DSAA2011 Machine Learning Project Report: Covertype Dataset

Binkai LIU	Chi Kit WONG	Zehan Lu
50012704	50012338	50012895

1 Introduction

The Covertype dataset contains 581,012 forest plot instances described by 54 features and categorized into 7 forest cover types. This project applies machine learning techniques to analyze, visualize, cluster, and classify these cover types using a systematic pipeline aligned with the course requirements. Key preprocessing steps include standardizing continuous features, preserving one-hot encoded soil and wilderness indicators, and applying both 2D and 3D t-SNE visualizations to examine the intrinsic structure of the high-dimensional feature space.

For unsupervised learning, we implemented three clustering approaches: **K-Means**, **Gaussian Mixture Models (GMM)**, and a **RandomTreesEmbedding + MiniBatchKMeans** pipeline. These methods allow us to evaluate both internal cluster structure and external alignment with the true cover-type labels. Their performance differences highlight the challenges of separating overlapping forest cover types based solely on unsupervised techniques.

For supervised learning, we employed three models capturing complementary inductive biases: **Logistic Regression** (linear baseline), **Random Forest** (nonlinear tree ensemble), and **Extra Trees** (a more randomized ensemble variant). Each model was evaluated on the training, testing, and full datasets using accuracy, precision, recall, F1-score, multiclass confusion matrices, and one-vs-rest ROC curves with macro-averaged AUC. Decision boundary visualizations were additionally generated via PCA projections to provide interpretable insights into each model's behavior.

In the open-ended exploration, we investigated methods for model improvement such as hyperparameter tuning of Random Forest, statistical feature selection, and extended model comparisons involving additional ensemble and boosting-based approaches. These explorations deepen our understanding of model robustness, dataset complexity, and the trade-offs between linear and nonlinear methods.

Through this end-to-end workflow, we obtained competitive predictive performance and gained meaningful insights into the structural characteristics of the Covertype dataset.

2 Introduction

The Covertype dataset contains 581,012 forest plot instances with 54 features and 7 cover type classes. This project applies machine learning techniques to analyze and classify forest cover types through data preprocessing, visualization, clustering, and supervised learning. Key steps include standardizing continuous features, preserving existing one-hot encoded wilderness and soil indicators, and applying t-SNE for 2D and 3D visualization to explore the intrinsic structure of the high-dimensional data. K-Means and Gaussian Mixture Models (GMM) were employed for clustering, enabling us to evaluate both internal and external clustering performance.

For supervised learning, Logistic Regression and Random Forest classifiers were implemented, and their performance was assessed across the training set, testing set, and the entire dataset. Evaluation metrics include accuracy, precision, recall, F1-score, confusion matrices, ROC curves, and

macro-averaged AUC values. Decision boundary visualizations were further generated to provide interpretable insights into model behavior under selected feature pairs.

In the open-ended exploration, we examined multiple techniques for model improvement, including hyperparameter tuning of Random Forest, feature selection using statistical scores, and extended model comparison involving Gradient Boosting, ExtraTrees, Histogram Gradient Boosting, and linear models. These experiments provide a broader perspective on model robustness and scalability for large-scale, mixed-type datasets such as Covertype. Through this comprehensive workflow, we achieved competitive classification performance and gained deeper understanding of both the dataset characteristics and the models' behavior.

3 Mandatory Tasks

3.1 Data Source and Preprocessing

The dataset employed in this study is the **Covertype Dataset**[1] obtained from the UCI Machine Learning Repository, which contains 581,012 forest plot instances across seven forest cover type classes. Each instance represents a geographical location in the Roosevelt National Forest, described by 54 numerical features, including 10 continuous topographic measurements such as elevation, slope, aspect, and various horizontal and vertical distances, as well as 44 binary indicators representing wilderness areas and soil types. The target variable *Cover_Type* consists of seven categories corresponding to tree species such as Spruce/Fir, Lodgepole Pine, Ponderosa Pine, and others.

According to the dataset documentation, no missing values were reported. This was further verified through our own preprocessing pipeline. Since all features in the Covertype dataset are already numeric—with wilderness areas and soil types encoded as one-hot indicator variables—no additional categorical encoding was required.

To ensure consistent feature scaling, all continuous numerical features were standardized to have zero mean and unit variance, while the existing binary features were preserved as-is. This standardization step is essential for distance-based algorithms such as t-SNE and K-Means, and it additionally stabilizes the optimization process for models such as Logistic Regression.

