Crop Predicting System using various ML Algorithms and Datasets

Khushi Thakur (05319051623)[1], Kriti Rastogi (20119051623)[1]

(1)-USAR, Guru Gobind Singh Indraprastha University-East Delhi campus, New Delhi, India

1. Abstract:

Agriculture is the pillar of the Indian economy and more than 50% of India's population are dependent on agriculture for their survival. Variations in weather, climate, and other such environmental conditions have become a major risk for the healthy existence of agriculture. Machine learning (ML) plays a significant role as it has decision support tool for Crop Prediction (CP). It involves estimating the number of crops that will be produced in a given area based on various factors such as soil type, weather conditions, and crop management practices. The research deals with a systematic review that extracts and synthesize the features used for CP and furthermore, there are a variety of methods that were developed to analyze crop yield prediction using artificial intelligence techniques.

Supervised learning techniques were incapable to capture the nonlinear bond between input and output variables faced a problem during the selection of fruits grading or sorting. Many studies were recommended for agriculture development and the goal was to create an accurate and efficient model for crop classification such as crop yield estimation based on the weather, crop disease, classification of crops based on the growing phase etc., This paper explores various ML techniques utilized in the field of crop yield estimation and provided a detailed analysis in terms of accuracy using the techniques.

2. Introduction

Predicting crop yield is crucial to addressing emerging challenges in food security, particularly in an era of global climate change. Accurate yield predictions not only help farmers make informed economic and management decisions but also support famine prevention efforts. In a country like India where farming is the main occupation, accurate prediction of the crops plays an important role.

Underlying crop prediction is a fundamental research question in plant biology, which is to understand how plant phenotype is determined. We propose a novel model, for crop prediction, which attempts to combine the strengths and avoid the limitations
of the soil and efforts being wasted. At the core of this model lies various algorithms so as to get a comparative study as to which predicted crop is most accurate and can be grown as per the mentioned conditions.

To ensure the explainability of the results, we trained our algorithms to find features and interactions that are spatially and temporally robust, which means that they should be consistently predictive of crop across all counties in all years and situations.

Many machine learning algorithms are scalable to large datasets and have reasonably high prediction accuracy. However, prediction accuracy is sensitive to model structure and parameter calibration, and it can prove difficult to explain why predictions are accurate or inaccurate.

The objectives of this work were to analyse the capability of ML methods for yield prediction in various crops using a collected dataset obtained with crop simulation models and to establish comparison for various ML algorithms in crop prediction.

3. Literature Review

The crop recommendation system utilizes a dataset, Crop_Recommendations.csv, that encompasses essential attributes such as nitrogen (N), phosphorus (P), potassium (K), temperature, humidity, pH, and rainfall to suggest the most suitable crops for varying environmental and soil conditions. A range of machine learning models, including K-Nearest Neighbors (KNN), Support Vector Machines (SVM), Decision Trees, Random Forests, and Gradient Boosting, are employed for both training and evaluation purposes. The system enhances prediction accuracy through feature scaling and model optimization techniques, such as grid search.

The system demonstrates high accuracy, achieving up to 98% with models such as Random Forests and Gradient Boosting. It also features a user-friendly interface that allows users to input soil and weather conditions to predict appropriate crops, showcasing its practical application in precision agriculture. The literature underscores the importance of integrating domain knowledge with data-driven insights, thereby establishing a solid foundation for effective and sustainable crop management.

4. Overview of ML Models:

4.1 K-Nearest Neighbour (KNN)

The **K-Nearest Neighbors (KNN) algorithm** is a supervised machine learning method employed to tackle classification and regression problems. Evelyn Fix and Joseph Hodges developed this algorithm in 1951, which was subsequently expanded by Thomas Cover. The article explores the fundamentals, workings, and implementation of the KNN algorithm.

KNN is one of the most basic yet essential classification algorithms in machine learning. It finds intense application in pattern recognition, data mining_and intrusion detection.

