AGRICULTURAL CROP YIELD PREDICTION USING MACHINE LEARNING TECHNIQUES

Simanta Hazra[a]
[a]Dept. of Computer Science and Engineering, Asansol Engineering College, West Bengal, India

Sunil Karforma[b]
[b]Dept. of Computer Science, University of Burdwan, West Bengal, India

Abhishek Bandyopadhyay[c]
[c]Dept. of Computer Science and Engineering (AIML), Asansol Engineering College, West Bengal, India

Shemim Begum[d]
*[d]Dept. of Computer Science and Engineering, Govt College of Engg and Textile Technology, Berhampore ,
West Bengal, India*

Debasis Chakraborty[e]
[e]Dept. of Computer Science and Engineering (IoT, CS, BCT), Asansol Engineering College, Asansol, India

***Corresponding Author: -** Simanta Hazra
*[a]Dept. of Computer Science and Engineering, Asansol Engineering College, West Bengal, India
Tel: + 919434188971, E-mail: simanta.cse@aecwb.edu.in,*

Abstract – In many developing nations, agriculture serves as the primary economic pillar. The sustenance and growth of the global population in the 21st century hinge upon a significant increase in food production. Hence, an essential aspect of agricultural expansion revolves around the precise prediction of crop yield and crop type, leveraging machine learning (ML) algorithms. This paper is dedicated to identifying the most effective predictive model that can empower farmers with accurate crop yield and crop type forecasting. Seven well-known algorithms, including K-Nearest Neighbor (KNN), Random Forest (RF) Classifier, Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine (LightGBM), Artificial Neural Networks (ANN), Support Vector Machine (SVM), and Multi-layer Perceptron (MLP), are meticulously examined using numerical crop datasets, and their performance is thoroughly analyzed. The experimental findings reveal that among these algorithms, XGBoost stands out as the top performer in terms of accuracy. Additionally, the study delves into the classification of crop types using transfer learning in Convolutional Neural Networks (CNN) on a separate crop image dataset. Various models, such as VGG-19 (Visual Geometry Group-19), VGG-16, and Inception v3, pretrained with weights from ImageNet, are explored. The results, evaluated based on accuracy and the kappa coefficient, demonstrate that VGG-19 surpasses other techniques in effectively identifying crop types from images.

Keywords – ML techniques, Extreme Gradient Boosting, Multi-Layer Perceptron, Convolutional Neural Networks, VGG-19, Image classification, Crop yield prediction.

1. INTRODUCTION

The agricultural industry plays a pivotal role in the economic development of any country, especially in the face of a growing global population, rapidly changing environmental conditions, and the challenge of meeting the nutritional needs of all. According to a study by Valin et al. [1], the demand for food is expected to surge by 59-98% by the year 2050, necessitating a substantial increase in agricultural output. Accurate yield forecasts are crucial for informed decision making by planning authorities in the realm of agriculture. To achieve this, agro-economists are tasked with devising simple yet precise computational methods for predicting crop yields. The prediction of crop yield is influenced by various input factors, including geographical location, irrigation techniques, temperature, and pesticide application.

Pesticides, while essential for boosting agricultural productivity, pose risks to non-target plants, aquatic life, birds, beneficial insects, soil quality, water bodies, crops, and non-target animals due to their chemical nature designed to combat pests and weeds. Environmental degradation often results from the unintended spread of pesticide contamination beyond their intended targets. Despite these challenges, pesticide usage remains necessary to meet food demands.

Machine Learning (ML) has emerged as a valuable tool for forecasting food production. ML approaches have greatly improved crop prediction research, as they have been applied to various industries, ranging from retail behavior analysis to predicting phone usage patterns. Crop forecasting, however, remains a complex issue within agriculture, given the multitude of factors that impact crop yields, including atmospheric conditions, fertilizer types, soil quality, and seed characteristics. Consequently, estimating agricultural output involves a series of intricate steps and is far from a straightforward process. While existing crop yield prediction technologies perform reasonably well, there is an ongoing quest for enhanced accuracy. The agricultural sector faces both new opportunities and challenges, necessitating the development of effective techniques for applications such as forestry, disaster management, and agriculture itself. For a more in-depth exploration of this topic, interested readers are encouraged to consult the comprehensive study conducted by Sheykhmousa et al. in their publication [2].