The dataset exhibits noticeable class imbalance, with certain cover types being significantly more frequent than others. To maintain representativeness, we applied stratified sampling when splitting the dataset into training and testing sets, preserving the original label distribution across subsets. This preprocessing pipeline establishes a reliable foundation for downstream visualization, clustering, and supervised learning tasks.

3.2 t-SNE Visualization

To explore the intrinsic structure of the Covertype dataset and assess the separability among the seven forest cover types, we applied t-distributed Stochastic Neighbor Embedding (t-SNE) for both 2D and 3D visualizations. As a nonlinear dimensionality reduction technique, t-SNE is particularly effective for revealing the local neighborhood structure of high-dimensional data, making it suitable for interpreting the mixed continuous and one-hot encoded features present in this dataset.

Given the large scale of the dataset, a random subset of 5,000 samples from the training set was selected to ensure computational efficiency while retaining representative class patterns. Prior to applying t-SNE, we applied Principal Component Analysis (PCA) to reduce the dimensionality to 50 components, which helps stabilize t-SNE and mitigate noise in the high-dimensional feature space.

The objective of this step was to obtain early insight into how distinct or overlapping the forest cover types are within the embedded space. The resulting 2D and 3D t-SNE projections (Figure 1) reveal substantial overlap among several cover types. In particular, Cover Type 2 (Lodgepole Pine) and Cover Type 3 (Ponderosa Pine) exhibit significant intermixing, suggesting these classes may be challenging to distinguish during classification. Conversely, Cover Type 1 (Spruce/Fir) forms a relatively more compact and separated region in the t-SNE space, indicating stronger intrinsic separability. These visual patterns provide useful preliminary guidance for both clustering analysis and supervised learning.

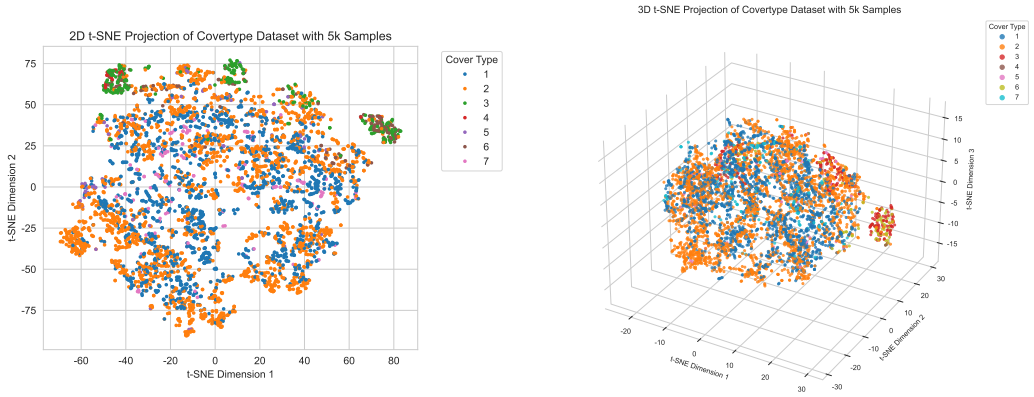


Figure 1: *2D and 3D t-SNE Projection of the Covertype Dataset with 5k Samples*

3.3 Clustering Analysis

To analyze latent group structures within the Covertype dataset without relying on cover-type labels, we applied three unsupervised clustering algorithms: **K-Means**, **Gaussian Mixture Model (GMM)**, and a tree-based representation method using **RandomTreesEmbedding followed by MiniBatchKMeans** (denoted as ExtraTrees+MBK). All methods were configured with $k = 7$ clusters to match the number of true cover type classes. A random subset of 15,000 samples from the training set was used to ensure computational feasibility while retaining the dataset's structural characteristics.

Clustering performance was evaluated using both external metrics—Adjusted Rand Index (ARI), Normalized Mutual Information (NMI), and Fowlkes–Mallows Index (FMI)—and internal metrics including Silhouette Score, Calinski–Harabasz Index, and Davies–Bouldin Index. The results are summarized in Table 1.