Distance Metrics Used in KNN Algorithm

As we know that the KNN algorithm helps us identify the nearest points or the groups for a query point. To determine the closest groups or the nearest points for a query point we need some metric. For this purpose, we use below distance metrics:

Euclidean Distance

This is the cartesian distance between the two points which are in the plane/hyperplane. It is the length of the straight line that joins the two points which are into consideration. This metric helps us calculate the net displacement done between the two states of an object.

$$\text{distance}(x, X_i) = \sqrt{\sum_{j=1}^{d} (x_j - X_{i_j})^2}]$$

Manhattan Distance

Manhattan Distance is generally used when we are interested in the total distance traveled by the object instead of the displacement. This metric is calculated by summing the absolute difference between the coordinates of the points in n-dimensions.

$$d\,(x, y) = \sum_{i=1}^{n} |x_i - y_i|$$

The above-discussed metrics are most common while dealing with a ML problem but there are other distance metrics as well which come in handy while dealing with problems that require overlapping comparisons between two vectors whose contents can be Boolean as well as string values.

4.2 SVM (Linear Kernel)

A linear kernel is a type of kernel function used in machine learning, including in SVMs (Support Vector Machines). It defines the dot product between the input vectors in the original feature space. The linear kernel can be defined as:

K(x, y) = x .y

Where x and y are the input feature vectors. The dot product of the input vectors is a measure of their similarity or distance in the original feature space. When using a linear kernel in an SVM, the decision boundary is a linear hyperplane that separates the different classes in the feature space. This linear boundary is useful when the data is already separable by a linear decision boundary or when dealing with high-dimensional data, where the use of more complex kernel functions may lead to Overfitting.

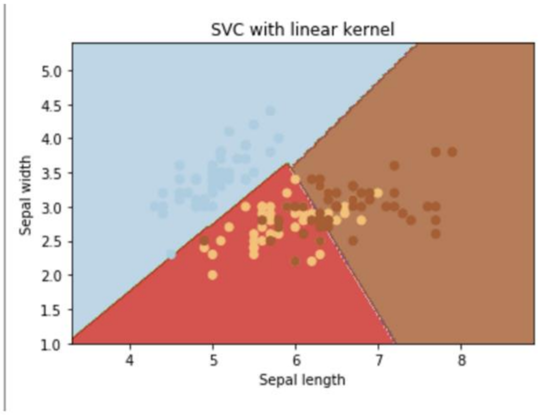


Fig.1: SVM graph with linear kernel

4.3 SVM (Gaussian (RBF) Kernel)

The Gaussian kernel, also known as the radial basis function (RBF) kernel, is a popular kernel function used in machine learning, particularly in SVMs (Support Vector Machines). It is a nonlinear kernel function that maps the input data into a higher-dimensional feature space using a Gaussian function.

The Gaussian kernel can be defined as:

K(x, y) = exp(-gamma * ||x - y||^2)

Where x and y are the input feature vectors, gamma is a parameter that controls the width of the Gaussian function, and ||x - y||^2 is the squared Euclidean distance between the input vectors.

When using a Gaussian kernel in an SVM, the decision boundary is a nonlinear hyper plane that can capture complex nonlinear relationships between the input features. One advantage of the Gaussian kernel is its ability to capture complex relationships in the data without the need for explicit feature engineering. The choice of the gamma parameter can be challenging, as a smaller value may result in under fitting, while a larger value may result in over fitting.

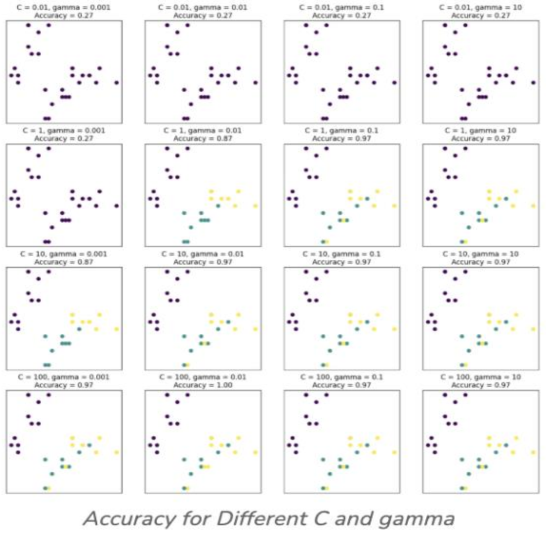


Fig.2: SVM graphs with RBF kernel

4.4 Decision Tree

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.

