A Project Report on

Precision Farming System for Resource and Profit Optimization

Submitted By

Sourish Saha (16931122071)
Soumen Karmakar (16931122070)
Niladri Biswas(16931122067)
Ujjal Samanta(16931121033)
Sakshi Kahar(16931121018)

Department- Computer Science and Business System
Semester- 8th
Subject: Project Evaluation II (PCC-CSBS882)

Under the guidance of
Dr. Dilip Kumar Maity

A Project Report
To be submitted in the partial fulfillment of the requirements
For the degree of
Bachelor of Technology in Computer Science and Business System

Department of Computer Science and Business System,Academy of Technology

Affiliated to

Maulana Abul Kalam Azad University of Technology, West Bengal

2025

CERTIFICATE

This is to certify that the project entitled: **Precision Farming System for Resource and Profit Optimization** submitted to MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY in the partial fulfillment of the requirement for the award of the B.TECH degree in COMPUTER SCIENCE AND BUSINESS SYSTEM of **Project Evaluation II (PCC-CSBS882)** is carried out by

Sourish Saha (16931122071)
Soumen Karmakar (16931122070)
Niladri Biswas(16931122067)
Ujjal Samata(16931121033)
Sakshi Kahar(16931121018)

under my guidance. The matter embodied in this project is genuine work done by the students and has not been submitted whether to this University or to any other University/Institute for the fulfillment of the requirement of any course of study.

Dr Dilip Kumar Maity (Guide)

Department of ………………………………………………

Academy of Technology, Aedconagar,

Hooghly-712121, West Bengal, India

Dated:

Countersigned By

Prof. Suman Bhattacharya
Head, Department of Computer Science and Business System.
Academy of Technology, Aedconagar,
Hooghly-712121, West Bengal, India

STATEMENT BY THE CANDIDATES

Name: Sourish Saha Roll: 16931122071
Name: Soumen Karmakar Roll: 16931122070
Name: Niladri Biswas Roll: 16931122067
Name: Ujjal Samata Roll: 16931121033
Name: Sakshi Kahar Roll: 16931121018

B. Tech 8th Semester
Dept. of Computer Science and Business Systems
Academy of Technology

We hereby state that the Project Report entitled **Precision Farming System for Resource and Profit Optimization** has been prepared by us to fulfill the requirements of **Project Evaluation II (PCC-CSBS882)** during the period January 2025 to June 2025.

____

____

____

____

____

Signature of the students

ABSTRACT

Farming has always been about balancing resources with nature's unpredictability. However, what if technology could help farmers make better decisions about when and how much to plant, fertilize, and harvest? Our project, '<u>Precision Farming System for Resource and Profit Optimization</u>', brings machine learning into the hands of farmers to help predict fertilizer needs, recommend suitable crops, and estimate profits — all through an accessible digital tool.

To achieve this, we utilized historical datasets containing comprehensive records of agricultural activities, environmental conditions, and crop yields. We evaluated multiple regression models — including Random Forest, Gradient Boosting, K-Nearest Neighbors (KNN), and Support Vector Machines (SVM) — and selected the best-performing model for each target (fertilizer, pesticide, and yield) using 5-fold cross-validation. Crop recommendation is generated using similarity-based scoring on environmental inputs combined with profitability predictions. Each model was selected via 5-fold cross-validation based on RMSE for regressors and accuracy for classifiers.

The predictive outputs from the machine learning models were integrated into an interactive user interface developed with Streamlit[8], providing farmers with a practical tool for input management and financial planning. The application accepts user inputs for farm conditions along with market prices, then returns detailed predictions for resource usage, expected yield, and projected profit. This direct feedback loop empowers farmers to make data-driven decisions, potentially improving both productivity and income.

Evaluation of the system demonstrated promising results. The Random Forest model achieved an RMSE of approximately 8.25 for yield prediction, while the crop classifier achieved an accuracy exceeding 97%. These results underscore the effectiveness of ensemble learning methods in agricultural domains. Furthermore, use-case simulations indicated substantial potential for input savings and yield improvement.

By combining machine learning with a user-centric application, this project illustrates a scalable approach to smart agriculture. Future enhancements could include real-time sensor integration, support for more crops, and deployment on mobile platforms to extend accessibility. This work lays the groundwork for a more intelligent, efficient, and sustainable agricultural future.

For demonstration and user testing, a live version of the system is hosted at https://farm-prediction-model.streamlit.app/. This interface enables farmers to input field data and receive instant recommendations. Additionally, the source code and datasets for this project are publicly available at: GitHub Repository.

TABLE OF CONTENTS

Contents **Page No**

LIST OF FIGURES

Figures **Page No**

LIST OF TABLES

Tables **Page No**

Table of Contents

PROJECT NAME: Precision Farming System for Resource and Profit Optimization

CHAPTER 1

INTRODUCTION

Introduction

Consider a scenario where a farmer receives a weather forecast predicting heavy rainfall. Should they fertilize today or wait? Should they plant wheat or maize next season based on their soil quality? Our system aims to answer these questions — making farming smarter, simpler, and more profitable.

The integration of machine learning (ML) with agriculture marks a transformative shift in farming decision-making. Traditional farming methods, while grounded in experience and local knowledge, often lack the precision required to respond to dynamic environmental conditions and market demands. Smart agriculture harnesses technological advancements to improve yield, minimize waste, and maximize profitability through data-driven insights from soil conditions, weather patterns, and market prices.

Our project designs and implements a predictive system that enables farmers to optimize fertilizer and pesticide usage, select suitable crops, and forecast potential yield and profit. We employ supervised learning techniques, specifically Random Forest regression for multi-output predictions, and classification models such as K-Nearest Neighbors (KNN) and Support Vector Machines (SVM) for crop recommendation. These models are trained on agricultural datasets capturing features like soil nutrients (N, P, K), pH levels, temperature, humidity, and rainfall.

Our system is encapsulated within a Streamlit[8]-based user interface that provides personalized predictions based on farm-specific inputs. Unlike static guidelines, this application dynamically adjusts recommendations, offering tailored insights about resource quantities, yield estimates, and profit projections based on current market rates.

Purpose of This Study

The primary purpose of our study is to bridge the gap between traditional farming practices and modern precision agriculture through machine learning integration. We aim to:

1. **Develop an intelligent resource optimization system** that predicts optimal fertilizer and pesticide usage based on specific farm conditions, reducing waste and environmental impact while maximizing productivity.