Given the importance of crop prediction, numerous suggestions have been made to improve its accuracy. Current research indicates that ML algorithms hold more promise in this regard than traditional statistical methods [3, 4, 5]. A wide array of classification methods, including those based on Artificial Neural Networks (ANNs), have shown promise in crop yield prediction [6, 7, 8, 9, 10]. Specifically, ANN architectures [11], CNN-RNN hybrids [12], hybrid MLR-ANN models [13], and hybrid CNN-LSTM techniques [15] have all demonstrated excellent performance.

This paper makes significant contributions in several areas. It explores crop damage prediction using various ML approaches, including XGBoost[16], MLP[17], LightGBM, KNN [18], SVM [19, 20], ANN[21] and Random Forest (RF)[22]. Through a comparative analysis on a crop dataset, it identifies XGBoost as the most accurate predictive approach, closely followed by MLP and LightGBM. Furthermore, the paper delves into the realm of CNN for large-scale crop image recognition. By increasing the depth of layers from 16 to 19, the study demonstrates that deeper architectures can outperform other ImageNet models, such as VGG-16 and Inception v3.

In the subsequent sections of this paper, we provide a concise overview of the diverse ML algorithms in Section 2. Section 3 details the crop dataset employed in our experiments, with the results presented in Section 4. Finally, Section 5 concludes the paper, highlighting avenues for future research.

2. Machine Learning Techniques

2.1 We examine seven machine learning methods, specifically XGBoost[16], MLP[17], Light-GBM, KNN[18], SVM, ANN, and Random Forest (RF)[22], to assess the empirical performance on a crop dataset. The algorithms are briefly introduced as follows:

1. MLP classifier: Essentially, it comprises two primary steps. In the forward pass, projected outputs are computed based on the provided inputs. During the backward pass, partial derivatives of the cost function with respect to individual parameters are propagated back through the network. A typical Multi-Layer Perceptron (MLP) network consists of an input layer composed of multiple source nodes, one or more hidden layers comprising computational nodes, and an output layer with nodes. For a more detailed understanding, readers can refer to [17].

2. LightGBM: This approach relies on the gradient boosting framework with a focus on the 'Decision Tree' component. It's a speedy, distributed, and efficient technique applicable to a wide range of machine learning tasks, encompassing both classification and regression. It is a gradient boosting framework that primarily employs decision trees as its base learners. These decision trees are an integral part of the algorithm's model building process. LightGBM builds decision trees sequentially during the training process, and each decision tree incrementally corrects errors made by the previous ones, ultimately leading to the construction of a strong ensemble model. While it enables significantly faster training, it should be noted that there is a potential risk of occasional overfitting.

3. K – Nearest Neighbor Classifier: K-Nearest Neighbors (KNN), as introduced by Tougonza in 1974 [18], stands as one of the fundamental machine learning algorithms. It operates on the principle that a new data instance can be placed into a category akin to the existing ones, assuming comparability between the new and previously observed cases. This algorithm achieves classification by leveraging the stored historical data, categorizing new data points based on their similarity to the existing data points. Consequently, KNN offers a reliable and swift means of classifying novel data instances. Notably, while KNN is predominantly used for classification tasks, it can also be adapted for regression. An essential characteristic of KNN is its non-parametric nature, signifying that it refrains from making any specific assumptions about the underlying data distribution. This quality renders KNN versatile and applicable to a wide range of data scenarios.

4. Random Forest: Random Forest (RF), introduced by Breiman in 2001 [22], is a technique used to build an ensemble of Decision Trees (DTs). Breiman adopted a randomization approach, compatible with both bagging and random subspace techniques, to introduce variability among the base DTs. This process involves using bootstrap samples from the training dataset. Here's a breakdown of the steps involved in generating each individual tree within the Random Forest:

i) **Bootstrap Sampling**: Given a training dataset with 'S' records, 'S' records are randomly and with replacement selected from the original data. These bootstrap samples serve as the training set for the tree.

ii) **Feature Randomization**: If there are 'S' input variables, 's' (where 's' is a fixed value) variables are chosen at random from the 'S' available variables at each node of the tree. The node is then divided using the best split found among these 's' features.

iii) **Full Potential without Pruning**: Unlike in some other tree-based algorithms, Random Forest does not prune the individual trees. Each tree is allowed to grow to its full extent, optimizing its performance on the training data.

The combination of multiple DTs, each trained on a bootstrap sample and utilizing random subsets of features, forms the Random Forest ensemble. The ensemble approach harnesses the wisdom of many individual trees to provide robust predictions while reducing the risk of overfitting.

