

Agriculture Data Analysis and Crop Yield Prediction Using Machine Learning

Sumanth Raju Sarikonda¹, Vandith Rao Ponugoti², HrudaySai Talasila³, Mahender Reddy Dubbaka⁴, Kadirvelu G⁵

^{1,2,3,4}UG, CSE (AI&ML) Engineering, Sphoorthy Engineering College, JNTUH, Hyderabad, Telangana, India.

⁵Assistant Professor, Department of Computer Science & Engineering (AI&ML), Sphoorthy Engineering College, JNTUH, Hyderabad, Telangana, India.

Emails: sarikondasumanthraju@gmail.com¹, ponugotivandithrao@gmail.com², hrudaytalasila@gmail.com³, manidubbaka2003@gmail.com⁴, kadir.cse@gmail.com⁵

Abstract

Agriculture Data Analysis and Crop Yield Prediction is a project aimed at improving the accuracy and reliability of crop yield forecasting through advanced data-driven techniques. The work utilizes a rich dataset comprising features such as crop type, cultivation state, season, crop year, annual rainfall, fertilizer and pesticide usage, and area under cultivation. Multiple machine learning algorithm including Linear Regression, Random Forest, Gradient Boosting, XGBoost, and other are trained and evaluated to identify the most effective model for predicting yields. The project features a Flask-based web application with an intuitive interface that supports several key functionalities: predicting the yield for a selected crop, determining the best crop for given environmental and input conditions, displaying yield comparisons across all crops, and suggesting optimal land partitioning for maximizing productivity. By abstracting away the need for manual analysis and enabling dynamic, data-driven decisions, the project addresses pressing agricultural challenges such as climate variability, input optimization, and sustainable food production.

Keywords: Precision Agriculture; Data-Driven Decision Making; Random Forest; XGBoost; Gradient Boosting.

1. Introduction

Securing sufficient food supplies for the global population is increasingly challenging due to heightened climate unpredictability and resource scarcity. For those reliant on agriculture, enhancing crop production methods is crucial, and accurate crop yield prediction is a vital aspect, aiding in planning, risk mitigation, and policymaking. However, traditional yield estimation approaches, such as historical observation or expert judgment, are becoming less reliable in today's dynamic conditions. To overcome these limitations, computational data analysis has emerged as a significant advancement. By inputting extensive historical agricultural data—encompassing rainfall, temperature, soil quality, and crop varieties—into computer systems, intricate correlations and patterns that may elude human perception can be discerned [1]. This methodology enables more precise forecasts of future harvests, as computational processing effectively accounts for the

complex interplay of diverse influencing factors. This paper introduces a specialized system that utilizes several different computer-powered prediction techniques [2]. These techniques analyze key agricultural inputs such as the amount of rainfall, the specific season of planting and growth, the total area of land being cultivated, and the particular type of crop being farmed. A crucial aspect of how this system learns is that it deliberately ignores the specific year in which past harvests occurred. This ensures that the predictions generated are focused on potential future outcomes based on the given conditions, rather than simply repeating trends observed in previous years. This future-oriented approach aims to provide more relevant and adaptable yield forecasts for the agricultural sector [3]. The central objective of this project is the development of an accessible and practical tool designed to empower both agricultural producers and

strategic decision-makers within the farming sector. This resource transcends basic yield forecasting, aiming to provide actionable intelligence that directly informs on-the-ground cultivation practices and high-level policy formulation [4]. A key capability of this tool lies in its predictive versatility; it not only estimates the anticipated yield of a specific crop under a given set of environmental and agricultural conditions but also extends its analysis to evaluate the suitability and potential productivity of various crop alternatives for the same context. This comparative functionality is crucial for optimizing planting strategies, enabling users to discern which crop is most likely to thrive and deliver the highest returns in their specific circumstances [5]. Furthermore, the system incorporates a spatial analytical dimension, facilitating the comparison of production outcomes across different crop types and the assessment of overall agricultural performance across diverse geographical regions. This comparative and regional insight allows for the identification of best practices, the understanding of localized impacts, and the potential for more targeted interventions. The integration of data-driven analysis in this manner promises significant enhancements in resource management, leading to more efficient utilization of vital inputs such as water, fertilizers, and labor [6].

