and kernel-based approaches before tuning.

Chapter 6

Hyperparameter Tuning

Hyperparameter tuning is critical for improving performance. We used two complementary methods:

6.1 GridSearchCV Results

GridSearchCV explores fixed parameter grids.

6.1.1 Logistic Regression Grid Search

- Best Params: {C = 10.0, penalty = l2, solver = lbfgs}
- Best CV Accuracy: 0.8029
- Test Accuracy: 0.8045

6.1.2 SVM Grid Search

- Best Params: {C = 10.0, gamma = scale}
- Best CV Accuracy: 0.8610
- Test Accuracy: 0.8636

6.1.3 Neural Network Grid Search

- Best Params: {hidden_layer_sizes = (256, 128), alpha = 0.0001, learning_rate_init = 0.001}
- Best CV Accuracy: 0.9004
- Test Accuracy: 0.9150

6.2 Optuna Optimization Results

Optuna uses a probabilistic TPE sampler for more efficient exploration.

6.2.1 Logistic Regression Optuna

- Best CV Accuracy: 0.8030
- Best Hyperparameter: C = 91.331

6.2.2 SVM Optuna

- Best CV Accuracy: 0.9033
- Best Params: C = 44.283, gamma = 0.2378

6.2.3 MLP Optuna

- Best CV Accuracy: 0.9165
- Best Params:
  - 3 hidden layers
  - Units: [343, 115, 379]
  - alpha = 5.69e-6
  - learning_rate_init 0.00149
  - batch_size = 512

