Forest Cover Type Prediction

Submitted by: Parveen Kashyap
Role: Machine Learning Intern
Internship ID: UMID13062542814
Company: Unified Mentor Pvt. Ltd.

Abstract

This project develops a supervised machine learning system to predict forest cover type for 30m × 30m land patches in the Roosevelt National Forest, Colorado. The model uses cartographic and environmental attributes such as elevation, slope, hydrology distances, hillshade indices, wilderness area, and soil type to classify each cell into one of seven cover types (Spruce-Fir, Lodgepole Pine, Ponderosa Pine, Cottonwood/Willow, Aspen, Douglas-fir, and Krummholz). The workflow includes exploratory data analysis (EDA), domain-driven feature engineering, preprocessing with feature scaling, benchmarking of multiple classification algorithms, and selection of a best-performing ensemble model. Experiments show that tree-based ensemble methods, especially ExtraTrees, achieve strong performance with an accuracy of about 90% on the held-out validation set, outperforming linear and distance-based models. The final system is exposed through a Streamlit web interface that allows interactive prediction from user-specified geographic inputs.

Introduction

Increasing pressure on forest ecosystems due to climate change, anthropogenic activities, and natural disturbances has created a strong demand for data-driven tools to support forest management. Accurately identifying the dominant forest cover type in a given geographical cell is a critical task for planning timber harvests, biodiversity conservation, and wildfire risk mitigation. Traditional field surveys are accurate but expensive and slow to scale. In contrast, machine learning models built on remotely sensed and cartographic variables can provide fast and repeatable predictions over large areas.

This project focuses on predicting the predominant forest cover type for a 30m × 30m land patch within the Roosevelt National Forest of northern Colorado. The goal is to learn a multi-class classifier from an analysis dataset prepared by the forest department and then deploy this classifier as a usable web application. The work covers the complete pipeline from data understanding and feature engineering to model selection, evaluation, and deployment. The final deliverable is a robust, production-ready model coupled with an interactive Streamlit user interface.

Problem Statement

The core problem is to classify each land cell into one of seven forest cover types using only tabular attributes derived from geographic information systems (GIS) and survey data. Formally, given a feature vector describing topography, hydrology, proximity to infrastructure, wilderness area, and soil characteristics, the task is to predict the categorical variable CoverType.

Key challenges include:

- Handling a relatively high-dimensional feature space (55 original columns including 4 wilderness-area indicators and 40 soil-type indicators)
- Capturing non-linear relationships between environmental gradients and cover type
- Ensuring that the model generalizes well and remains interpretable for domain experts
- Building a training pipeline that can be re-executed to regenerate the scaler and model artifacts

The project aims to build, benchmark, and select a supervised learning model that addresses these challenges and can be served in real time for individual predictions.

Dataset Description

The dataset comes from an analysis performed in the Roosevelt National Forest of northern Colorado. Each record corresponds to a 30m × 30m patch of land and contains 55 columns. All rows in the working subset used here are complete (no missing values), with 15,120 observations and 55 attributes.

Target Variable

The target variable is CoverType, an integer-encoded forest cover type:

1. Spruce-Fir
2. Lodgepole Pine
3. Ponderosa Pine
4. Cottonwood/Willow
5. Aspen
6. Douglas-fir
7. Krummholz

For certain algorithms (such as XGBoost), labels are remapped from 1–7 to 0–6 for compatibility.

Main Continuous Features

- **Elevation**: Elevation in meters
- **Aspect**: Aspect in degrees (azimuth)
- **Slope**: Slope in degrees
- **HorizontalDistanceToHydrology**: Horizontal distance to nearest surface water features
- **VerticalDistanceToHydrology**: Vertical distance to nearest surface water features

- **HorizontalDistanceToRoadways**: Horizontal distance to nearest roadway
- **Hillshade9am, HillshadeNoon, Hillshade3pm**: Hillshade indices (0–255) at 9 am, noon, and 3 pm on summer solstice
- **HorizontalDistanceToFirePoints**: Horizontal distance to nearest wildfire ignition points

Categorical and Binary Features

- **WildernessArea1–4**: Four binary columns (0/1) representing different wilderness area designations
- **SoilType1–40**: Forty binary columns (0/1) encoding soil type designations

The dataset is a mix of continuous geographic variables and sparse one-hot encoded categorical variables. Initial inspection confirmed there are no null values, and class distributions show all seven cover types are represented, though with some imbalance.