Method	ARI	NMI	FMI	Silhouette	Calinski–Harabasz	Davies–Bouldin
K-Means	0.022250	0.065013	0.272895	0.133091	2010.778121	1.790876
GMM	0.121351	0.202422	0.391169	0.017213	570.649249	3.388818
ExtraTrees+MBK	0.044197	0.129558	0.296775	0.036020	823.148257	3.199565

Table 1: *Internal and External Metrics for Three Clustering Methods*

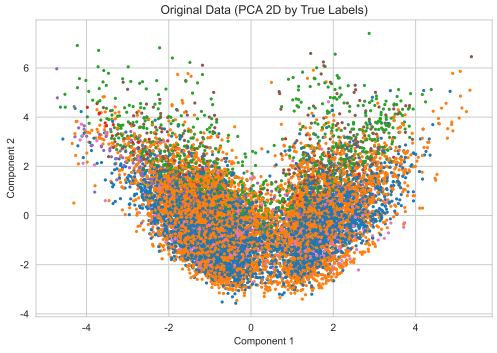
Figure 2 displays the true labels projected into 2D PCA space, followed by the clustering assignments from all three algorithms.

Based on Table 1, the Gaussian Mixture Model (GMM) achieves the highest scores across all external metrics (ARI, NMI, FMI), indicating that its clustering assignments align most closely with the true class labels. In contrast, K-Means obtains the strongest internal metrics (Silhouette and Calinski–Harabasz), suggesting that it forms more compact and well-separated clusters in geometric terms.

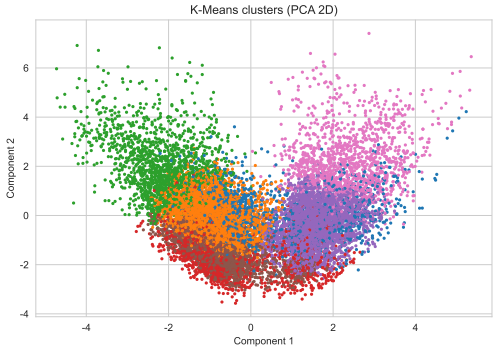
The ExtraTrees+MBK method performs in between the other two algorithms: its external metrics surpass K-Means but fall short of GMM, while its internal metrics are weaker than K-Means but stronger than GMM. This indicates that the tree-based embedding introduces useful nonlinear structure but also increases cluster overlap.

Despite these differences, all three methods struggle to distinguish several cover types. The PCA visualizations (Figure 2) reveal substantial overlap, particularly among Cover Types 2, 3, and 7. These patterns reflect the intrinsic difficulty of the dataset, where many classes share similar elevation profiles and soil-type vectors, leading to weak unsupervised separability.

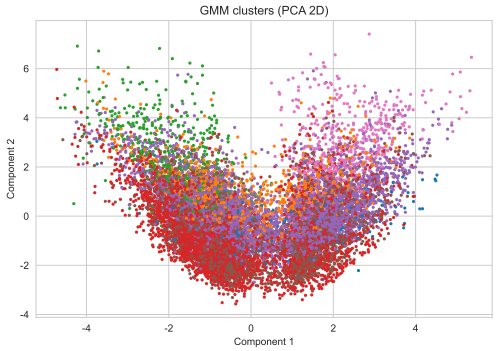
Overall, GMM provides the best label-recovery performance, while K-Means yields the most compact clusters. However, none of the methods achieve strong separability, suggesting that supervised learning is necessary to capture the complex interactions within the Covertype features.



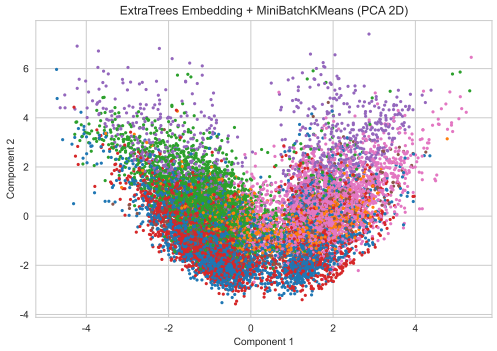
(a) True Labels in PCA 2D



(b) K-Means Clusters



(c) GMM Clusters



(d) ExtraTrees Embedding + MiniBatchKMeans Clusters

Figure 2: *Clustering Assignments Projected onto PCA 2D*

3.4 Prediction Models Analysis

3.4.1 Model Selection

The target variable in the Covertype dataset is the forest cover type, consisting of 7 classes representing different dominant tree species. For the supervised prediction task, we selected three model classes that provide complementary perspectives on the data: **Logistic Regression**, **Random Forest**, and **Extra Trees**. These choices align with the project requirement of employing multiple distinct model types while also reflecting the characteristics of the dataset.