5. **Support Vector Machines:** Support Vector Machines (SVMs) are powerful supervised learning models used for classification and regression tasks. They are particularly effective in high-dimensional spaces and are widely used in various fields such as image classification, bioinformatics, text mining, agriculture and more. It aims to find the hyperplane that best separates different classes in the feature space. The algorithm finds the hyperplane with the maximum margin, which is the distance between the hyperplane and the nearest data point from either class, also known as support vectors. SVM can handle non-linear decision boundaries by transforming the input space into a higher-dimensional space using kernel functions (e.g., polynomial, radial basis function (RBF), sigmoid), allowing it to learn complex relationships.

6. **Artificial Neural Network:** Artificial Neural Networks (ANNs) are a fundamental component of machine learning and artificial intelligence, inspired by the structure and function of biological neural networks in the human brain. ANNs are a computational model composed of interconnected nodes, called neurons or units, organized in layers. Each neuron receives input signals, processes them, and produces an output signal that is transmitted to other neurons. The strength of connections between neurons, known as weights, is adjusted during the learning process to enable the network to solve specific tasks. Structure of Artificial Neural Networks:

i) Input Layer: The first layer of the network, where input data is introduced. Each neuron in this layer represents a feature or attribute of the input data.

ii) **Hidden Layers**: Intermediate layers between the input and output layers where the processing of data occurs. These layers are called "hidden" because their activities are not directly observable from the outside. Deep Neural Networks contain multiple hidden layers, allowing for complex transformations of data.

iii) **Output Layer**: The final layer of the network that produces the output or prediction. The number of neurons in this layer depends on the nature of the task (e.g., binary classification, multi-class classification, regression).

7. Extreme Gradient Boosting Algorithm

XGBoost, a groundbreaking tree-based approach, has garnered significant attention as an exceptionally effective method for data classification. XGBoost employs an ensemble learning technique rooted in decision trees (DTs) and excels in both classification and regression tasks, leveraging a gradient boosting framework [16, 26]. The approach taken by the developers is characterized by a clearly articulated philosophy, and here's how the algorithm operates is shown in Algorithm 1. The readers may refer to [16, 25] for details.

Input: training set $\{(x_i, y_i)\}_{i=1}^{n}$, a differentiable loss function L(y, F(x)), number of iterations M.

Algorithm 1:

1. Initialize model with a constant value:

F0(x) = arg min $\sum_{i=1}^{n} L(y_i, \gamma)$

2. For m = 1 to M:

Compute so-called pseudo-residuals:

$r_{im} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}$ for i – 1,…,n.

3. Fit a base learner (e.g. tree) hm(x) to pseudo-residuals, i.e. train it using the training set $\{(x_i, r_{im})\}_{i=1}^{n}$

4. Compute multiplier γ_m by solving the following problem:

$$\gamma_m = \arg\min \sum_{i=1}^{n} L(y_i, F_{m-1}(xi) + \gamma h_m(x_i))$$

5. Update the model:

$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x)$

6. Output FM(x).

2.2. Transfer Learning in CNN using VGG-16, VGG-19

depth of the network.

i) **Input Layer**: This layer accepts the input image of size (224×224) pixels

ii) Convolutional Layers: The network's initial two sets of convolutional layers are followed by the max-pooling layer. Two convolutional layers having 64 filters make up the first set, while next two layers with 128 filters make up the second.

iii) Convolutional Layers: The pattern is repeated for the following two sets of

In this section, VGG-16 is briefly described as follows:

1. VGG-16 and VGG19: Visual Geometry Group-16, is a CNN architecture [23] which is a very deep convolutional networks for large scale image recognition. VGG-16 is known for its simplicity and effectiveness in image classification tasks. It consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. The network's architecture is characterized by using a series of 3x3 convolutional filters throughout the entire depth of the network, which leads to a better representation of the input data. It is famous for being both straightforward and efficient in picture classification tasks. It has 16 layers, including 3 fully linked layers and 13 convolutional layers. To augment the representation of the data input, the network's architecture uses a succession of (3 ×3) convolutional filters over the whole convolutional layers. Three convolutional layers with 256 filters make up the third set, while next three layers having 512 filters make up the fourth.

iv) Convolutional Layers: Three layers containing 512 filters make up the fifth set.

v) Completely Connected Layers: VGG-16's final three layers are completely connected layers. The last completely connected layer, which serves as the output layer, has the number of matching neurons to the number of classes in the classification job, while the first two fully connected layers each have 4,096 neurons.