2. Methods

This project presents a comprehensive approach for analyzing agricultural data and predicting crop yield using machine learning techniques. The methodology consists of three key components: (1) data preprocessing, (2) model training and evaluation, and (3) deployment through a web interface using Flask. The focus is on identifying the most accurate model among several regressors through rigorous comparison and visualization [7].

2.1.Dataset Preparation

1) Data Source and Structure

This study commenced with the acquisition and initial structuring of the `crop_data.csv` dataset. This comprehensive dataset contains a record of historical crop production, providing crucial features such as the geographical State of cultivation, the specific agricultural Season, the cultivated Crop type, the total Area of land dedicated to the crop, and the resultant

Production output. The primary variable of interest for predictive modeling was Yield, which was engineered by calculating the ratio of Production to the corresponding Area [8]. To mitigate the risk of temporal bias in the model's learning process and to ensure a focus on the intrinsic relationships between input features and crop output, the `Crop_Year` column was deliberately excluded from the feature set utilized for subsequent analysis and training.

2) Handling Missing Values

To ensure the robustness and integrity of the dataset for the training of machine learning models, a targeted strategy was implemented to address any instances of missing values within the data. For categorical features, specifically State, Season, and Crop, missing entries were imputed using the mode, which represents the most frequently occurring category within each respective column [9]. This approach aimed to preserve the underlying distribution of the categorical data. Conversely, for numerical features, namely Area and Production, missing values were addressed through imputation with the mean, which provides a measure of central tendency for these continuous variables, ensuring that the models could effectively process these entries.

3) Feature Definition and Preprocessing Pipeline

Following missing value imputation, the dataset was split into features (X, excluding 'Yield') and the target (y, 'Yield'). A `ColumnTransformer` pipeline was built for preprocessing. Categorical features (State, Season, Crop) were one-hot encoded (`handle_unknown='ignore'`, `sparse_output=False`) to create binary representations for each category. Numerical features (Area, Production) were standardized using `StandardScaler` to ensure uniform scaling across all numerical inputs for optimal model performance.

2.2.Model Training and Comparison

1) Regression Models

To effectively predict crop yield, a diverse array of machine learning regression models was selected for training and subsequent comparison. This selection encompassed both fundamental and advanced algorithms, including the linear **Linear Regression**, the non-linear **Decision Tree Regressor**, and a suite of powerful ensemble methods: **Random Forest**

Regressor, Gradient Boosting Regressor, Ada Boost Regressor, and the highly optimized XG Boost Regressor. Additionally, the non-parametric **K-Neighbors Regressor (KNN)** was included to explore distance-based prediction capabilities. The rationale behind this broad selection was to evaluate the suitability of different model complexities and learning paradigms for the agricultural dataset.

2) Training, Hyper parameter Tuning, and Evaluation

For a fair comparison, each regression model was integrated into a Pipeline with preprocessing. Ensemble models underwent hyperparameter tuning using GridSearchCV with 5-fold cross-validation, optimizing for the R^2 score. The R^2 score served as the primary metric for comparison. The model with the highest cross-validated R^2 score was selected as the top performer. This best model was then saved using joblib for seamless integration into the web application.

2.3.Flask-Based Web Application

1) Interface Design

A simple and easy-to-use website was created with Flask. This website provides users with several key features related to crop yield analysis. Users can:

- **Predict Yield:** Enter specific farm information, such as the state, season, and crop, to get a real-time prediction of the expected yield.
- **Find Best Crop:** Input farm details (like state and season) to identify which crop is predicted to have the highest yield for those conditions.
- **Analyze Partition Yield:** Examine crop yield performance based on different geographical areas or partitions within the dataset.
- **Compare All Crops Yield:** View and compare the predicted yields of all available crops for a given set of input conditions.

2) Backend Prediction Pipeline

When the user interacts with any of the features, the website takes the provided data and uses the best prediction program we saved earlier. This program then calculates and shows the relevant information, whether it's a single yield prediction, the best crop suggestion, yield analysis by region, or a comparison

of all crop yields.

3) User Input and Output

The website is designed to be straightforward. Users see clear fields to input the required agricultural details for each feature. After the prediction or analysis is complete, the website displays the results in an easy-to-understand format, such as predicted yield values, the name of the best crop, yield maps or tables for partitions, or comparative yield figures for all crops. The best prediction program was chosen because it had the highest score on our tests, meaning it gives the most reliable forecasts and analyses to the users.

