BATCH NUMBER-7B

ANALYSIS OF AGRICULTURAL DATA USING DATA MINING TECHNIQUES

A Project report submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE ENGINEERING

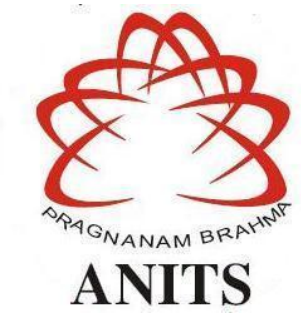
Submitted by

G. SARMISTANJALI	317126510070
M. SUDEEPTHI SWATHI	317126510092
K. PHANINDRA KUMAR	318126510L18
S. GOWTHAM KUMAR	317126510115

Under the guidance of

S.SNL PRIYANKA
(ASSISTANT PROFESSOR)

COMPUTER SCIENCE & ENGINEERING



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES
(UGC AUTONOMOUS)

(*Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade*)
Sangivalasa, Bheemili Mandal, Visakhapatnam dist. (A.P)

2020-2021

ACKNOWLEDGEMENT

We would like to express our deep gratitude to our project guide **S.S N L PRIYANKA,** Assistant Professor, Department of Computer Science and Engineering, ANITS, for his/her guidance with unsurpassed knowledge and immense encouragement. We are grateful to **Dr. R. Sivaranjani,** Head of the Department, Computer Science and Engineering, for providing us with the required facilities for the completion of the project work.

We are very much thankful to the **Principal and Management, ANITS, Sangivalasa,** for their encouragement and cooperation to carry out this work.

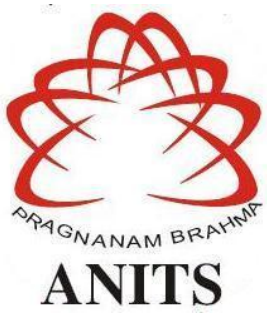
We express our thanks to Project Coordinator **Dr. K. S. Deepthi**, for his/her Continuous support and encouragement. We thank all **teaching faculty** of the Department of CSE, whose suggestions during reviews helped us in accomplishment of our project.

We would like to thank our parents, friends, and classmates for their encouragement throughout our project period. At last but not the least, we thank everyone for supporting us directly or indirectly in completing this project successfully.

PROJECT STUDENTS

G. SARMISTANJALI	317126510070
M. SUDEEPTHI SWATHI	317126510092
K. PHANINDRA KUMAR	318126510L18
S. GOWTHAM KUMAR	317126510115

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES
(UGC AUTONOMOUS)
(*Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade*)
Sangivalasa, Bheemili Mandal, Visakhapatnam dist.(A.P)



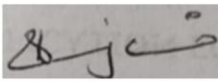
CERTIFICATE

This is to certify that the project report entitled "**ANALYSIS OF AGRICULTURE DATA USING DATA MINING TECHNIQUES"** submitted by **G. Sarmistanjali (317126510070), M. Sudeepthi Swathi (317126510092), K. Phanindra Kumar (318126510L18), S. Gowtham Kumar (317126510115)** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science Engineering** of Anil Neerukonda Institute of technology and sciences (A), Visakhapatnam is a record of bonafide work carried out under my guidance and supervision.

Project Guide

S. S N L Priyanka
Assistant Professor
Department of CSE
ANITS

Head of the Department



Dr.R. Sivaranjani
Professor
Department of CSE
ANITS

DECLARATION

We, **G. Sarmistanjali (317126510070), M. Sudeepthi Swathi (317126510092), K. Phanindra Kumar (318126510L18), S. Gowtham Kumar (317126510115)** , of final semester B.Tech., in the department of Computer Science and Engineering from ANITS, Visakhapatnam, hereby declare that the project work entitled **ANALYSIS OF AGRICULTURE DATA USING DATA MINING TECHNIQUES** is carried out by us and submitted in partial fulfillment of the requirements for the award of **Bachelor of Technology in Computer Science Engineering** , under Anil Neerukonda Institute of Technology & Sciences(A) during the academic year 2017-2021 and has not been submitted to any other university for the award of any kind of degree.