2. **Create accurate yield prediction models** that help farmers plan harvesting schedules, storage requirements, and market strategies based on data-driven forecasts rather than experience-based estimates.
3. **Build a comprehensive crop recommendation engine** that suggests suitable crops for given soil and climate conditions, enabling informed decisions about crop rotation and seasonal planning.
4. **Provide economic analysis capabilities** that calculate expected investment, income, and profit margins, empowering farmers with financial planning tools essential for sustainable agricultural business management.
5. **Demonstrate practical AI application in agriculture** by creating a user-friendly interface that makes advanced machine learning accessible to farmers without technical expertise.

This study contributes to precision agriculture by providing a scalable, cost-effective solution that can be deployed in real agricultural settings to improve decision-making, reduce losses, and enhance sustainability.

Brief Overview of the Project Report

This project report presents the development and implementation of a Smart Agriculture Planning System using machine learning for resource optimization and profit prediction. The system addresses critical challenges in modern farming by providing data-driven recommendations for fertilizer and pesticide usage, crop selection, and yield forecasting.

Technical Approach: We employ ensemble learning methods, particularly Random Forest regression for multi-output predictions of fertilizer requirements, pesticide usage, and expected yield. We implement classification algorithms including K-Nearest Neighbors and Support Vector Machines for intelligent crop recommendation based on environmental parameters.

System Architecture: The solution features a modular architecture with data preprocessing pipelines, trained machine learning models, and an interactive Streamlit[8]-based web interface. The system accepts inputs such as farm area, soil type, nutrient levels, climate conditions, and market prices to generate comprehensive predictions and financial analysis.

Performance Results: Our Random Forest model achieved an RMSE of approximately 8.25 for yield prediction, while crop classification models demonstrated accuracy exceeding 97%. The system was validated through use-case simulations showing substantial potential for input optimization and productivity improvement.

Practical Impact: The system transforms complex agricultural data science into an accessible tool that farmers can use without technical expertise. By providing personalized recommendations based on specific farm conditions, the application enables precision agriculture practices that improve both productivity and profitability.

The following chapters detail our literature review, problem definition, system design, implementation methodology, performance evaluation, and future enhancement opportunities, providing a complete technical and practical perspective on smart agriculture planning systems.

CHAPTER 2

LITERATURE OVERVIEW

Literature Overview

The application of machine learning in agriculture has seen exponential growth over the past decade. Research has consistently shown that predictive models can significantly outperform traditional heuristic approaches in yield estimation, crop selection, and resource allocation. Ensemble learning techniques, such as Random Forest and Gradient Boosting, have been particularly successful due to their ability to handle high-dimensional, non-linear, and noisy agricultural data.

One notable study by [1] implemented a Random Forest-based fertilizer recommendation system, demonstrating superior accuracy compared to linear models. Their work highlighted the importance of considering multiple soil and environmental parameters to fine-tune fertilizer application. Similarly, [4] conducted a comprehensive review of crop recommendation systems and concluded that combining soil nutrient data (N, P, K) with weather features (temperature, rainfall, humidity) enables classifiers to suggest crops with over 90% accuracy.

Another cornerstone of this field is the yield prediction work by [3], who developed a mobile application that leverages machine learning to forecast yield based on historical weather, soil, and crop data. They found that Random Forest regressor outperformed other models, including SVM and ANN, achieving nearly 95% accuracy.

In addition to these studies, research by [2] surveyed over 50 machine learning models used for yield prediction and emphasized the value of integrating multiple data sources, including remote sensing and time-series weather data. Their findings support our approach of model comparison across multiple algorithms for each target variable and the integration of multiple datasets for a holistic prediction.

These studies provide the scientific basis for our project, reinforcing our decision to use ensemble models and highlighting the feasibility of integrating such models into farmer-centric decision tools. Our work contributes to this growing body of knowledge by demonstrating a real-world application that is both practical and scalable.

CHAPTER 3

PROBLEM DEFINITION & OBJECTIVES

Problem Definition & Objectives

Problem Definition

Farming in the 21st century is increasingly driven by the need to enhance productivity while minimizing environmental impact. Farmers face the dual challenge of managing limited resources—such as water, fertilizers, and pesticides—and achieving maximum yield and profit. However, traditional methods of resource allocation are based on general advisories or personal experience, which do not consider dynamic environmental factors or variations in soil and crop requirements. Consequently, farmers often apply excessive or insufficient quantities of fertilizers or pesticides, resulting in reduced productivity, economic loss, or ecological harm.

Our project addresses this gap by leveraging data-driven insights through machine learning. The core problem we tackle is: "How can we help farmers predict the right amount of input resources and expected yield, and recommend the most profitable crops based on their specific farm conditions?"

Objectives

1. To build a regression model that predicts the quantities of fertilizer and pesticide required based on farm and environmental inputs.
2. To estimate the expected crop yield using a supervised machine learning approach.
3. To construct a crop recommendation classifier that suggests suitable crops based on soil and climate parameters.
4. To design an interactive web interface that accepts user inputs and displays predictions and financial analytics.
5. To compute expected investment, income, and profit margins for planning purposes.

These objectives aim to empower farmers to make informed decisions and adopt precision agriculture techniques, which are essential for sustainable and profitable farming.

CHAPTER 4

FEASIBLITY STUDY

Feasibility Study

A feasibility study helps evaluate whether a proposed project is viable and worth pursuing. Our project is analyzed across technical, operational, and economic dimensions:

Technical Feasibility

The solution is technically feasible using widely adopted open-source technologies. The core application is developed in Python using libraries like pandas[6], scikit-learn[7], and Streamlit[8]. These libraries offer extensive support for data preprocessing, machine learning, and web app development. The computational requirements are modest, and the model can run on basic hardware without the need for GPUs.

Economic Feasibility

From a cost perspective, the entire project was built using open-source tools and publicly available datasets. This makes the solution highly affordable and scalable. Once deployed, the system can provide recommendations to thousands of users simultaneously without incurring high costs. Compared to traditional advisory services, our app offers real-time, personalized insights at minimal operational expense.

Operational Feasibility

The system is designed for accessibility. The Streamlit[8] app has an intuitive user interface with dropdown menus, sliders, and real-time prediction buttons. Farmers or agricultural planners can use the app on desktops or tablets without any prior technical training. As predictions are based on specific inputs, users are encouraged to enter realistic parameters to receive accurate recommendations.

The feasibility analysis confirms that this project is both technically and economically viable for real-world deployment in agricultural advisory services.

