Data Report: GEO-AI Cropland Mapping Project

1. Project Overview

This project was developed as part of the GEO-AI challenge to delineate cropland extent (crop vs non-crop) for target regions using multi-temporal satellite imagery. The goal was to generate per-location cropland probabilities or labels formatted for Zindi competition submissions.

2. Objectives and Constraints

Primary Objective: Maximize the competition score using the official metric.
Secondary Objective: Produce interpretable region-wise performance reports and a reproducible inference pipeline.

Constraints included:
- Limited labelled samples and regional variability.
- Risk of spatial leakage (handled via spatially aware cross-validation).
- Weak labels and the need for data augmentation.

3. Data Preparation

Data preparation involved loading and integrating multiple geospatial datasets including Sentinel-1 and Sentinel-2 imagery and shapefiles for target regions. Key steps included:
- Reading shapefiles using GeoPandas.
- Managing raster data using Rasterio.
- Transforming coordinate reference systems (pyproj).
- Handling missing values via scikit-learn's SimpleImputer.
- Encoding categorical labels with LabelEncoder.
- Scaling features with StandardScaler.

4. Exploratory Data Analysis (EDA)

EDA was performed to understand spatial data distribution and class balance. Visualization tools included matplotlib, seaborn, and folium for interactive mapping.

5. Feature Engineering

Custom features were engineered from multi-temporal spectral indices and spatial attributes. SelectKBest with mutual_info_classif was used to identify relevant predictors.

6. Modeling Approach

The modeling pipeline used scikit-learn Pipelines to combine preprocessing, feature selection, and model fitting. Cross-validation was implemented using StratifiedKFold with spatial considerations.
Models explored included:
- Random Forest Classifier

- Gradient Boosting Classifier
- Logistic Regression (baseline)

Evaluation metrics included accuracy, precision, recall, F1-score, and ROC-AUC.

7. Deployment

The final model and inference pipeline were deployed via Streamlit to enable interactive predictions. This provided a user-friendly interface for uploading new geospatial data and visualizing predictions.

8. Tableau Visualizations

Exploratory and summary visualizations were also developed in Tableau to present spatial patterns, class distribution, and model predictions to stakeholders.

9. Tools and Libraries

Python (Jupyter Notebook), GeoPandas, Rasterio, pyproj, Folium, NumPy, Pandas, Matplotlib, Seaborn, scikit-learn, Streamlit, Tableau.