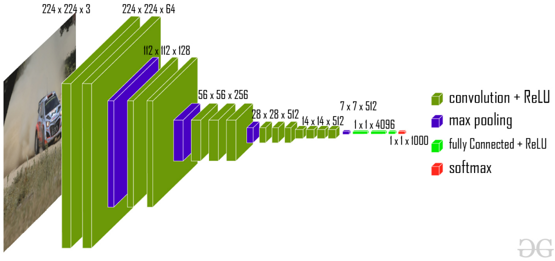


Figure 1: The architecture of VGG-16 [24]

vi) Activation Function: Beside the output layer, each convolutional and fully connected layer is followed by the ReLU (Rectified Linear Unit) activation function.

A common baseline model for a number of computer vision applications, such as image classification, object identification, and picture segmentation, is the VGG-16 architecture. Deep learning researchers and practitioners favour it because of its ease of use and strong performance on benchmark datasets. VGG-19 is an extended version of VGG-16 having 19 layers, including 3 fully linked layers and 16 convolutional layers. VGG-16 architecture [24] is depicted in Fig. 1.

2. InceptionV3: This CNN architecture that was developed by Google researchers in 2015. It is part of the Inception series of CNN architectures, which are designed to address the problem of efficiently and effectively training deeper neural networks for image classification tasks. The key innovation of Inception V3 lies in its use of so-called ”Inception modules,” which are small subnetworks that perform multiple parallel convolutions of different sizes on the input data and then concatenate the results together. This approach allows the network to capture features at various spatial scales and resolutions, enabling it to recognize patterns in images of different sizes and complexities.

3. Dataset Used

In this paper, two crop datasets are investigated. First one is numerical data which is downloaded crop dataset from https://github.com/Shrey-B/AV-Janatahack-Machine-Learning-in-Agriculture obtained from US field data collection. The dataset contains 88858 labeled samples, eight features and three classes. The details of the dataset is given in Table 1. Second one is a collection of crop image data. Data plays a crucial role to build any effective model using any ML / deep learning technique, because the model is trained based on the ground truth image data, so the data must be insightful and meaningful to solve our task. This Data is downloaded from https://www.kaggle.com/datasets/aman2000jaiswal/agriculture crop images dataset contains 1005 samples having five type of images namely maize, wheat, jute, rice and sugarcane. Fig. 2 depicts the dataset in detail. Sample crop image (The size of each image :224 × 224 pixels) for each category is illustrated in Fig. 3.

Table 1: Description of the dataset

Feature name	Definition (feature values)
Estimated Insects Count	Estimated insects count per square meter
Type of Crop	Category of Crop (0, 1)
Type of Soil	Category of Soil (0, 1)
Type of Pesticide	Pesticides use category (1- Never, 2-Previously Used 3-Currently Using)
# Doses per Week	Number of doses per week
# Weeks Used	Number of weeks used
# Weeks Quit	Number of weeks quit
Season	Season Category (1, 2, 3)
Label of Crop Damage	Crop Damage Category (0=alive, 1=Damage for other reasons 2=Damage for use of Pesticides)

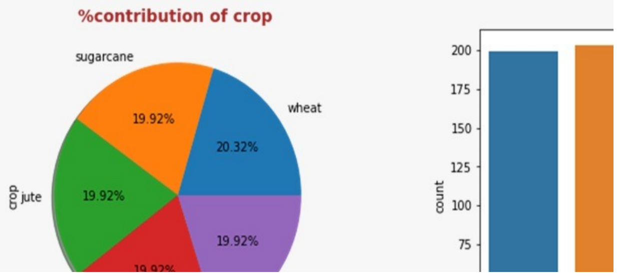


Figure 2: Crop label contains the five crops namely sugarcane, wheat, rice, maize, jute .Wheat crop contains the 204 images and remaining crops contain the 200 images for each crop type.

4. Experimental Results

Python 3.9.13 (Anaconda IDE) is used to implement the algorithms in our study while Windows 11 is the operating system. On the crop dataset, trials are run to compare XGBoost's performance to that of other algorithms as MLP, SVM, ANN, LightGBM, RF, and K-NN classifier. The pertinent recommendations from the literature were used to set the essential parameters of the DT-based classifiers.

