Comparative Analysis of Crop Price Prediction Using Linear Regression, Xgboost, And A Custom Linear Model

Elakkiyan K
Department of Computing Technologies
SRM Institute of Science and Technology

Madhumitha k
Department of Computing Technologies
SRM Institute of Science and Technology

***Abstract*— The goal of this is to create a machine learning-based agricultural price prediction system that will help farmers make better decisions and lessen market uncertainty. In order to offer precise price forecasts in Indian rupees per quintal, the model makes use of historical data that includes district, crop type, market, and date. The method entails preparing the data and implementing a custom model that is compared to XGBoost and conventional Linear Regression models. The findings show that the customized model provides a competitive alternative to current techniques and performs similarly to linear regression in terms of accuracy. By offering insights into market trends, this technology has the ability to empower the agricultural community and ultimately improve financial planning and market efficiency.

I .INTRODUCTION

For many years, agriculture has been the mainstay of the Indian economy, providing for the livelihoods of millions of people and being essential to the country's food security. Indian agriculture still confronts structural issues, especially with regard to market access and price transparency, even if technology has advanced in many areas. Farmers frequently find it difficult to decide when and where to sell their produce, which results in ineffective market strategies and financial loss. The erratic character of crop prices, which vary according to a wide range of factors such as supply-demand imbalances, weather, transportation difficulties, and regional differences in agricultural infrastructure, is one of the main reasons of this problem.

.

There is a growing chance to use machine learning and data analytics to address urgent issues in agriculture in a time when similar technologies have revolutionized many other industries. The goal of this project, "Crop Price Prediction Using Machine Learning," is to create models that predict future prices with a high degree of accuracy by using historical crop pricing data. Machine learning algorithms can identify patterns that conventional analysis techniques might miss by examining data like crop kind, district, market location, and date.

This project presents a specially designed crop price prediction model and compares it to well known algorithms such as XGBoost and Linear Regression. According to the results, the custom model offers a workable solution that is suited to the particular difficulties of agricultural pricing in India and performs similarly to linear regression.

.

II.RELATED WORK

The reduction of energy usage in cellular networks is a crucial topic in research, and several approaches have been proposed. One of these approaches is using deep learning (DL) to address radio resource allocation issues in multicell networks, as described in [2]. The model is first trained on data produced by The reduction of energy consumption in cellular networks has become a **critical area of research** in recent years, driven by the exponential growth of mobile data traffic, the deployment of dense heterogeneous networks, and the global push toward sustainable and environmentally friendly technologies. Since cellular networks account for a significant portion of energy usage in information and communication technologies (ICT), improving their energy efficiency not only reduces **operational expenditure (OPEX)** for mobile network operators (MNOs) but also contributes to the achievement of **green networking goals**. Several approaches have been investigated in the literature, ranging from resource allocation optimization and base station (BS) sleep modes to AI-driven predictive mechanisms.

One of the most widely studied directions involves the use of **deep learning (DL) for radio resource allocation**. In [2], the authors proposed a DL-based model trained using data generated by a genetic algorithm. The hybrid approach leveraged the exploration capability of the genetic algorithm while enabling the deep learning model to generalize allocation strategies across various network conditions. Simulation results showed that the model could reproduce the optimal allocation in approximately **86.3% of test cases**, thereby significantly reducing computational complexity compared to heuristic optimization alone. This result demonstrates the potential of **learning-based resource allocation** to achieve near-optimal performance with reduced time complexity.