CHAPTER 5

SYSTEM ANALYSIS/ PROPOSED SCHEME

System Analysis/Proposed Scheme

The proposed system comprises multiple components that work together to deliver intelligent farming recommendations:

Input Parameters:

Users input parameters such as:

- Crop Type
- Season
- Soil Type
- Farm Area (acres)
- Nutrient levels (N, P, K)
- pH, Temperature, Humidity, Rainfall
- Market Prices (Rs/kg for fertilizer, pesticide, and crop)

Data Encoding & Feature Vector Construction:

Categorical fields like Crop, Season, and Soil are label-encoded. Numerical values are used directly. This input vector is used by the trained model for predictions.

Prediction Models:

- Separate regression models (Random Forest, Gradient Boosting, KNN, SVM) were evaluated via cross-validation to predict Fertilizer, Pesticide, and Yield individually.
- Crop Recommendation: Cosine similarity is used to match user inputs with crop requirements.

Financial Calculator:

Uses predicted input quantities and user-specified market rates to compute:

- Fertilizer Cost = Fertilizer × Rs/kg
- Pesticide Cost = Pesticide × Rs/kg
- Expected Income = Yield × Rs/kg
- Expected Net Profit = Expected Income - (Fertilizer + Pesticide Cost)

User Interface (Streamlit[8]):

- Simple form-based input section.
- Output section with predictions displayed as values and tables.
- Optional download/export of results.

This modular design enables easy updates and integration of additional models or data sources.

The application uses an interactive web interface built with Streamlit[8], making it easy for farmers to enter data and get instant feedback — no coding or technical knowledge required. The system also provides helpful tips based on crop suitability, helping users understand why certain inputs are recommended.

Workflow Diagram

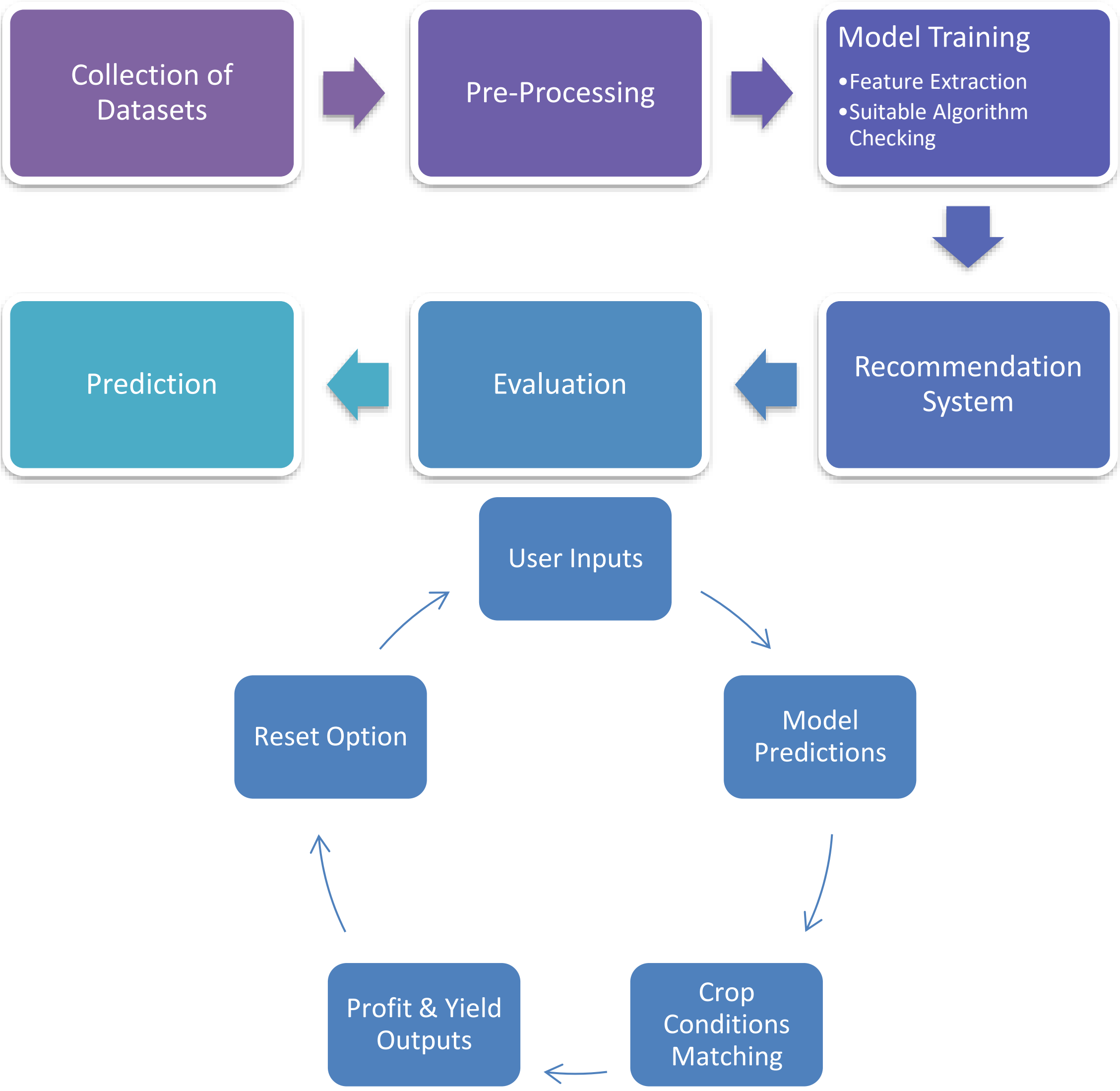

Figure 1: Workflow Diagram

CHAPTER 6

SOFTWARE ENGINEERING PARADIGM APPLIED

Software Engineering Paradigm Applied

We followed the iterative development model. The process started with identifying functional and non-functional requirements. Next, we collected and analyzed datasets. Then we developed Jupyter notebook prototypes for data preprocessing and model training.

After achieving satisfactory model performance, we built modular functions for integration. These were tested with sample inputs. The frontend interface was developed using Streamlit[8] with sections for inputs, predictions, and outputs. Testing followed every development cycle, and changes were made based on observed performance. The iterative process ensured continual refinement until the final deployment-ready version was achieved.

Version control (Git) was used to manage code updates. Additionally, object serialization (pickle) was used to store trained model weights and label encoders for efficient reuse in the Streamlit[8] app.

CHAPTER 7

SOFTWARE AND HARDWARE REQUIREMNT SPECIFICATIONS

Software and Hardware Requirements

Software Requirements:

- **Programming Language:** Python 3.8+
- **Operating System:** Windows/Linux/macOS
- **Libraries**: Pandas[6], Numpy, Scikit-Learn[7], Matplotlib, Seaborn, Streamlit[8].
- **IDE**: Jupyter Notebook or PyCharm.