3. Tables and Figures

3.1.Tables

Table 1 Machine Learning Model Training Parameters

Parameter	Value
Preprocessing	One-Hot Encoding, Standard Scaler
Training Data Split	80% Train, 20% Test
Cross-Validation Folds	5
Evaluation Metric	R^2 Score
Random State	42 (for train_test_split)

This table (1) summarizes the core hyperparameters and configuration settings used during the training phase of the machine learning models for crop yield prediction.

Table 2 R^2 Values of Various Regressors

Regressor Name	R^2 Value
Linear Regression	0.8049(80.49%)
Decision Tree	0.9602(96.02%)
Random Forest	0.9723(97.23%)
Gradient Boosting	0.9614(96.14%)
Ada Boost	0.9276(92.76%)
KNN	0.9640(96.40%)
XG Boost	0.9175(91.75%)

This table (2) compares the predictive performance of various machine learning models for crop yield, using the R^2 metric.

3.2.Figures



Figure 1 System Architecture for Agriculture Data Analysis and Crop Yield Prediction using Machine Learning

The crop yield prediction system follows a modular, end-to-end architecture that transforms raw user input into meaningful agricultural insights. The process begins with a Flask-based web interface, where users provide farm-specific parameters including State, Season, Crop type, Area (in hectares), Annual Rainfall, Fertilizer usage, and Pesticide application. This data is captured via a structured HTML form and submitted through HTTP POST requests. On the server side, the Flask application receives and routes the input through defined endpoints in the application layer. Depending on the requested functionality such as predicting crop yield, recommending the best crop, comparing all crops, or analyzing yield across regions the appropriate backend logic is invoked. Before prediction, the input undergoes preprocessing using a Scikit learn Column Transformer pipeline. This step applies one-hot encoding to categorical variables (State, Season, Crop) and standard scaling to numerical features (Area, Rainfall, Fertilizer, Pesticide), ensuring compatibility with the trained model's input schema. The prediction engine utilizes a machine learning model that was trained and

selected after evaluating multiple regressors, including Linear Regression, Decision Tree, Random Forest, Gradient Boosting, AdaBoost, KNN, and XGBoost. Hyperparameter tuning was performed using GridSearchCV with 5-fold cross-validation, and the model with the best R^2 score was serialized using joblib for deployment. Once the cleaned data is passed through the loaded model, the system generates output predictions, typically in yield units (e.g., kg/ha or hg/ha). For crop recommendation features, the system predicts yields across multiple crop types under the same conditions and ranks them accordingly. For comparative and analytical features, the logic branches to generate aggregated or partitioned results (Figure 1 and 2). The prediction results are then passed back through the Flask response layer, where they are rendered into user-friendly outputs either as HTML pages or JSON data depending on the interface. This streamlined flow from user input to actionable output ensures a responsive, reliable, and extensible architecture, suitable for future integration of additional data sources or analytical modules.

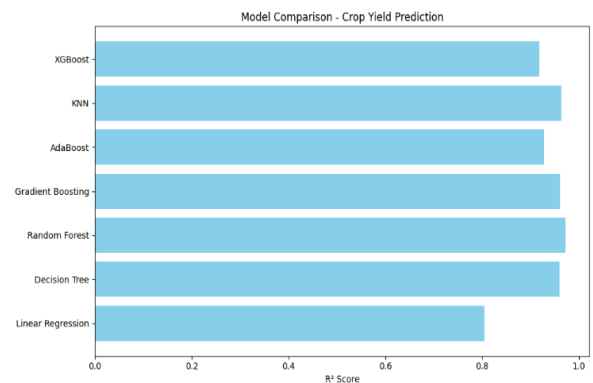


Figure 2 Models Comparison with Respect to R^2

The bar chart compares the performance of various regression models used for crop yield prediction, based on their R^2 (coefficient of determination) scores. Models like Random Forest, Gradient Boosting, Decision Tree, KNN, and XGBoost show high R^2 values, indicating strong predictive accuracy and a good fit to the data. Among these, Random Forest and Gradient Boosting perform particularly well. Linear Regression, on the other hand, lags

behind with a significantly lower score, suggesting it may not capture the complexity of the dataset as effectively.