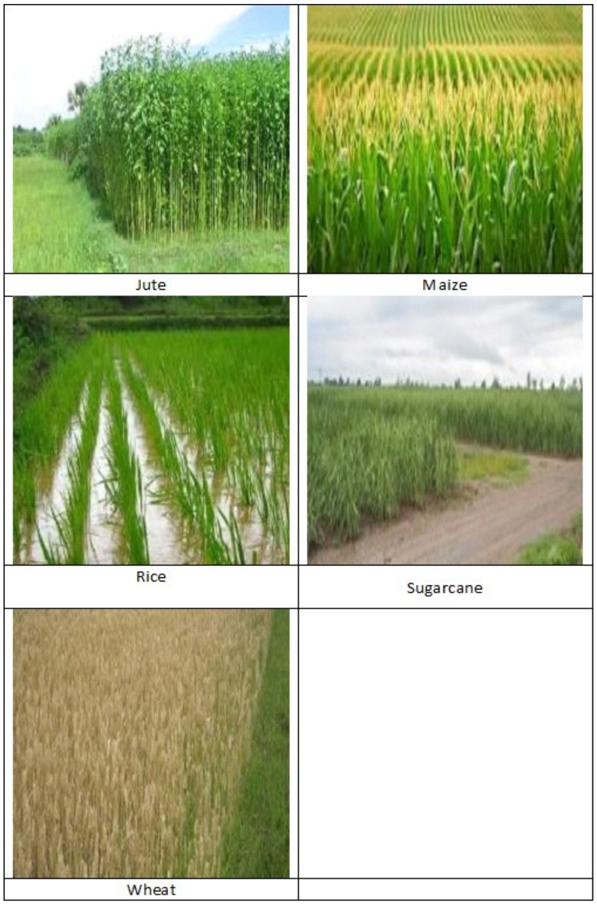


Figure 3: Sample Crop Image

For k-NN classifier, k is set to 7. K-fold cross validation is used to estimate the average accuracy and standard deviation. To compare the empirical performance of XGBoost with other algorithms including MLP, SVM, ANN, LightGBM, RF, and K-NN classifier, experimentation is done with the crop dataset. The pertinent recommendations from the literature are used to set the essential parameters of the DT-based RF classifiers. The k value for the KNN classifier is considered as 7. The average accuracy and standard deviation are calculated using K-fold cross validation.

4.1. Evaluation Criteria

The predicted efficacy of various algorithms is assessed using the standard deviation (SD), boxplot, and 10 fold cross validation accuracy.

1. K-fold cross validation: The steps in this process are as follows. First, 10(K =10) subgroups of equal size are selected at random from the training samples. The classifier is then trained ten times, using the remaining nine subsets to train the classifier while holding out one subset at a time. The classifier is then put to the test on the subset that was held out, and the classification accuracy is noted. To get the average accuracy, the accuracies are averaged.

2. Standard deviation (SD): The degree to which a set of numbers can fluctuate is estimated. A low SD indicates that the values tend to be closer to the established mean, whereas a large SD suggests that the values are distributed across a wider range.

3. Boxplot : A type of chart called a boxplot is often used in descriptive statistics' explanatory data analysis. Boxplots employ the quartiles (or percentiles) and averages of the data to show the distribution and skewness of numerical data.

4. Cohen's [27] kappa index: It is statistic for evaluating the degree of agreement between two raters is the kappa index. The categorization accuracy is estimated using this statistic. Each entry of the confusion matrix, Cij, represents the number of records from cluster i that have been mapped to cluster j in order to measure the agreement with kappa. As a result, entries in the diagonal display the correctly mapped counts. Eqn 1 is used to calculate the overall accuracy and kappa index is computed using Eqn. 2.

$$\%\text{correct}=\frac{\sum_{k} C_{kk}}{n}\times 100 \qquad (1)$$

$$\text{Kappa}=\frac{n\times\sum_{k} C_{kk}-\sum_{k} C_{k+}C_{+k}}{n^2-\sum_{k} C_{k+}C_{+k}} \qquad (2)$$

where $C_{k+}=\sum_{j} C_{kj}$, $C_{+k}=\sum_{i} C_{jk}$ and n is the number of total samples. Kappa ranges from 0 to 1. Larger value of kappa indicates better accuracy.