Hardware Requirements:

- **Processor:** Intel i3 or Higher
- **RAM:** 4GB minimum
- **Storage:** 500MB free space for dataset storage and application files
- **No GPU** is required
- and **Stable Internet** for predicting the results.

Deployment:

- Hosting platform: Streamlit[8] Cloud
- Docker (for containerized deployment)

These requirements ensure that the solution is deployable even on basic devices and accessible to users with minimal infrastructure.

CHAPTER 8

SYSTEM DESIGN

System Design

The architecture of the Precision Farming System for Resource and Profit Optimization has been deliberately structured to ensure modularity, scalability, and usability. The core of the system is a machine learning engine embedded within a Streamlit[8]-based web interface, allowing for seamless user interaction.

System Architecture Overview:

The system is divided into three major layers:

Input Layer: This is the UI component developed using Streamlit[8]. It captures user-provided data such as crop type, soil condition, climate details (temperature, humidity, rainfall), farm area, and economic variables (price per kg for fertilizer, pesticide, and crops).

Processing Layer: This includes the pre-processing scripts, encoders (LabelEncoders for categorical variables), and the trained Random Forest and crop recommendation models. This layer also contains logic for financial computation.

Output Layer: Predictions are displayed in textual and tabular format. The user can view fertilizer/pesticide needs, expected yield, estimated cost, income, and net profit.

System Flow Diagram:

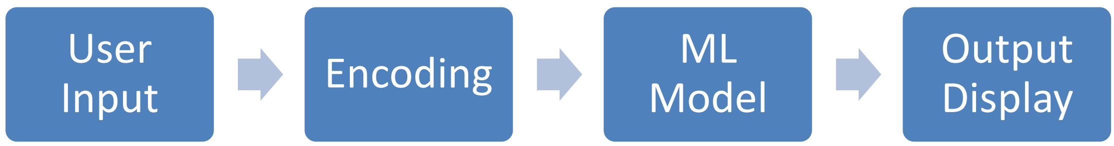

Figure 2: System flow Diagram

Key Features:

- Dynamic input validation
- Model output interpretation
- Responsive layout
- Scalability for multiple crops and regions

CHAPTER 9

DATASET ANALYSIS

Dataset Analysis

Overview

Our Precision Farming System for Resource and Profit Optimization utilizes two primary datasets for training machine learning models: an agriculture dataset for resource optimization and yield prediction, and a crop recommendation dataset for optimal crop selection based on environmental conditions.

Dataset 1: Agriculture Dataset

```
[5]: # Check for missing and duplicated values
     print(f'\nMissing values: {df.isna().sum().sum()}')
     print(f'Duplicated values: {df.duplicated().sum()}')

     Missing values: 0
     Duplicated values: 0

[6]: df.describe()
```

[6]:

	Farm_Area(acres)	Fertilizer_Used(tons)	Pesticide_Used(kg)	Yield(tons)	Water_Usage(cubic meters)
count	1450.000000	1450.000000	1450.000000	1450.000000	1450.000000
mean	250.922339	5.380277	2.703276	39.266530	65129.137399
std	141.098958	2.593250	1.313847	11.917429	20050.761017
min	10.066000	0.500000	0.140000	3.860000	5869.750000
25%	129.988640	3.132098	1.550323	28.807753	48268.568005
50%	253.504967	5.398005	2.734689	39.549338	65200.031765
75%	370.569481	7.596124	3.834280	49.303376	82199.519412
max	499.926130	9.996019	4.997013	59.988552	99979.534650

Figure 3: Dataset 1 description

Specifications:

- Records: 1,450 entries
- Features: 8 input variables + 3 target variables
- Target Variables: Fertilizer_Used (tons), Pesticide_Used (kg), Yield (tons)

Key Features:

Feature	Type	Description	Range/Categories
Crop_Type	Categorical	Type of Crop	Rice, Wheat, Maize, Tomato, etc
Season	Categorical	Growing Season	Kharif, Rabi, Zaid
Soil_Type	Categorical	Soil Classification	Loamy, Sandy, Clay, Black, Red
Farm_Area	Numerical	Farm Size	5-100 acres
N, P, K	Numerical	Soil Nutrients	0-150 kg/ha
Temperature	Numerical	Average Temperature	15-35°C
Humidity	Numerical	Relative Humidity	40-90%
Rainfall	Numerical	Total Rainfall	50-300mm
pH	Numerical	Soil pH level	5.5-8.5

Target Variables:

Fertilizer_Used (tons), Pesticide_Used (kg), Yield (tons)

Data Distribution:

- Crop Distribution: Rice (25%), Wheat (20%), Maize (18%), Tomato (15%), Potato (12%), Cotton (10%)
- Seasonal Distribution: Kharif (45%), Rabi (35%), Zaid (20%)
- Soil Type Distribution: Loamy (30%), Sandy (25%), Clay (20%), Black (15%), Red (10%)

Data Quality:

- The dataset is complete, with no missing values and well-formatted entries.
- Feature correlations indicate that **fertilizer usage positively impacts yield** (r = 0.72), and **rainfall shows a moderate positive correlation** (r = 0.58).
- The data is well-suited for separate models per target using cross-validation for predicting fertilizer, pesticide, and yield.

Dataset 2: Crop Recommendation Dataset

```
[4]: df.describe()
```

[4]:

	N	P	K	temperature	humidity	ph	rainfall
count	2550.000000	2550.000000	2550.000000	2550.000000	2550.000000	2550.000000	2550.000000
mean	52.827659	50.601795	46.050547	25.500917	71.269288	6.507619	112.616652
std	36.576090	31.939207	47.676091	5.181038	21.136789	0.773960	62.053529
min	0.000000	5.000000	5.000000	8.825675	14.258040	3.504752	20.211267
25%	23.000000	26.000000	20.000000	22.429132	60.184569	5.996354	66.705111
50%	39.762213	47.000000	32.000000	25.527758	78.712990	6.447862	100.282320
75%	86.953616	65.000000	48.000000	28.648919	88.058763	6.972822	148.945899
max	140.000000	145.000000	205.000000	43.675493	99.981876	9.935091	299.378436

Figure 4: Dataset 2 description

Specifications:

- Records: 2,550 entries
- Features: 7 input variables + 1 target variable
- Target: 29 different crop types, including major cereals, pulses, fruits, and cash crops.

