**Technical Report: Land Cover Classification**

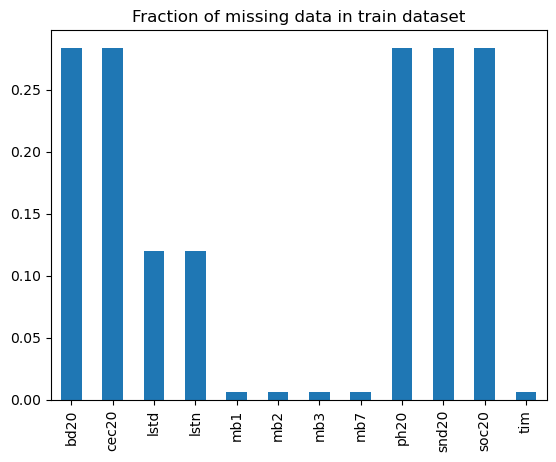
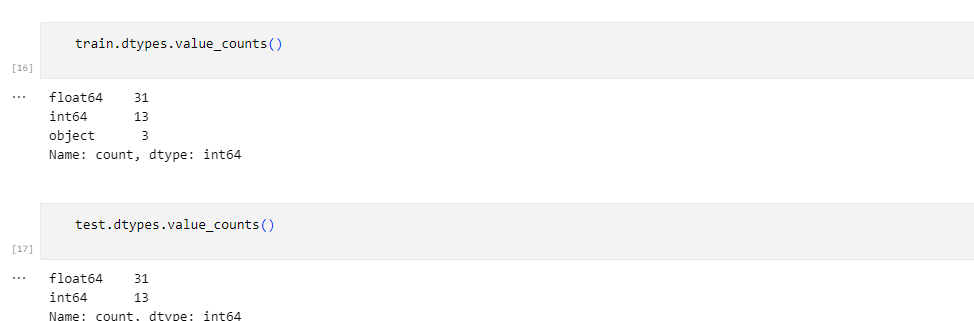