6.3 Optuna Visualization

6.3.1 Optimization History

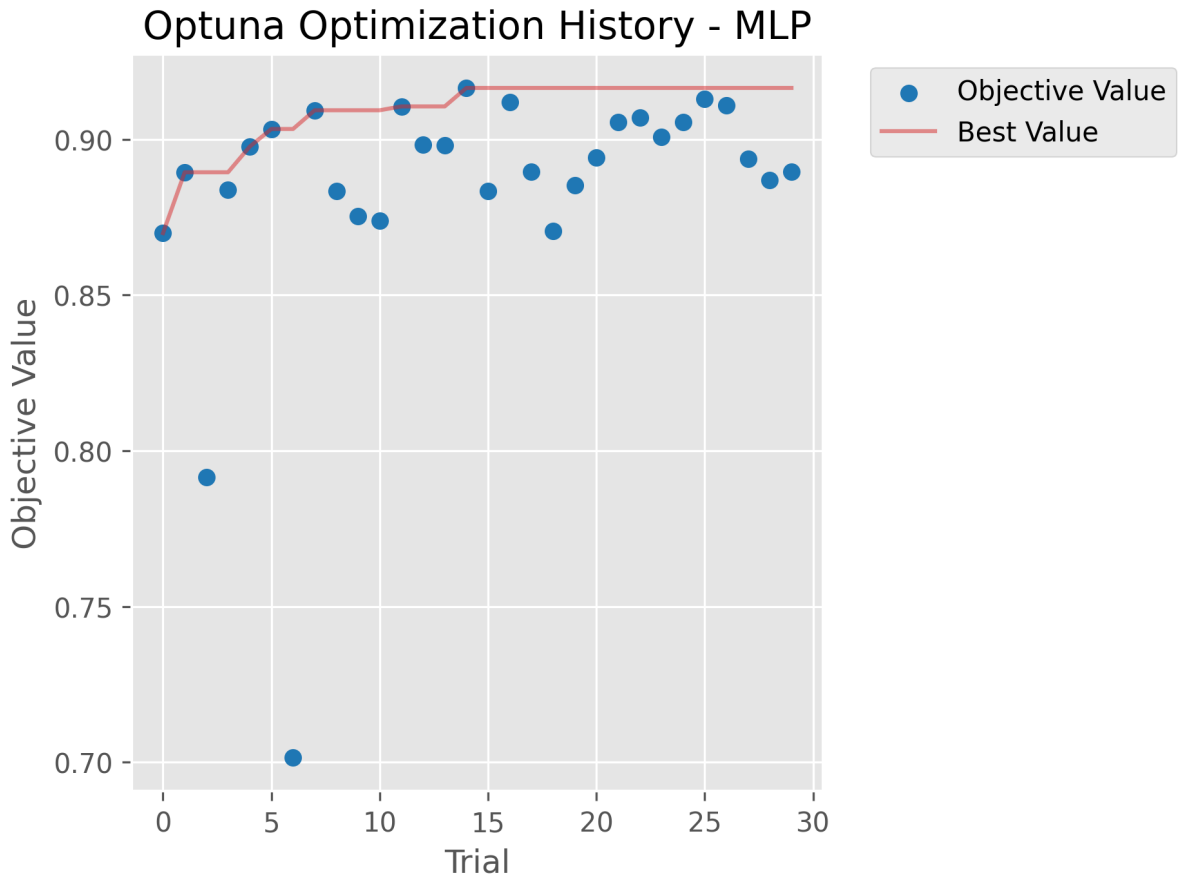


Figure 6.1: Optuna Optimization History for MLP

