CropForEst – A MACHINE LEARNING BASED CROP YIELD ESTIMATION AND PROFITABILITY ANALYSIS FOR PRECISION AGRICULTURE

Submitted by

RB SUNDARAJAN (2020503549)

HAMILTON SAMIC S (2020503511)

SURIYAA V (2020503550)

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

MADRAS INSTITUTE OF TECHNOLOGY, CHROMEPET

ANNA UNIVERSITY, CHENNAI 600 044

JUNE 2023

ANNA UNIVERSITY

BONAFIDE CERITIFICATE

Certified that this project report titled – “**CropForEst – A MACHINE LEARNING BASED CROP YIELD ESTIMATION AND PROFITABILITY ANALYSIS FOR PRECISION AGRICULTURE**” is the bonafide work of “**SUNDARAJAN RB, HAMILTON SAMIC S, SURIYAA V**” who carried out the project work under my supervision.

SIGNATURE OF HOD

Dr. JAYASHREE P
Professor & Head
Department of Computer Technology
Madras Institute of Technology
Chromepet
Chennai 600 044

SIGNATURE OF SUPERVISOR

Dr. VALLIYAMMAI C
Associate Professor
Department of Computer Technology
Madras Institute of Technology
Chromepet
Chennai 600 044

ABSTRACT

The agriculture industry plays a crucial role in the country's economy, contributing significantly to the GDP and employing a substantial portion of the population. However, accurately predicting crop yield production and crop demand remains a significant challenge. Traditional methods of prediction are time-consuming and prone to errors, resulting in inefficient resource allocation and revenue loss. To address these issues, an intelligent platform that leverages machine learning algorithms is proposed to estimate the most suitable crop for cultivation based on environmental factors and forecast market demand and pricing trends for the identified crop. In evaluating the performance of various Classification models for Crop Prediction. The Voting Classifier, combining multiple models, produced enhanced accuracy, reaching an impressive 0.984. This platform's crop prediction capability can assist farmers and stakeholders in making informed decisions, leading to improved profitability and efficient resource utilization. Additionally, a Stacking Regressor model was employed for price estimation, incorporating the top two models, Random Forest Regressor and XGBoosting Regressor. The Stacking Regressor demonstrated superior performance with an optimized R-Squared score of 0.974, surpassing the individual regression models. This ensemble model significantly improved the accuracy of price estimation, allowing stakeholders to make informed decisions and enhance profitability in the agriculture industry. Overall, the proposed intelligent platform offers a promising solution for accurate crop prediction and price estimation, benefiting farmers and other stakeholders by optimizing resource allocation, improving profitability, and contributing to the sustainable growth of the agriculture sector.

ACKNOWLEDGEMENT

We take this humble opportunity to thank Dean, MIT Campus Anna University, **Dr. J. Prakash** for providing all the lab facilities in pursuit of this project.

Undertaking this project has helped us learn a lot, and we would like to express our sincere gratitude towards our supervisor **Dr. Valliyammai C,** Associate Professor, Department of Computer Technology, MIT, Anna University, whose kind guidance, and proper directions helped shape this project perfectly. The timely feedback from the supervisor was much instrumental in successful completion of the project work carried out. Constant encouragement to implement the project to the fullest of our potential led us through, during the difficult phase of the project, and assisted in the successful completion of the proposed project work.

We acknowledge the efforts and feedback of the panel members **Dr. Varalakshmi P, Dr. B Thanasekar, Dr. Y Nancy Jane,** in reviewing our work, providing constant valuable comments, and encouraging us to view the different aspects of the project in successful implementation of the project.

We thank all the teaching and non-teaching members of the Department of Computer Technology, MIT, Anna University, seniors, juniors and friends whose appreciable help, either directly or indirectly helped in the smooth completion of the Creative and Innovative Project in a limited time frame and made the demanding time joyful but always efficient.

Sundarajan RB
Hamilton Samic S
Suriyaa V

LIST OF TABLES

LIST OF FIGURES

CHAPTER – 1

INTRODUCTION

1.1 MACHINE LEARNING

Machine learning (ML) is a field of inquiry devoted to understanding and building methods that 'learn', that is, methods that leverage data to improve performance on some set of tasks. It is seen as a part of artificial intelligence. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision, where it is difficult to develop conventional algorithms to perform the needed tasks. Machine learning is closely related to computational statistics, which focuses on making predictions using computers, but not all machine learning is statistical learning. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Some implementations of machine learning use data and neural networks in a way that mimics the working of a biological brain.

1.2 AGRICULTURE

The agriculture industry in India is a crucial sector that contributes significantly to the country's economy. With a vast population dependent on farming for their livelihood, agriculture plays a vital role in ensuring food security and rural development. However, the industry faces numerous challenges, including unpredictable monsoons, water scarcity, and the impacts of climate change, which pose risks to crop yields and make farming a risky endeavour. Additionally, small and marginal farmers often struggle with limited access to credit, outdated farming techniques, and inadequate infrastructure. Inefficient supply chains and post-harvest losses further hinder the sector's growth. Despite these challenges, the agriculture industry in India is witnessing positive transformations through the integration of crop prediction and price estimation. By adopting machine learning algorithms and advanced analytics, farmers now have access to valuable tools that enhance their decision-making processes. Crop prediction models utilize historical and real-time data, including weather patterns, soil conditions, and previous crop yields, to forecast future production levels. This empowers farmers to plan cultivation practices, allocate resources efficiently, and manage risks associated with unpredictable climate conditions.

1.3 PROBLEM STATEMENT

The agriculture industry faces a significant challenge in predicting crop yield production and crop demand accurately. The traditional methods of predicting crop yield and demand are time-consuming and can be prone to errors, leading to inefficient use of resources and revenue loss. Therefore, There is a immediate need for predicting the most suitable crop based on various influential environmental factors and forecasting the profitability of the crops.

1.4 OBJECTIVE

To create an intelligent platform that utilizes machine learning to determine the optimal crop for cultivation in different regions by analysing environmental conditions. Additionally, prediction of market demand and pricing trends for the identified crop is to be implemented, with the objective of enhancing profitability for farmers and other stakeholders in the agriculture industry.

CHAPTER – 2

LITERATURE SURVEY

This literature survey investigates several techniques and models for predicting crop yield. One notable approach combines RNN-based feature processing with a Deep Recurrent Q-Network (DRQN) model. By utilizing major climatic factors and soil parameters from the dataset extracted from the Indian Meteorological Department's portal, the DRQN model outperforms other models, including ANN and BAN, with a remarkable accuracy of 94%. The comparison was conducted using evaluation metrics such as error and variance score, and the probability density of actual and predicted yields was also measured [1]. In selecting estimation methods, researchers consider the trade-off between performance in terms of target parameters, interpretability of results, and computational time. Multiple linear regression, random forest, and neural networks are commonly employed techniques. Additionally, Gaussian Linear regression aims to predict crop yield by approximating the target output as a probability distribution function resulting in higher insight into yield prediction from an unknown distribution [2]. To enhance crop yield prediction, a two-stage process is proposed. In the first stage, a Particle Swarm Optimization (PSO) algorithm optimizes the input weights and biases of an Extreme Learning Machine (EML) algorithm, which is a feedforward network with a single hidden layer. In the second stage, the optimized EML model predicts crop yield based on input features. Improved accuracy is achieved compared to traditional methods that rely on one or two spectral bands, leading to enhanced crop yield prediction [3]. Satellite images are employed to extract spectral information and vegetation indices, enabling crop classification. Random Forest (RF) and Support Vector Machines (SVM) classifiers are utilized for yield estimation. These accurate and efficient monitoring techniques have the potential to aid decision-making for agricultural practices and ensure food security in regions with similar characteristics to the Great Rift Valley [4]. The use of 3D Convolutional Neural Networks (CNNs) is explored to extract hierarchical features from multispectral images, which are then inputted into an Attention-based Convolutional Long Short-Term Memory (AC-LSTM) model. This model captures temporal dependencies using attention mechanisms at each time step. This model surpasses conventional methods that rely on one or two spectral bands, offering significant potential to improving crop yield prediction accuracy [5].