In a Decision tree, there are two nodes, which are the **Decision Node** and **Leaf Node.** Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.

The decisions or the test are performed on the basis of features of the given dataset. It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.

It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.

A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.

This diagram explains the general structure of a decision tree:

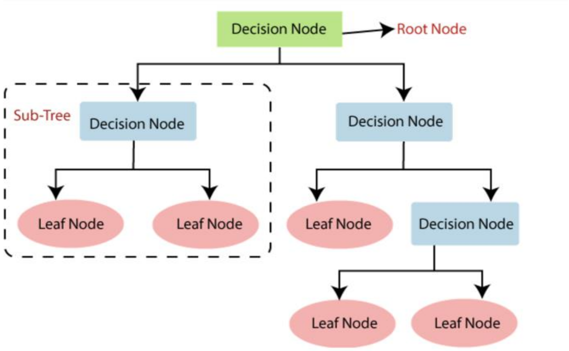


Fig.3: General structure of a decision tree

4.5 Random Forest Algorithm

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of **ensemble learning,** which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. It is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting. The below diagram explains the working of the Random Forest algorithm:

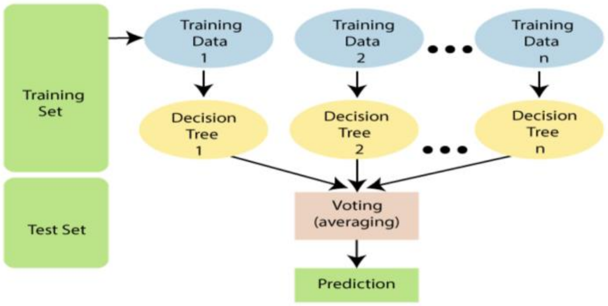


Fig.4: working of random forest algorithm

4.6 Gradient Boosting

Gradient Boosting is a powerful boosting algorithm in which each new model is trained to minimize the loss function. In each iteration, the algorithm computes the gradient of the loss function with respect to the predictions of the current ensemble and then trains a new weak model to minimize this gradient. The predictions of the new model are then added to the ensemble, and the process is repeated until a stopping criterion is met.

The weights of the training instances are not tweaked, instead, each predictor is trained using the residual errors of the predecessor as labels. There is a technique called the Gradient Boosted Trees whose base learner is CART (Classification and Regression Trees). The below diagram explains how gradient-boosted trees are trained for regression problems.

5. Methodology

5.1 Analyzing the dataset

The crop recommendation system is based on a dataset comprising around 2000 entries, taken from Kaggle, characterized by eight essential attributes: Nitrogen (N), Phosphorus (P), Potassium (K), temperature, humidity, pH, rainfall, and the target variable, "label," which indicates the type of crop. This dataset includes 22 distinct crops, with 100 samples allocated to each, thereby maintaining a balanced representation. The nutrient levels exhibit considerable variation, with nitrogen levels spanning from 0 to 140, phosphorus from 5 to 145, and potassium from 5 to 205. Additionally, environmental parameters such as temperature (ranging from 8.8 to 43.7°C), humidity (from 14.3 to 99.9%), pH (between 3.5 and 9.9), and rainfall (from 20.2 to 298.6 mm) reflect a wide array of agricultural conditions.

