Optimization of Crop Production using Machine Learning

Ayush Sagore*, Mohit Rakhade**, Riddhi Jha***, Rushikesh Kate****

(Department of Information Technology, Government College of Engineering,Amravati - 444601,

Maharashtra, India

Abstract

There is always a need to enhance food production to meet the food needs of the world. Agriculture suffers due to a lack of water management, land allocation, crop selection, climate changes, transportation, labour, marketing, and other resources, leading to low crop production.
Optimization of crop production software is the integration of technology and digital mechanisms into agricultural processes for more efficient output. Optimization of Crop Production is a WEBSITE application and a trained model which provided a machine learning–aided software for farmland optimization, using various inputs such as location, crop type, soil type, soil pH, amount of Nitrogen, Phosphorus, and Potassium, location temperature, humidity and spacing between crops.
Keywords – Machine Learning, Deep Learning, TensorFlow, Keras, K Means Clustering, Convolution Neural Network, Logistic Regression.

I. INTRODUCTION

Agriculture is the art and science of growing plants, and crops and raising animals for food and other needs of human or economic gain and it is the most active activity in the world. An increase in the world population has increased the food needed. Also, Agriculture suffers due to a lack of water management, land allocation, crop selection, climate changes, transportation, labour, marketing, and other resources, leading to low crop production. So, there is always a need to enhance food production to meet the food needs of the world. Optimization of crop production software is the integration of technology and digital mechanisms into agricultural processes for more efficient output.

Optimization of Crop Production is a WEBSITE application and a trained model which provided a machine learning–aided software for farmland optimization, using various inputs such as location, crop type, soil type, soil pH, amount of Nitrogen, Phosphorus, and Potassium, location temperature, humidity and spacing between crops.

The entire application has three modules Crop prediction module, crop cluster module, and crop requirement module. The crop prediction module is embedded with a model which predicts crops that grow in a specific condition. The crop cluster module is used to find out the list of the crops that can grow with a given crop. The last crop requirement module provides us with certain requirements and conditions to grow the specific crop.

II. METHODOLOGY

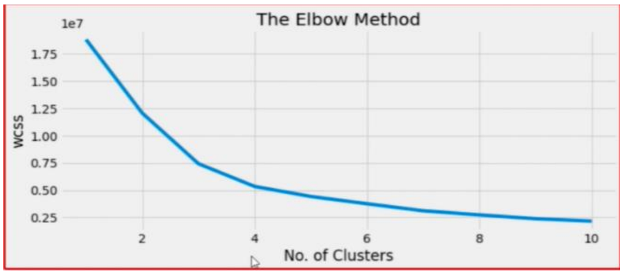
A. Dataset

The data set used in the project for the CNN model consists of 22803 images, including plant (leaf) with disease images and plant(leaf) without disease images. All of the images were sourced from Kaggle datasets. The images in the dataset cover diverse types of plant disease images. In our approach, we have dedicated 80% of the dataset as the training data and the remaining 20% as the testing (and validation) data, which makes the split ratio 0.8:0.2 of the train to test set.

The data set used for the Clustering model contains 2200 rows and 8 columns namely N (Nitrogen), P (Phosphorous), K (Potassium), temperature, humidity, rainfall pH, and label. We have dedicated 80% of the dataset as the training data and the remaining 20% as the testing (and validation) data, which makes the split ratio 0.8:0.2 of the train to test set.

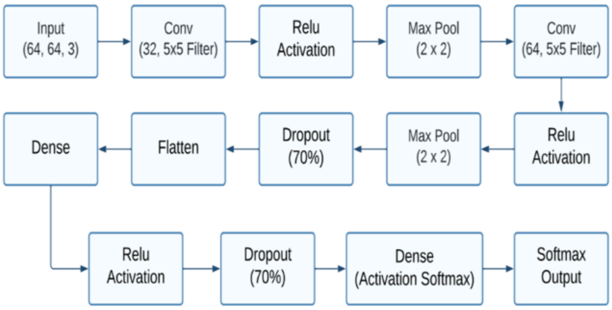
B. Architecture

In Similar crop Prediction, we use K Means Clustering Algorithm from Sklearn.cluster. We determine the number of clusters using the Elbow method and get it as 4. Then we perform K-means clustering and put the threshold value as 50 for each label and build clusters of similar crops.