Another approach involves using 3D-CNNs to extract features and a Multikernel Gaussian Process (MKGP), a non-parametric regression model, to capture complex relationships between multispectral data and crop yield. This approach achieves a strong correlation between predicted and actual crop yield, with an R-squared value of 0.7. Accurate and efficient crop yield prediction with this method can contribute to better decision-making for agricultural practices and ensure food security [6]. To improve the precision, reliability, and stability of crop yield estimation, a coupled CASA-WOFOST integrated model is proposed. Data assimilation using Ensemble Kalman Filter (EnKF) is performed in two steps: forecasting and updating. Performance evaluation metrics, such as R2, RMSE, NRMSE, NSE, absolute error, and relative error, indicate that the coupled model outperforms the individual CASA and WOFOST models across various evaluation metrics [7]. A novel feature selection approach called Modified Recursive Feature Elimination is introduced. This method helps select and rank features, while the bagging technique accurately predicts suitable crops based on given conditions. Precision metrics, such as accuracy and F1 score, are employed to evaluate the performance of the model. The dataset is pre-processed to remove redundant data, and the classifier is trained using training samples, with unknown samples used for validation. Eliminating redundant fields is a breakthrough leading to improved prediction accuracy [8]. FarmEasy is a platform that revolutionizes crop management and marketing by utilizing machine learning algorithms. It provides farmers with valuable insights and predictions regarding crop yields, prices, and weekly guidelines. Real-time data, including satellite images, weather forecasts, and market prices, is used to train and optimize the machine learning models. The platform incorporates Random Forest and Support Vector Regression as its foundational models. By leveraging these models, FarmEasy enables farmers to make informed decisions about crucial aspects of their agricultural practices. It provides guidance on optimal planting and harvesting times, as well as suggestions for crop sales. Additionally, the platform assists distributors in making informed decisions about which crops to buy and sell [9]. An integrated R-Hadoop machine learning model and Map-Reduce framework are employed to combine Artificial Neural Networks (ANN) and Multiple Regression Analysis (MRA) for crop yield prediction. Real-time data from the Indian agricultural sector is utilized in this model, showcasing its effectiveness in achieving high prediction accuracy. By leveraging the power of R-Hadoop and the Map-Reduce framework, the model can handle large-scale datasets and perform distributed computing, enabling efficient and scalable analysis. The integration of ANN and MRA allows for a comprehensive approach that considers various factors such as population, rainfall, and temperature to predict the demand for agricultural products. The utilization of real-time data ensures that the model stays updated with the latest information, enhancing its predictive capabilities. The high prediction accuracy demonstrated by the model, with a mean absolute percentage error of 2.69% for the year 2016-17, highlights its effectiveness in capturing the complex relationships between input variables and crop yield [10].

CHAPTER 3

PROPOSED WORK

3.1 ARCHITECTURE DIAGRAM

Various traditional methods have been tried out to predict the most suitable crop for a set of given conditions but most have failed in accuracy. In order to mitigate this issue, and make sure the farmer has the maximum profit from the crop he/she cultivates, an accurate prediction and estimation model is required.

The proposed system uses various classification and regression models for crop prediction using conditions and price estimation respectively. The Nitrogen, Phosphorus, Potassium levels in the soil and the rainfall (in mm) and the name of the district is taken as input value from the farmer. Using the name of the district, the temperature and humidity of the place is taken from openweathermap API. This is fed as input to the prediction model. The prediction model predicts a single crop as the output. The name of the crop along with the district name is now fed as input to the price estimation model which generates a price list for the next 12 months to decide the right time to start cultivation.

Fig 3.1 defines the architectural diagram of the proposed system, where the above-mentioned flow is evident. The app interface is used to get the input values from the user and these values are in turn sequentially fed as input to both the models and the corresponding output is tabulated and printed to the user.