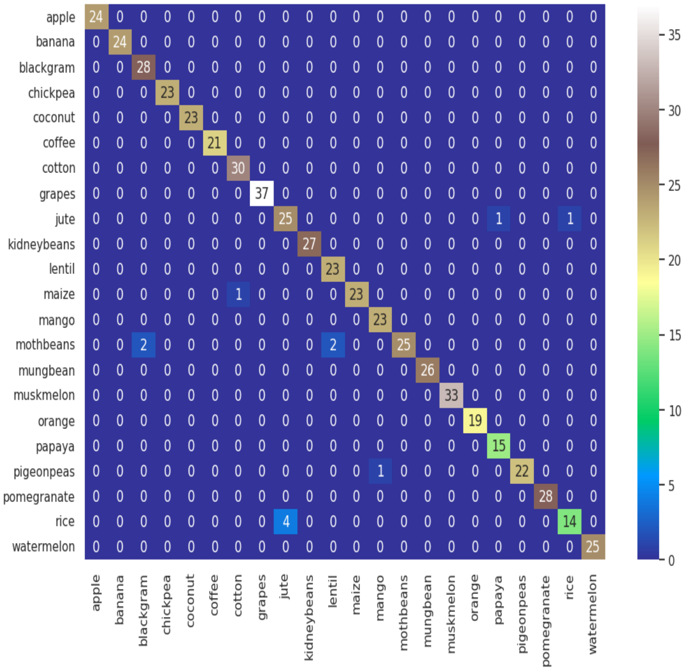
This dataset is well-suited for the development of machine learning models, utilizing features such as soil characteristics, climatic conditions, and rainfall patterns to suggest the most suitable crops for particular environments. Its design also supports integration with database management systems, allowing for effective data storage and retrieval. The extensive variety of attributes provides a thorough basis for analysis, establishing a solid groundwork for predictive modeling and classification endeavors in the field of agriculture.

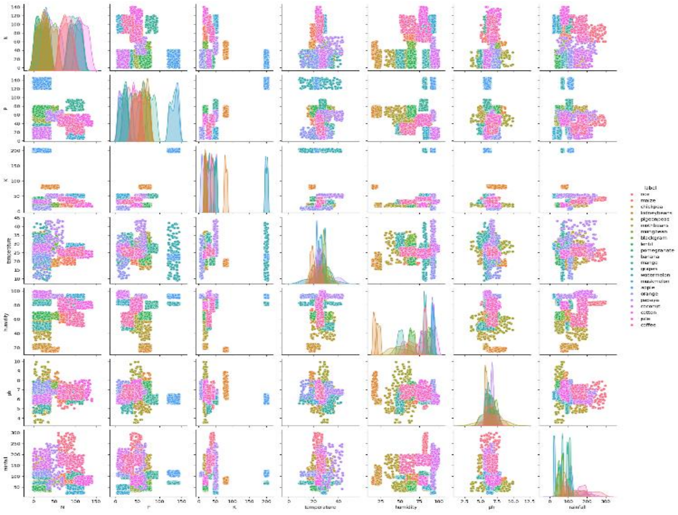
5.2 Training and Testing the Dataset

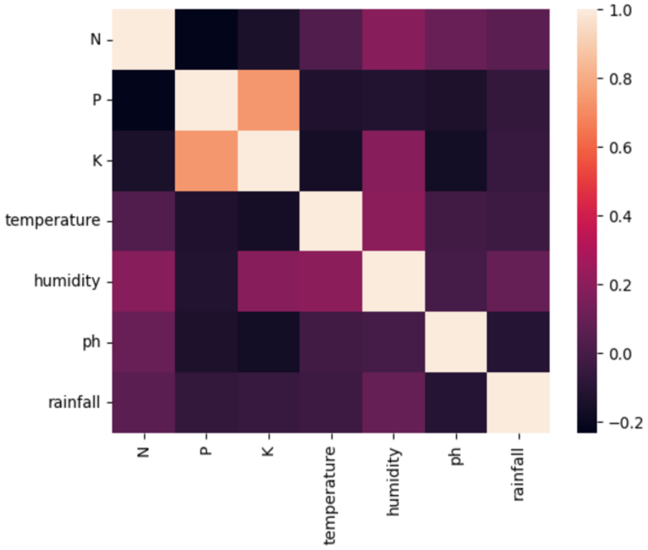
The crop recommendation system employs an extensive dataset to train and assess various machine learning models aimed at delivering precise predictions. This dataset encompasses key attributes such as nitrogen (N), phosphorus (P), potassium (K), temperature, humidity, pH, and rainfall, which undergo preprocessing to maintain high data integrity.