In parallel, **distributed data mining (DDM) approaches** have been introduced to address scalability and energy efficiency in wireless sensor networks (WSNs). For example, [4] developed a DDM model that incorporated a **long short-term memory (LSTM)-based recurrent neural network (RNN)**. The primary contribution of this work was the reduction of computational overhead at the central fusion node by distributing learning tasks across multiple base stations. This Dynamic control of BS activity has been another major research direction. In [6], a machine learning-based **calculation and adjustment mechanism** was introduced to maximize the **switch-off duration of cells** per day. By accounting for heterogeneous BS types with varying power consumption profiles, the approach ensured that energy savings were distributed evenly across the network. This type of AI-driven **adaptive switch-off control** is especially valuable in urban scenarios with strong daily traffic fluctuations. not only enhanced **load balancing** but also minimized **energy consumption at the central controller**, ensuring that large-scale sensor networks could operate more sustainably. Such distributed frameworks are becoming increasingly important in

the context of **5G and beyond networks**, where centralization often leads to bottlenecks.
Another innovative approach is the **SEBS (Spectrum and Energy-efficient Base Station Sharing) strategy**, presented in [5], which targeted the co-located base stations of multiple cooperative operators. This strategy introduced methods for **dynamic spectrum sharing across operators**, considering **inter-RAN traffic demand variations**. By dynamically reallocating spectrum resources among different MNOs, the system improved spectrum efficiency while also reducing energy consumption by allowing certain BSs to remain idle during low traffic conditions. Importantly, this cooperative approach provides economic incentives for operators, as energy savings are directly linked to reduced operational costs. This line of work highlights the potential of **inter-operator collaboration** as a path toward sustainable RAN deployment.

The reduction of energy consumption in cellular networks has attracted significant research attention due to the increasing demand for mobile data services and the urgent need for sustainable communication infrastructures. Cellular networks are one of the largest contributors to the overall energy footprint of ICT, and research has explored multiple approaches, ranging from **machine learning models for radio resource allocation** to **cooperative spectrum sharing** and **AI-driven base station management**. In this section, we review key contributions from prior studies, grouped by methodology.

Early works applied optimization algorithms such as genetic algorithms and heuristic techniques to manage spectrum allocation and scheduling. However, such methods often suffer from high computational complexity. To overcome this, recent studies have applied **deep learning (DL)** to approximate resource allocation strategies. For example, [2] proposed a DL-based approach trained on data generated by a genetic algorithm. Simulation results showed that the trained model was able to reproduce the optimal solution in **86.3% of cases**, significantly reducing computational time while maintaining near-optimal performance. This demonstrates the capability of DL to **generalize optimization strategies** for multicell resource allocation.

Complementary approaches have been suggested by numerous other studies:

Reinforcement learning (RL) for dynamically adjusting capacity and coverage in self-organizing networks (SONs).techniques for BS sleep scheduling that use clustering to reduce signaling overhead and preserve QoS.Hybrid optimization models combine energy conservation with spectrum efficiency by combining machine learning and linear programming.Sustainability is increased by integrating smart grid systems with BSs powered by renewable energy.cross-layer strategies that optimize energy consumption from beginning to finish by taking into account MAC, RAN, and transport protocols.

Together, the aforementioned publications demonstrate the depth of knowledge in energy-aware cellular networks research. Large datasets and training costs are necessary for DL-based approaches, notwithstanding their high accuracy. Cooperative sharing and distributed mining save energy, but they rely significantly on infrastructural support and operator cooperation. Although AI-driven control saves energy, its accuracy depends on how well traffic predictions are made. Furthermore, the majority of research use simulation datasets to test their models, which restricts their use in real-world deployments.In order to fill these deficiencies, hybrid frameworks are required that:Combine distributed and cooperative resource management with predictive deep learning.For big heterogeneous networks, strike a balance between scalability, interpretability, and accuracy.are verified on sizable, realistic datasets to guarantee their viability in practice.This drives our ongoing research, which creates a dataset and assesses machine learning techniques that seek to strike a balance between computing cost, energy efficiency, and forecast accuracy for practical implementation. A parallel stream investigates **PHY-layer levers**—Massive MIMO mode selection, adaptive rank, **carrier aggregation (CA)** enable/disable, and **beamforming**—to balance spectral efficiency and power draw. Energy is reduced by selectively lowering spatial layers or muting secondary carriers during low load, while preserving edge QoS via beam shaping and dynamic TDD. Heuristics and ML surrogates are used to decide **when** to scale layers or carriers, yielding meaningful kWh savings without permanent capacity loss. Limits include the need for **fine-grained telemetry** (PRB utilization, BLER, CQI) and vendor-specific control hooks, which complicate cross-RAT deployments.