Logistic Regression serves as a strong linear baseline for multiclass classification, offering interpretability and efficient training on large-scale datasets. Random Forest and Extra Trees, on the other hand, are nonlinear ensemble methods based on decision trees. Both are capable of capturing complex feature interactions, making them well-suited for high-dimensional datasets that include both continuous terrain measurements and numerous one-hot encoded soil indicators. Extra Trees introduces stronger randomization at split selection, which can further reduce variance and provide a useful contrast to the Random Forest model.

Table 2 summarizes the models chosen for this task.

Model	Description
Logistic Regression	Linear baseline for multiclass classification
Random Forest	Nonlinear ensemble model capturing complex feature interactions
Extra Trees	Extremely randomized tree ensemble with higher split randomness

Table 2: *Prediction Models Used for Covertype Classification*

3.4.2 Training Procedure

The processed Covertype dataset contains 581,012 samples with 54 features, including 10 continuous topographic attributes and 44 one-hot encoded wilderness and soil indicators. After preprocessing and standardizing the continuous features, the dataset was divided into training and testing subsets using a 70%–30% stratified split ($X_{\text{train}}, y_{\text{train}}$ and $X_{\text{test}}, y_{\text{test}}$). Stratified sampling was applied to preserve the original class distribution across all seven cover types.

All three models were initialized with configurations tailored to the dataset. Logistic Regression was trained with multinomial loss, the LBFGS optimizer, and class-balancing enabled to address label imbalance. Random Forest was trained as a nonlinear ensemble model with 200 trees and class-weight adjustment for robust performance across majority and minority classes. Extra Trees was configured with 250 extremely randomized trees, also with class-weight balancing, to further explore the effect of increased split randomness on performance and stability. All models were trained on the standardized training data.

A comprehensive set of evaluation metrics was used to assess performance. For the training set, testing set, and the combined full dataset, accuracy, precision, recall, and F1-scores were computed. To further examine model behavior across individual classes, confusion matrices were generated for each model. In addition, Receiver Operating Characteristic (ROC) curves were plotted under a one-vs-rest scheme, and the macro-averaged Area Under the Curve (AUC) was calculated to provide a holistic view of multiclass classification performance.

3.4.3 Logistic Regression

Logistic Regression was selected for the Covertype classification task as a strong linear baseline. Given the large scale of the dataset (581,012 samples and 54 features), its computational efficiency and well-understood optimization behavior make it suitable for establishing a reference performance level. The model was trained using multinomial logistic regression with the LBFGS optimizer, along with L2 regularization and class-weight balancing to mitigate the class imbalance inherent in the dataset.

As shown in Figure 4, we applied PCA to reduce the feature space to two dimensions and plotted the decision boundaries to gain interpretability. Figure 3 summarizes the overall performance metrics across the training, testing, and full datasets, while Figure 6 presents the corresponding confusion matrices. The one-vs-rest ROC curves are shown in Figure 5, from which the macro-averaged AUC score was calculated.

The Logistic Regression model demonstrates stable performance across all dataset splits, with balanced values of accuracy, precision, recall, and F1-score. This consistency suggests that the use of L2 regularization and class-weight adjustment effectively mitigated overfitting and improved generalization. However, as expected from a linear classifier, its performance is limited in capturing the nonlinear interactions present in the Covertype dataset, which includes complex combinations of elevation, slope, and soil-type features.

The ROC curves reveal relatively high AUC values across all seven classes, indicating good ranking performance even in cases where decision boundaries may not fully capture nonlinear relationships. The confusion matrices further illustrate the model's weaknesses: substantial misclassification occurs between Cover Types 2 and 3 (Lodgepole Pine vs. Ponderosa Pine), as well as between Types 6 and 7, reflecting the overlapping structure observed earlier in the t-SNE projection. These confusions highlight intrinsic challenges of the dataset, rather than solely model limitations.