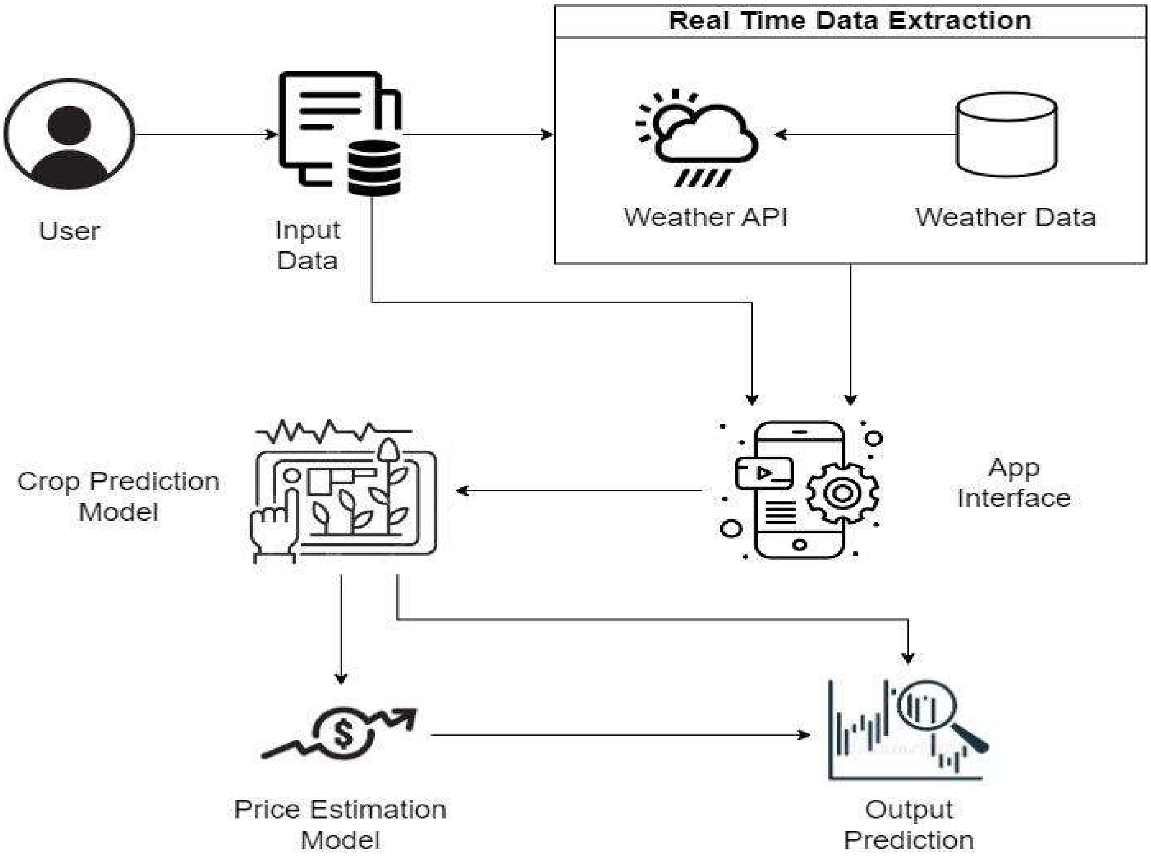


Fig 3.1 Proposed Crop Prediction and Price Estimation model

3.2 CROP PREDICTION MODEL

This module involves predicting the details of a crop which grows in a particular region in Tamil Nadu. Prediction is done using Nitrogen, Phosphorous, Potassium values along with Humidity, Average Rainfall, Temperature as the independent variables obtained using datasets extracted and synthesized from https://data.gov.in, www.indiastat.com and OpenWeatherMap API. This data is fed into a Crop prediction model which is essentially a voting classification ensembler. The data obtained was cleaned to remove duplicates, irrelevant data, and handle missing values, rounding of values, Removing outliers etc. Normalization or scaling was done to improve the model's convergence and performance. The classification models used are KNN Classifier, Support Vector Machine, Decision Tree Classifier and Naïve-Bayes Classifier.

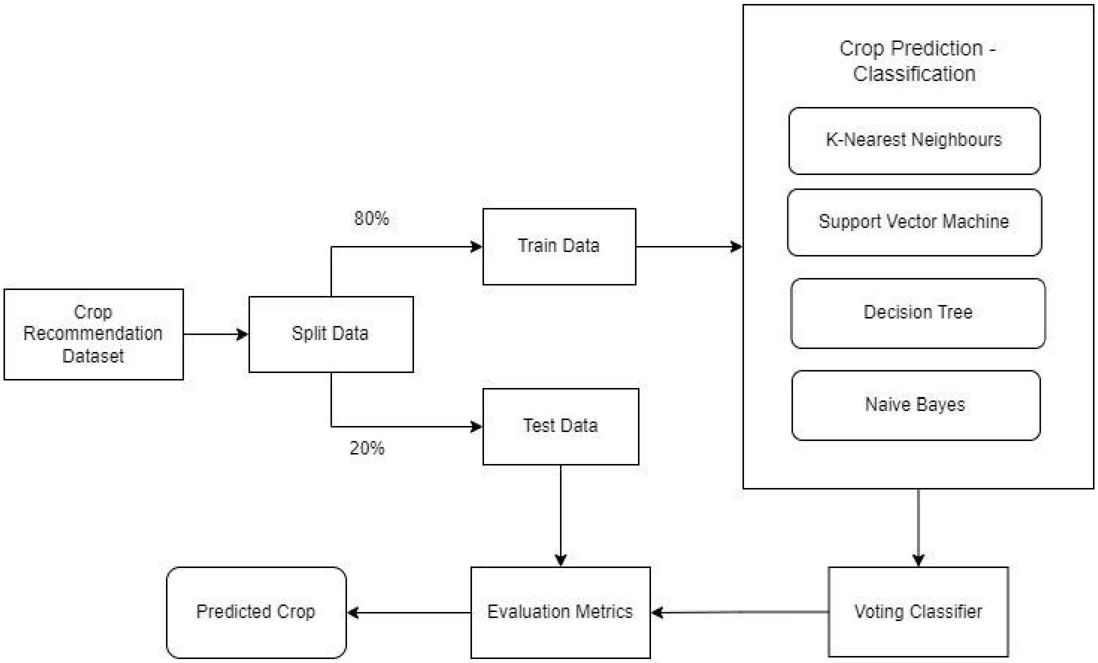


Fig 3.2 Workflow of Crop Prediction

3.3 CROP PRICE ESTIMATION MODEL

This module involves predicting the prices of the crop for the next 12 months in that district. Prediction is done using the crop name and the district name as the independent variable obtained using datasets extracted and synthesised from https://data.gov.in and www.indiastat.com. This data is fed into the Price Estimation model which is essentially a stacking regressor ensembler. The data obtained was cleaned to remove duplicates, irrelevant data and handle the missing values, rounding of values, removing outliers etc. Encoding was done using Label

Encoder for the labelled classes. The regression model used here are XGBoost regression, Random Forest regression, Gradient boosting.

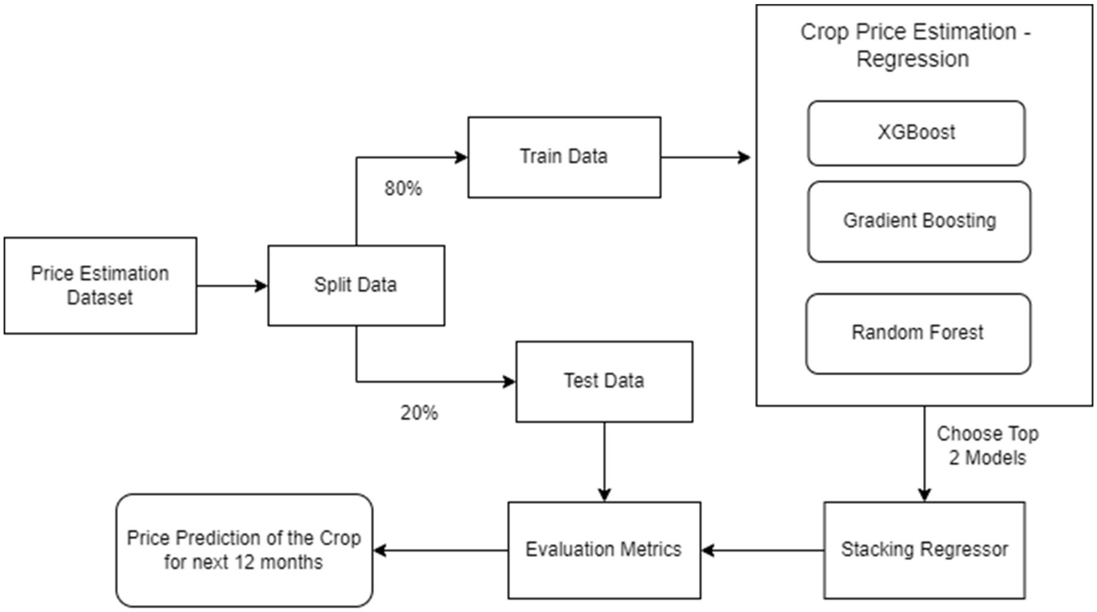


Fig 3.3 Workflow for Price Estimation

3.4 WEB APP INTERFACE

The frontend of the web app is designed to make it easy for farmers to input their data and receive predictions about the most suitable crop to grow in their specific location. The app is built using a combination of tools including Python Virtual Environment, Flask, HTML/CSS, and Bootstrap. These tools are essential in creating a user-friendly interface that is both highly customizable and visually appealing. The input page of the app is designed to collect important information from the farmer, such as Nitrogen, Phosphorous, Potassium (NPK), District, Rainfall, and a submit button is present to submit the input form. This input data is then used by the trained ML models running in the backend to make predictions about the most suitable crop for the farmer to grow. Once the farmer submits their input data, the app takes them to the prediction page, which has two parts. The first part is a price prediction table for the predicted crop. This table displays the predicted prices of the crop over the next 12 months. The second part is a chart or graph that displays the predicted prices of the crop over the next 12 months in a more visual manner. The use of these visualizations allows farmers to easily interpret the data and make informed decisions about when to sell their crops.