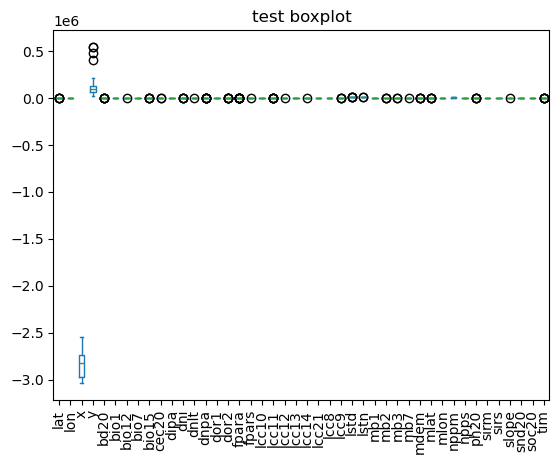
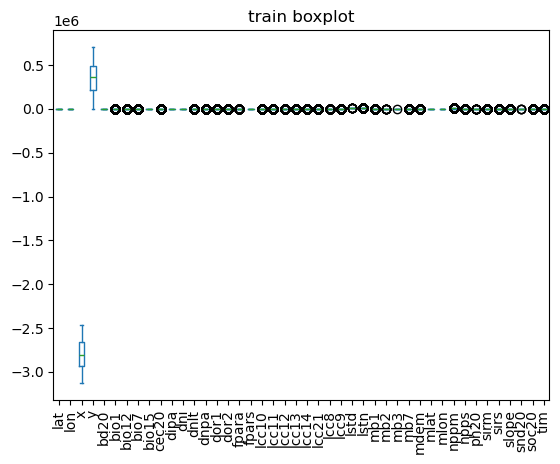
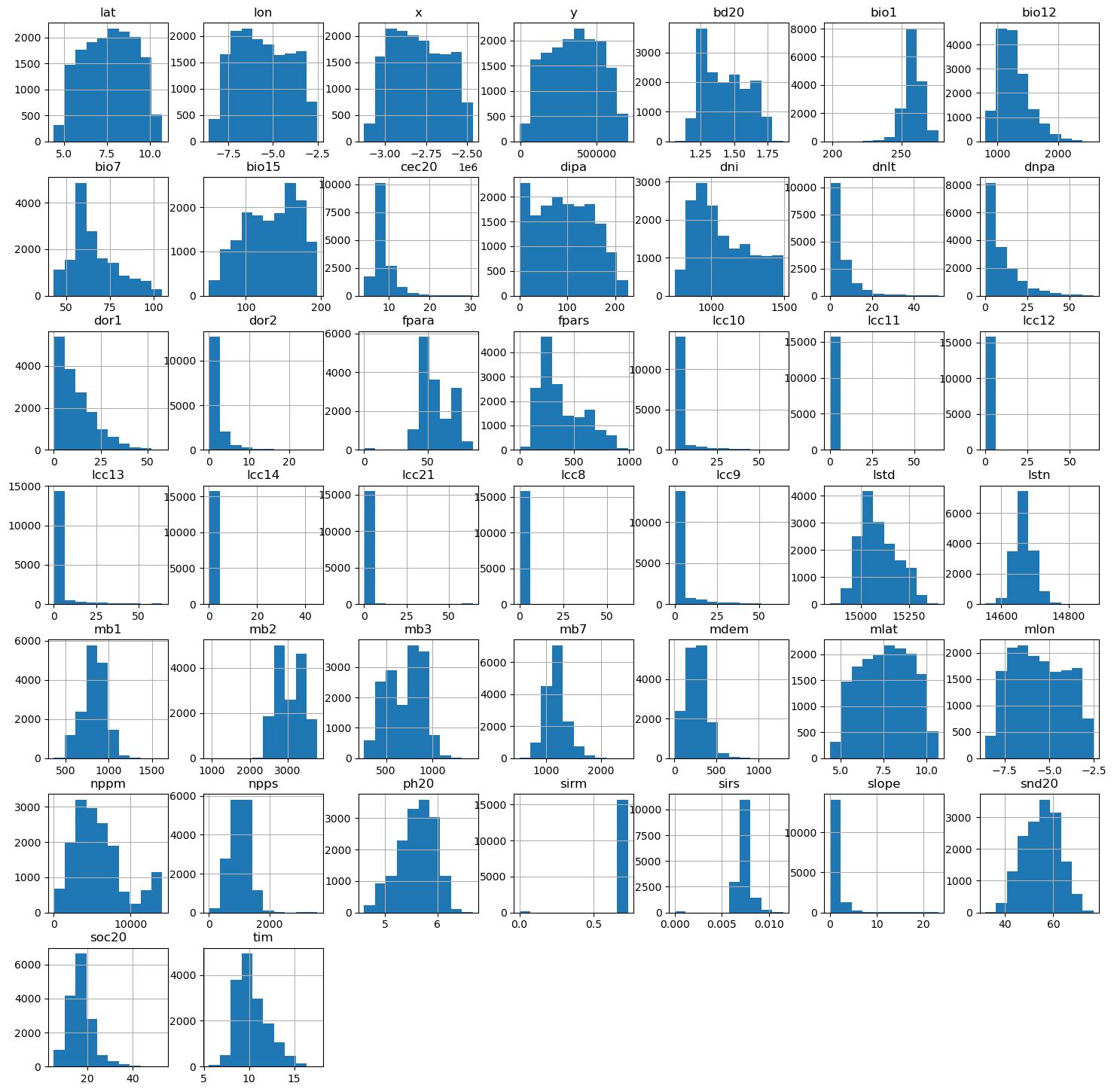
**1. Introduction**

