**Technical Report: Land Cover Classification Using CatBoost**

### **1. Introduction**

This report outlines the methodology, findings, and recommendations for predicting land cover types (building, cropland, and wcover) using machine learning. The main objective was to develop a model that classifies these 3 categories and provides probability estimates for each class.

### **2. Methodology**

#### **2.1 Data Loading and Preprocessing**

Imported the necessary libraries for the data computation then loaded the training and test datasets, ensuring consistency in features. The target variables included building, cropland, and wcover. Features included geospatial and environmental attributes such as lat, lon, bio-climatic indicators, soil properties, and land cover classifications. I transformed the dataframe to geopandas dataframe to better manipulate the geospatial data. The training data had 15856 rows and 50 columns, the test data had 35 rows and 47 columns including the geometry column created after gdf creation. Missing values were handled appropriately by dropping the columns that had over 70% missing values then filled in the rest with 0. I performed visualization which helped in understanding the geographic spread and clustering of data to better understand the data, box- plots to explore the relationship between a numerical feature and target variables, and Pearson Correlation that helped to identify relationships (high positive or negative correlations) between features.

#### **2.2 Feature Engineering**

The training and test datasets were concatenated for uniform processing. The geometry column was removed to retain numerical features only. Additional transformations included having the wcover column transformed to Yes for values >60% and No for <30%, >30% then label encoding categorical variables for all the three target variables.

#### **2.3 Model Selection and Training**

For the model CatBoostClassifier was chosen due to its efficiency with categorical and structured data. I performed Hyperparameter using Optuna, optimizing for log-loss. The search focused on the following params; learning rate, max depth, L2 regularization, and bootstrap type, improving model generalization.To maintain class balance across training and validation sets, the data was split using Multilabel Stratified K-Fold (5 folds), ensuring robust learning across different samples. The final model training utilized StratifiedShuffleSplit (5 folds), further enhancing model stability and generalization.

#### **2.4 Predictions and Submission Format**

The trained model predicted probabilities for each class (building, cropland, wcover). The prediction results were formatted in the required submission format, containing:

* + subid: Unique identifier for each test sample.
  + building\_prob: Probability of belonging to the building class.
  + cropland\_prob: Probability of belonging to the cropland class.
  + wcover\_prob: Probability of belonging to the wcover class.

### **3. Key Findings**

The models trained for predicting building, cropland, and wcover performed well in the multilabel classification task. The feature engineering process, which included creating new features like lat\_lon and bcount\_x\_bio1, added value to the model by capturing important relationships in the data. The Multilabel Stratified K-Fold cross-validation ensured that the distribution of labels was consistent across the training and validation sets, which helped avoid bias in model evaluation. The CatBoostClassifier, with early stopping and fine-tuned hyperparameters including; learning rate, max depth, L2 regularization, and bootstrap type, which were optimized using Optuna to enhance model performance and demonstrated strong results across key metrics such as accuracy, log loss, ROC AUC, and F1 score.

### **4. Recommendations**

To further improve model performance, more extensive hyperparameter tuning for the CatBoostClassifier should be conducted, focusing on parameters like learning rate, depth, and iterations. Additionally, more advanced feature engineering techniques, such as exploring non-linear interactions or incorporating external data sources, could provide further improvements. Using ensemble methods, such as stacking or blending multiple models, could help enhance predictive accuracy and robustness. Exploring other models like Random Forest or XGBoost might uncover better-performing algorithms for this multilabel classification task.

Work by: Eunice Wangechi Muriithi