Dataset	Accuracy	Precision	Recall	Score
Training	0.599	0.704	0.599	0.628
Testing	0.600	0.704	0.600	0.629
Overall	0.599	0.704	0.599	0.628

Figure 3: *Overall Performance Metrics*

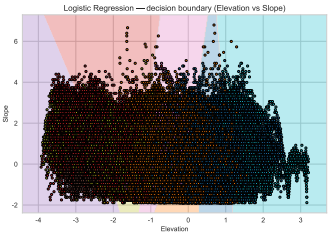


Figure 4: *Decision Boundary (PCA 2D)*

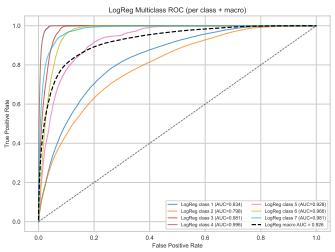


Figure 5: *ROC Curves by Class*

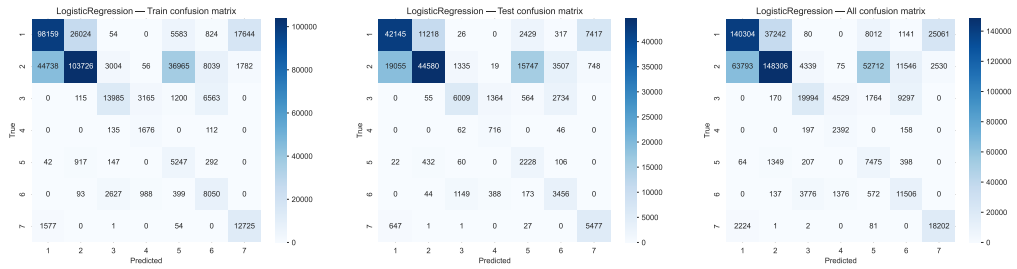


Figure 6: *Confusion Matrices of Logistic Regression*

3.4.4 Random Forest

Random Forest was selected as a nonlinear ensemble model capable of capturing complex feature interactions within the Covertype dataset. Unlike Logistic Regression, which relies on linear decision boundaries, Random Forest aggregates predictions from multiple decision trees trained on bootstrapped samples with randomized feature subsets, allowing it to learn highly flexible and nonlinear patterns. This makes the model particularly suitable for the mixture of continuous terrain variables and high-dimensional one-hot encoded soil-type indicators present in this dataset.

The model was trained using 200 estimators with class-weight balancing enabled to address the skewed distribution across the seven cover types. This setup provides a robust compromise between model complexity and computational efficiency on the large-scale dataset consisting of over half a million samples.

Figure 7 summarizes the overall performance metrics on the training, testing, and full datasets. The decision boundary in a reduced PCA space is shown in Figure 8, illustrating the model's ability to form nonlinear separation regions when projected into two dimensions. The one-vs-rest ROC curves in Figure 9 further demonstrate its strong ranking performance across all classes. Finally, Figure 10 provides the confusion matrices for all three dataset splits.

Compared with Logistic Regression, the Random Forest model achieves notably higher accuracy and F1-scores, particularly on the testing set, demonstrating better generalization to unseen data. The model shows strong robustness across all classes, though misclassifications remain frequent between Cover Types 2 and 3, as well as Types 6 and 7—patterns consistent with the earlier t-SNE visualization and clustering analysis. These error patterns suggest that some forest cover types share intrinsic feature similarities that challenge both linear and nonlinear classifiers.

Overall, Random Forest provides one of the best performances among the models implemented in the mandatory tasks, offering a strong balance between expressive power and generalization.

Dataset	Accuracy	Precision	Recall	Score
Training	1.000	1.000	1.000	1.000
Testing	0.952	0.952	0.952	0.952
Overall	0.986	0.986	0.986	0.986

Figure 7: *Overall Performance Metrics*

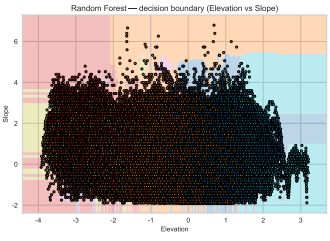


Figure 8: *Decision Boundary (PCA 2D)*

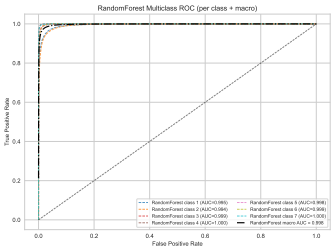


Figure 9: *ROC Curves by Class*

3.4.5 Extra Trees

The Extra Trees (Extremely Randomized Trees) classifier extends the idea of Random Forest by introducing stronger randomness at each split. Instead of searching for the best split threshold on a selected feature, Extra Trees samples thresholds at random and chooses among them, which typically reduces variance and can improve robustness on noisy or highly redundant features. This

