

## Agricultural Production Prediction Using Ensemble Models

---

### Overview

This project focuses on predicting agricultural **production (in tonnes)** based on multiple features from **FAOSTAT data**. The workflow includes preprocessing, feature engineering, model training, ensemble evaluation, and a Streamlit-based interactive interface.

---

### Data Preprocessing

#### 1. Source:

- FAOSTAT Excel file with columns:  
*Domain Code, Area, Element, Item, Year, Value*

#### 2. Filtering:

- Retain only:  
Area harvested, Yield, Production, Area, Item, Year

#### 3. Pivoting:

- Pivot Element column with:  
index = ['Domain Code', 'Area', 'Item', 'Year']

#### 4. Cleaning and Imputation:

- Rename for clarity
- Convert Yield (kg/ha → tonnes/ha)
- Use:  $\text{Production} = \text{Area} * \text{Yield}$  for filling missing
- Fill other nulls with median
- Since the data is skewed converted all the data in to log scale for reducing the skewness

#### 5. Encoding & Scaling:

- Label Encode: Area, Item
  - Log1p transformation applied to features since some of the data was zero. Since log of zero is infinity that would not make any valuable data for training the model, hence used log1p to get rid of this issue and target to stabilize variance and improve model performance.
- 

### Model Training




#### 1. Features:

- Year, Yield, Area Harvested, Item, Area

## 2. Target:

- Production (tonnes)

## 3. Models Used:

-  GradientBoostingRegressor
-  XGBRegressor
-  Ensemble (average of both)

## 4. Training Details:

- n\_estimators=1000, random\_state=42
- 80/20 train-test split


## 5. Persistence:

- Save models using joblib
  - Cleaned data exported to CSV for future reference
- 



## Streamlit Interface

A real-time, interactive prediction tool built using Streamlit.

### Sidebar Inputs:

- Year, Yield, Area Harvested, Crop/Item, Area
-  Predict Button

### Main Area Displays:

-  Predicted Production (tonnes)
-   $R^2$  Scores for:
  - Gradient Boosting
  - XGB Regressor
  - Ensemble Model





### Model Comparison Plots:

- Scatter plots (Actual vs Predicted)
  - Log-scaled axes with diagonal "perfect line"
-

## Model Insights

Model	R <sup>2</sup> Score	Notes
Gradient Boosting	~0.937	Slightly underperforms vs XGB
XGB Regressor	~0.989	★ Best accuracy overall
Ensemble (Average)	~0.977	High but slightly reduced due to GB underperform.

## Conclusion:

-  XGB Regressor = most stable & accurate
-  Ensemble slightly lowered by GB performance
-  Tried other objectives (e.g., pseudohubererror), but R<sup>2</sup> dropped
-  Random Forest tested but dropped due to high compute demand

---

## Strengths

- ✓ High R<sup>2</sup> (> 0.95)
  - ✓ Smart handling of missing values
  - ✓ Intuitive and easy-to-use interface
  - ✓ Portable model using joblib + Streamlit
-