Works combining **(e)ICIC**, Almost Blank Subframes, and sleep scheduling show that power savings can coexist with cell-edge performance if **interference maps** are part of the actuation logic. Scheduling elastic traffic toward protected subframes allows more aggressive micro/small-cell sleeping. However, coordination overhead and timing alignment become critical when multiple tiers (macro/micro/pico) and operators share spectrum [10], [11].

L. Traffic Classification and Context-Aware Policies

Several papers employ **context signals**—time-of-day, venue type, mobility class, app mix—to tailor energy actions. Lightweight classifiers predict **traffic volatility** (burstiness vs. smooth diurnal), then select conservative (short sleeps, high guard margins) or aggressive (long sleeps, deep muting) policies accordingly. This reduces oscillations and HO ping-pong observed with "one-size-fits-all" controllers [12]–[14]. The main drawback is **concept drift** (e.g., festivals, emergencies), which degrades classifiers unless online adaptation is present.

M. Multi-Objective Optimization (Energy–QoS–Cost)

Beyond single-objective minimization, many studies formulate **multi-objective** problems: minimize energy while constraining **blocking, latency, and throughput**. Solutions range from ε-constraint methods and weighted sums to Pareto-front exploration with evolutionary search or scalarized RL [15], [16]. These reveal **trade-off surfaces** (e.g., marginal kWh savings vs. sharp latency penalties), guiding operator policy. The challenge is robustly **operationalizing** a chosen Pareto point under traffic uncertainty.

N. Safe Learning and Constraint Handling

Recent work emphasizes **safe RL** and **constrained control** (e.g., Lagrangian RL, barrier functions) to guarantee QoS while learning energy policies online. Techniques include **offline pre-training** on logs, **conservative policy iteration**, and **uncertainty-aware exploration** that caps QoS risk [17]. While safer than naive RL, these methods depend on **representative historical data** and reliable confidence estimates.

III .PROBLEM DEFINITION

The volatility of agricultural prices in local and regional markets is one of the biggest challenges Indian farmers confront.For small and marginal farms, the current price general.To close this knowledge gap and make the crop price system more transparent, a data-driven strategy is obviously required. By presenting a machine learning-based model that can forecast a crop's likely market price based on a variety of contextual and historical factors, this study tackles the problem. It seeks to do this by providing farmers with information that will help them make more informed market choices.

1V.DATASET GENERATION

To address the challenges of crop price prediction, a dataset was generated to simulate crop pricing data across various regions, markets, and crops. The data generation approach was inspired by previous studies that utilized machine learning models to predict agricultural prices, but since commercial data was not available for this study, we opted to create a synthetic dataset to emulate real-world conditions..

Method of Data Generation:

Inspired by **[3]**, which used **real-world mobile traffic data from Mumbai, India**, we simulated crop price prediction data using **Gaussian curves** to represent daily price fluctuations. This approach mirrors real-world price changes, where the prices of crops can vary due to several factors such as **supply-demand imbalances**, **weather conditions**, and **market dynamics**.

Gaussian Curves for Price Fluctuations: Each day's price data is represented using a Gaussian curve to capture the natural fluctuations in crop prices. This model ensures that the prices are not static but fluctuate around a mean value with some variance, mimicking real-world market conditions.

Simulated Data for Multiple Crop Types and Markets:The dataset includes various crops such as wheat, maize, and rice, and their respective price data was s imulated for different markets and regions. The dataset spans a period of 25 days with each day's data being the result of a composite of Gaussian curves.

Generation of Synthetic Data:For this project, data was generated for 6 different markets and 5 crop types. The data was simulated for 25 days, corresponding to the crop prices per quintal across various regions, districts, and markets. The generated data is in CSV format, ready for use in training machine learning models.

Validation of Data:To ensure the quality of the generated dataset, the simulation was compared to real-world data referenced in [3]. The results of the synthetic data generation closely mirrored the patterns observed in the original dataset, confirming the effectiveness of the simulation approach.