Methodology

The project follows a comprehensive machine learning workflow implemented in Jupyter notebook (Forest_Cover_Prediction.ipynb) and Streamlit app (app.py). Key stages include:

1. **Data Loading and Inspection**: Load training data from train.csv, drop Id column if present, inspect shape/types (15,120 rows × 55 columns)
2. **Exploratory Data Analysis**: Analyze distributions of continuous variables by cover type, visualize cover types across wilderness areas, examine soil type frequencies, generate correlation heatmap, study bivariate relationships
3. **Feature Engineering**: Create interaction and composite features capturing non-linear relationships; retain original binary indicators
4. **Preprocessing and Scaling**: Split 80/20 train-validation, apply StandardScaler only to continuous features, keep binary indicators unscaled, persist scaler as bestscaler.pkl
5. **Model Benchmarking**: Train and evaluate seven algorithms (Logistic Regression, SVM, KNN, Decision Tree, Random Forest, ExtraTrees, XGBoost); compute accuracy, weighted precision, recall, F1-score
6. **Model Selection**: Select best-performing model based on validation accuracy and F1-score; save as bestforestmodel.pkl
7. **Deployment**: Integrate scaler and model into Streamlit web application with sidebar inputs for elevation, aspect, slope, hydrology distances, wilderness area, soil type

This pipeline ensures all steps—from raw data to web-based prediction—are reproducible.

Model Explanation

Several classification algorithms were benchmarked to balance interpretability, capacity to model complex interactions, and computational efficiency.

Baseline and Linear Models

Logistic Regression: Multinomial logistic regression with max_iter=1000 provides baseline and some interpretability via feature coefficients, but is limited in capturing non-linear structure.

Kernel and Distance-Based Models

Support Vector Machine (SVM): C-SVC classifier can handle non-linear decision boundaries with appropriate kernels but is relatively expensive for large datasets and sensitive to hyperparameter choices.

K-Nearest Neighbors (KNN): KNN with n_neighbors=5 is simple and non-parametric but performs less competitively in high-dimensional spaces and is affected by feature scaling.

Tree-Based Models

Decision Tree: DecisionTreeClassifier provides baseline tree-based approach with interpretability but tends to overfit and delivers moderate accuracy.

Random Forest: Ensemble of 200 decision trees substantially improves performance by averaging over many de-correlated trees.

ExtraTrees Classifier: Ensemble of 200 extremely randomized trees further improves accuracy and F1-score by randomizing both feature selection and cut points.

Gradient Boosting

XGBoost: XGBClassifier with 200 estimators and learning rate 0.1 performs strongly as gradient-boosted baseline.

Ensemble tree methods, especially ExtraTrees and Random Forest, outperform linear, kernel, and distance-based models. ExtraTrees was chosen as production model due to highest accuracy, strong F1-score, and robustness.