The goal of this project is to build a robust predictive model to classify land cover into three categories: Buildings, Cropland, and Woody vegetation cover (>60%). The model should output occurrence probabilities for each class. This report outlines the detailed methodology, approach, critical findings, and recommendations.

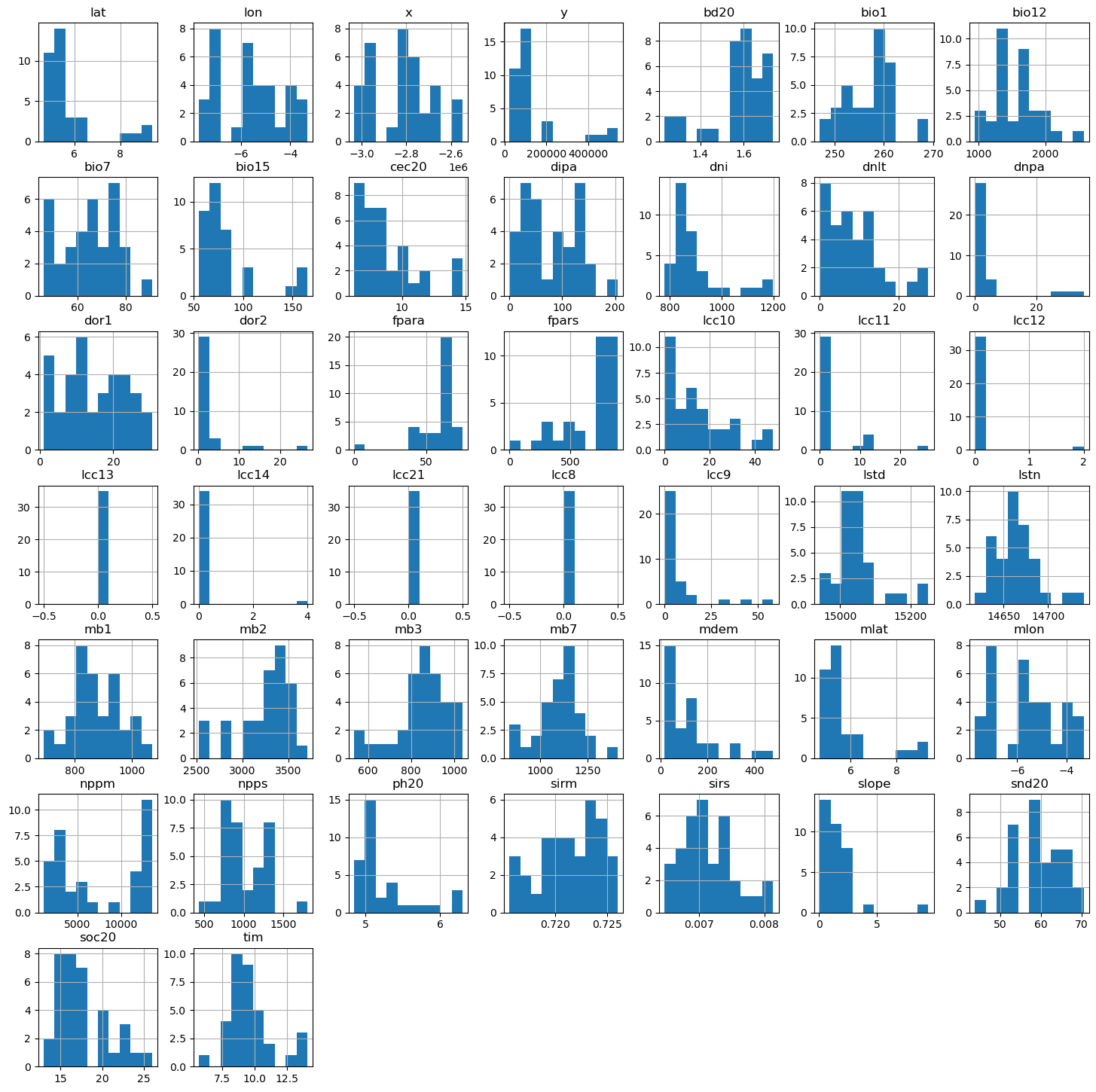
**2. Methodology and Approach:**

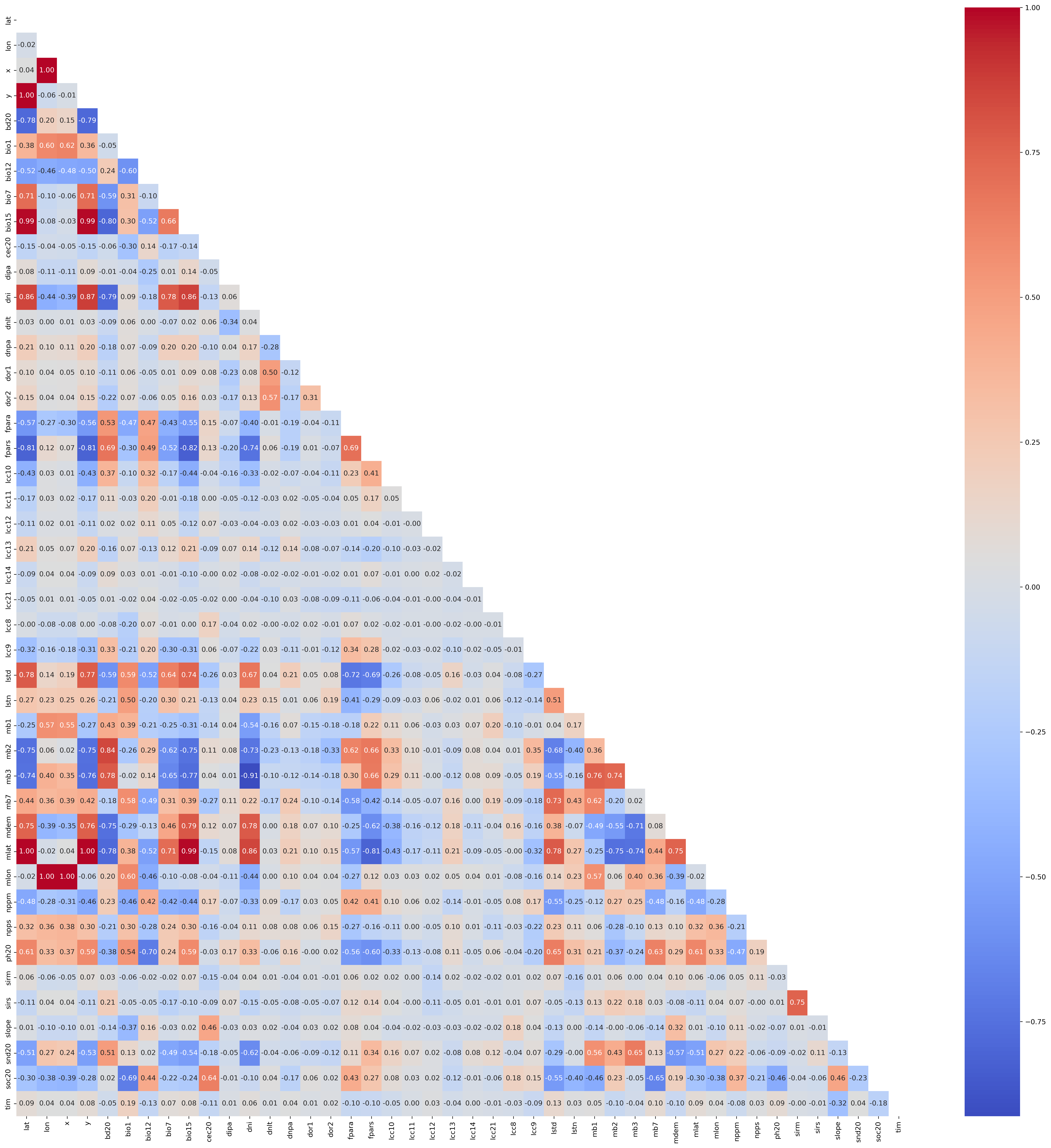
This section outlines the steps followed to complete this technical assessment.

**2.1 Exploratory Data Analysis (EDA)**

* **Data Loading**: The datasets were loaded. The shapes of the datasets were checked to understand the number of samples and features. I
* **Missing Data Analysis**: The percentage of missing data in both the train and test sets was calculated and visualized. The “bcount” column was dropped from both the train and test set as it was completely empty in the test set. It was also noted that columns with the same fraction of missing data came from the same source, such mb1, mb2, mb3 and mb7. This could hint as errors with devices used to collate data. a
  + The **data types** of the features were checked, and the percentage occurrence of each target variable (Buildings, Cropland, Woody vegetation) in the training set was analyzed. This resulted in finding out that the dataset is imbalanced for all target classes.
* **Outlier Detection and Summary Statistics**: Outliers were visualized, and summary statistics for numerical features were generated using the describe() function. train histogram