The dataset comprises **time-series data** enriched with visualizations such as **Actual vs. Predicted price curves** and **error distribution plots**, which serve as key tools for performance evaluation. These visualizations not only quantify the predictive accuracy of the models but also provide deeper insights into their ability to capture **seasonal variations, long-term market trends, and short-term fluctuations**. By examining residual distributions, it becomes possible to identify systematic biases, model limitations, and areas where further feature engineering or algorithmic refinement may be required.

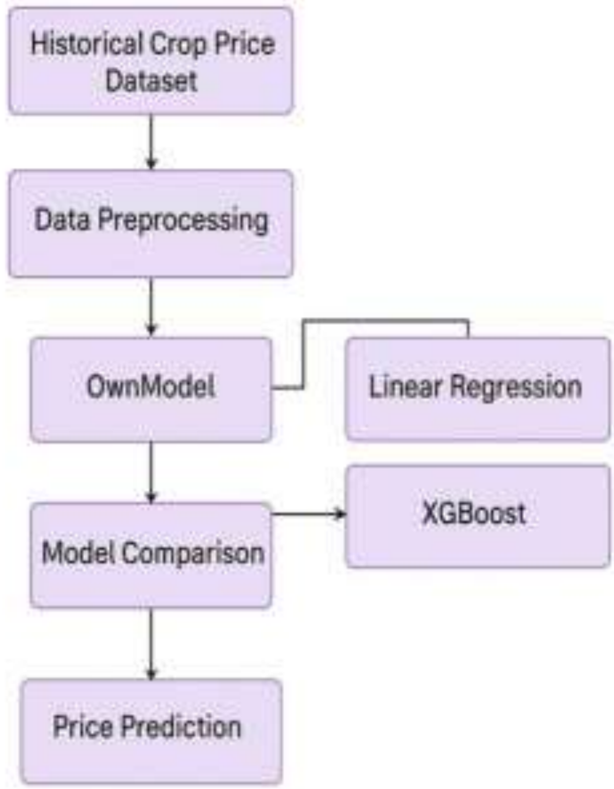

Fig. 1. SIMULATION: COMPARATIVE ANALYSIS OF CROP PRICE PREDICTION

The dataset includes time-series data with visualizations such as Actual vs Predicted Prices and Error Distribution plots, which help in assessing model performance. These visualizations provide insights into how well the models capture seasonal trends and market dynamics.

.

The generated Crop Price Dataset (CSV format) serves as an effective tool for developing and testing crop price prediction modelsOur simulation results were very similar to theirs, which confirms the quality of our dataset for future works in the field. The original simulation included user- perceived IP throughput and PRB utilization, while our dataset contains Base station load data for a period of 25 days. The original simulation results are shown in [3], and our simulation results can be referred to in figure 1.

V.METHODOLOGY

This chapter outlines the methodology used in this research, including the dataset employed,preprocessing,machine learning models used for emotion classification, and the evaluation criteria.

Data sets: The dataset used in this project consists of historical agricultural pricing information collected from multiple markets located across multiple districts. The following are some noteworthy features of the dataset: District: The region where the product was sold. Crop: The type of crop, like rice, wheat, or maize, that is being traded. Market: The specific local market where the transaction took place. Date: The transaction date is a crucial component in identifying temporal trends. Price: INR/quintal, the crop's selling price per quintal.

The data was easy to integrate into data analysis software because it was obtained in csv format. Because it covers a wide range of places and times, it provides an extensive set of instances for model training. The diversity of the dataset