G. SARMISTANJALI	317126510070
M. SUDEEPTHI SWATHI	317126510092
K. PHANINDRA KUMAR	318126510L18
S. GOWTHAM KUMAR	317126510115

ABSTRACT

Agriculture is undoubtedly the largest livelihood provider in India and contributes a significant figure to the economy of our Country. The technological factors affecting the crop production includes practices used and also managerial decisions. So, predicting the crop yield prior to its harvest would help farmers to take appropriate steps. We attempt to resolve the issue by building an user-friendly prediction system. The results of the prediction are suggested to the farmer such that suitable changes can be made to improve the produce. There are different techniques or algorithms which help to predict crop yield. By analyzing all the parameters like location, soil nutrients, pH value, rainfall, moisture a potential solution can be obtained to overcome the situation faced by farmers. This paper focuses on the analysis of the agriculture data and finding optimal yield to provide an insight before the actual crop production using data mining techniques and Machine Learning algorithms.

Keywords: Yield, Random forest regressor, Decision Tree regressor, GDP, Digitalization.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

1.1 Introduction

Today, India is one of the main makers across the world in the farming area. Horticulture is the broadest monetary area and assumes a remarkable part in the financial piece of India. Horticulture is an unconventional business crop creation which is impacted by numerous environment and monetary factors. Andhra Pradesh, fundamentally being an agro-Based economy offers over 29% of the Gross domestic product as against 17% in the nation's Gross domestic product. Periodical guidance to the ranchers either as far as improved farming procedures or headways in factors influencing the creation of harvests may fortify the state in the horticulture sector. Yield forecast is one among the rural progressions. Because of these sorts of developments farming is driving the interest of present-day man. In the past ranchers used to anticipate their yield from past encounters. Digitalization in cultivating gives mindfulness about the development of the yields at the perfect time and at the perfect spot even to youthful ranchers. These sorts of headways need the utilization of information analytics. This is one such framework that can be utilized to address yield forecasts.

1.1.1 Introduction to Data Mining

Data mining is the computing process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics, and database systems. Data mining involves five-steps. They are:

1. Identifying the source information.
2. Picking the data points that need to be analyzed.
3. Extracting the relevant information from the data.
4. Identifying the key values from the extracted data set.
5. Interpreting and reporting the results.

Data mining is a process used by companies to turn raw data into useful information. By using software to look for patterns in large batches of data, businesses can learn more about their customers to develop more effective marketing strategies, increase sales and decrease costs.

HOW DATA MINING IS USED IN AGRICULTURE SECTOR

Data mining techniques are used in performing several activities in the agricultural sector such as pest identification, detection and classification and prediction of crop diseases. It can also be used in yield prediction, input management (planning of irrigation and pesticides), fertilizer suggestion and predicting soil. In a world full of data, data mining is the computational process for discovering new patterns. Data mining techniques provide a major advantage in agriculture for detection and prediction for optimizing the pesticides. Techniques for agriculture related activities provide a lot of information. The yield of agriculture primarily depends on diseases, pests, weather conditions, planning of various crops for the harvest productivity is the results.

Crop production for reliable and timely requirements for various decisions for agriculture marketing. Predictions are very useful for agriculture data. For instance, by applying data mining techniques, the government can fully benefit from data about farmers' buying patterns and to achieve a superior understanding of their land to achieve more profit on the farmer's part.

Data mining techniques followed in two ways:

1. Descriptive data mining.
2. Predictive data mining.

Descriptive data mining tasks characterize the final properties of the info within the database while predictive data mining is employed to predict the direct values supported patterns determined from known results. Prediction involves using some variables or fields within the database to predict unknown or future values of other variables of interest. As far as data mining techniques are concerned, in most cases predictive data mining approaches are employed. Predictive data mining techniques are employed to predict future crop, forecasting, pesticides, and fertilizers to be used, revenue to be generated and so on.

These techniques are used for pre-harvest forecasting for the agriculture field and can provide a lot of data on agricultural-related activities. Data of agriculture in data mining can be presented in the form of datasets.