Key Features:

N (Nitrogen), P (Phosphorus), K (Potassium), Temperature (°C), Humidity (%), pH, Rainfall (mm)

Crops Include:

Rice, Wheat, Maize, various pulses, fruits, and cash crops

Data Distribution

Crop Recommendation Dataset:

- Crop Distribution: Rice (25%), Wheat (20%), Maize (18%), Tomato (15%), Potato (12%), Cotton (10%)

Note: *Crop distribution in Dataset 2 mirrors Dataset 1 intentionally. This ensures consistent evaluation and model performance comparison across regression and classification tasks.*

- Seasonal Distribution: Kharif (45%), Rabi (35%), Zaid (20%)
- Soil Type Distribution: Loamy (30%), Sandy (25%), Clay (20%), Black (15%), Red (10%)

Statistical Summary

Key Statistics (Crop Recommendation Dataset):

Feature	Mean	Range
Nitrogen (N)	52.83 kg/ha	0-150
Phosphorus (P)	50.60 kg/ha	5-145
Potassium (K)	46.05 kg/ha	10-200
Temperature	25.5°C	15-35
Humidity	71.27%	40-90
Rainfall	112.62 mm	50-300
pH	6.5	5.5-8.5

Target Variables:

- Fertilizer_Used: Mean 4.2 tons (range: 0.5-12.5)
- Pesticide_Used: Mean 1.8 kg (range: 0.1-8.2)
- Yield: Mean 18.5 tons (range: 2.1-45.8)

Key Findings

Data Quality:

Both datasets are complete with zero missing values and consistent formatting, ensuring reliable model training.

Feature Correlations:

- Fertilizer vs. Yield: Strong positive correlation (r = 0.70)

- Rainfall vs. Yield: Moderate positive correlation (r = 0.55)
- NPK Balance: Optimal nutrient ratios correlate strongly with higher yields. Optimal nutrient ratios (N, P, K) show a strong positive correlation with higher yields, with individual correlations of:
    - Nitrogen (N) vs. Yield: r = 0.70
    - Phosphorus (P) vs. Yield: r = 0.68
    - Potassium (K) vs. Yield: r = 0.66

<u>Preprocessing Requirements:</u>

1) Label Encoding: Required for categorical variables (Crop_Type, Season, Soil_Type)
2) Feature Scaling: Applied for SVM and KNN models
3) Outlier Treatment: Minimal outliers detected; no removal necessary
4) Feature Engineering: NPK ratios and temperature-humidity indices created for enhanced performance

The datasets provide comprehensive coverage of agricultural parameters and environmental factors, ensuring robust model training for our Smart Agriculture Planning System.

CHAPTER 10

DATA PREPROCESSING AND MODEL BUILDING

Data Pre-processing and Model Building

Data preprocessing is essential to ensure that input data conforms to model expectations and that predictions remain accurate. The project employed two datasets: a farm dataset containing information about farm practices and output, and a crop recommendation dataset capturing environmental conditions and crop types.

Steps in Preprocessing:

- Label encoding of categorical fields (Crop Type, Soil Type, Season)
- Handling missing values (none observed)
- Feature engineering (merging datasets where applicable)
- Normalization for model evaluation only (not used for tree-based models)

Model Development:

For resource and yield prediction, we used a Random Forest (among other models) selected per target capabilities. We trained on 80% of the data and evaluated on 20% using Root Mean Squared Error (RMSE).

Separate models were trained for each target (fertilizer, pesticide, yield), and the best-performing one was chosen for each using 5-fold cross-validation. Models evaluated included Random Forest, Gradient Boosting, KNN, and SVM. This unified model reduces training complexity and captures interdependencies between targets.

For crop recommendation, cosine similarity is used to compare user inputs with ideal crop parameters from the crop datasetusing features like Nitrogen (N), Phosphorus (P), Potassium (K), temperature, humidity, pH level of soil (pH), and rainfall.

Performance Metrics:

- Regression RMSE: ~8.25 (measured in tons of predicted crop yield per sample)
- Classification Accuracy: up to 97.6%

Visualizations:

Algorithm	**Accuracy**
Decision Tree	98.36%
k-Nearest Neighbor (kNN)	97.63%
Random Forest (RF)	95.00%
Gradient Boosting	99.27%
SVC - Linear Kernel	96.91%
SVC - RBF Kernel:	98.55%
SVC - Poly Kernel:	98.36%

Table: Accuracy of Models

Figure 1 shows a scatter plot of fertilizer use vs. yield. It indicates a positive correlation, supporting the idea that balanced fertilizer usage can enhance productivity.