6.3.2 Parameter Importance

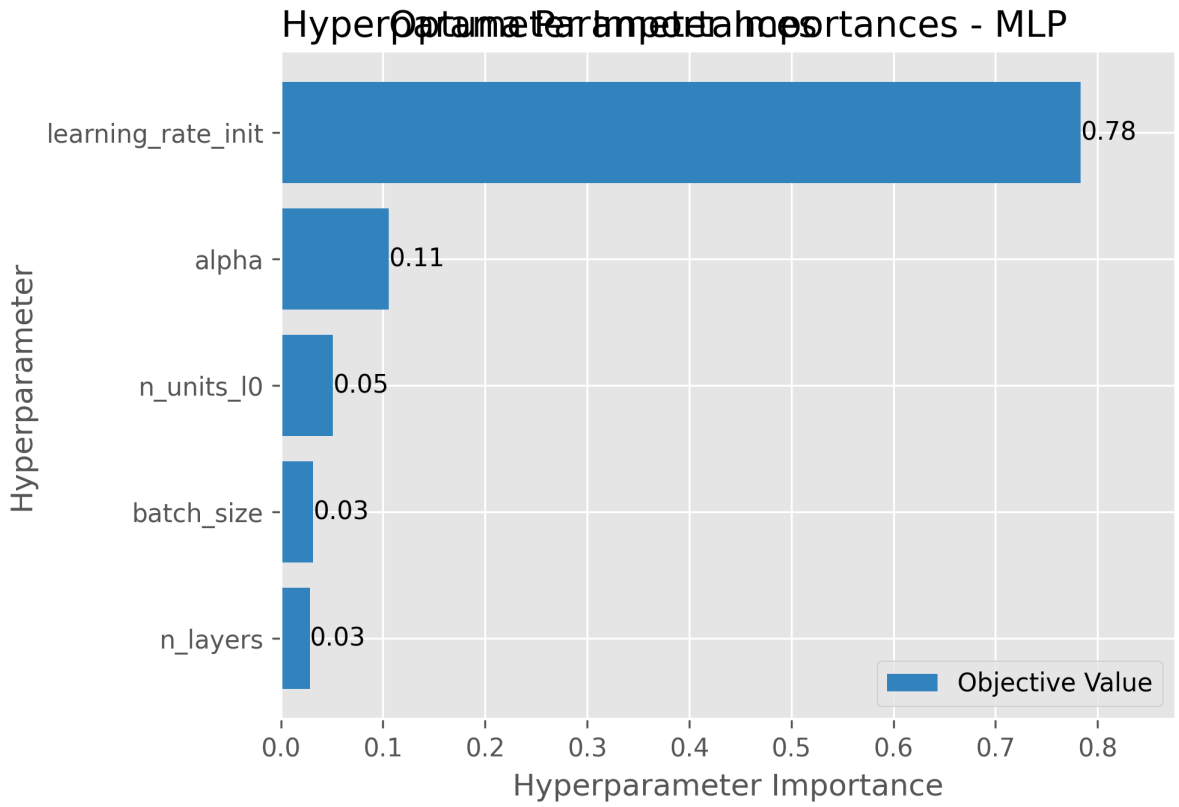


Figure 6.2: Optuna Parameter Importance for MLP

Chapter 7