**Test histogram**

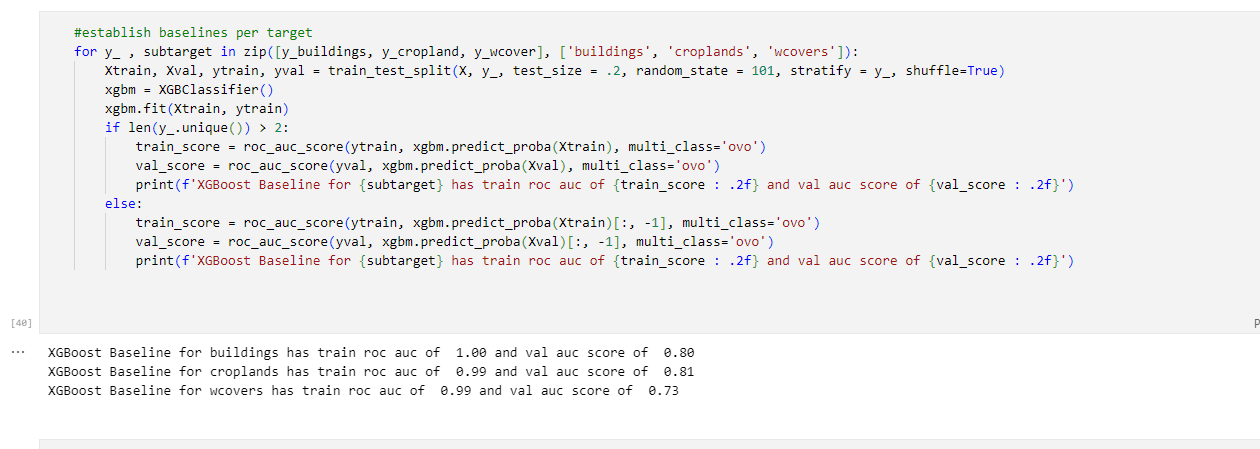


* Correlated Features: Perfectly correlated features (mlong, mlat, x, y) were identified and dropped as they were redundant with Lon and Lat. 

2.2 **Feature Engineering**

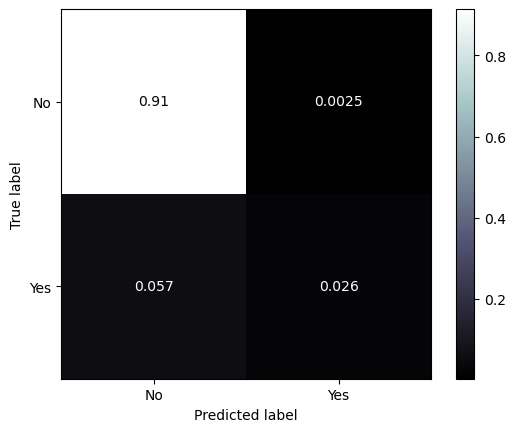
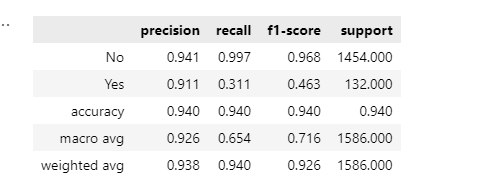
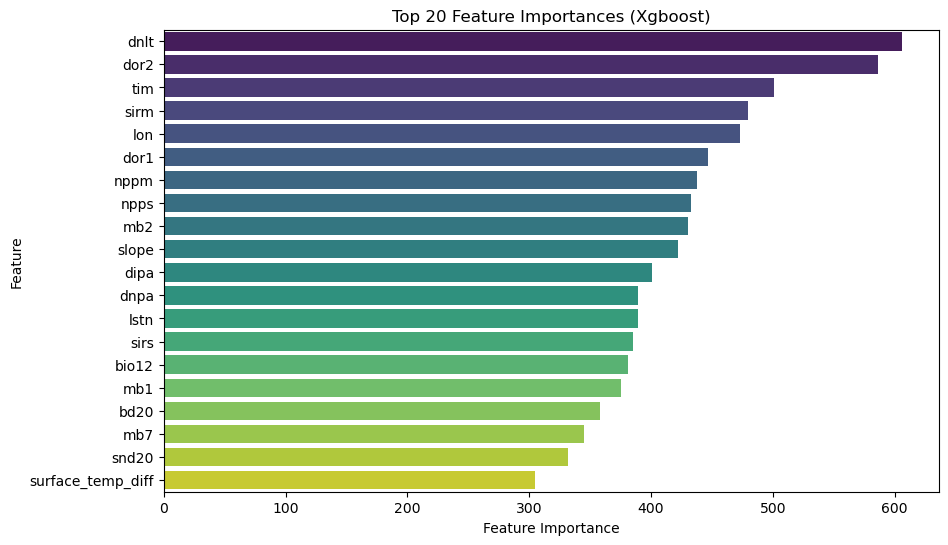
* **Handling Missing Data**: Missing values in the training set were filled with 0 to ensure no data loss and to maintain consistency across the dataset.
* **Splitting Data**: The data was split into features (X) and target (Y).
* **Adding New Features**: Added surface temperature difference column which is the difference between temperature during the day and temperature at night.