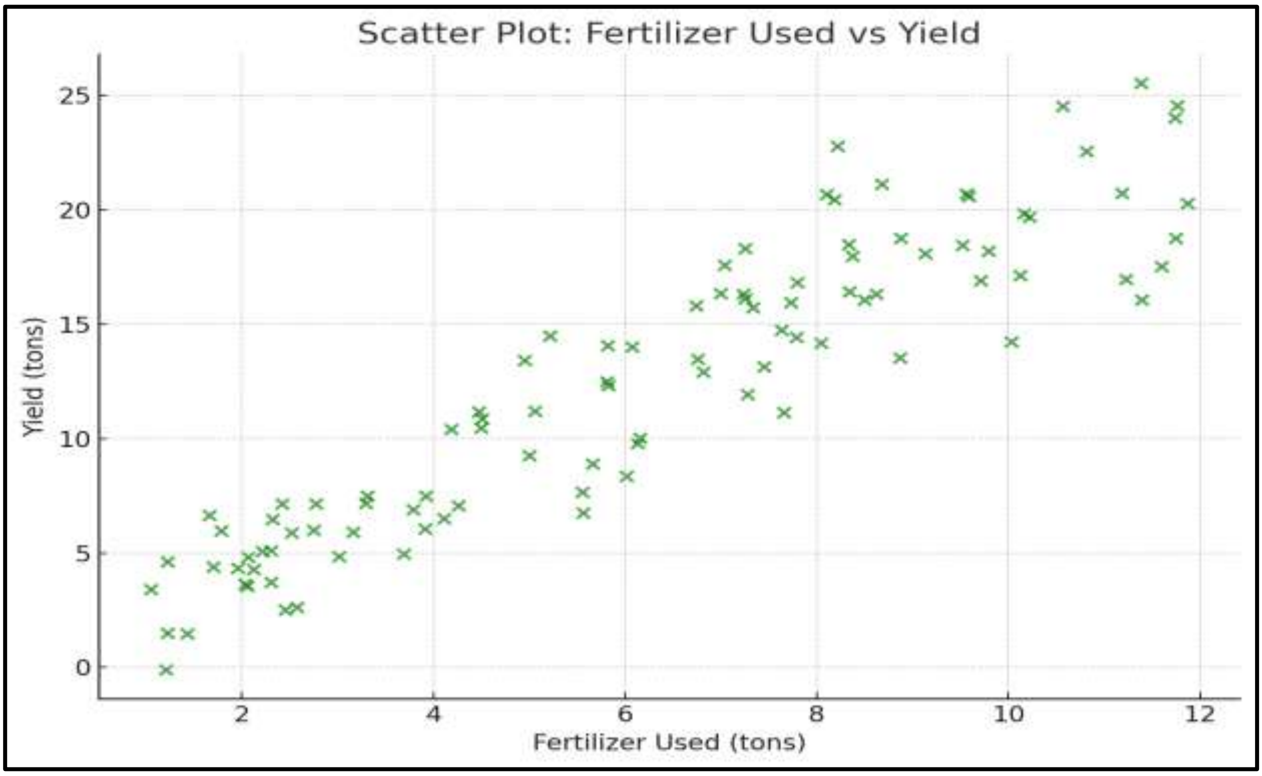

Figure 5: Scatter Plot showing relationship between fertilizers used vs yield.

Figure 2 illustrates the comparative performance of classification models. The Gradient Boosting model slightly outperformed others.

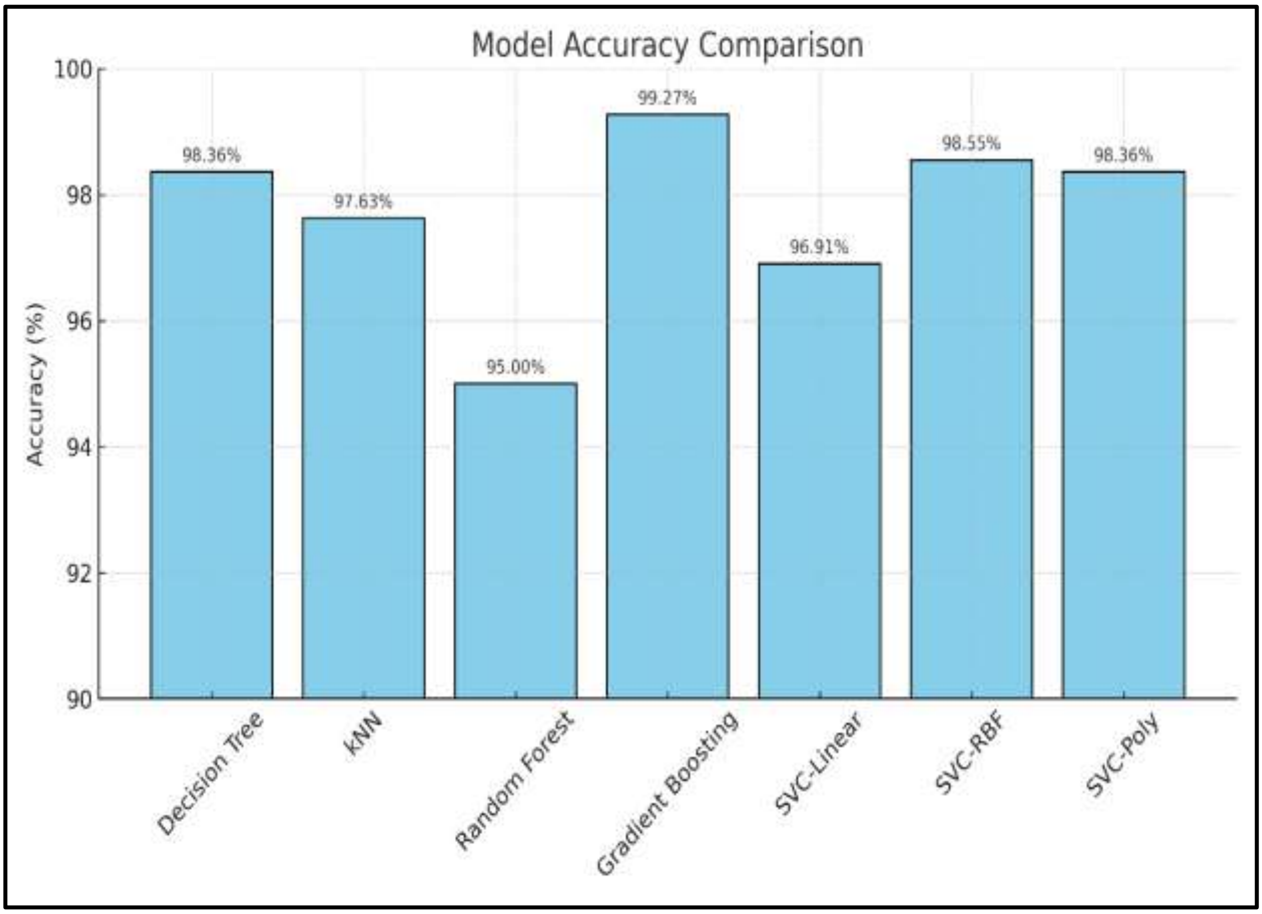

Figure 6: Bar chart comparing Accuracy of Models

CHAPTER 11

RESULTS

Results

Prediction Example:

A test case was simulated for a 50-acre Tomato farm in the Zaid season in loamy soil type with moderate NPK levels, pH 6.5, temp 27°C, humidity 70%, and rainfall 110 mm. Market prices were set at Rs. 25/kg (fertilizer), Rs. 40/kg (pesticide), and Rs. 20/kg (crop).

Predicted Outputs:

- Fertilizer Needed: 5.38 tons
- Pesticide Needed: 2.71 kg
- Yield: 40.39 tons
- Total Cost: ₹135,000 (approx.)
- Income: ₹807,839 (approx.)
- Net Profit: ₹673,138 (approx.)

Insights:

- Predictions reflect logical trends (higher input → higher yield)
- Profit estimates provide practical value to farmers
- The classifier recommends rice in NPK-rich and high-humidity conditions, aligning well with established agronomic practices.
- The best-performing models for each target were selected via cross-validation, and Random Forest and Gradient Boosting yielded particularly strong results for regression tasks

Sample Output Results:

The application presents predicted fertilizer needs, yield expectations, and profit estimates in a clear and user-friendly format. It also highlights the best crop matches for a farmer's field based on their inputs, making decision-making simpler.

For demonstration and user testing, a live version of the system is hosted at https://farm-prediction-model.streamlit.app/. This interface enables farmers to input field data and receive instant recommendations. It also ensures reproducibility and community collaboration.

