

Uganda Christian University
Department of Computing and Technology
Artificial Intelligence Project Exam

PROJECT REPORT
AI-BASED CROP RECOMMENDATION SYSTEM FOR
ETHIOPIAN AGRICULTURE

Submitted By:

DANIEL MAGERO MUGULO

REG. NO: M24B38/017

A Project Submitted in Partial Fulfillment of the Requirements
for the Award of the Degree of Bachelor of Science in Information Technology

1.0 Abstract

Background:

Agriculture is the backbone of Ethiopia's economy, yet smallholder farmers often struggle with crop selection due to unpredictable climate variability and soil degradation. Traditional methods of crop planning rely on intuition or generalized data, which can lead to sub-optimal yields. Artificial Intelligence (AI) offers a precision agriculture approach, utilizing historical soil and climate data to recommend the most suitable crops for specific environmental conditions.

Objective:

This project aims to develop a machine learning-based crop recommendation system that predicts the optimal cereal crop to plant based on localized soil parameters (nutrients, pH, moisture) and seasonal climate features (temperature, precipitation, humidity).

Methods:

The study utilizes a dataset of **3,867 samples** combining soil data from the Ethiopian Agricultural Transformation Agency (ATA) and climate data from NASA POWER. Data pre-processing involved outlier detection using the Interquartile Range (IQR) method, label encoding, and standardization. Feature selection was conducted using Mutual Information (MI) and Lasso Regression to identify key predictors. Three distinct AI models were implemented and evaluated: Random Forest Regressor, XGBoost Regressor, and a Feedforward Neural Network (FNN) Classifier.

Results:

Feature selection analysis revealed that seasonal temperature extremes and humidity levels were significant predictors, while soil nutrients showed weaker individual correlations. The regression models (Random Forest and XGBoost) yielded low R^2 scores (≈ 0.12), indicating difficulties in capturing the variance. The Feedforward Neural Network classifier achieved the best relative performance with an accuracy of approximately **50%** on the test set.

Conclusion:

While the FNN outperformed traditional ensemble methods, the overall predictive accuracy suggests highly complex, non-linear relationships in the agricultural data that simple nutrient-weather correlations cannot fully explain. Future work should focus on expanding the dataset and incorporating time-series analysis for better temporal resolution.

2.0 Introduction

2.1 Background

The agricultural sector in Ethiopia is highly sensitive to environmental changes. With a growing population and shrinking arable land, maximizing productivity per hectare is essential. Farmers constantly face the decision of which crop to plant to maximize yield and minimize failure risk. This decision is complicated by fluctuating weather patterns driven by climate change and varying soil health across regions.

2.2 Problem Statement

Farmers lack access to site-specific, data-driven advice for crop selection. Reliance on traditional knowledge fails to account for rapid climatic shifts. There is a need for an automated system that can analyze complex interactions between soil properties (like nutrients: N, P, K, pH) and climatic variables (like rainfall, temperature) to recommend the specific cereal crop most likely to succeed in a given location.

2.3 Objectives

The primary objective is to build an AI model for crop recommendation. Specific goals include:

1. **Analyze** the relationship between soil/climate features and crop types using statistical methods.
2. **Select** the most relevant features using Mutual Information and Lasso Regression to reduce dimensionality and noise.
3. **Develop** and compare three machine learning models: Random Forest, XGBoost, and Neural Networks.
4. **Evaluate** the models using appropriate metrics (Accuracy, F1-score for classification; MSE, R^2 for regression models) to determine the most effective approach.

3.0 Literature Review

Machine learning in precision agriculture has gained significant traction. Studies have demonstrated the utility of algorithms like **Random Forest (RF)** and **Support Vector Machines (SVM)** for crop selection. For instance, research by [1] highlights RF's robustness in handling noisy agricultural data due to its ensemble nature. Similarly, **XGBoost** has been praised for its speed and performance in structured datasets [2].

However, traditional models often struggle with high-dimensional interactions found in ecological data. Recent advancements suggest that **Neural Networks (NN)** can capture complex non-linear relationships better than tree-based models, though they require larger datasets to generalize effectively [3].

Feature selection is also critical. **Mutual Information (MI)** is widely used to capture non-linear dependencies between features and targets, as noted in agricultural yield prediction studies [4]. This project builds on these foundations by applying these specific techniques to the unique context of Ethiopian cereal production, integrating distinct data sources (ATA soil surveys and NASA satellite climatology).