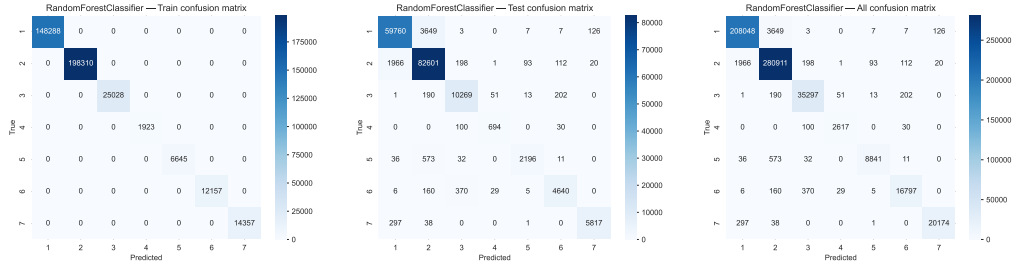


Figure 10: *Confusion Matrices of Random Forest*

characteristic makes it a natural candidate for comparison with Random Forest on the Covertype dataset.

In our experiments, the Extra Trees model was configured with 250 trees, unrestricted depth, and class-weight balancing. It achieved a testing accuracy of 0.9509, which is only slightly lower than Random Forest in terms of accuracy and weighted F1-score, but marginally higher in macro precision, recall, and F1-score. This suggests that Extra Trees may handle minority classes slightly more evenly, at the cost of a negligible drop in overall accuracy. The confusion matrices exhibit patterns very similar to those of Random Forest, with strong diagonal dominance and only a small number of residual confusions among the known difficult class pairs.

Figure 14 shows the confusion matrices for Extra Trees on the training, testing, and full datasets, illustrating its consistently strong per-class performance.

Dataset	Accuracy	Precision	Recall	Score
Training	1.000	1.000	1.000	1.000
Testing	0.951	0.951	0.951	0.951
Overall	0.985	0.985	0.985	0.985

Figure 11: *Overall Performance Metrics*

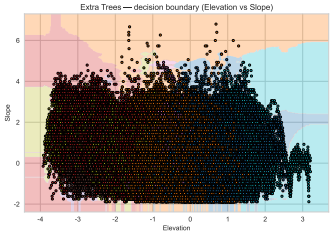


Figure 12: *Decision Boundary (PCA 2D)*

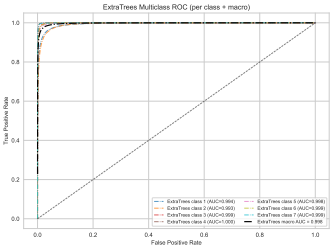


Figure 13: *ROC Curves by Class*

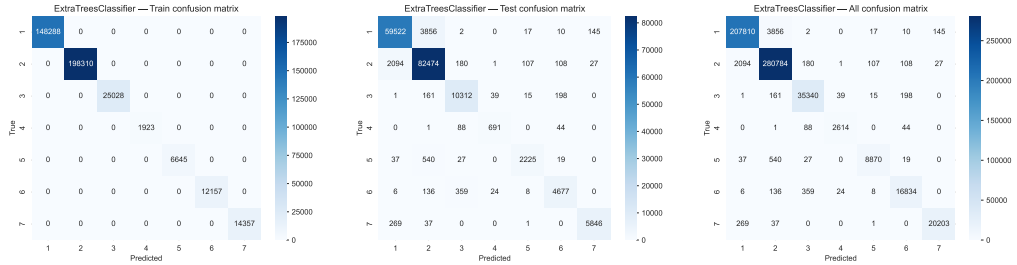


Figure 14: *Confusion Matrices of Extra Trees*

3.4.6 Model Comparison

Table 3 summarizes the test-set evaluation metrics for the three models. Random Forest and Extra Trees clearly outperform Logistic Regression across all metrics, demonstrating the importance of nonlinear modeling for the Covertype dataset. Between the two ensemble models, differences are small: Random Forest yields slightly higher accuracy and weighted F1-score, while Extra Trees achieves marginally better macro-level scores, indicating a slightly more balanced treatment of majority and minority classes.