A range of machine learning algorithms is utilized to assess their effectiveness, including K-Nearest Neighbors (KNN), Support Vector Machines (SVM) with linear, polynomial, and radial basis function (RBF) kernels, Decision Trees, Random Forests, and Gradient Boosting. The KNN algorithm, refined through the evaluation of different k values, consistently demonstrates high accuracy, while SVM models exhibit strong performance across various kernel configurations. Random Forests, configured with a maximum depth of 4 and 100 estimators, yield outstanding results, achieving accuracies of 98% on the test dataset. Likewise, Gradient Boosting is applied to enhance classification outcomes, resulting in competitive performance.

During the training phase, hyperparameter tuning is conducted using GridSearchCV, focusing on optimizing parameters such as C and gamma for SVM models to enhance model generalization. The evaluation of model performance is facilitated through the generation of confusion matrices and classification reports, which provide comprehensive insights into accuracy and precision at the class level. An analysis of feature importance, particularly within tree-based models, highlights the most critical factors affecting crop suitability, including pH, rainfall, and nutrient levels. The models are subsequently tested on previous data, where their high accuracy rates affirm their robustness and applicability in real-world scenarios. Furthermore, the system incorporates an interactive module that allows users to input environmental and soil parameters, empowering farmers to determine the most appropriate crop for specific conditions. This fusion of machine learning and agricultural practices enhances decision-making in crop selection.







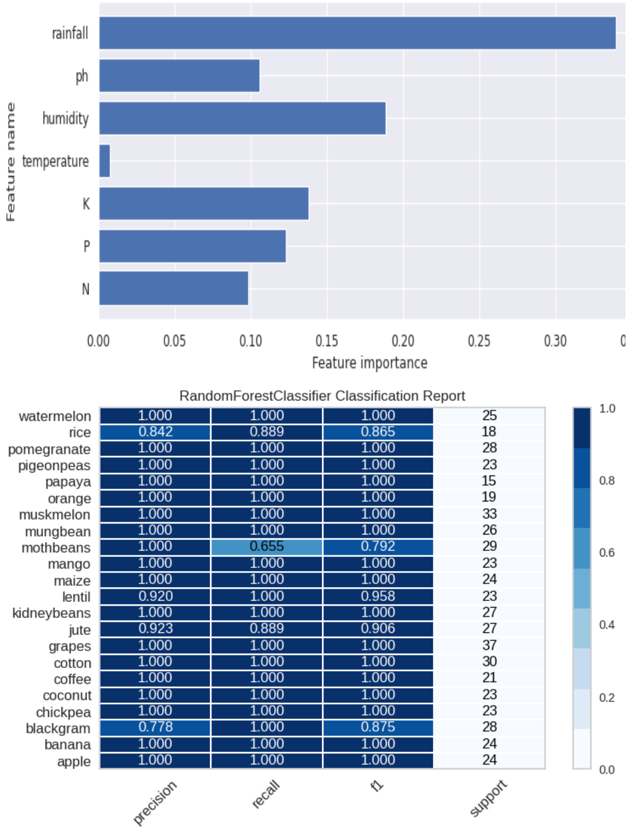
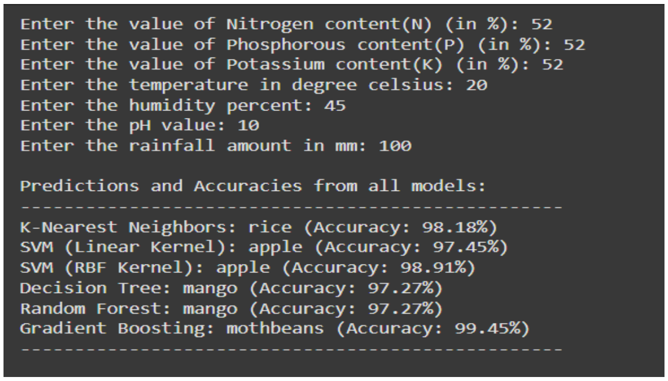