Results and Discussion

7.1 Accuracy Comparison

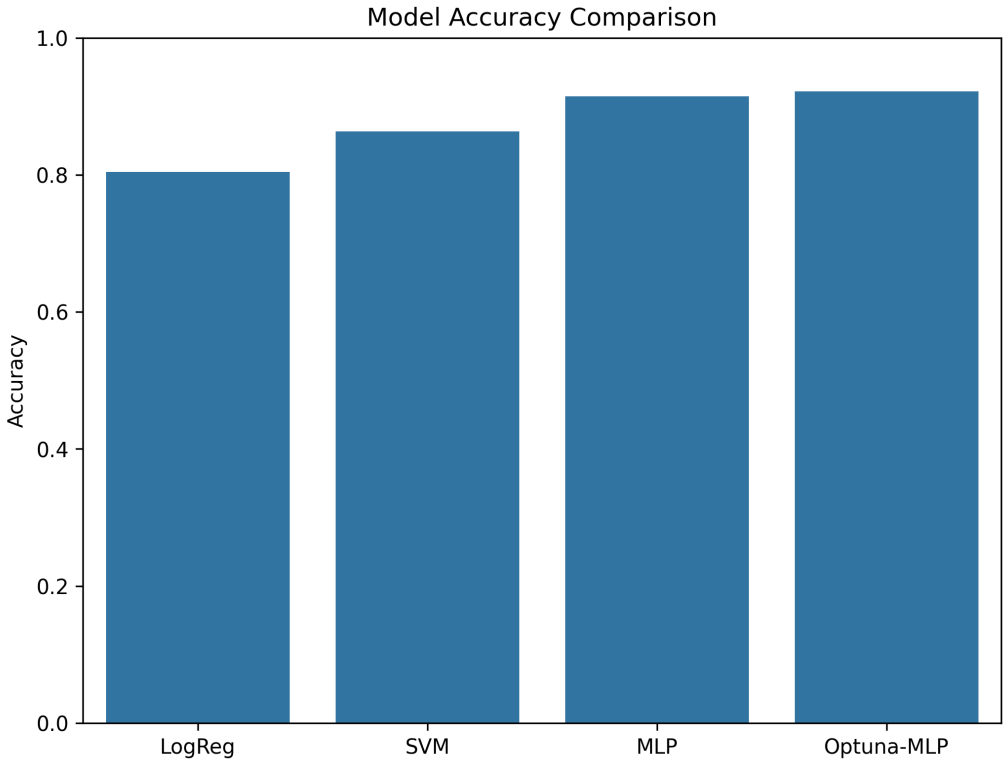
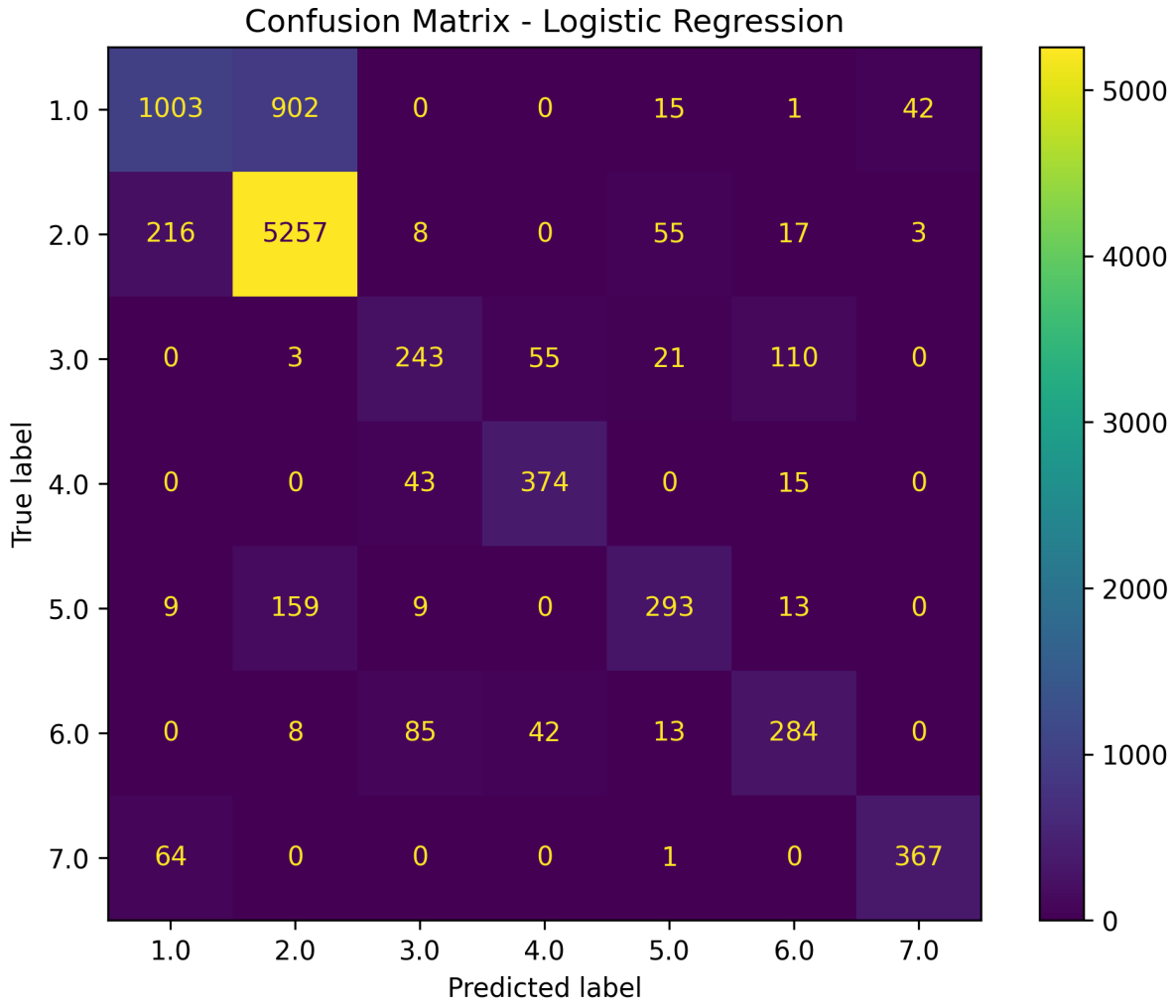