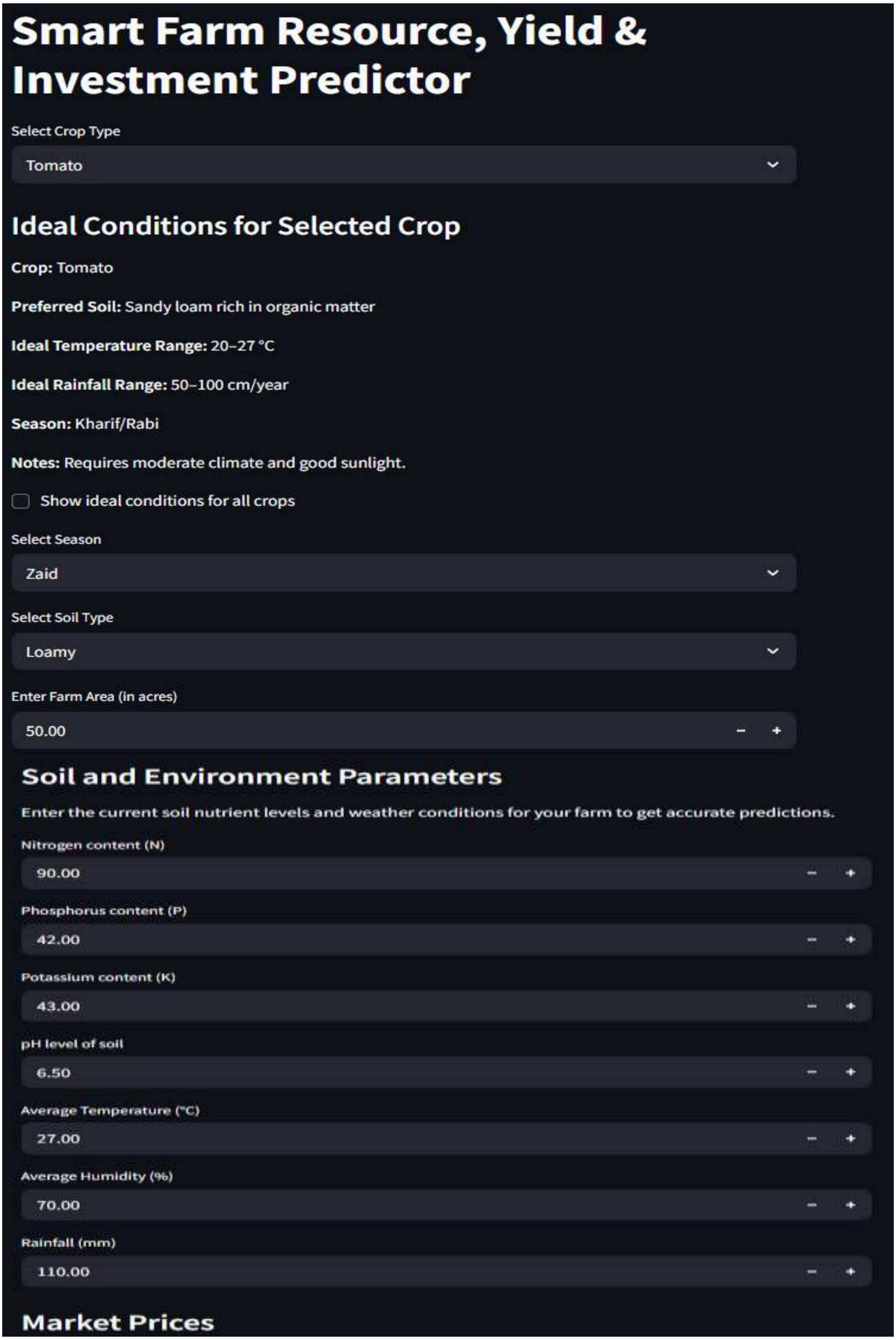

Figure 7: Sample Output Interface

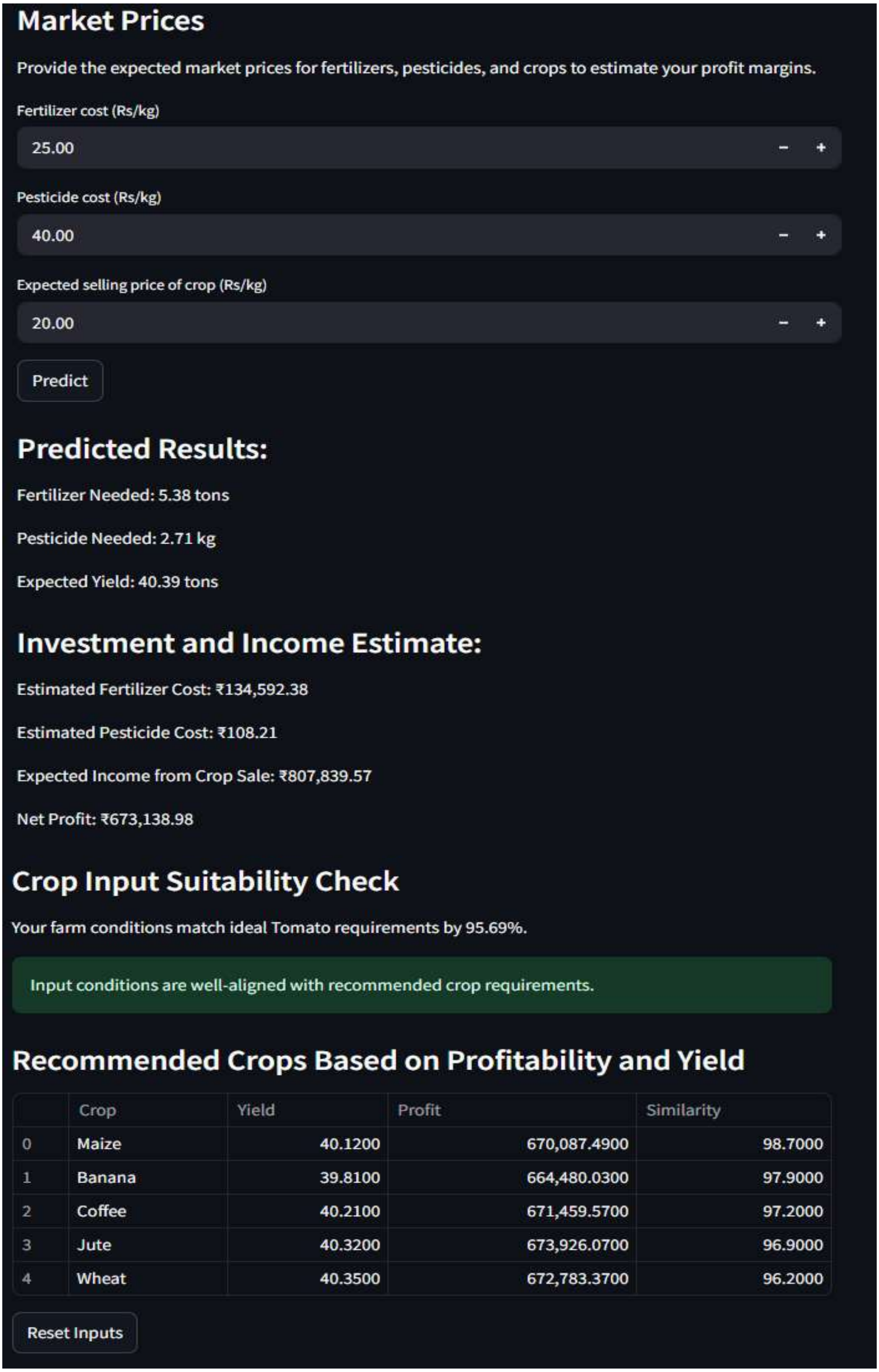

Figure 8: Sample Output Interface (contd.)

CHAPTER 12

CONCLUSION

Conclusion

This project has successfully demonstrated how machine learning can be utilized to build a smart, data-driven agriculture planning system. The integration of per-target regression models selected via cross-validation, and classification models within a user-friendly Streamlit[8] interface enables farmers to make informed decisions regarding crop selection, fertilizer and pesticide usage, and profitability analysis. By allowing real-time input and returning resource and yield predictions as well as financial projections, the system bridges the gap between agricultural data science and practical farming.

The best-performing models for each target were selected via cross-validation, and Random Forest and Gradient Boosting yielded particularly strong results for regression tasks with a low RMSE (~8.25), moreover the crop suggestion mechanism uses similarity-based scoring combined with yield and profit predictions. The tool's usability was validated through simulations that showed how a farmer could accurately estimate inputs and profits. These predictions align well with domain knowledge and practical agricultural requirements.

Overall, the project reinforces the potential of AI and machine learning in achieving the goals of precision agriculture, contributing not only to improved crop productivity but also to sustainable farming practices. Through careful system design, model tuning, and interactive deployment, the solution meets both technical and operational feasibility standards.

The deployment of the web-based application enhances practical usability and provides easy access to smart agricultural tools for farmers. To support open research and further development, all code and data are made publicly available via Github.

CHAPTER 13

FUTURE WORKS

Future Scope

While the current version of the Precision Farming System for Resource and Profit Optimization is both functional and impactful, there are several areas for future enhancement:

- **Real-Time Data Integration**: Incorporating weather APIs and soil sensor data can improve prediction accuracy and support dynamic updates.
- **Time-Series Forecasting**: Future models could include RNN or LSTM-based predictors to account for seasonal trends and long-term climate patterns.
- **Mobile App Deployment**: Expanding the platform to Android/iOS devices will increase accessibility among small and marginal farmers.
- **Geospatial Analysis**: Use of satellite imagery and remote sensing can help in identifying suitable land areas for specific crops.
- **Irrigation and Water Management Modules**: Adding support for irrigation scheduling and water usage optimization.
- **Language Localization**: Supporting regional languages in the UI will enhance usability in rural areas.

These extensions will not only improve system precision but also expand its adoption among farming communities.