In the plant prediction model, we initially build a predictive model using logistic regression importing it from sklearn.linear_model and fit the training data set into the model. We predict the testing dataset using predict function. We then evaluate its performance using a confusion matrix and check the classification report from sklearn.metrics. Classification report includes precision,f1-score, and recall of crop.Then finally we use the model to predict the crop specifying the conditions.

In the Plant Disease Detection, the model is trained with the labeled dataset. We used Keras in conjunction with TensorFlow to train our model. The first part of the training includes storing all image labels in a Dataframe and the corresponding images are also reshaped for the model base (64, 64, 3) with PIL.Image.ANTIALIAS filter. Model architecture is composed of 2 convolutional layers with a combination of max-pooling layers, 2 fully connected layers, and 2 dropout layers. All layers use the Relu activation function. The Softmax activation function has been used in the output layer. The total number of parameters in this architecture is around 10,886,712.



C. Loss Function

The term "loss" refers to a prediction error made by a Neural Net. Loss Function is the method for calculating the loss. The gradients are calculated using the Loss, to put it another way. In the Plant Disease Detection, we are using a categorical cross-entropy loss function. The categorical cross-entropy loss function computes an example's loss by adding the following values:

$$\text{Loss} = -\sum_{i=1}^{\substack{\text{output}\\\text{size}}} y_i \cdot \log \hat{y}_i$$

Where ŷi is the i-th scalar value in the model output, yi is the corresponding target value, and the output size is the number of scalar values in the model output. This loss is a good indicator of how easily two discrete probability distributions can be distinguished from one another. In this context, yi represents the likelihood that event I will occur, and the sum of all yi equals 1, implying that only one event will occur. When the distributions get smaller, the minus sign ensures that the loss gets smaller.

D. Methods

1. K-Means Clustering

The abbreviation K stands for Clustering is a vector quantization approach that divides n observations into k clusters, with each observation belonging to the cluster with the nearest mean (cluster centroid), which serves as the cluster's prototype. The letter 'K' in K-means stands for the number of clusters found from data by the approach. Data points are grouped in such a way that the total distance between the data points and the centroid is as small as possible.

To solve the problem, K-means applies the Expectation-Maximization methodology. The Expectation stage assigns data points to the closest cluster, whereas the Maximization stage computes the centroid of each cluster. The main function

$$J = \sum_{i=1}^{m} \sum_{k=1}^{K} w_{ik} \|x^i - \mu_k\|^2 \qquad (1)$$

is:

where the value of wik=1 for data point xi if it belongs to cluster k; otherwise its value is, wik=0. Also, µk is the centroid of xi's cluster.

It's a two-part minimization problem. We start by minimizing J concerning wik and treating µk as constant. Then we reduce J concerning µk and consider wik as constant. Technically, we differentiate J based on wik first and then update cluster assignments (E-step). Then, after the cluster assignments from the previous phase, we differentiate J w.r.t. µk and recomputed the centroids (M-step). Therefore, E-step is:

$$\frac{\partial J}{\partial w_{ik}} = \sum_{i=1}^{m} \sum_{k=1}^{K} \|x^i - \mu_k\|^2$$

$$\Rightarrow w_{ik} = \begin{cases} 1 & \text{if } k = argmin_j \|x^i - \mu_j\|^2 \\ 0 & \text{otherwise.} \end{cases} \qquad (2)$$

To put it another way, assign data point xi to the cluster with the shortest sum of squared distance from the cluster's centroid.

And M-step is:

$$\frac{\partial J}{\partial \mu_k} = 2\sum_{i=1}^{m} w_{ik}(x^i - \mu_k) = 0$$

$$\Rightarrow \mu_k = \frac{\sum_{i=1}^{m} w_{ik}x^i}{\sum_{i=1}^{m} w_{ik}} \qquad (3)$$

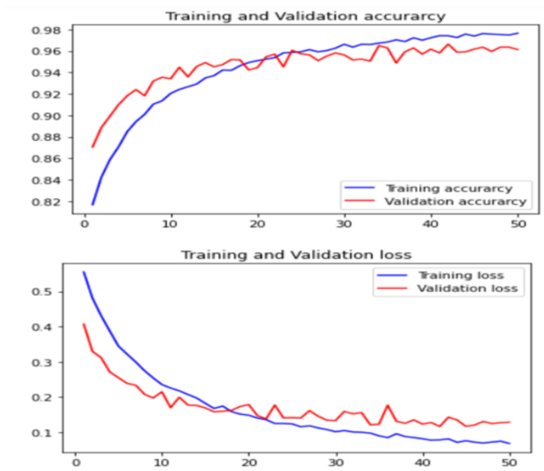
This means that the centroid of each cluster must be recalculated to reflect the new assignments.