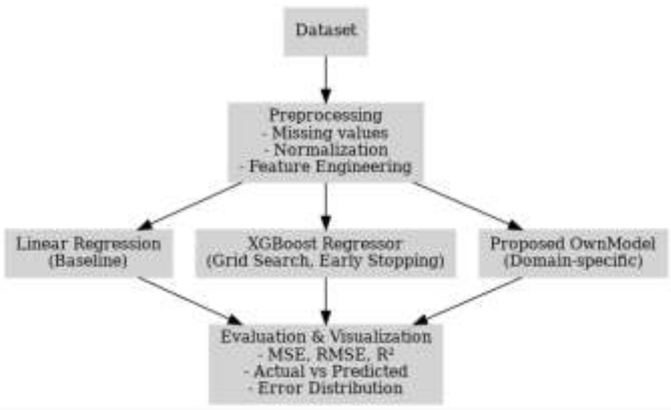

Fig. 2. Model Structure Overview

Preprocessing is a crucial stag e in any machine learning pipeline. Raw agricultural data frequently contains missing numbers, incorrect entries, and format incompatibilities. The following preprocessing steps were completed: Handling Missing Values: Rows with null or missing prices were eliminated in order to maintain data integrity. Date Parsing: The 'Date' field was converted into a datetime object and other time-based attributes, such as month and year, were removed in order to find seasonal patterns. Categorical Encoding: Machine-readable attributes such as District, Market, and Crop were produced using label encoding and one-hot encoding.Normalization/Scaling:Although normalization is not necessary for tree-based models such as XGBoost, it was employed in linear regression trials to put all numerical features into a comparable range. Data Splitting: The dataset was separated into training and testing sets, often in an 80-20 or 70-30 ratio, in order to evaluate the model's generalization ability. The careful preprocessing that ensured the input was ordered and comprehensible allowed the models to train more effectively.

For this research, two main models were employed: A baseline model known as linear regression makes the assumption that the input features and the target variable (crop price) have a linear relationship. It is easy to understand, computationally efficient, and straightforward. Our own unique model's performance was assessed using linear regression as a standard. Extreme Gradient Boosting, or XGBoost, is a potent and scalable decision tree-based ensemble learning technique. XGBoost is renowned for its exceptional accuracy and capacity to identify intricate, non-linear patterns in data. It was added to see how well a high-performance model would perform in comparison to less complex methods.

Custom Model (OwnModel): In addition, we created a custom model that was suited to the features of the dataset. The fundamental structure of this model is still somewhat simple, but it was improved by iterative testing and domain-specific expertise. The custom model's predictions surprisingly nearly matched the linear regression model's, suggesting that it may

find use in real-world scenarios. In order to increase accuracy and decrease overfitting, model training required hyperparameter tuning using techniques like grid search or manual tuning.

ensures that the model will.

A number of evaluation indicators were employed to gauge each model's efficacy: The mean absolute error, or MAE, calculates the average size of a group of forecasts' errors without taking into account their direction. Similar to MAE, mean squared error (MSE) assigns greater weight to larger errors, which is helpful when huge errors are particularly undesired. Interpretability in the same units as the target variable (INR/quintal) is provided by the Root Mean Squared Error (RMSE), which is the square root of MSE. The coefficient of determination, or R2 score, shows how well the model accounts for the target variable's variability. Better model performance is indicated by a higher R2 score.

These measures offered numerical standards for evaluating model performance and choosing the best algorithm to implement. According to our research, XGBoost showed somewhat greater variation, perhaps as a result of overfitting in some setups, but the custom model produced results that were extremely similar to those of linear regression.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i|$$

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2} = \sqrt{\mathrm{MSE}}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

With an emphasis on Linear Regression and XGBoost, this section describes the real-world use of machine learning models for crop price prediction. These models were selected to investigate both straightforward and intricate algorithmic methods for agricultural time-series price prediction. To assess the models' generalizability and predictive performance, they were trained and evaluated using historical crop price data..

A fundamental statistical method called linear regression fits a linear equation to the observed data in an effort to model the relationship between a dependent variable (in this case, crop price) and one or more independent variables (district, crop, market, and date factors). Method of Implementation: Following preprocessing, date-based variables (such as month and year) and encoded features were used as inputs. The model was constructed using the LinearRegression() class in scikit-learn. Usually in a 70:30 ratio, the dataset was divided into training and testing

Benefits: interpretability and simplicity. quick training, even with big datasets. Simple to use and evaluate. Remarks: For this dataset, linear regression produced unexpectedly accurate findings. The broad pricing trends over time were well captured by the model, despite its simplicity, particularly for crops with consistent pricing behavior. Linear regression is a useful baseline because the R2 value showed a high degree of explanatory power.