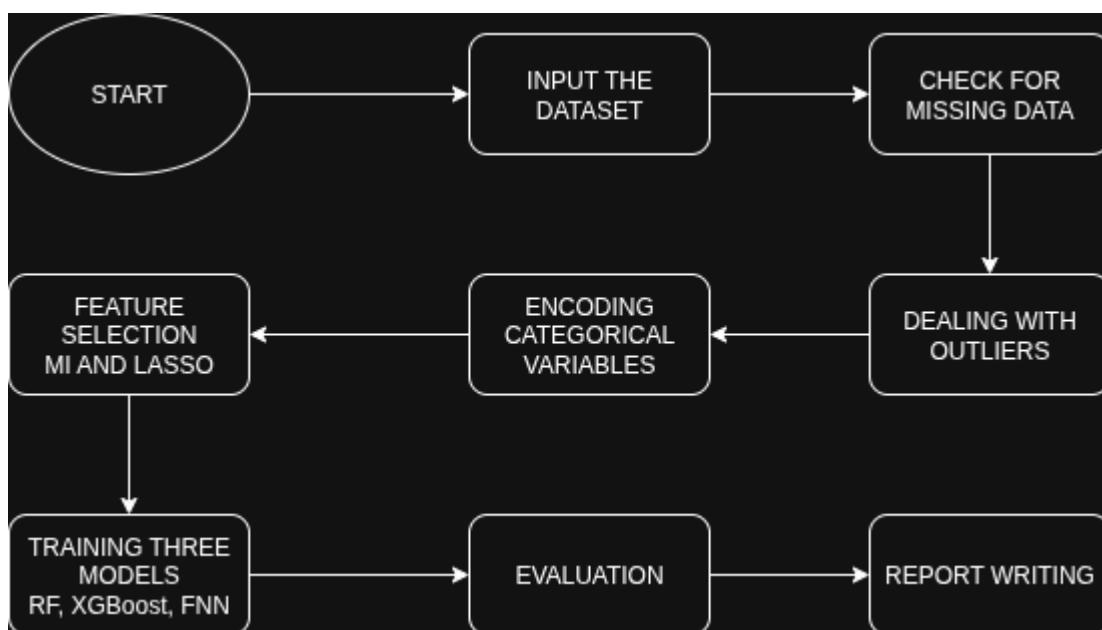
4.0 Materials and Methods

4.1 AI Methods

- **Feature Selection:** Two methods were employed:
 - **Mutual Information (MI):** Measures the dependency between variables. It quantifies the amount of information obtained about the target (Crop Type) by observing a feature (e.g., Soil pH).
 - **Lasso Regression (L1 Regularization):** Adds a penalty equal to the absolute value of the magnitude of coefficients. This shrinks coefficients of less important features to zero, effectively performing feature selection.
- **Modeling:**
 - **Random Forest:** An ensemble method using bagging to create multiple decision trees and merge their outputs to reduce variance.
 - **XGBoost:** A gradient boosting framework that builds trees sequentially, with each new tree correcting errors made by the previous ones.
 - **Feedforward Neural Network (FNN):** A deep learning architecture comprising input, hidden, and output layers to model complex non-linear mappings.

4.2 Workflow

1. **Input Dataset** (Soil & Climate Data)
2. **Preprocessing** (Outlier Capping, Encoding, Scaling)
3. **Feature Selection** (Mutual Information & Lasso)
4. **Model Training** (RF, XGBoost, FNN)
5. **Evaluation** (Metrics & Analysis)



4.3 Dataset Description

The dataset comprises **3,867 instances** of secondary data derived from:

- **Soil Data:** Sourced from the Ethiopian Agricultural Transformation Agency (ATA). Features include Soil pH, Potassium (K), Phosphorus (P), Nitrogen (N), Zinc (Zn), Sulfur (S), and Soil Color.
- **Climate Data:** Sourced from the NASA POWER project. Features include Humidity (QV2M), Temperature (T2M), Precipitation (PRECTOTCORR), and Wind Speed, aggregated seasonally (Winter, Spring, Summer, Autumn).
- **Target Variable:** `label` (Crop Type), consisting of cereal crops such as Barley and Wheat.

4.4 Preprocessing Steps

1. **Outlier Handling:** Outliers were detected using the Interquartile Range (IQR) method. Values falling outside $1.5 \times IQR$ were capped at the 5th and 95th percentiles to preserve data while reducing noise.
2. **Encoding:** Categorical variables (`Soilcolor`, `label`) were converted into numerical format using Label Encoding.
3. **Data Splitting:** The data was split into training (70%) and testing (30%) sets.
4. **Scaling:** Features were standardized using `StandardScaler` (z-score normalization) to ensure equal contribution to distance-based calculations and gradient descent.

4.5 Evaluation Metrics

- **Regression (RF & XGBoost):** Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R^2).
- **Classification (Neural Network):** Accuracy, Precision, Recall, and F1-score.

4.6 Parameter Settings

- **Random Forest:** `n_estimators=100`, `random_state=42`.
- **XGBoost:** `n_estimators=300`, `learning_rate=0.05`, `max_depth=5`.
- **Neural Network:**
 - **Architecture:** Input Layer → Dense (256 units, ReLU) → Dense (128 units, ReLU) → Dropout (0.3) → Dense (64 units, ReLU) → Dropout (0.2) → Output (Softmax).
 - **Optimizer:** Adam.
 - **Loss Function:** Sparse Categorical Crossentropy.
 - **Epochs:** 50 (with Early Stopping).