Table 2: Standard deviation and average accuracy produced by different algorithms

Dataset	Method	SD	Average accuracy (%)
Crop data	XGBoost	0.0019	84.79
	MLP	0.0011	84.49
	LightGBM	0.0010	84.54
	SVM	0.0051	84.07
	ANN	0.0012	83.54
	RF	0.0003	82.48
	K-NN	0.0003	82.81

Table 3: Accuracy of different methods on the test set of crop image data

Dataset	Method	Accuracy	Kappa index	Agreement
Crop image	VGG-19	89.55	0.869	Almost perfect
	VGG-16	86.56	0.832	Almost perfect
	Inception v3	82.08	0.795	Substantial

accuracy, while MLP and LightGBM are very close competitors. It is to be noted that RF has lowest performance among seven algorithms. Moreover, for the purpose of illustration, Fig.4 illustrate the boxplot representing %accuracy obtained by seven algorithms. It is clear from the figure that the boxplot corresponding to XGBoost is situated at the upper left corner, which indicates that XGBoost results in higher accuracy score than those produced by other methods. Next, we reported ROC Curve of XGBoost in Fig.5 to illustrate one versus rest classification. However, ROC curves of other algorithms are not included in this paper.

Subsequently, crop image data containing total 1005 image samples is explored. The dataset is first partitioned to training and test dataset having 804 and 201 images, respectively. The training set contains 804 samples consisting of jute(160 samples), maize(160 images), rice(160 images), sugarcane(160 images) and wheat(164 images) while, test set has jute(40 images), maize(40 images),rice(40 images), sugarcane(40 images) and wheat(41 images). Sample image of each category is depicted in Fig. 3. Table 3 shows accuracy, kappa index and agreement of three CNN models. It can be observed from the table that VGG-19 has the best performance compared to other two models. Confusion matrix for three models viz, VGG-19, VGG-16 and InceptionV3 are presented in tables 4, 5 and 6, respectively.

In Fig. 6, we have depicted train and validation accuracy versus epoch as well as train and validation loss versus epoch for three techniques VGG-19, VGG-16 and Inception v3, respectively. It is to be noted that accuracy does not necessarily increase with epoch size. In a neural network, all of the training data is applied to fine-tune the model parameters after one epoch. Up to a certain point, epoch size might increase precision; after that point, the model starts to overfit the data. A bad fit will also occur from having a really low level. The ideal number of epochs, which should be reached when deep learning accuracy stops advancing, is often between 1 and 10. In our investigation, the number of epochs is found to be four, after that point overfitting occurs.

4.2. Classifiers' Performance

We have investigated the effectiveness of the different algorithms using 10 fold cross validation. Average accuracy and SD are reported in Table 2. From the table, we observe that XGBoost has the highest average.

Table 4: Confusion matrix of VGG-19 on validation set of size 201

	Predicted					
Actual		Jute	Maize	Rice	Sugarcane	Wheat
	Jute	35	0	2	2	1
	Maize	0	37	0	1	2
	Rice	0	1	33	0	6
	Sugarcane	0	2	2	35	1
	Wheat	0	1	0	0	40

Table 5: Confusion matrix of VGG-16 on validation set of size 201

	Predicted					
Actual		Jute	Maize	Rice	Sugarcane	Wheat
	Jute	35	0	0	3	1
	Maize	0	34	0	6	0
	Rice	0	0	25	7	8
	Sugarcane	0	0	0	40	0
	Wheat	0	0	1	1	39

Table 6: Confusion matrix of Inception v3 on validation set of size 201

	Predicted					
l Actual		Jute	Maize	Rice	Sugarcane	Wheat
	Jute	36	3	0	0	1
	Maize	0	38	1	0	1
	Rice	1	3	28	1	7

	Sugarcane	1	7	1	30	1
	Wheat	0	4	1	0	36

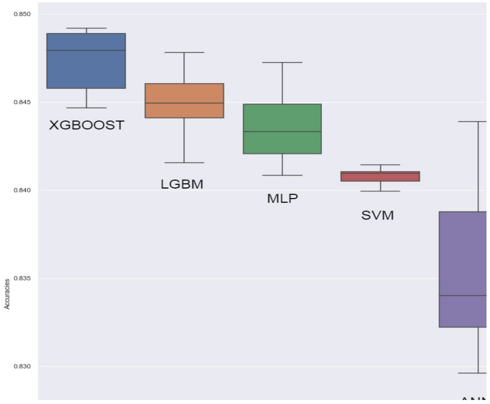


Figure 4: The boxplot showing the accuracies produced by the different algorithms using 10 fold cross validation