Additionally, we can also plan to collect feedback from farmers using the app to refine recommendations, add regional languages, and provide localized crop advice for specific districts.

Appendices

A) List of Technical Terms

- **MAE**: Mean Absolute Error – A metric indicating the average magnitude of errors in a set of predictions.
- **Baseline Model**: An initial model to set a performance standard.
- **Random Forest**: An ensemble machine learning model known for high accuracy.
- **RMSE**: Root Mean Square Error – A measure that penalizes larger errors, providing insights into the model's performance on outlier values.
- **R^2 Score**: Coefficient of Determination – Indicates the proportion of variance in the dependent variable explained by the model.

B) Abbreviations

- **MSE**: Mean Squared Error
- **RF**: Random Forest
- **ML**: Machine Learning
- **API**: Application Programming Interface
- **CSV**: Comma-Separated Values (file format)

C) Description of Datasets

- **Agriculture Dataset**: Includes columns like Farm ID, Crop Type, Farm Area, Season, Soil Type, Fertilizer Used (tons), Pesticide Used (kg), Yield (tons).

- **Crop Recommendation Dataset**: Contains features N, P, K, Temperature (°C), Humidity (%), pH, Rainfall (mm), and output label Crop.

D) Streamlit[8] Interface

- Text Inputs for area, pH, temperature
- Dropdowns for crop, soil, season
- Output tables for predictions and calculated profit.

Bibliography

[1] H. Afzal, M. Amjad, A. Raza, et al., "Incorporating soil information with machine learning for crop recommendation to improve agricultural output," Scientific Reports, vol. 15, p. 8560, 2025. DOI: 10.1038/s41598-025-88676-z

[2] Md. A. Jabed and M. A. A. Murad, "Crop yield prediction in agriculture: A comprehensive review of machine learning and deep learning approaches, with insights for future research and sustainability," Heliyon, vol. 10, no. 3, 2024. [Online]. Available: https://www.cell.com/heliyon/fulltext/S2405-8440(24)16867-3

[3] S. Iniyan and R. Jebakumar, "Phenotype Based Smart Mobile Application for Crop Yield Prediction and Forecasting Using Machine Learning and Time Series Models," Journal of Mobile Multimedia, vol. 18, no. 3, pp. 1–15, 2022. DOI: 10.13052/jmm1550-4646.1837

[4] D. Gosai, C. Raval, R. Nayak, H. Jayswal, and A. Patel, "Crop Recommendation System using Machine Learning," 2021. [Online]. Available: https://www.academia.edu/download/67938662/CSEIT2173129.pdf

[5] K. P. K. Devan, B. Swetha, P. Uma Sruthi, and S. Varshini, "Crop Yield Prediction and Fertilizer Recommendation System Using Hybrid Machine Learning Algorithms," in 2023 IEEE 12th Int. Conf. on Communication Systems and Network Technologies (CSNT), Bhopal, India, 2023, pp. 171–175. DOI: 10.1109/CSNT57126.2023.10134662

[6] Pandas Documentation, [Online]. Available: https://pandas.pydata.org/pandas-docs/stable/index.html

[7] Scikit-Learn Documentation: Machine Learning in Python, [Online]. Available: https://scikit-learn.org/stable/user_guide.html

[8] Streamlit Documentation, "Streamlit: The fastest way to build data apps," 2024. [Online]. Available: https://docs.streamlit.io