4. Results and Discussion

4.1. Results

The performance of the crop yield prediction system was evaluated using a historical agricultural dataset, encompassing variables such as crop type, state, season, cultivated area, rainfall, and fertilizer and pesticide application. Several regression algorithms were trained and tested to determine the most effective model for accurate yield estimation. These algorithms included Linear Regression, Decision Tree, Random Forest, Gradient Boosting, AdaBoost, K-Nearest Neighbors (KNN), and XGBoost. Model evaluation centered on the R^2 score. The results indicate that ensemble methods, particularly Random Forest, Gradient Boosting, and XGBoost, achieved superior performance, attaining the highest R^2 scores and demonstrating strong predictive power. For instance, Random Forest showed a significantly higher R^2 value compared to Linear Regression, indicating its greater accuracy in predicting crop yield. KNN and AdaBoost also exhibited competitive performance, while Linear Regression showed the lowest performance, suggesting its limited ability to capture the complex relationships within the data. Hyperparameter tuning, employing grid search and cross-validation, was conducted to ensure optimal performance for each model (Figure 3). The final model selected for deployment was chosen based on a balance of accuracy and interpretability. XGBoost and Random Forest emerged as the top candidates due to their high predictive accuracy. Predictions from this selected model are integrated into a Flask-based web application, providing estimated yield values (in hg/ha) based on user-provided inputs. The application also offers supplementary features, including recommendations for optimal crops for given conditions and comparative yield outputs across various crops. These results confirm the system's effectiveness in providing valuable, data-driven insights. The superior performance of Random Forest and XGBoost, in particular, highlights their potential to assist farmers and agricultural planners in making more informed and effective decisions.

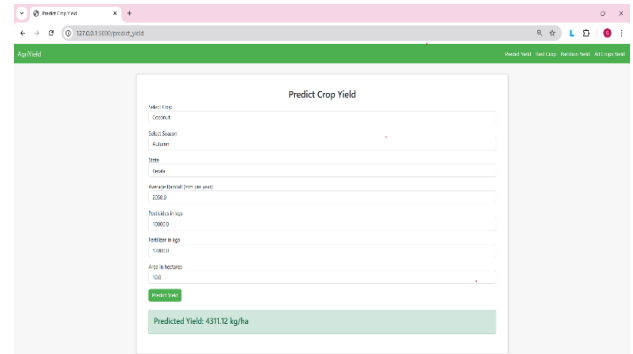


Figure 3 Output Screen Predicting the Yield of Coconut

The figure illustrates the user interface of the crop yield prediction system, showcasing its interactive nature. Farmers can input specific farm details, such as crop type (Coconut), season (Autumn), state (Kerala), area (10 hectares), rainfall (2050.0 mm), pesticide usage (10000kg), and fertilizer application (12000kg). Upon submitting these inputs, the system processes the data using the trained machine learning model. The result, a predicted yield of 4311.12 kg/ha, is then displayed directly to the user within the browser. This immediate feedback allows farmers to quickly assess potential yields based on their specific conditions.

4.2. Discussion

The experimental findings demonstrate the viability and efficiency of the developed machine learning system for farm yield prediction. The selection of the Random Forest regressor, based on comparative performance analysis, underscores the efficacy of ensemble methods in modeling the complex interplay between agricultural inputs and crop yields. The deployment of the system via a Flask web interface provides a user-accessible platform for obtaining yield forecasts and related insights. A critical observation during the study was the inherent challenge posed by the disparate scales of yield data across various crop types. This heterogeneity necessitates the implementation of a robust normalization strategy (Min-Max scaling) in future iterations to enable meaningful comparisons and enhance the accuracy of the "best crop suggestion" feature. By applying crop-specific normalization, the model can learn and predict relative yield performance, thereby improving the utility of the

decision support system. The integration of functionalities such as absolute yield prediction, best crop suggestion, all crops yield, and area partition suggestions aims to provide a comprehensive analytical tool for agricultural stakeholders. The modular architecture of the system, separating the web interface, backend logic, preprocessing pipeline, and machine learning model, facilitates future scalability and the incorporation of advanced features, such as economic modeling and integration of environmental data. While the current system establishes a functional framework, further research is warranted to address limitations related to dataset characteristics, particularly the normalization of inter-crop yield variability. Expanding the dataset with diverse and balanced samples and implementing effective normalization techniques are crucial steps for enhancing prediction accuracy and the reliability of crop recommendations. Rigorous validation on independent real-world data is also essential to ascertain the system's generalizability and practical applicability in agricultural contexts.