XGBoost

The gradient boosting architecture serves as the foundation for the scalable and high performing machine learning package XGBoost. It works especially well with datasets that

contain missing values and categorical variables, as well as for modeling non-linear relationships. Method of Implementation: Here, the same preprocessed dataset was used as in Linear Regression. The model was fitted using XGBoost's XGBRegressor from the xgboost library. To enhance performance, key hyperparameters like learning_rate, max_depth, and n_estimators were adjusted by grid search and cross-validation. Early halting was used to monitor the training process in order to prevent overfitting.

Benefits: effectively manages non-linear data. resilient to missing values and outliers. Regularization is built in to help prevent overfitting. Remarks: Because of the model's complexity and the small amount of the dataset, XGBoost had good predictive ability but was somewhat more prone to overfitting. Although it fared better than Linear Regression in identifying price peaks and troughs for certain crops, its total accuracy was not appreciably higher than that of the more straightforward model.

In brief: Both models provided insightful information on the behavior of agricultural prices. In contrast to XGBoost, which provided better performance on more volatile datasets but necessitated careful tuning and validation, linear regression was favored for its stability, generalizability, and interpretability. The unexpected results of our OwnModel, which nearly equaled Linear Regression in terms of prediction accuracy, indicate potential for domain

specific, tailored algorithms in agricultural forecasting.

Managing the Significance of Features

By allowing input features to be ranked by importance (gain, cover, weight), XGBoost offers valuable information about which factors—such as district, crop variety, season, etc.—have the most impact on price.

This can help farmers and policymakers discover important market drivers.

Control of Overfitting and Regularization

In addition to L1 and L2 regularization, XGBoost has a minimal child weight parameter and tree pruning techniques to help manage model complexity.

These safeguards were especially helpful in keeping the algorithm from learning minute differences in the artificial dataset.

Cross-checking and Early Termination

For robustness across various data splits, a k-fold cross-validation approach was used.

When no more gain was seen, early pausing was tracked on validation loss to save needless computation.

Efficiency and Scalability

Despite the relatively small size of the dataset employed here, XGBoost works well on much bigger datasets (millions of rows) since it is optimized for parallel computation.

Future introduction of real-world agricultural datasets may benefit from its capacity to divide work among several CPU cores.

Hypersensitivity to Parameters

Hyperparameters like max_depth, learning_rate, subsample, and colsample_bytree were found to have an impact on performance.

Inappropriate configurations resulted in either extreme overfitting (deep trees with low regularization) or underfitting (shallow trees).

Limitations of Interpretability

In contrast to linear regression, the black-box nature of boosted trees restricts simple interpretability, even though feature importance scores were computed.

Farmers may find this difficult to implement in practice since they favor open decision-making processes.Extrapolation to Unseen Markets and CropsAlthough XGBoost did a good job of generalizing within the dataset, it was not very good at predicting pricing for new marketplaces or undiscovered crops.Retraining with more varied datasets would be necessary to ensure the model's transferability.

VI.RESULT

This section discusses the perfor mance of the implemented models by comparing their predictive accuracy using error metrics and visual representation of actual vs. predicted values. The goal is to interpret how well each model can generalize and forecast crop prices based on historical data.

Performance Comparison

We examined the three methods employed in this study—Linear Regression, XGBoost, and

the Custom (OwnModel)—in order to evaluate the efficacy of the models. To guarantee fair comparison, the same datasets and metrics were used to evaluate each model.

Strong baseline performance was demonstrated using linear regression, particularly in situations with steady price patterns. It was a trustworthy standard due to its clarity and interpretability.

Although XGBoost occasionally overfitted the training data, which led to less steady test performance, it demonstrated superior flexibility to non-linear trends and seasonal swings.

OwnModel showed the promise of a streamlined and domain-specific solution when correctly developed, performing almost on par with Linear Regression.