**2.3 Modeling**

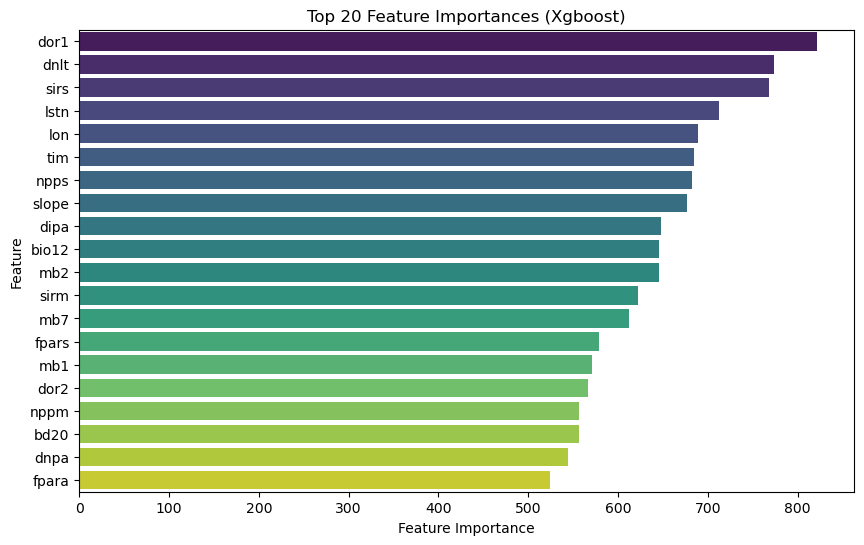
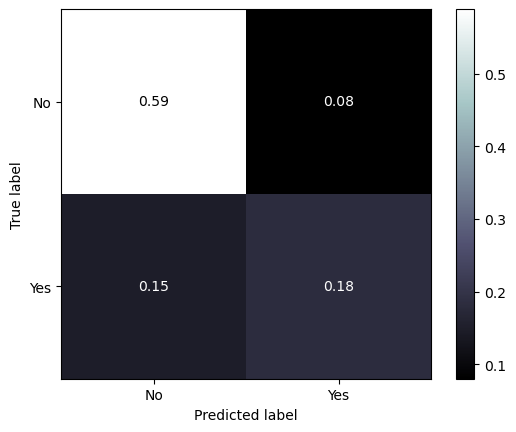
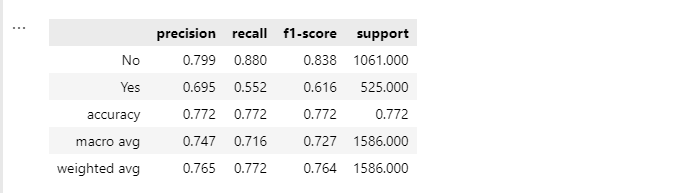
* **Multi-Classification Approach**: A multi-classification approach was adopted by training separate models for each target variable to optimize performance for each class, using roc-auc as the evaluation metric.
* **Baseline Model**: **XGBoost** was used as the baseline model for all target classes
* **Hyperparameter Tuning**: The training data was split into train, validation, and test subsets. The train and validation sets were used for hyperparameter tuning, while the test set was reserved for evaluating the tuned models. Optuna was used for hyperparameter tuning, with the objective of maximizing roc-auc score. The idea, was to perform five fold cross validation s on the smaller train subset, and validate on the created validation set, and evaluate models ion the test subset and measure metrics like roc-auc, classification report(precision, recall, accuracy, etc) and also view the confusion matrix for the task. The best model for each task is retrained on the entire dataset to get the feature importance and make predictions on the actual test set.
* **Model** **Evaluation**: The best model from hyperparameter tuning was evaluated on the test set using metrics such as accuracy, precision, recall, and F1-score. A confusion matrix was generated to understand the model's performance across different classes.
* **Final Model Training**: The best model was retrained on the entire training dataset (train + validation + test) to ensure it learned from all available data.
* **Feature Importance**: The feature importances for the best model were visualized to understand which features contributed most to the predictions.