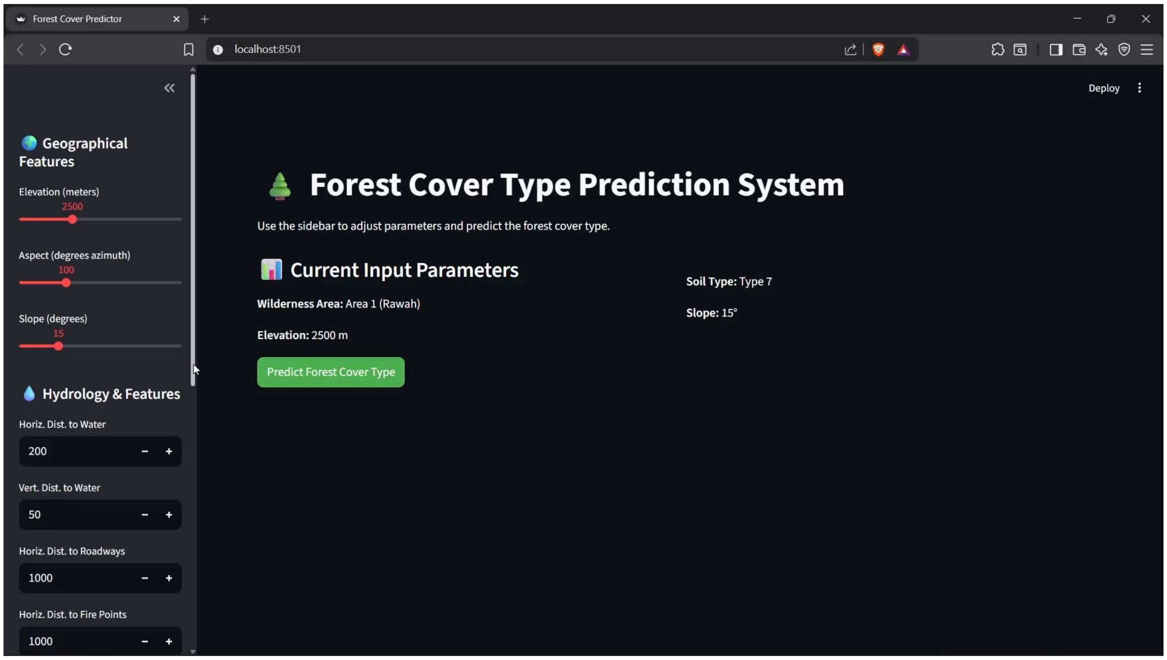
Evaluation Metrics

Multiple metrics were used to obtain balanced view of performance across all classes:

- **Accuracy**: Overall proportion of correctly classified instances
- **Precision (Weighted)**: Class-wise precision averaged and weighted by support
- **Recall (Weighted)**: Class-wise recall averaged and weighted by support
- **F1-Score (Weighted)**: Harmonic mean of precision and recall, weighted by class frequencies

Metrics were computed on validation set using scikit-learn's metrics module. Classification reports provided per-class performance; confusion matrices analyzed misclassification patterns.

Results and Discussion



Geographical Features

Elevation (meters): 2500

Aspect (degrees azimuth): 100

Slope (degrees): 15

Hydrology & Features

Horiz. Dist. to Water: 200

Vert. Dist. to Water: 50

Horiz. Dist. to Roadways: 1000

Horiz. Dist. to Fire Points: 1000

Forest Cover Type Prediction System

Use the sidebar to adjust parameters and predict the forest cover type.

Current Input Parameters

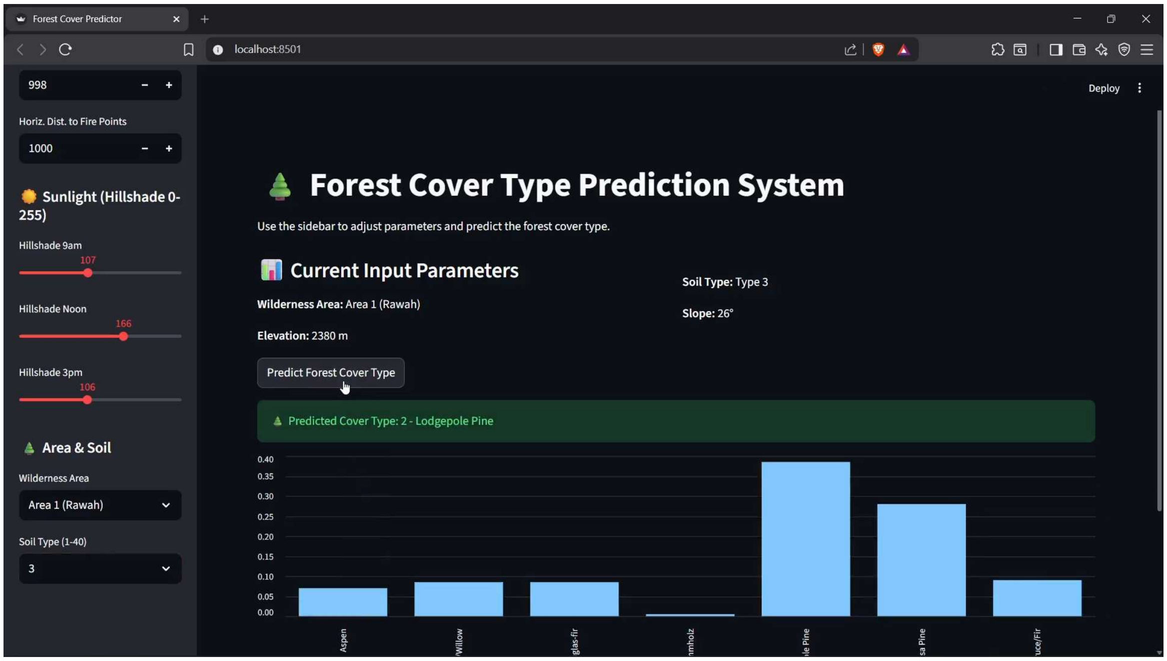
Wilderness Area: Area 1 (Rawah)