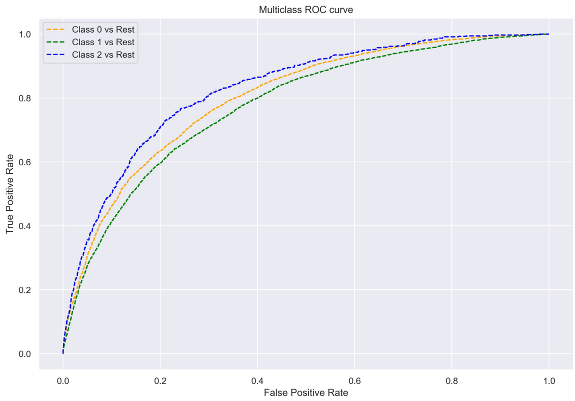
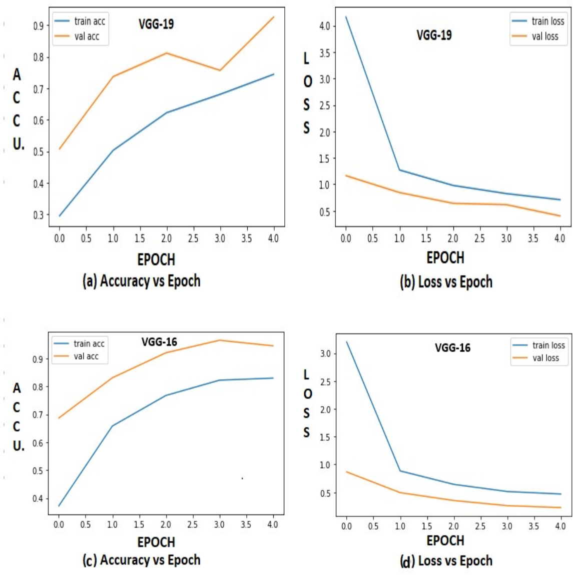


Figure 5: ROC curve obtained for XGBoost

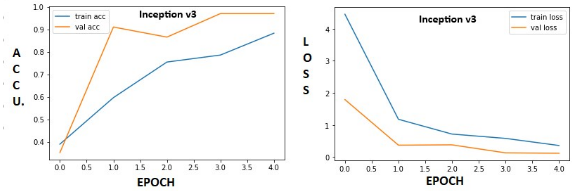


(a) Accuracy vs Epoch

(b) Loss vs Epoch

(c) Accuracy vs Epoch

(d) Loss vs Epoch



(e) Accuracy vs Epoch

(f) Loss vs Epoch

Figure 6: VGG-19: (a)Train and validation accuracy versus epoch and (b) Train and validation loss versus epoch; VGG-16: (c) Train and validation accuracy versus epoch and (d) Train and validation loss versus epoch and Inception v3: (e) Train and validation accuracy versus epoch and (f) Train and validation loss versus epoch.

5. Conclusion

Seven distinct supervised machine learning models (XgBoost, SVM, ANN, LightGBM, MLP, KNN, and RF) are compared in order to forecast crop yield for given soil type, crop type, pesticide use, etc. as features. These predictions can assist farmers in making decisions regarding the usage of pesticides, crop varieties, soil types, and other factors that will help them grow crops more effectively. Finally, we draw the conclusion that the XGBoost Classifier exhibits the best accuracy with the crop dataset. On the other hand, RF and KNN offer poor prediction performance, while MLP and LightGBM are in second position in terms of accuracy. We have also addressed the issue of crop image recognition using three CNN models (VGG-19, VGG-16 and Inception v3). Study on crop image reveals that VGG-19 has the best performance among three. As a scope for further work, there is need to investigate crop type prediction using ensemble of ML/CNN models to achieve better results.