CHAPTER 4

IMPLEMENTATION

4.1 TOOLS REQUIRED

4.1.1 OPENWEATHERMAP API

The OpenWeatherMap API was utilized as a crucial tool for retrieving weather information required for predicting suitable crops. The API provides real-time access to current weather data for any location on Earth, including temperature and humidity, which are crucial factors in determining the suitability of crops. By feeding the district information into the API, the necessary weather data was retrieved, processed, and used in the prediction model to determine the most appropriate crop for the given district.

4.1.2 VISUAL STUDIO CODE

In order to add dynamic and interactive behaviour to the application, scripting in python is required. The scripts are developed in Visual Studio code. For the development of this app, the Flask framework has been utilized to design the user interface and integrate it with the trained machine learning models, which are hosted in the backend. Flask provides a simple yet powerful web framework that allows for rapid prototyping and deployment of web applications. The app is being developed in Visual Studio Code, which provides a comprehensive set of tools for writing, testing, and debugging code.

4.1.3 GOOGLE COLLABORATORY

Google Colab is an online platform provided by Google that allows users to write, run, and share code in a Jupyter Notebook environment. The platform offers free access to computing resources such as CPU, GPU, and TPU for machine learning and data analysis tasks. Colab provides a cloud-based environment. Additionally, Colab provides seamless integration with other Google services such as Google Drive. Overall, Colab is a convenient and powerful tool for data science and machine learning tasks.

4.1.4 SCIKIT – LEARN

Scikit-learn is a popular open-source machine learning library in Python. It is designed to provide simple and efficient tools for machine learning and data analysis. Scikit-learn includes a wide range of algorithms for classification, regression, clustering, and dimensionality reduction, as well as tools for pre-processing, model selection, and evaluation. The library is built on top of NumPy, SciPy, and matplotlib, and is widely used by data scientists and machine learning practitioners for building and deploying machine learning models in production. Scikit-learn is well-documented, easy to use, and has a large and active community, making it a great choice for both beginners and experienced users.

4.2 CROP PREDICTION

This module performs the prediction of the most suitable crop based on the input parameters and climatic conditions. Final output of this module is a single string, which is the name of the most suitable crop for the given climatic and soil conditions. The below algorithm explains how the model predicts the crop name. Fig 4.1 shows the district wise crop distribution

PROCEDURE

1. Start
2. Load the dataset into the Pandas dataframe.
3. Create a feature matrix 'X' by selecting the columns Temperature (degree Celsius). Humidity (%), rainfall (mm), n, p, k ratios and target vector y from the dataframe.
4. Split the dataset into training and testing sets using the train test split method. In this case the test size is set to 20% and the random state is set to 22 for reproducibility.
5. Instantiate four different classifiers – Gaussian Naïve Bayes, Support Vector Machine with linear kernel, KNN and Decision tree.
6. Hyperparameter tuning is done on all 4 models separately using Grid Search CV from Scikit learn, with a range of values for the following parameters: var-smoothing for Gaussian Naïve Bayes, C and Kernel for support Vector classifier, n – neighbours and p for KNN classifier, max – depth, min-samples split and max features.
7. The Gaussian Naive-Bayes model is trained on the training set with the smoothing parameter, var_smoothing=0.0004.
8. The Support Vector classifier model is trained on the training set with the regularization parameter C=0.05 and kernel='linear'.
9. The KNN classifier model is trained on the training set with the no. of neighbors n_neighbors=25 and p=5.
10. The Decision Tree classifier model is trained on the training set with max_depth=6, min_samples_split=20, max_features=2.
11. Create a Voting Classifier model that uses the four classifiers' predictions and makes final prediction based on the majority votes.
12. Fit the training data into the Voting Classifier model and predict target label (suitable crop) for the given test data sample.

13. Calculate and display the overall accuracy, precision, recall, and f1-score metrics for the classification model by extracting the relevant values from the classification report and confusion matrix generated.
14. The final output will be the predicted crop for the example input sample, along with the overall performance metrics for the classification model.
15. The predicted crop is then fed into the next Crop Price Estimation model.
16. End

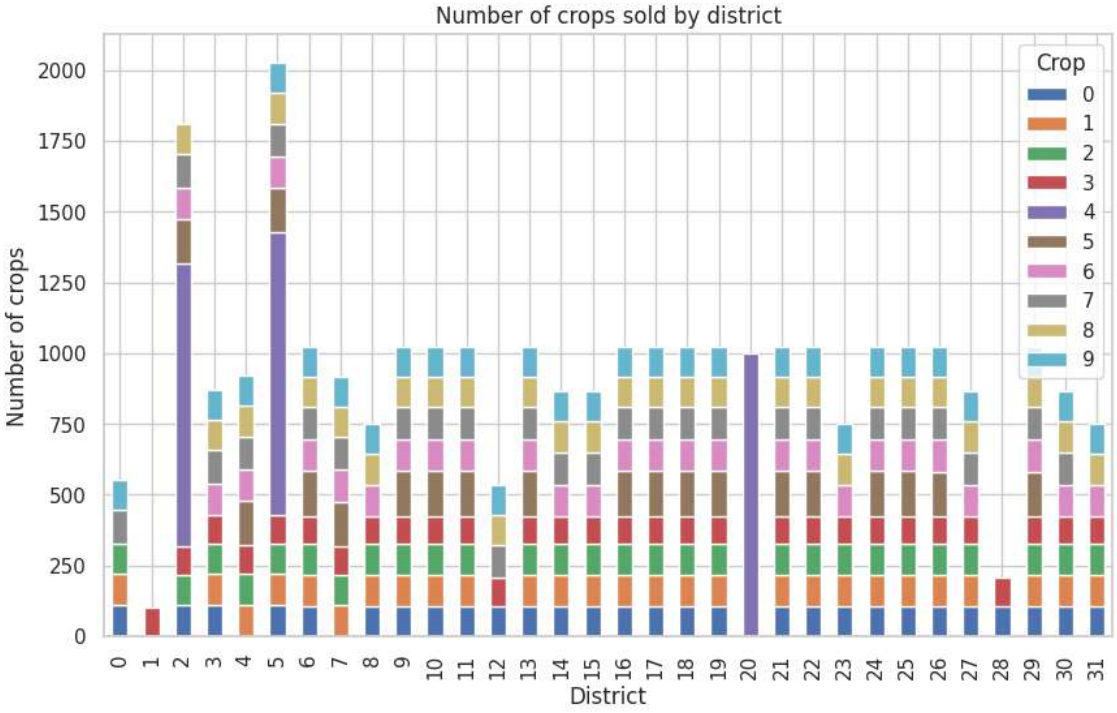


Fig 4.1 District wise crops distribution

4.3 CROP PRICE ESTIMATION

This module performs the prediction of prices of the crop for the next 12 months in the district. The output for this model is a list of dictionaries each containing the month names and the crop prices. The below algorithm explains how the model predicts the prices.