Fig.5: Training the dataset and forming various graphs and charts to understand the data trend using heatmap, confusion matrix, classification reports and other graphs.

6. Results and Observations

The findings from the crop recommendation system illustrate the relative effectiveness of various machine learning models in forecasting optimal crops based on environmental and soil parameters. Among the models assessed, the K-Nearest Neighbors (KNN) classifier, attained an accuracy ranging from 97% to 98%, contingent upon the selected value of k. The Support Vector Machine (SVM) models, tested with different kernel functions, produced competitive outcomes, with the linear kernel achieving an accuracy of 98%. In contrast, the polynomial and radial basis function (RBF) kernels yielded slightly lower accuracies, underscoring the linear kernel's appropriateness for this particular dataset. Tree-based models also demonstrated commendable performance, with the Decision Tree classifier recording an accuracy of 97% on the test dataset. Nevertheless, its performance was marginally eclipsed by the Random Forest classifier, which attained an impressive accuracy of 98%. Additionally, Gradient Boosting, exhibited a comparable accuracy of 98%, highlighting its capability to effectively manage complex data patterns.

A comparative evaluation of these models underscores the benefits of ensemble methods such as Random Forests and Gradient Boosting over simpler algorithms like KNN. These sophisticated techniques leverage their capacity to mitigate overfitting and enhance generalization to new data. This study illustrates the potential of machine learning in agricultural decision-making, offering farmers reliable guidance for optimizing crop selection and enhancing yield efficiency.



```
Enter the value of Nitrogen content(N) (in %): 52
Enter the value of Phosphorous content(P) (in %): 52
Enter the value of Potassium content(K) (in %): 52
Enter the temperature in degree celsius: 20
Enter the humidity percent: 45
Enter the pH value: 10
Enter the rainfall amount in mm: 100

Predictions and Accuracies from all models:
--------------------------------------------------
K-Nearest Neighbors: rice (Accuracy: 98.18%)
SVM (Linear Kernel): apple (Accuracy: 97.45%)
SVM (RBF Kernel): apple (Accuracy: 98.91%)
Decision Tree: mango (Accuracy: 97.27%)
Random Forest: mango (Accuracy: 97.27%)
Gradient Boosting: mothbeans (Accuracy: 99.45%)
--------------------------------------------------
```

Fig.6: the received output on Google Collab, showing all the algorithms, their predicted crop recommendation and their accuracy percentages.

7. Challenges

There can be various challenges while implement this system among various domains. Some of them include-

Data Quality and Availability: Insufficient historical or real-time data on soil, climate, crop yields, and agricultural practices, especially in developing regions.

Data inconsistency: Missing, biased, or noisy data can lead to inaccurate predictions.

Feature Engineering: Identifying and integrating diverse features such as soil nutrients, water availability, and pest infestations is complex.

Climate Change: The unpredictability and extremes of weather patterns due to climate change add complexity to prediction models.

Cost and Accessibility: Developing and

deploying advanced ML systems is expensive, and cost-effective solutions are crucial for small-scale farmers.

8. Future Developments and Practical Implementations

Integration of Advanced Data Sources: Utilizing satellite imagery, Internet of Things (IoT) sensors, and drones to gather high-resolution information regarding soil conditions, weather patterns, and crop health. This approach facilitates real-time data collection, enabling the development of dynamic and adaptive models.

Use of Hybrid Models: Merging machine learning (ML) techniques with domain-specific knowledge, such as agronomy, enhances both interpretability and accuracy. This includes the integration of statistical and physics-based models alongside ML methodologies.