4.7 Implementation Details

The project was implemented in **Python** using:

- `pandas` and `numpy` for data manipulation.
- `matplotlib` and `seaborn` for visualization.
- `scikit-learn` for preprocessing, feature selection, and traditional ML models.

- tensorflow (Keras) for building the Neural Network.

5.0 Results and Discussion

5.1 Feature Selection Analysis

Mutual Information scores revealed generally weak dependencies between individual features and the target.

- **Top Features:** QV2M-Sp (Spring Humidity), QV2M-Su (Summer Humidity), and QV2M-W (Winter Humidity).
- **Insight:** Climate variables, particularly seasonal humidity and temperature extremes, were more informative than soil nutrients (N, P, K). This suggests that macro-climatic conditions are the primary drivers for crop suitability in this dataset.

5.2 Model Performance

Model	Metric	Value
Random Forest	R^2 Score	0.12
	RMSE	3.94
XGBoost	R^2 Score	0.1155
	RMSE	3.94
Neural Network	Accuracy	~49.87%
	Macro F1-Score	0.17

5.3 Discussion

- **Low Regression Performance:** The low R^2 scores for Random Forest and XGBoost indicate that the features do not have a strong linear or simple non-linear correlation with the crop label when treated as a regression problem.
- **Neural Network Superiority:** The FNN achieved an accuracy of ~50%, which, while not perfect, is significantly better than random guessing (which would be ~8% for 12 classes). The confusion matrix showed the model performed well for majority classes (e.g., Label 10) but struggled with minority classes (Labels 2, 3, 5, 6, 7, 8), indicating a **class imbalance** issue.
- **Key Insight:** The "Support" column in the classification report revealed a severe imbalance (e.g., 380 samples for class 10 vs. only 7 for class 3). This heavily biased the models toward the dominant crops.

6.0 Conclusion and Future Work

This project successfully developed a pipeline for analyzing agricultural data to recommend crops. The results demonstrated that while climatic factors like humidity and temperature are critical, soil nutrients alone are insufficient predictors in this specific dataset. The Neural Network model showed the most promise, achieving ~50% accuracy.

Link to the project: <https://github.com/DanielMagero/AIProject#>

Future Work:

1. **Address Class Imbalance:** Implement techniques like SMOTE (Synthetic Minority Over-sampling Technique) to balance the dataset.
2. **Hyperparameter Tuning:** Perform Grid Search or Bayesian Optimization for the Neural Network.
3. **Data Enrichment:** Incorporate additional features such as soil texture, terrain slope, or satellite imagery (NDVI).
4. **Hybrid Models:** Explore ensemble voting classifiers combining RF and Neural Networks.

7.0 References

1. P. S. Maya Gopal and R. Bhargavi, "Performance Evaluation of Best Feature Subset Selection for Crop Yield Prediction Using Machine Learning Algorithms," *Journal of Big Data*, vol. 6, 2019.
2. T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.
3. A. Kamilaris and F. X. Prenafeta-Boldú, "Deep Learning in Agriculture: A Survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70-90, 2018.
4. R. Battiti, "Using Mutual Information for Selecting Features in Supervised Neural Net Learning," *IEEE Transactions on Neural Networks*, vol. 5, no. 4, 1994.
5. NASA POWER Project, "NASA Prediction of Worldwide Energy Resources," [Online]. Available: <https://power.larc.nasa.gov/>.
6. S. Alemu, "Crop Recommendation using Soil Properties and Weather Prediction Dataset." Sep. 04, 2024. <https://doi.org/10.17632/8v757rr4st.1>.

8.0 Appendices

A. Feature Selection Code Snippet

```
from sklearn.feature_selection import mutual_info_regression
# Calculate MI scores
mi_scores = mutual_info_regression(X, y, random_state=42)
mi_ranked = pd.DataFrame({
    'Feature Name': X.columns,
    'MI Score': mi_scores
}).sort_values(by='MI Score', ascending=False)
```

B. Neural Network Architecture

```
def create_model_config():

    custom_adam = Adam(learning_rate=0.0001)
    l2_reg_factor = 0.001 # Regularization strength

    model = Sequential()

    # Apply L2 Regularization to hidden layers
    model.add(Dense(256, input_dim=X_train_scaled.shape[1],
activation='relu', kernel_regularizer=l2(l2_reg_factor)))

    model.add(Dense(128,
activation='relu',kernel_regularizer=l2(l2_reg_factor)))
    model.add(Dropout(0.3))

    model.add(Dense(64, activation='relu',
kernel_regularizer=l2(l2_reg_factor)))
    model.add(Dropout(0.2))

    model.add(Dense(n_classes, activation='softmax'))
    model.compile(loss='sparse_categorical_crossentropy',
                  optimizer=custom_adam,#Use custom optimizer
                  metrics=['accuracy'])

return model
```