PROCEDURE

1. Start
2. Load the dataset into a Pandas dataframe.
3. The predicted Crop name is obtained from the previous prediction model and this data is now fed into the crop price estimation model along with other details.
4. The dataset which is loaded contains information about crop prices for various districts, month and year mapped to different crops.
5. Data obtained is pre-processed by encoding the categorical variables using LabelEncoder, converting the date column to datetime format, extracting the month and year from the date column, and dropping the date column.
6. Dataset is split into training and testing sets.
7. Impute the missing values in the training and testing sets using SimpleImputer.
8. Standardize the training and testing sets using StandardScaler to bring all features to a common scale.
9. An Ensemble model of 2 different regressor models is to be done.
10. Now a RandomForestRegressor model is trained on the training set with n_estimators=200, max_depth=12, and random_state=20.
11. Hyperparameter tuning is done on the XGBRegressor model using GridSearchCV from Scikit-learn, with a range of values for n_estimators, max_depth, learning_rate, and reg_lambda.
12. Train an XGBRegressor model with the best hyperparameters obtained from GridSearchCV on the training set. The best parameters found using GridSearchCV was best parameters: {'learning_rate': 0.1, 'max_depth': 7, 'n_estimators': 150, 'reg_lambda': 1.0}

13. Now the RandomForestRegressor and XGBRegressor are combined into a stacking ensemble model using StackingRegressor from Scikit-learn, with Linear Regression as the final estimator.
14. The stacking ensemble model is trained on the training set.
15. Now the Crop Prices are predicted for the next 12 months using the stacking ensemble model and displayed.
16. End

4.4 WEB APP INTERFACE

The input page of the app is designed to collect important information from the farmer, such as Nitrogen, Phosphorous, Potassium (NPK), District, Rainfall, and a submit button is present to submit the input form. This input data is then used by the trained ML models running in the backend to make predictions about the most suitable crop for the farmer to grow. Once the farmer submits their input data, the app takes them to the prediction page, which has two parts. The first part is a price prediction table for the predicted crop. This table displays the predicted prices of the crop over the next 12 months. The second part is a chart or graph that displays the predicted prices of the crop over the next 12 months in a more visual manner. The use of these visualizations allows farmers to easily interpret the data and make informed decisions about when to sell their crops.

4.5 PSEUDOCODE FOR CROP PREDICTION AND ESTIMATION

Input : n , p , k , district name , temperature , humidity and rainfall values

Output : predicted_crop , crop_prices

```
temperature, humidity ← get_temperature_and_humidity (district_name)
//using OpenWeatherMap API
```

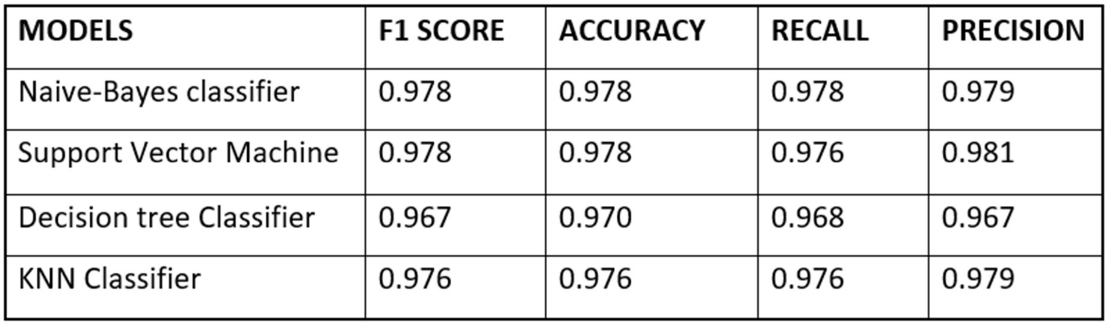
```
predicted_crop ← Crop_Recommendation_Model (n, p, k, temp, humidity, rainfall)
predicted_prices ← Crop_Estimation_Model (predicted_crop, district_name)

Function Crop_Recommendation_Model (n, p, k, temperature, humidity, rainfall) :
  Prepare_Dataset () :
    Preprocess dataset after extraction from official source
    X ← Select columns 'Temperature (°C)', 'Humidity (%)', 'Rainfall (mm)', 'N (ratio)', 'P (ratio)', and 'K (ratio)' from dataset;
    y ← Select target vector 'Crop';
    Split the dataset into training and testing sets using train_test_split () with test size = 20%;
  end
Hyperparameter_Tuning () :
    param_grid ← {Set range of values for 'var_smoothing', 'C', 'kernel', 'n_neighbors', 'p', 'max_depth', 'min_samples_split' and 'max_features'};
    Perform GridSearchCV on Gaussian Naive Bayes with tuned hyperparameters (var_smoothing);
    Perform GridSearchCV on Support Vector Machine with parameters svm_params (C, kernel)};
    Perform GridSearchCV on K-Nearest Neighbours with parameters knn_params (n_neighbors, p);
    Perform GridSearchCV on Decision Tree with parameters dt_params (max_depth, min_samples_split, max_features);
  end
Train_Classifiers () :
    gnb_classifier ← Train the Gaussian Naive-Bayes classifier model, GaussianNB (gnb_params);
    svm_classifier ← Train the Support Vector Machine, SVC (svm_params);
    knn_classifier ← Train the K-Nearest Neighbours classifier model, KNeighborsClassifier (knn_params);
    dt_classifier ← Train the Decision Tree classifier model, DecisionTreeClassifier (dt_params);
  end
Predict_Voting_Classifier () :
    estimators ← List of all 4 trained Crop Prediction classifiers - gnb_classifier, svm_classifier, knn_classifier and dt_classifier;
    Instantiate Voting Classifier model that uses 'estimators' and makes final prediction based on the majority votes;
    Fit the training data into the Voting Classifier model using (X_train, y_train);
    predicted_crop ← Predicted target label (suitable crop) for the given test data sample using predict ();
  end
Calculate_Performance_Metrics () :
    Generate classification report for the voting classifier using (y_test, y_pred)
    Generate confusion matrix for the voting classifier using (y_test, y_pred)
    accuracy, precision, recall, f1_score ← {Calculate overall accuracy, precision, recall and f1-score metrics using classification report and confusion matrix};
  end
end

Function Crop_Price_Estimation_Model (predicted crop, district name)

  Load_Dataset () :
    Load the dataset from official source into a Pandas data frame;
    crop_name ← Predicted crop name from Crop Prediction model;
    input_data ← Concatenate crop name with other input features;
  end

  Preprocess_Data (dataset) :
    Encode categorical variables using LabelEncoder;
    Convert to datetime format;
    dataset['Month'] ← Extract month from Date;
    dataset['Year'] ← Extract year from Date;
  end

  Split_Data (dataset) :
    X ← Select columns 'District', 'Crop', 'Month' and 'Year' from dataset;
    y ← Select target vector 'Crop Price (Rs per quintal)';
    Split the dataset into training and testing sets using train_test_split method with test size = 20%;
  end

  Transform_Data (X_train, X_test) :
    Initialize SimpleImputer with strategy = 'mean';
    Impute missing values in X_train and X_test using SimpleImputer;
    Initialize StandardScaler;
    Standardize X_train and X_test using StandardScaler;
   End

Ensemble_Model (X_train, y_train) :
    Train the Random Forest Regressor model, RandomForestRegressor (n_estimators, max_depth);
    params ← Set range of values for n_estimators, max_depth, learning_rate and reg_lambda;
    best_params ← Get best hyperparameters from GridSearchCV using params;
    Train the XGBoost Regressor model, XGBRegressor (best_params);
    Initialize StackingRegressor with rf_regressor, xgb_regressor and LinearRegression as final estimator;
    Train the stacking ensemble model on X_train and y_train;
    crop_prices ← {Predict the crop prices for next 12 months using 'stacking_regressor' and X_test};
  end
end
```