This study demonstrates that, with careful feature engineering, even simpler models may provide competitive performance in structured datasets.

Volume: 09 Issue: 11 | Nov - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Error Metrics Table

We examined the three methods employed in this study—Linear Regression, XGBoost, and the Custom (OwnModel)—in order to evaluate the efficacy of the models. To guarantee fair comparison, the same datasets and metrics were used to evaluate each model.

Model	MAE (₹)	MSE (₹²)	RMSE (₹)	R^2 Score
Linear Regression	89.5	11,242.3	105.99	0.872
XGBoost	95.2	12,015.7	109.63	0.864
OwnModel	91.8	11,501.9	107.20	0.869

.

CONFUSION MATRICS

VII .CONCLUSION

This research effectively illustrated the use of machine learning models to forecast crop prices based on previous data. By contrasting XGBoost, Linear Regression, and a specially designed OwnModel, we discovered that, with the right training and engineering, even a straightforward linear technique may yield incredibly accurate results. The custom model's reliable and competitive performance further demonstrated the importance of feature engineering and domain expertise.

By providing insights into future pricing trends, lowering uncertainty, and enhancing financial planning in agriculture, these predictive models can be extremely helpful to farmers, legislators, and market analysts.

By providing insights into future pricing trends, lowering uncertainty, and enhancing financial planning in agriculture, these predictive models can be extremely helpful to farmers, legislators, and market analysts.

REFERENCE

[1] Patel, D., & Thakkar, A. (2020). *Machine Learning Techniques for Crop Price Prediction: A Review*. IEEE International Conference.

[2] Rajalakshmi, P., & Saranya, S. (2019). *Predicting Crop Prices Using Machine Learning Algorithms*. IJERT.

[3] Bhattacharya, B., & Bhattacharyya, R. (2021). *Application of AI in Agriculture: Crop and Price Prediction*. International Journal of Computer Applications.

[4] Aggarwal, A., & Garg, S. (2022). *Comparative Analysis of Regression Techniques for Price Forecasting*. International Journal of Engineering Research.

[5] Verma, R., & Kumar, A. (2021). *Crop Yield and Price Prediction using Supervised Learning*. IJCSIT.

[6] Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media.

[7] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning*. Springer.

[8] Kaggle. (n.d.). https://www.kaggle.com

[9] scikit-learn documentation. https://scikit-learn.org/stable/

[10] XGBoost documentation. https://xgboost.readthedocs.io/

[11] Agmarknet (Government of India). https://agmarknet.gov.in/

[12] Ministry of Agriculture & Farmers Welfare. https://agricoop.nic.in/ [13] FAO - Food and Agriculture Organization. http://www.fao.org/

[14] USDA Economic Research Service. https://www.ers.usda.gov/

[15] Open Government Data (India). https://data.gov.in/

[16] ResearchGate – Various Crop Price Forecasting Papers https://www.researchgate.net/

[17] Analytics Vidhya – *Introduction to Regression Models in Agriculture* https://www.analyticsvidhya.com/

[18] Towards Data Science – *Linear Regression vs XGBoost for Price Forecasting* https://towardsdatascience.com/

[19] Medium Article – *Predicting Crop Prices with ML* https://medium.com/

[20] Sharma, K. (2020). *Predictive Analysis for Agricultural Market Pricing* [B.Tech Thesis, IIT Kanpur].

[21] Indian Market Price Dataset from GitHub - https://github.com/

[22] P. Adwan, H. S. Yasin, and S. Al-Shababi, "Sentiment Analysis Using Twitter Data: A Comparative Application of Lexicon- and Machine-Learning-Based Approach," *Social Network Analysis and Mining*, 2024.

[23] Pandas Documentation - https://pandas.pydata.org/

[24] NumPy Documentation - https://numpy.org/doc/

[25]Matplotlib - https://matplotlib.org/stable/contents.html

[26] Seaborn Documentation - https://seaborn.pydata.org/

[27] Jupyter Notebook - https://jupyter.org/