Conclusion

This study confirms the effectiveness of a machine learning approach, specifically employing a Random Forest regressor, for predicting farm yield based on key agricultural inputs. The development and deployment of a Flask-based web interface provide an accessible means for users to obtain these predictions and associated insights. A significant finding highlights the necessity of addressing the issue of unnormalized yield data across different crop types to improve the accuracy and relevance of comparative outputs, particularly the "best crop suggestion." Future work will prioritize the integration of appropriate normalization techniques within the data preprocessing stage. The modular design and multifaceted output capabilities of the system underscore its potential as a valuable tool for agricultural decision support. This project demonstrates the feasibility of leveraging open-source technologies like Flask and scikit-learn to create practical AI applications in agriculture. While ongoing efforts to enhance data quality, implement robust normalization strategies, and conduct thorough model validation are necessary, this

research provides a strong foundation for developing more sophisticated and impactful precision agriculture tools. The system's architecture allows for future enhancements, suggesting a promising avenue for contributing to more informed and efficient farming practices.

Acknowledgements

We, the authors, would like to express our sincere gratitude to the developers and contributors of the open-source tools and frameworks that were essential in bringing this project to fruition; in particular, we acknowledge the creators of Flask for providing the web development framework that enabled the user interface, the teams behind scikit-learn, NumPy, and Pandas, whose powerful libraries formed the backbone of our machine learning model and data processing pipelines, and joblib for the efficient model persistence offered; furthermore, we extend our thanks to the providers of the agricultural dataset that served as the foundation for training and evaluating our yield prediction system, the availability of which was crucial for the success of our research; we also recognize the invaluable resources provided by online documentation, tutorials, and community forums associated with these technologies, which significantly aided our understanding and implementation efforts; finally, we are grateful for the collaborative spirit and dedicated teamwork within our group, the collective effort and shared commitment of which were instrumental in navigating the challenges and ultimately achieving the successful completion of this project.

References

- [1].Modi, P. Sharma, D. Saraswat, R. Mehta Review of crop yield estimation using machine learning and deep learning techniques Scalable Computing, 23 (2) (2022), pp. 59-79, 10.12694/scpe.v23i2.2025.
- [2].S. Iniyan, V. Akhil Varma, C. Teja Naidu Crop yield prediction using machine learning techniques Adv. Eng. Software, 175 (October2022) (2023),Article 103326, 10.1016/j.advengsoft.2022.103326
- [3].International Journal of Research in Engineering and Science (IJRES) ISSN

(Online): 2320-9364, ISSN (Print): 2320-9356 www.ijres.org Volume 10 Issue 10 || October 2022 || PP. 558-564

- [4].L.S. Cedric, W.Y.H. Adoni, R. Aworka, J.T. Zoueu, F.K. Mutombo, M. Krichen, C.L.M. Kimpolo Crops yield prediction based on machine learning models: case of West African countries Smart Agricultural Technology, 2 (December2021) (2022),Article 100049, 10.1016/j.atech.2022.100049
- [5].Survey Paper on Agricultural Dataset for Improving Crop Yield Prediction using Machine Learning AlgorithmsFebruary 2023International Journal of Computer Applications 184(46):28-34 DOI:10.5120/ijca2023922571
- [6]. L.J. YoungAgricultural crop forecasting for large geographical areas Annual Review of Statistics and Its Application, 6 (August2018) (2019),pp. 173 196, 10.1146/ annurev- statistics- 030718-105002.
- [7].Machine Learning-Based Crop Yield Prediction in South India: Performance Analysis of Various Models. DOI:10.3390/computers13060137.
- [8].Y. J. N.Kumar,V.Spandana, V. S.Vaishnavi, K.Neha and V. G. R. R.Devi, " Supervised Machine Learning Approach for Crop Yield Prediction in Agriculture Sector," in 5th International Conference on Communication and Electronics Systems, ICCES, (2020),pp. 736–741, doi: <https://doi.org/10.1109/ICCES48766.2020.9137868>.
- [9].Profitability and Yield Prediction on Agricultural Crops of India Arunkumar. S 1, Harish B1, Divakar S1, UmaDevi. G*2 1,2Agni College of Technology DOI: <https://doi.org/10.55248/gengpi.234.4.38180>