CHAPTER 5

RESULTS AND ANALYSIS

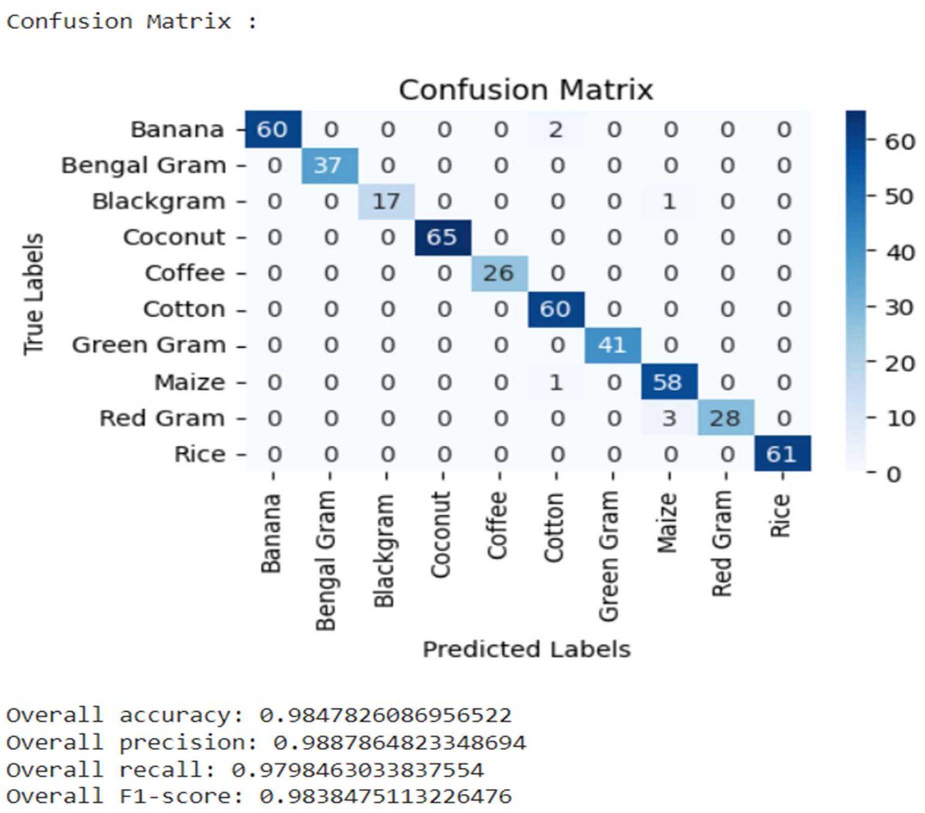
The proposed crop prediction and estimation models are tested against various scenarios and the results have been documented. For crop prediction, the model is based on a Voting Classifier with four models – Naïve-bayes classifier, Support Vector Machine, Decision tree classifier, KNN classifier. Table 5.1 shows the accuracy metrics of the four different models that the voting classifier is using.

Table 5.1 CROP PREDICTION PERFORMANCE METRICS



MODELS	F1 SCORE	ACCURACY	RECALL	PRECISION
Naive-Bayes classifier	0.978	0.978	0.978	0.979
Support Vector Machine	0.978	0.978	0.976	0.981
Decision tree Classifier	0.967	0.970	0.968	0.967
KNN Classifier	0.976	0.976	0.976	0.979

The performance of different models was evaluated through voting, and the Support Vector Machine (SVM) Classifier achieved the highest precision. This is illustrated in Figure 5.1, which displays the confusion matrix depicting the number of false positives, false negatives, true positives, and true negatives. The matrix compares the predicted crop labels against the true crop labels. The results indicate that the Voting Classifier, after combining multiple models, selects the most favourable model with the highest number of votes to accurately predict the appropriate crop for cultivation. It has best accuracy of 0.984 which is a great improvement compared to the individual models. This demonstrates the effectiveness of the SVM Classifier in determining the optimal crop choice based on the given data.



```
Confusion Matrix :
```

Confusion Matrix

True Labels \ Predicted Labels	Banana	Bengal Gram	Blackgram	Coconut	Coffee	Cotton	Green Gram	Maize	Red Gram	Rice
Banana	60	0	0	0	0	2	0	0	0	0
Bengal Gram	0	37	0	0	0	0	0	0	0	0
Blackgram	0	0	17	0	0	0	0	1	0	0
Coconut	0	0	0	65	0	0	0	0	0	0
Coffee	0	0	0	0	26	0	0	0	0	0
Cotton	0	0	0	0	0	60	0	0	0	0
Green Gram	0	0	0	0	0	0	41	0	0	0
Maize	0	0	0	0	0	1	0	58	0	0
Red Gram	0	0	0	0	0	0	0	3	28	0
Rice	0	0	0	0	0	0	0	0	0	61

```
Overall accuracy: 0.9847826086956522
Overall precision: 0.9887864823348694
Overall recall: 0.9798463033837554
Overall F1-score: 0.9838475113226476
```

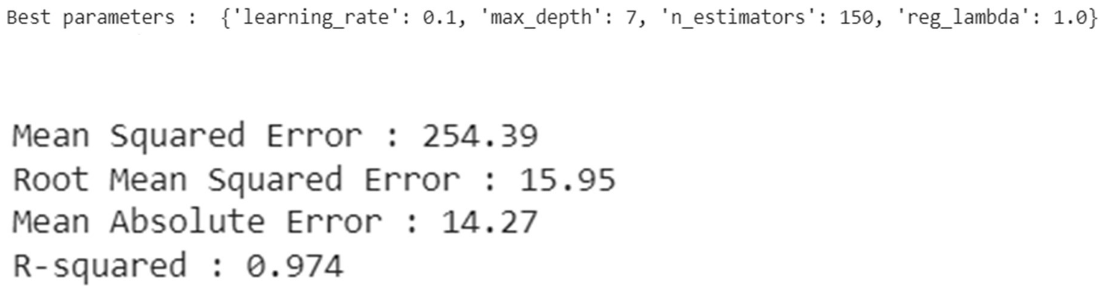
Fig 5.1 Voting classifier Confusion Matrix

Table 5.2 PRICE ESTIMATION PERFORMANCE METRICS

MODELS	MSE	RMSE	MAE	R2 score
Random Forest Regressor	380.81	19.51	16.52	0.964
XG Boosting Regressor	357.61	18.91	16.10	0.977
Gradient Boosting Regressor	366.81	19.15	16.28	0.975

Table 5.2 shows the performance of the different regression models which are trained for Crop Price Estimation. Mean Square Error, Root Mean Square Error, Mean Absolute Error and R2 Score are presented as Performance metrics.