Elevation: 2500 m

Soil Type: Type 7

Slope: 15°

Predict Forest Cover Type



998

Horiz. Dist. to Fire Points: 1000

Sunlight (Hillshade 0-255)

Hillshade 9am: 107

Hillshade Noon: 166

Hillshade 3pm: 106

Area & Soil

Wilderness Area: Area 1 (Rawah)

Soil Type (1-40): 3

Forest Cover Type Prediction System

Use the sidebar to adjust parameters and predict the forest cover type.

Current Input Parameters

Wilderness Area: Area 1 (Rawah)

Elevation: 2380 m

Soil Type: Type 3

Slope: 26°

Predict Forest Cover Type

Predicted Cover Type: 2 - Lodgepole Pine

Quantitative Results

Benchmark results for all models on validation set:

- **Logistic Regression**: Accuracy ≈ 0.71
- **SVM**: Accuracy ≈ 0.80
- **KNN**: Accuracy ≈ 0.82

- **Decision Tree**: Accuracy ≈ 0.81
- **Random Forest**: Accuracy ≈ 0.90
- **ExtraTrees Classifier**: Accuracy ≈ 0.9024, precision ≈ 0.90, recall ≈ 0.90, F1-score ≈ 0.90
- **XGBoost**: Accuracy ≈ 0.88

ExtraTrees achieved best accuracy (approximately 90.2%) with strong balance between precision, recall, and F1-score across all seven classes. Random Forest closely trailed ExtraTrees.

Qualitative Analysis

From confusion matrices, most misclassifications occur between ecologically similar cover types (e.g., conifer-dominated classes), suggesting model learns meaningful patterns but reaches inherent limits of separability given feature set.

Feature engineering significantly contributed to performance. Composite variables such as combined hydrology distances and interactions between hydrology, fire points, and roadways provided additional separation power. The model benefits from full set of soil and wilderness indicators encoding important ecological context.

Limitations

Project has several limitations:

1. **Dataset Scope**: Model trained exclusively on Roosevelt National Forest data in northern Colorado; generalization to other regions not validated
2. **Static Data**: Dataset represents static snapshot without temporal dynamics, disturbances, or climate variability
3. **Feature Set Constraints**: Uses only provided cartographic variables; additional remote-sensing features or climate covariates might improve accuracy
4. **Class Imbalance**: Some cover types less frequent; no explicit rebalancing strategy applied
5. **Limited Hyperparameter Tuning**: More exhaustive search, especially for XGBoost and Random Forest, could yield further gains

Conclusion

This project delivers a complete, practical solution for predicting forest cover type in Roosevelt National Forest using supervised machine learning. Starting from rich tabular dataset of topographic, hydrologic, wilderness, and soil attributes, the workflow performs thorough EDA, domain-informed feature engineering, careful preprocessing, and rigorous model benchmarking. Ensemble tree methods, particularly ExtraTrees, achieve approximately 90% validation accuracy with strong balanced metrics across all cover types.

The final ExtraTrees model and scaler are serialized and integrated into Streamlit web application, enabling users to input geographical characteristics and receive real-time predictions. This demonstrates feasibility of deploying production-quality ML model for forest management tasks and offers foundation for future extensions.

Future work could explore domain adaptation to other regions, incorporate additional remote sensing layers, mitigate class imbalance, and perform targeted hyperparameter optimization.

References

- Forest Cover Type dataset description and internal project documentation (Roosevelt National Forest analysis data)
- Project notebook: Forest_Cover_Prediction.ipynb (exploratory data analysis, feature engineering, model training, benchmarking)
- Project README: "Forest Cover Type Prediction – Project Overview" (methodology and deployment details)
- Streamlit application source code: app.py (model serving and user interface)