References

1. Valin et al, ”The future of food demand: understanding differences in global economic models,”. Agric. Econ., 45(1): pp. 51-67,2014.
2. M. Sheykhmousa et al., ”Support Vector Machine Versus Random Forest for Remote Sensing Image Classification: A Meta-Analysis and Systematic Review,” IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 6308-25,2020.
3. X. E. Pantazi, D. Moshou, T. Alexandridis, R. L. Whetton, and A. M. Mouazen, “Wheat yield prediction using machine learning and advanced sensing techniques,” Comput. Electron. Agricult., vol. 121, pp. 57–65, Feb. 2016.
4. D. Elavarasan, D. R. Vincent, V. Sharma, A. Y. Zomaya, and K. Srinivasan, “Forecasting yield by integrating agrarian factors and machine learning models: A survey,” Comput. Electron. Agricult., vol. 155, pp. 257–282, Dec. 2018.
5. T. U. Rehman, M. S. Mahmud, Y. K. Chang, J. Jin, and J. Shin, “Current and future applications of statistical machine learning algorithms for agricultural machine vision systems,” Comput. Electron. Agricult., vol. 156, pp. 585–605, Jan. 2019.
6. M. D. Johnson, W. W. Hsieh, A. J. Cannon, A. Davidson, and F. B´edard, “Crop yield forecasting on the Canadian prairies by remotely sensed vegetation indices and machine learning methods,” Agricult. Forest Meteorol., vols. 218–219, pp. 74–84, Mar. 2016.
7. M. Shahhosseini, G. Hu, and S. V. Archontoulis, “Forecasting corn yield with machine learning ensembles,” Frontiers Plant Sci., vol. 11, p. 1120, Jul. 2020
8. H. Yalcin, “An approximation for a relative crop yield estimate from field images using deep learning,” in Proc.

8th Int. Conf. Agro-Geoinf. (AgroGeoinf.), Jul. 2019, pp. 1–6.

9. S. Khaki and L. Wang, "Crop yield prediction using deep neural networks," Frontiers vol. 10, p. 621, May 2019
10. N. Prasad, N. Patel, and A. Danodia, "Crop yield prediction in cotton for regional level using random forest approach," Spatial Inf. Res., vol. 29, pp. 1–12, Jul. 2020.
11. W.W. Guo and H. Xue, "Crop yield forecasting using artificial neural networks: A comparison between spatial and temporal models," Math. Problems Eng., vol. 2014, pp. 1–7, Jan. 2014.
12. S. Khaki, L. Wang, and S. V. Archontoulis, "A CNN-RNN framework for crop yield prediction," Frontiers Plant Sci., vol. 10, p. 1750, Jan. 2020.
13. P. S. M. Gopal and R. Bhargavi, "A novel approach for efficient crop yield prediction," Comput. Electron. Agricult., vol. 165, Oct. 2019.
14. M. Rashid, B. S. Bari, Y. Yusup, M. A. Kamaruddin and N. Khan, "A Comprehensive Review of Crop Yield Prediction Using Machine Learning Approaches With Special Emphasis on Palm Oil Yield Prediction," in IEEE Access, vol. 9, pp. 63406-63439, 2021, doi: 10.1109/ACCESS.2021.3075159.
15. J. Sun, L. Di, Z. Sun, Y. Shen, and Z. Lai, "County-level soybean yield prediction using deep CNN-LSTM model," Sensors, vol. 19, no. 20, p. 4363, Oct. 2019.
16. T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system. In Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August,pp. 785–794, 2016.
17. R. Hecht-Nielsen, "Theory of the backpropagation neural network," in Proc. Int. Joint Conf. Neural Networks (IJCNN), 1989, pp. 445–448.
18. J. T. Tou and R. C. Gonzalez, Pattern Recognition Principles, Reading: Addison-Wesley, 1974.
19. V. Vapnik, Statistical Learning Theory, New York, Wiley, 1998.
20. C. J. C. Burges, "A tutorial on support vector machines for pattern cecognition," Knowledge Discovery and Data Mining, vol. 2, pp. 121–167, 1998.
21. N. Gandhi, O. Petkar and L. J. Armstrong, "Rice crop yield prediction using artificial neural networks," 2016 IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR), Chennai, India, 2016, pp. 105-110, doi: 10.1109/TIAR.2016.7801222.
22. L. Breiman, "Random forests," Machine learning, vol. 45, no. 1, pp. 5–32, 2001.
23. K. Simonyan and A Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", International Conference on Learning Representations (ICLR 2015), https://doi.org/10.48550/arXiv.1409.1556.
24. website : https://www.geeksforgeeks.org/vgg-16-cnn-model/
25. J. Parashar, Sumiti, M. Rai,"Breast cancer images classification by clustering of ROI and mapping of features by CNN with XGBOOST learning. Materials Today: Proceedings,2020, http://dx.doi.org/10.1016/j.matpr.2020.09.650.
26. T. Chen, T.T. He, and M. Benesty, xgboost: Extreme Gradient Boosting; R Package Version 0.3-0; Technical Report; R Foundation for Statistical Computing: Vienna, Austria.
27. J. Cohen, "A coefficient of agreement for nominal scales," Educational and Psychological Measurements, vol. 20, no. 12, pp. 37–46, 1960.