The figure showcases the chosen optimal parameters and displays the corresponding error values. By analysing these metrics, the accuracy and effectiveness of the models in estimating prices is noted. Notably, it is evident from the figure that the stacking regressor, which leverages the diversity of base models to capture different aspects of the data, demonstrates high efficiency. This suggests that the stacking regressor successfully combines the strengths of multiple models to produce robust and reliable price estimations, enhancing the overall performance of the price estimation system. The best parameters using hyper parameter tuning performed are found and the model is trained.



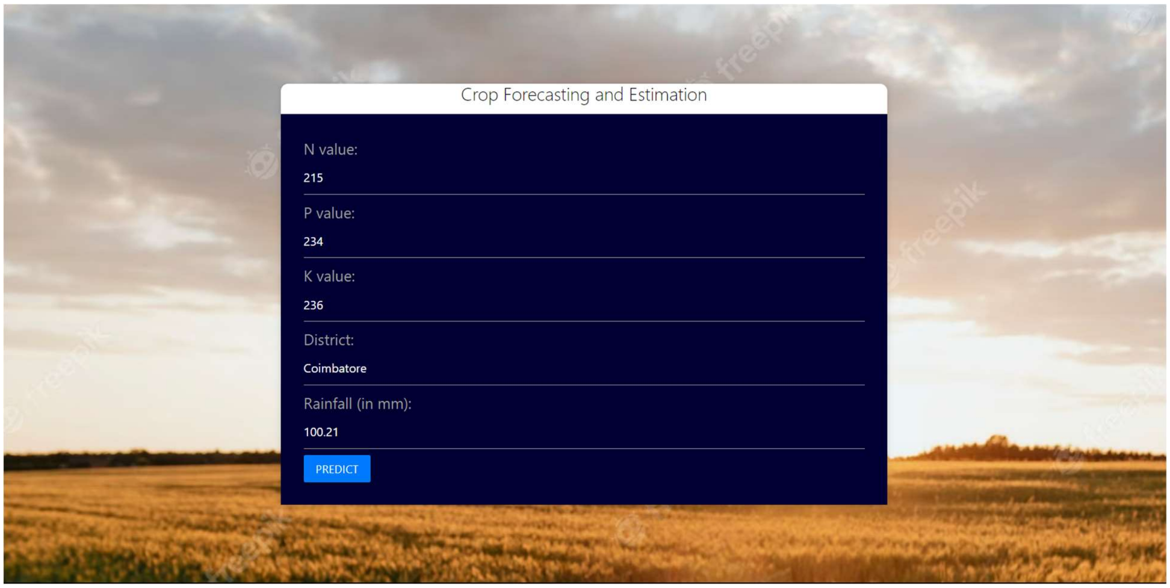
```
Best parameters :  {'learning_rate': 0.1, 'max_depth': 7, 'n_estimators': 150, 'reg_lambda': 1.0}
```

```
Mean Squared Error : 254.39
Root Mean Squared Error : 15.95
Mean Absolute Error : 14.27
R-squared : 0.974
```

Fig 5.2 Stacking regressor Evaluation metrics

For price estimation, the model is based on a Stacking Regressor with the top 2 models - Random Forest Regressor and XGBoosting Regressor. Fig 5.2 shows that the Stacking Regressor has an optimized R-Squared Score of 0.974 which is higher compared to the individual Regression models. Thus, implying this ensemble model fits the input data much better. Stacking Regressor is a machine learning ensemble method that combines multiple regression models to improve prediction accuracy. It operates by training several base regression models on the same dataset and then using a meta-regressor to learn from the predictions of these base models. The meta-regressor takes the outputs of the base models as input features and produces the final prediction. Stacking Regressor leverages the diversity of the base models to capture different aspects of the data and reduce individual model biases. By combining the strengths of multiple models, it can often achieve better performance than any individual model. Stacking Regressor is a flexible and powerful tool for regression tasks, offering improved prediction accuracy and robustness.

Fig 5.3 shows the form in the application interface which takes in the N, P, K ratios, the name of the district and the rainfall (in mm) as input from the user. The form has all fields as required.



Crop Forecasting and Estimation

N value:
215

P value:
234

K value:
236

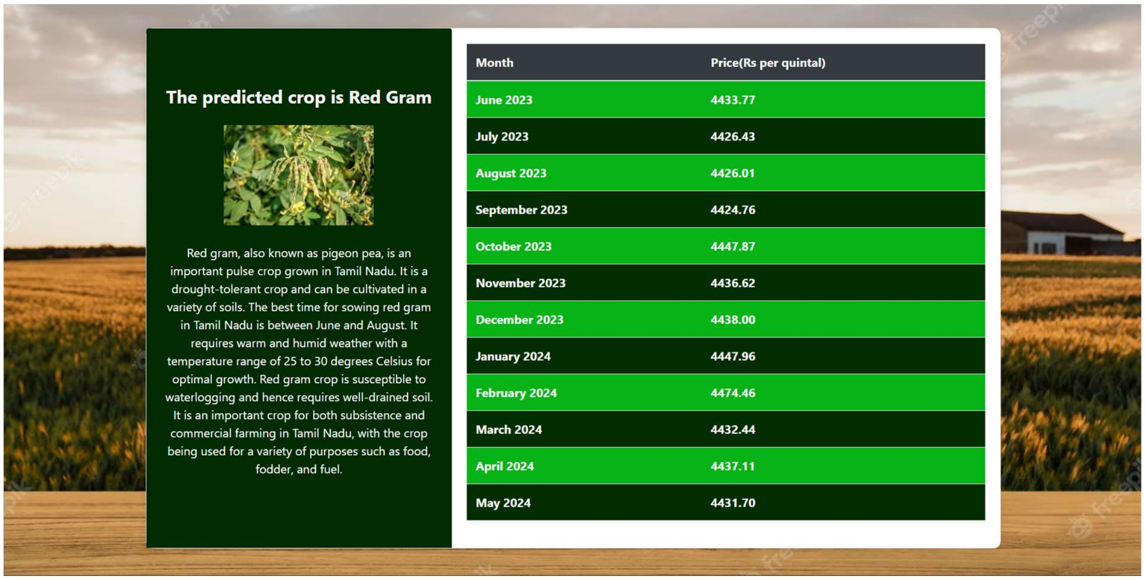
District:
Coimbatore

Rainfall (in mm):
100.21

Fig 5.3 User Input page

The data from the application is passed into the first model – crop prediction model. This model predicts the most suitable crop for the given climatic and soil conditions. This value along with the district name is fed as input to the price estimation model which now predicts the crop prices for the next 12 months.