Efforts are made to incorporate interpretability within ML models, thereby providing actionable insights for farmers. Additionally, the creation of user-friendly dashboards and visualization tools is prioritized.

Predictive Analytics for Sustainable Farming: The development of models capable of forecasting optimal planting schedules, irrigation requirements, and fertilizer application to enhance yield while minimizing environmental repercussions.

Adapting to Climate Change: The implementation of dynamic models that adjust to evolving climatic conditions, utilizing climate simulations and forecasting techniques. Recommendations for resilient crop varieties are provided for areas susceptible to extreme weather phenomena.

Edge Computing and IoT: The use of edge devices for on-site computation allows for the processing of real-time sensor data, thereby decreasing reliance on cloud-based systems. Mobile applications and portable devices are designed to deliver predictions directly to farmers.

Education and Training: Initiatives are in place to educate farmers and extension workers on the application of AI and ML tools, empowering them to make informed agricultural decisions.

9. Conclusion

India's prominence in the dataset, with the highest number of crops, suggests that it is a logical location for crop cultivation. Following this, rainfall and temperature are also significant factors. The model's predictions for crops were notably influenced by these variables, confirming the initial hypothesis regarding their importance.

In summary, the application of machine learning in crop yield prediction holds the potential to transform the agricultural sector. By delivering more precise forecasts, enhancing decision-making processes, boosting efficiency, and promoting sustainability, this technology can assist farmers in achieving improved yields and more lucrative operations. Although there are certain challenges associated with implementing machine learning for yield prediction, the advantages are evident, and we can anticipate ongoing progress in this area in the future.

Machine learning algorithms demonstrated limited effectiveness in predicting sunflower and wheat yields across various regions of Spain when compared to a basic average-yield benchmark. The random division of data for training and testing tends to underestimate model errors, in contrast to a time-dependent partitioning approach. While Random Forest (RF) models are generally easier and quicker to execute, they will consistently perform at least as well as the average yield estimate based on historical data, a reliability that is not assured with linear models unless there is an adequate amount of data available.

10. References

1. Ansarifar, J., Wang, L., & Archontoulis, S. V. (2023). An interaction regression model for crop yield prediction. *Scientific Reports*. Retrieved from https://doi.org

2. Crop Yield Prediction Using Machine Learning. Javatpoint. Retrieved from https://www.javatpoint.com/crop-yield-prediction-using-machine-learning

3. Crop Recommendation Dataset. Retrieved from Kaggle.

https://www.kaggle.com/code/theeyeschico/crop-analysis-and-prediction/notebook

4. Patil, P., Athavale, P., Bothara, M., Tambolkar, S., & More, A. (2023). Crop selection and yield prediction using machine learning approach. *Current Agriculture Research Journal*, *11*(3). Retrieved from http://dx.doi.org/10.12944/CARJ.11.3.26

5. Reddy, D. J., & Kumar, M. R. Crop Yield Prediction Using Machine Learning Algorithm. *PDF*.

6. K-Nearest Neighbours. GeeksforGeeks. Retrieved from https://www.geeksforgeeks.org/k-nearest-neighbours/

7. Support Vector Machine Algorithm. GeeksforGeeks. Retrieved from https://www.geeksforgeeks.org/support-vector-machine-algorithm/

8. Creating Linear Kernel SVM in Python. GeeksforGeeks. Retrieved from https://www.geeksforgeeks.org/creating-linear-kernel-svm-in-python/

9. Gradient Boosting. GeeksforGeeks. Retrieved from https://www.geeksforgeeks.org/ml-gradient-boosting/

10. Machine Learning: Random Forest Algorithm. Javatpoint. Retrieved from https://www.javatpoint.com/machine-learning-random-forest-algorithm

11. Technical Advances in Plant Science. (2023). *Frontiers in Plant Science, 14*. Retrieved from https://doi.org/10.3389/fpls.2023.1128388