Model	Accuracy	P_weighted	R_weighted	F1_weighted	P_macro	R_macro	F1_macro
Logistic Regression	0.600021	0.704523	0.600021	0.628772	0.471699	0.707670	0.506241
Random Forest	0.952858	0.952998	0.952858	0.952627	0.941822	0.902901	0.920822
Extra Trees	0.950908	0.951037	0.950908	0.950682	0.943448	0.904882	0.922699

Table 3: *Comparison of Prediction Model Performance on the Test Set*

Overall, the ensemble methods provide strong and reliable performance for the classification task, with Random Forest and Extra Trees both achieving high accuracy and balanced class-wise metrics. Logistic Regression remains a useful baseline for interpretability and efficiency but is insufficient to fully capture the complex, nonlinear feature relationships in the Covertype dataset.

4 Open-ended Exploration

Beyond the mandatory workflow, we extended our analysis through feature engineering, model improvement, hyperparameter tuning, and multi-model comparison. These experiments provide additional insights into how different modeling choices affect performance on the Covertype dataset.

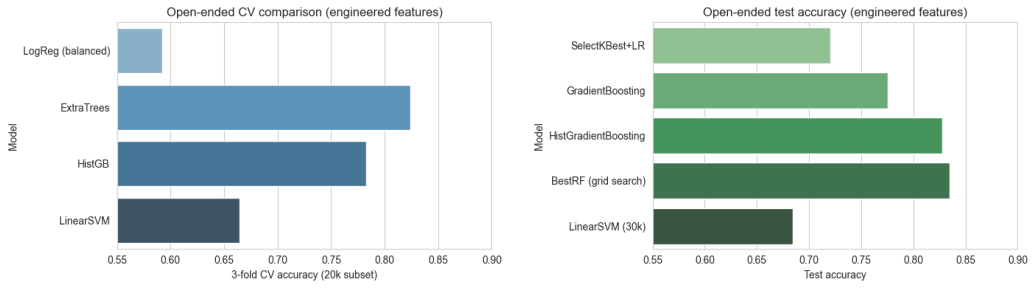
4.1 Feature Engineering and Logistic Regression

We introduced several engineered features inspired by domain characteristics, including total hydrology distance, a water-above indicator, a combined hillshade feature, and soil-derived climatic and geological zone codes. After scaling continuous attributes and applying SelectKBest with $k = 30$, Logistic Regression achieved a test accuracy of **0.7206**, showing that meaningful structure is retained even after aggressive dimensionality reduction.

4.2 Model Improvement with Boosting Methods

To evaluate the effect of increased model expressiveness, we applied two boosting-based models on the engineered features. Gradient Boosting achieved a test accuracy of **0.7758**, while the histogram-based HistGradientBoosting classifier reached **0.8277**. This improvement demonstrates the effectiveness of nonlinear boosting methods and their suitability for mixed-type, high-dimensional data.

Figure 15 summarizes 3-fold CV accuracy and test-set accuracy for the primary models evaluated during the exploration.



(a) 3-fold CV accuracy on 20k subset

(b) Test accuracy on full test set

Figure 15: *Cross-validation and test accuracy for models in the open-ended exploration.*

4.3 Random Forest Hyperparameter Tuning

To further improve ensemble performance, we performed a lightweight grid search over $n\_estimators$ and max_depth using a 20,000-sample training subset. The best configuration—**200 trees**, unlimited depth, and **sqrt** feature sampling—achieved a cross-validation accuracy of **0.8202** and a test accuracy of **0.8398**.

A broader parameter sweep on a 15,000-sample subset explored the interaction between tree depth and the number of estimators, with the resulting CV accuracies shown in Figure 16. Deeper trees

and moderately large ensembles yield the strongest results, with diminishing returns beyond 200 estimators.