3. **Critical Findings**

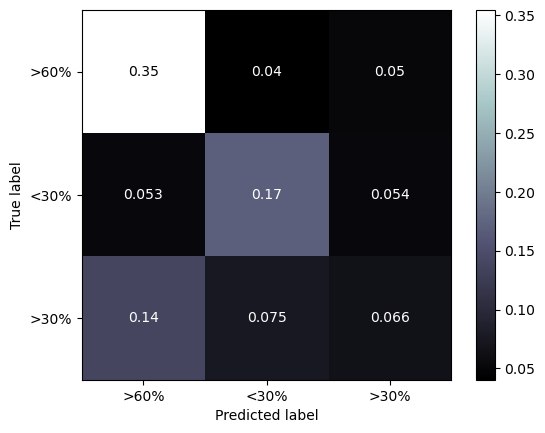
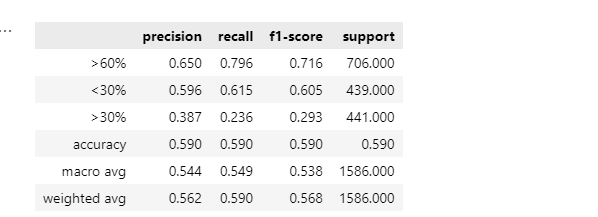
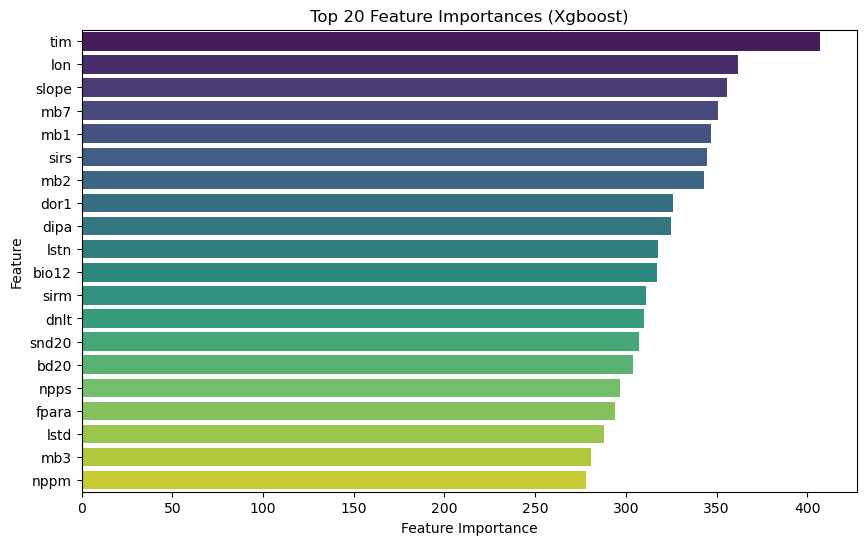
3.1 **Model Performance**

* For the building target, xgboost performed best with roc-auc score of 0.843

The model is able to predict 91% of the no class correctly, but only predicts 2.6% percent of the yes-class correctly

* For the cropland class, xgboost also had the best roc-auc score of 0.837 

Here, th e model struggles to accurately predict both classes.

* For the Woody vegetation cover, xgboost was also the test performing model with roc-auc of 0.75
* The confusion matrices showed that the models struggled with the minority classes in the datasets across each target.
* Limited domain knowledge restricted the creation of more meaningful features, which could have further improved model performance.

**4. Recommendations**

**4.1 Feature Engineering**

* Further domain knowledge could help in creating more meaningful features, potentially improving model performance.
* Advanced feature engineering techniques, such as clustering or dimensionality reduction, could be explored.
* **Longer turning**: Each model was tuned for only 2 iterations.

5**. Conclusion**

This project successfully built a robust predictive model for land cover classification using a multi-classification approach with XGBoost. The model was tuned using Optuna and evaluated on a holdout test set, achieving strong performance metrics. The final model was retrained on the entire dataset, and feature importances were visualized to gain insights into the model's decision-making process. Further improvements can be made by incorporating domain knowledge, exploring advanced models, and collecting additional data