2. Convolutional Neural Network (CNN):

ConvNet, also known as CNN, is a type of deep learning neural network. They're used to take an input image, assign importance (learnable weights and biases) to various aspects/objects in the image, and differentiate between them. In comparison to other classification algorithms, the amount of pre-processing required by a ConvNet is significantly less. While filters are hand-engineered in primitive methods, ConvNets can learn these filters/characteristics with enough training. Surprisingly, most convolutional neural networks are only equivariant to translation, not invariant. They are used in image and video recognition, recommender systems, image classification, image segmentation, medical image analysis, natural language processing, brain-computer interfaces, and financial time series and interfaces, among other applications.

III. RESULTS:

- The accuracy of the Crop Prediction Model is 97%.
- The CNN model is 96 percent accurate. It's also visible in the accuracy loss graph below.
- The loss value is below 0.2.
- It can be observed that with every iteration of the model training, the loss is getting gradually decreased and the accuracy is getting increased.



IV. Conclusion:

The connection among diverse links in the agri-food supply chain is transforming, with long-term deals that force new planning techniques to fulfil the request. In this paper, a new crop planning crisis under fixed and known demand was specified, and sported using a biobjective optimization strategy that first underrates cost and then, allowing for some cost growths, maximizes the geographic diversification of the crops to ease local threats
Therefore, it seems that, very large models, it is necessary to develop a more efficient metaheuristic solution method, given the inherent impracticalities in trying to solve the optimality of such instances

The model results demonstrate the ability of K-Means clustering and CNN as applied to multi objective constrained problem of crop planning. We have shown that K-Means clustering , CNN (accuracy of the model is 96%.) & logistic regression (accuracy of the model is 97%.) can be successfully employed to search the feasible solution space for a complex cropping pattern that involves multiple objectives and constraints

ACKNOWLEDGMENT:

Prof. S. R. Wankhede, Dept. of Information Technology (I.T.), Government College of Engineering, Amravati, has been a source of support and guidance to the authors throughout this study.
We are also obliged to our Head of Department (Dept.of Information Technology), Prof. A. W. Bhade for giving us this wonderful opportunity. We are thankful for their collaboration during the course of our assignment.

REFERENCE:

1. Cid-Garcia, N.M.; Bravo-Lozano, A.G.; Rios-Solis, Y. A crop planning and real-time irrigation method based on site-specific
management zones and linear programming. Comput. Electron. Agric. 2014,
https://www.sciencedirect.com/science/article/abs/pii/S0168169914001513?via%3Dihub

2. Amini, A. Application of Fuzzy Multi-Objective Programming in Optimization of Crop Production Planning. Asian J. Agric. Res.
2015
https://scialert.net/abstract/?doi=ajar.2015.208.222

3. Optimization Method for Crop Growth Characteristics Based on Improved Locality Preserving Projection
Jia Dongyao

https://www.hindawi.com/journals/jam/2014/809597/

3. Optimization of soil medium for better crop growth.
Dhananchezhiyan Perumal
https://www.researchgate.net/figure/Optimization-of-soil-medium-for-better-crop-growth_tbl1_347427424

4. Predicting optimum crop designs using crop models and seasonal climate forecasts
D. Rodriguez, P. de Voil, D. Hudson
https://www.nature.com/articles/s41598-018-20628-2

5. Optimum Crop Planning using Multi-Objective Differential Evolution Algorithm
https://scialert.net/fulltext/?doi=jas.2009.3780.3791

6. "Scikit-Lear.org" | Available at:
https://scikitlearn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html

7. "Numpy" | Available at: https://numpy.org/

8. "M\atplotlib – Visualization using Python" | Available at: https://matplotlib.org/

9. "Seaborn – Statistical data visualization" | Available at: https://seaborn.pydata.org/

10. "Pandas – Python Data Analysis Library" | Available at: https://pandas.pydata.org/