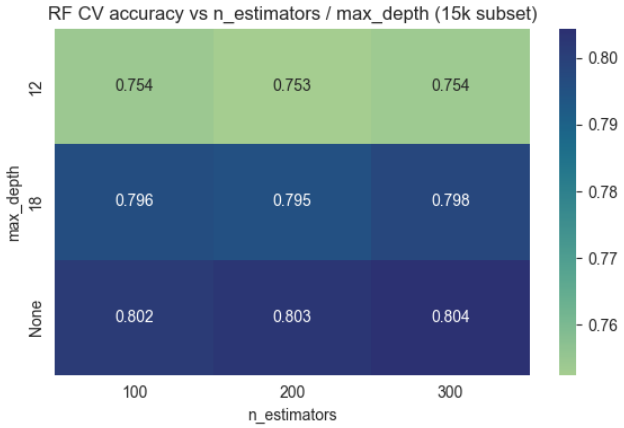


Figure 16: *Random Forest CV accuracy vs. number of trees and maximum depth (15k subset).*

In addition, feature importance analysis on the tuned Random Forest highlights the dominant influence of Elevation, hydrology-related distances, and hillshade variables, along with the engineered climatic and geological zone codes.

4.4 Model Comparison Across Classifiers

We conducted a unified 3-fold cross-validation comparison across four model families—balanced Logistic Regression, Extra Trees, HistGradientBoosting, and Linear SVM—using a 20,000-sample subset of the engineered features. The mean CV accuracies were:

- Extra Trees: **0.8240** $\pm$ 0.0039
- HistGradientBoosting: **0.7585** $\pm$ 0.0008
- Linear SVM: **0.6768** $\pm$ 0.0076
- Logistic Regression (balanced): **0.5983** $\pm$ 0.0024

Extra Trees achieved the strongest cross-validation performance, closely followed by the tuned Random Forest. A Linear SVM trained on a 30,000-sample subset reached a test accuracy of only **0.6777**, reinforcing the limitations of purely linear models on this dataset.

For HistGradientBoosting, we also tested different combinations of learning rates and depths. Lower learning rates (0.05–0.1) combined with moderate depths (8–12) performed best. Because the overall pattern is monotonic, we omit the heatmap figure for brevity and report the qualitative trend instead.

4.5 Summary of Insights

Across all experiments, a consistent trend emerges: performance improves as model complexity increases from Logistic Regression to Gradient Boosting, Random Forest, and Extra Trees. The engineered features proved useful in enhancing both linear and nonlinear models, while targeted hyperparameter tuning delivered further gains for tree ensembles. These results satisfy the open-ended exploration objectives and highlight the importance of expressiveness, feature design, and tuning when modeling large-scale environmental datasets.

5 Conclusion

In this project, we explored a complete machine learning workflow using the Covertype dataset. We applied data preprocessing, visualization, clustering, and supervised classification to analyze and predict forest cover types. The clustering results from K-Means, GMM, and ExtraTrees-based embedding highlighted the inherent overlap among classes, while supervised models—Logistic

Regression, Random Forest, and Extra Trees—demonstrated clear differences in predictive capability, with tree-based ensembles achieving the strongest performance.

Through open-ended exploration, we further improved the models using feature engineering, hyperparameter tuning, and extended comparisons with additional classifiers. These experiments provided deeper insight into model behavior on large-scale environmental data and reinforced the importance of expressive models and well-designed features.

Overall, the project offered a comprehensive hands-on understanding of the machine learning process and demonstrated how different modeling choices influence performance on a complex, mixed-type dataset.

References

[1] Jock Blackard. Covertype. UCI Machine Learning Repository, 1998. DOI: https://doi.org/10.24432/C50K5N.

AI Declaration

1. Receiving and revising suggestions for improving the report's writing quality with assistance from ChatGPT 5.1 by OpenAI.
2. Using ChatGPT 5.1 by OpenAI to explain functions for generating visualizations and the important parameters involved.
3. Explaining some functions for creating visualizations and their key parameters with ChatGPT5.1 by OpenAI.
4. Enhancing code readability with refinement support from ChatGPT 5.1 by OpenAI.

Credit

- **Binkai LIU**:
- **Chi Kit WONG**: Report writing, adding visualizations to the code, and delivering the presentation on the data preprocessing component.
- **Zehan Lu**:
- **GenAI (ChatGPT5.1)**: Suggestions for improving the report's presentation, explanations of the key functions used for generating visualizations, clarifications on important Scikit-learn functions, guidance on inserting figures in LaTeX, recommendations on problem-solving strategies, and refinements to improve code readability.

We affirm that all team members made equal contributions and worked collaboratively throughout every phase of this project to ensure its successful completion. The use of AI tools remained below 30% and was limited strictly to supportive tasks. All AI-assisted content was reviewed, revised, and refined before being incorporated into the project.