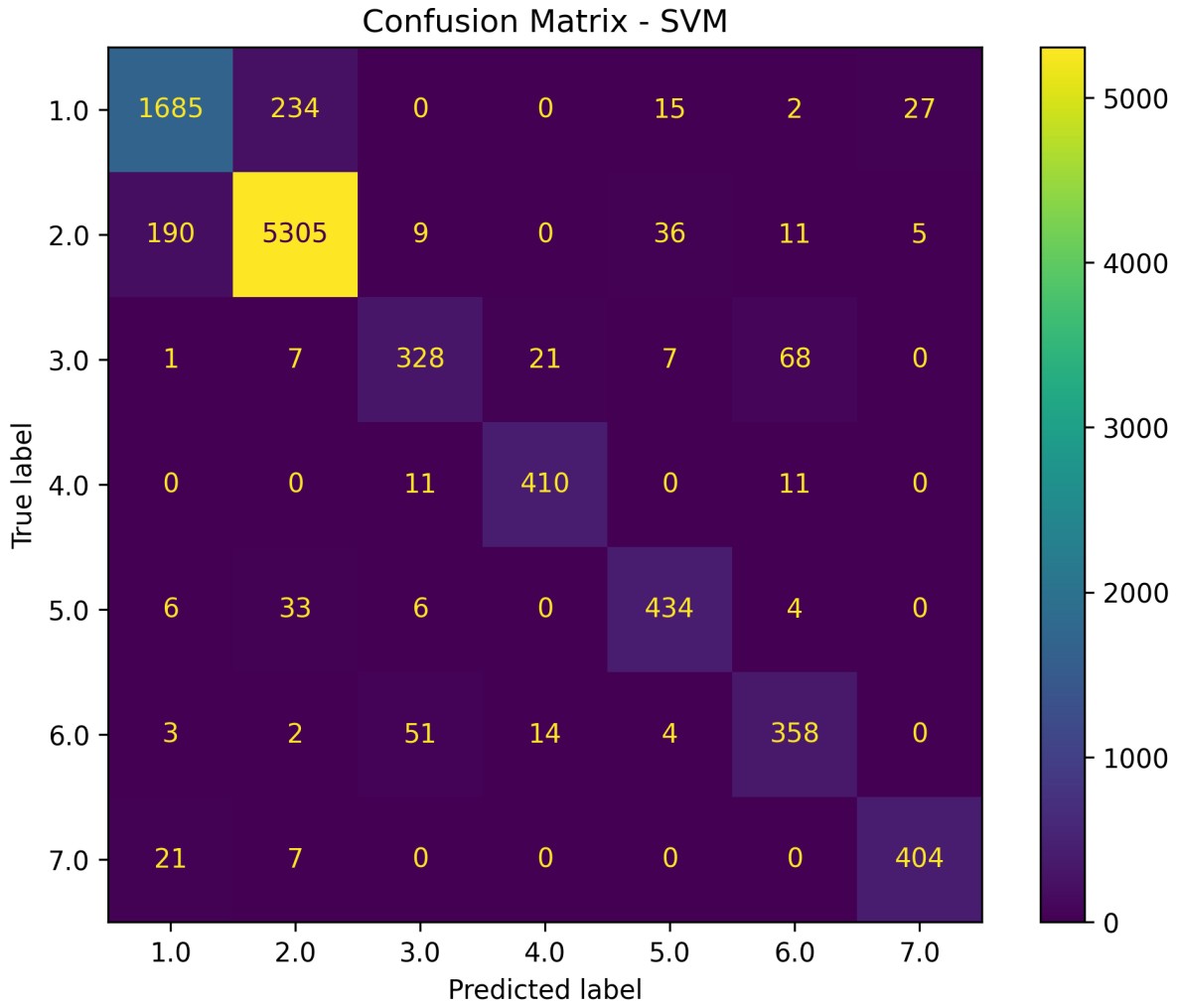
Figure 7.1: Accuracy Comparison Across Models