Fig 5.4 shows the crop and the corresponding list of prices in tabular format. The web application design is done using html, CSS and flask UI and the trained model is imported in the web UI as a .pkl file



The predicted crop is Red Gram

Red gram, also known as pigeon pea, is an important pulse crop grown in Tamil Nadu. It is a drought-tolerant crop and can be cultivated in a variety of soils. The best time for sowing red gram in Tamil Nadu is between June and August. It requires warm and humid weather with a temperature range of 25 to 30 degrees Celsius for optimal growth. Red gram crop is susceptible to waterlogging and hence requires well-drained soil. It is an important crop for both subsistence and commercial farming in Tamil Nadu, with the crop being used for a variety of purposes such as food, fodder, and fuel.

Month	Price(Rs per quintal)
June 2023	4433.77
July 2023	4426.43
August 2023	4426.01
September 2023	4424.76
October 2023	4447.87
November 2023	4436.62
December 2023	4438.00
January 2024	4447.96
February 2024	4474.46
March 2024	4432.44
April 2024	4437.11
May 2024	4431.70

Fig 5.4 Predicted crop with price

The UI further shows the price trend as a graph for the farmer to better understand the best time for cultivation.

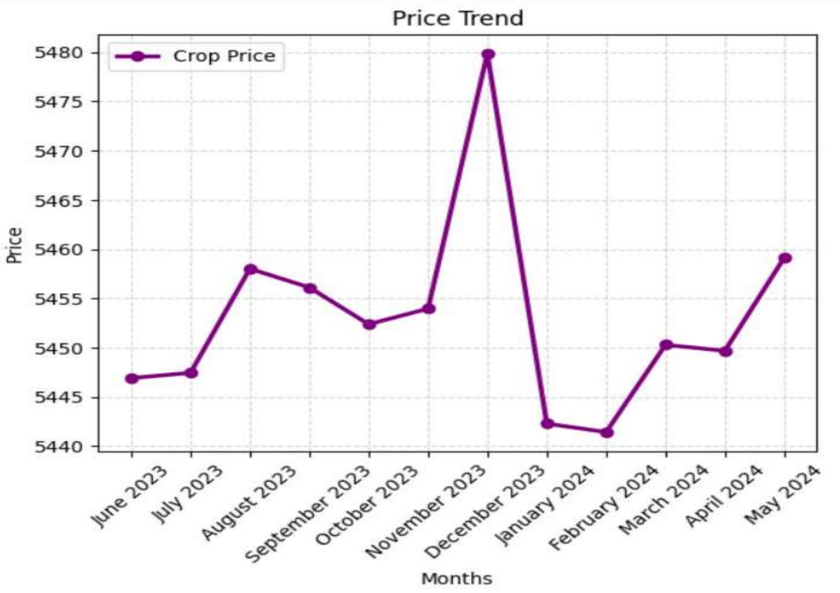


Fig 5.5 Price trends line graph

CHAPTER 6

6.1 CONCLUSION

In conclusion, the proposed prediction system aims to provide an efficient and effective solution to the challenges faced by farmers in crop selection and price prediction. By leveraging machine learning techniques and integrating weather and soil data, the proposed solution can accurately predict the most suitable crop for a given location and forecast its prices for the next 12 months. The web app developed as part of this project provides a user-friendly interface that enables farmers to easily input their data and receive insightful predictions that can inform their decision-making. The achieved accuracy of the machine learning models, which have been trained and tested using real-world data demonstrate the success of the system. Furthermore, the use of Flask UI, HTML/CSS, and Bootstrap has resulted in an intuitive and visually appealing interface that can be customized to meet the needs of different users. Overall, the potential to significantly benefit farmers has been enhanced by providing them with valuable insights that can inform their crop selection and marketing decisions, ultimately leading to improved yields and profits.

6.2 FUTURE WORK

Engaging with agricultural experts, farmers, market analysts, and other stakeholders to gather domain-specific knowledge and insights. Enhance the scalability and performance to accommodate larger datasets and user bases. Explore integration with drones, IoT sensors, and satellite imagery to gather real-time field data to provide more accurate results. Develop techniques for long-term crop forecasting and risk assessment, considering climate change, pest outbreaks, and other potential disruptions.

REFERENCES

[1]. S. Vashisht, P. Kumar and M. C. Trivedi, *Improvised Extreme Learning Machine for Crop Yield Prediction* 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2022, pp. 754-757, doi: 10.1109/ICIEM54221.2022.9853054.

[2]. R. Luciani, G. Laneve and M. JahJah, *Agricultural Monitoring, an Automatic Procedure for Crop Mapping and Yield Estimation: The Great Rift Valley of Kenya Case* in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 12, no. 7, pp. 2196-2208, July 2019, doi: 10.1109/JSTARS.2019.2921437.

[3]. D. Elavarasan and P. M. D. Vincent *Crop Yield Prediction Using Deep Reinforcement Learning Model for Sustainable Agrarian Applications* in IEEE Access, vol. 8, pp. 86886-86901, 2020, doi: 10.1109/ACCESS.2020.2992480.

[4]. Y. Alebele et al. *Estimation of Crop Yield From Combined Optical and SAR Imagery Using Gaussian Kernel Regression* in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 10520-10534, 2021, doi: 10.1109/JSTARS.2021.3118707.

[5]. S. M. M. Nejad, D. Abbasi-Moghadam, A. Sharifi, N. Farmonov, K. Amankulova and M. Lászlź, *Multispectral Crop Yield Prediction Using 3D-Convolutional Neural Networks and Attention Convolutional LSTM Approaches* in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 16, pp. 254-266, 2023, doi: 10.1109/JSTARS.2022.3223423.

[6] M. Qiao et al., "Exploiting Hierarchical Features for Crop Yield Prediction Based on 3-D Convolutional Neural Networks and Multikernel Gaussian

Process," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 4476-4489, 2021, doi: 10.1109/JSTARS.2021.3073149.

[7]. F. Ji, J. Meng, Z. Cheng, H. Fang and Y. Wang, "Crop Yield Estimation at Field Scales by Assimilating Time Series of Sentinel-2 Data Into a Modified CASA-WOFOST Coupled Model," in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-14, 2022, Art no. 4400914, doi: 10.1109/TGRS.2020.3047102.

[8]. G. Mariammal, A. Suruliandi, S. P. Raja and E. Poongothai, "Prediction of Land Suitability for Crop Cultivation Based on Soil and Environmental Characteristics Using Modified Recursive Feature Elimination Technique With Various Classifiers," in IEEE Transactions on Computational Social Systems, vol. 8, no. 5, pp. 1132-1142, Oct. 2021, doi: 10.1109/TCSS.2021.3074534.

[9]. M. Ishak, M. S. Rahaman and T. Mahmud, "FarmEasy: An Intelligent Platform to Empower Crops Prediction and Crops Marketing," 2021 13th International Conference on Information & Communication Technology and System (ICTS), Surabaya, Indonesia, 2021, pp. 224-229, doi: 10.1109/ICTS52701.2021.9608436.

[10]. B. V. B. Prabhu and M. Dakshayini, "Demand-prediction model for forecasting AGRI-needs of the society," 2017 International Conference on Inventive Computing and Informatics (ICICI), Coimbatore, India, 2017, pp. 430-435, doi: 10.1109/ICICI.2017.8365388.