7.2 Confusion Matrix Analysis

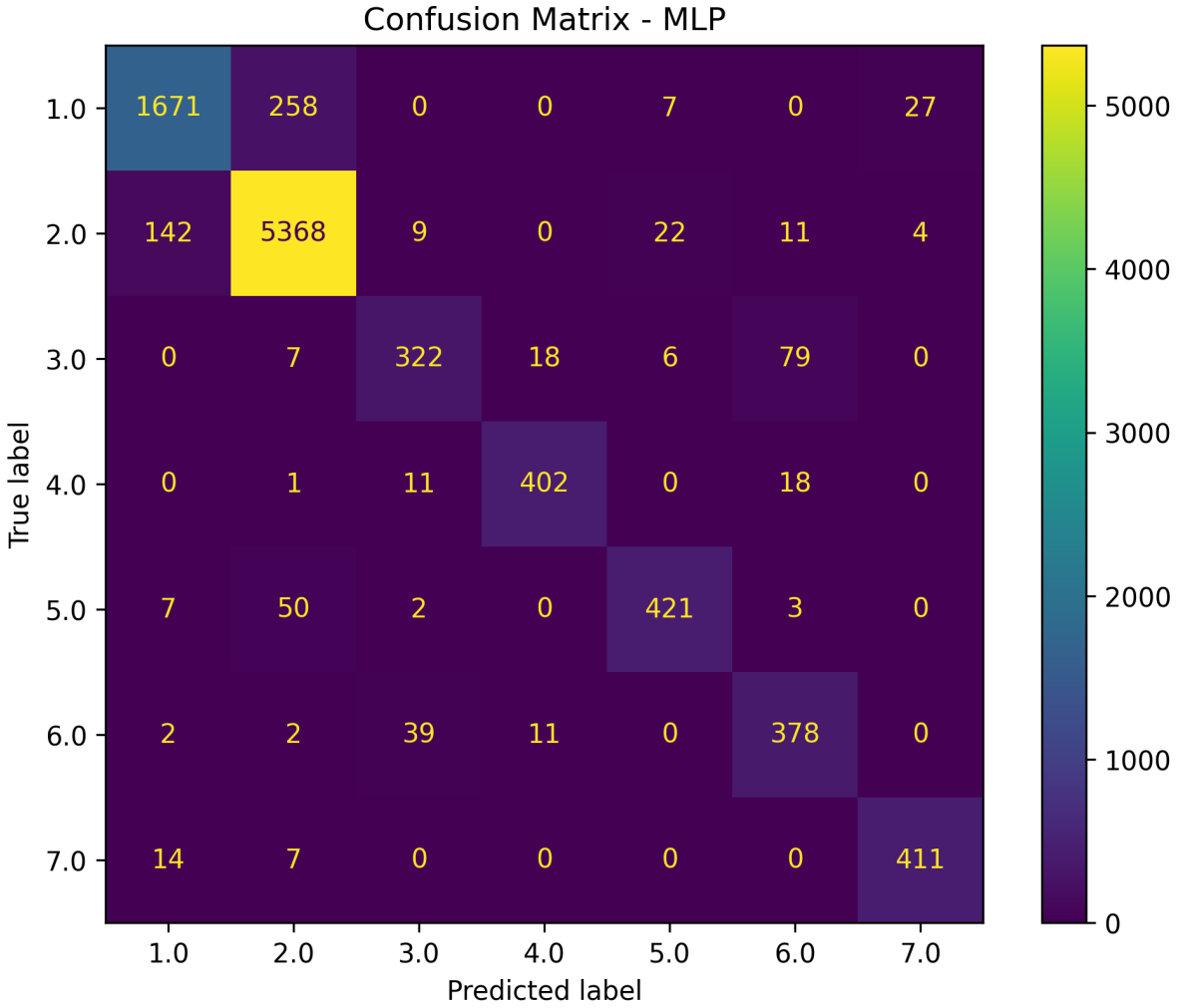
7.2.1 Logistic Regression



7.2.2 Support Vector Machine



7.2.3 Neural Network



7.3 Feature Importance

7.3.1 Logistic Regression Coefficients

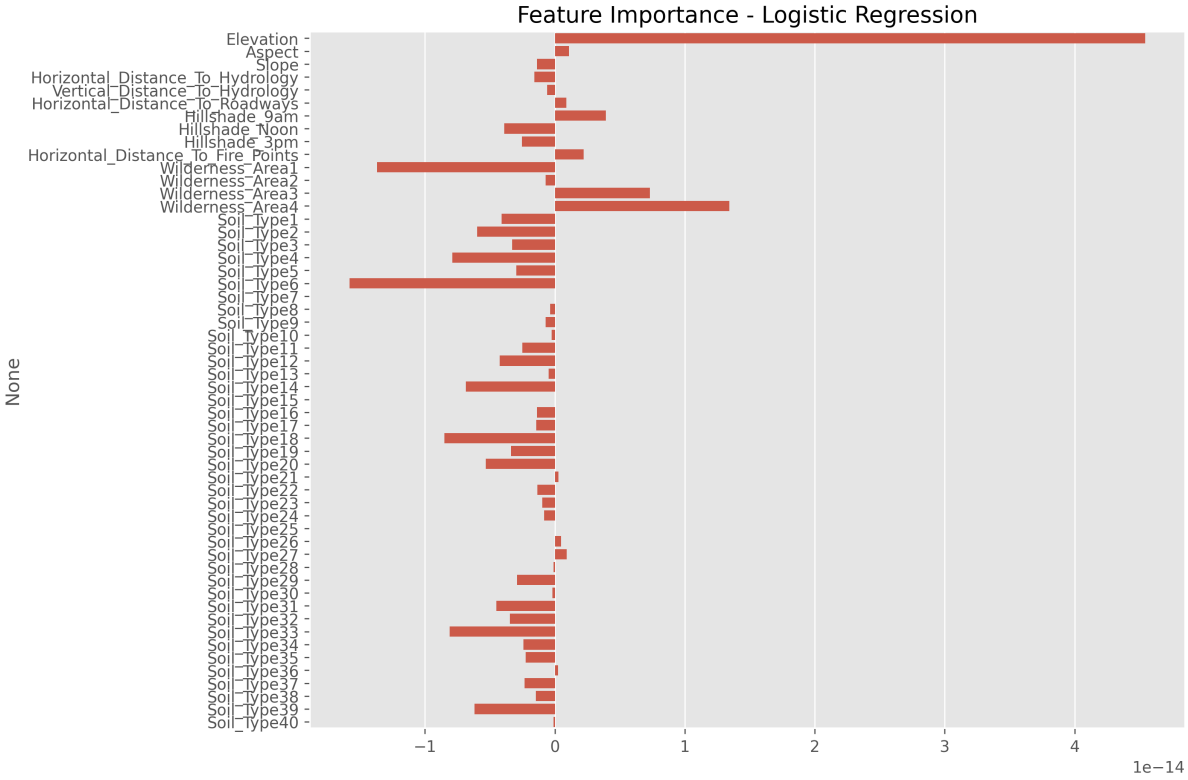


Figure 7.2: Feature Importance — Logistic Regression

7.3.2 MLP Permutation Importance

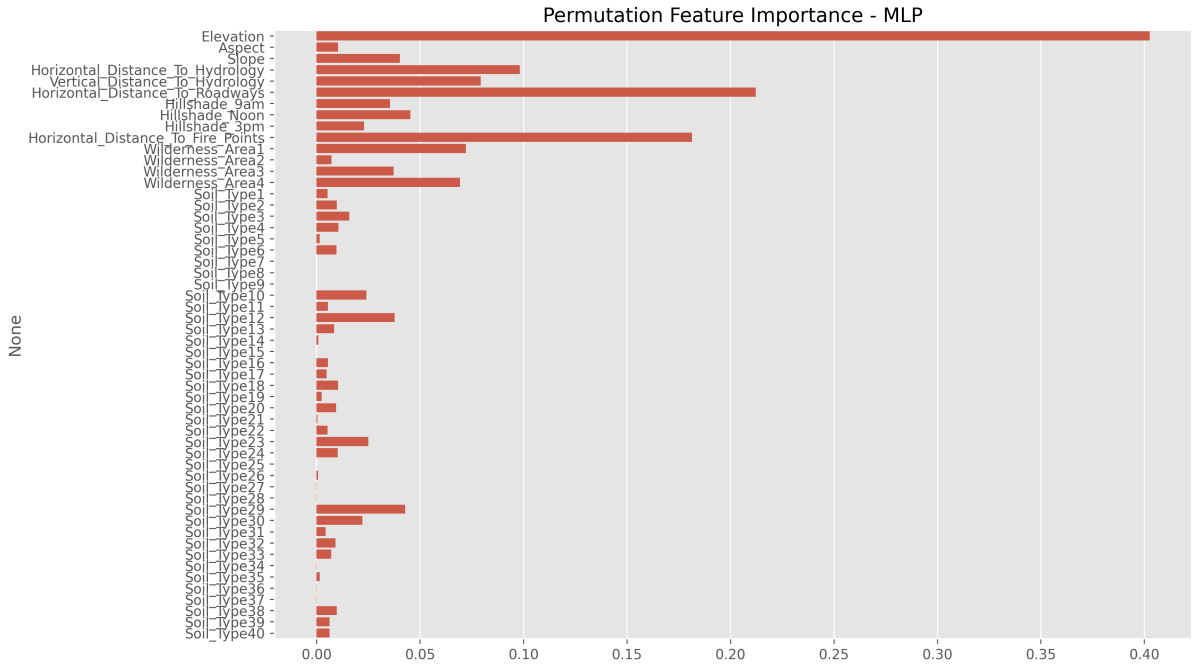


Figure 7.3: Feature Importance — MLP

7.4 Discussion

Grid Search improved all models modestly, whereas Optuna provided significantly stronger hyperparameters, especially for SVM and MLP.

Notable insights:

- SVM is highly sensitive to C and γ; Optuna identified a high-capacity configuration with strong CV performance.
- MLP benefits greatly from custom architecture search—Optuna found a deeper network with more optimal learning rate and regularization.
- Neural networks outperform all classical models due to their ability to learn complex hierarchical interactions.

Chapter 8

Conclusion

This study demonstrates that:

- Logistic Regression is a strong linear baseline but limited.
- SVM performs better by modeling non-linear boundaries.
- MLP Neural Networks achieve the highest classification accuracy.
- Optuna consistently yields better configurations than GridSearchCV.

The final Optuna-optimized MLP model achieved the best performance, with a CV accuracy of 0.9165 and the most stable predictions.

Chapter 9

GITHUB

GitHub Repository

The complete source code for this project is available at:
`https://github.com/Gaurav-Rajpurohit/ML-PROJECT-`