1.2 Motivation of the work

Agriculture is undoubtedly the largest livelihood provider in India and contributes a significant figure to the economy of our Country. But it was neglected over a period, farmers' effort was not appreciated. The world has recognized farming in several world conferences and countries are

focusing on the development of their respective agriculture sector. As a part of digital India campaign farmers are encouraged to adopt digital practices in their farming strategies. The technological factors affecting the crop production includes practices used and managerial decisions. Crop production for reliable and timely requirements for various decisions for agriculture marketing. Predictions are very useful for agriculture data.

By applying data mining techniques, the government can fully benefit from data about farmers' buying patterns and to achieve a superior understanding of their land to achieve more profit on the farmer's part. So, predicting the crop yield prior to its harvest would help farmers to take appropriate steps .We attempt to resolve the issue by building an user-friendly prediction system. The results of the prediction are suggested to the farmer such that suitable changes can be made to improve the produce.

1.3 Problem Statement

This project will analyze the agriculture data and find optimal parameters to maximize the crop production using data mining techniques like PAM, CLARA, DBSCAN and Multiple Linear Regression. The dataset consists of features like year, District, crop, season, area, production (in tons), nitrogen(kg/Ha), phosphorus (Kg/Ha), Potassium (Kg/Ha) etc. The major goal of the proposed system is understanding data mining techniques and applying it to the dataset.

1.4 Organization of Thesis

The chapters of this document describe the following:

Chapter-1:

This is about the introduction of our project where we have given clear insights about our project domain and other related concepts.

Chapter-2:

This Chapter specifies a literature survey where all different existing methods and models are examined.

Chapter-3:

This Chapter specifies the proposed system with a system architecture along with detailed explanations of each module.

Chapter-4:

This Chapter specifies the requirements which include both software and hardware requirements.

Chapter-5:

This contains the sample code of the project.

Chapter-6:

This Chapter specifies the experimental analysis of our system along with performance measures and comparisons between different models. It also specifies the user interface also.

Chapter-7:

This Chapter gives the conclusion to our work with an insight for the future scope.

CHAPTER 2 -LITERATURE SURVEY

Clustering or **cluster analysis** is an unsupervised learning problem. It is often used as a data analysis technique for discovering interesting patterns in data, such as groups of customers based on their behavior. Many clustering algorithms have been developed for different purposes. Clustering techniques can be categorized into Partitioning clustering (iteratively reallocating objects to improve the quality of clustering results), Hierarchical clustering algorithms (assign objects in tree structured clusters) , Density-based clustering algorithm is that, for each point of a cluster, the neighborhood of a given unit distance has to contain at least a minimum number of points. Other types are Grid-based methods and Model based clustering methods.

There are different forecasting methodologies developed and evaluated by the researchers all over the world in the field of agriculture. Some of such studies are: Two persons namely Lakkana Ruekkasaem, Montalee Sasanian from Department of Industrial Engineering, Faculty of Engineering, Thammasat University, Pathumthani 12120, Thailand had worked on data obtained from January 2013 – December 2017, for a total of 60 months. The data were analyzed with times series analysis by using the following seven methods: the Least Square Method, the Moving Average Method (3 months, 5 months, and 7 months), the Single Exponential Method, the Double Exponential Method, and Winter's Method.

Thompson (1986) used a statistical type model to determine the impact of climate change and weather variability on corn production in five Midwestern states in the USA. He found pre-season precipitation (September –June), June temperature, and temperature and rainfall in July and August to be closely correlated with corn yield variations from the trend. This approach significantly improved predictions of historical yields of corn and soybean. Lobell et al. (2011, 2013) used statistical models to determine the effects of increases in temperature on maize yield in the USA concluding that temperature increase will play a large role in yield decrease under climate change.

Yield forecast using agrometeorological inputs into a statistical regression is rather common and used in many yield forecasts research and programs (NASS, 2006; Lobell et al., 2009). In general, a simple statistical model is built using a matrix with historic yield and several agrometeorological parameters (e.g. temperature and rainfall). Then, a regression equation is derived between yields as a function of one or several agrometeorological parameters. The NASS (2006) program uses a statistical model to forecast crop yield and production.

CHAPTER 3 PROPOSED METHODOLOGY

3.1 Proposed System

Though there are many yield prediction models, they are neither fully functional nor implemented fully in the real world. So, we have thought to make our proposed system fully functional and also develop in a simple manner.

3.1.1 System Architecture

The below diagram depicts the system architecture of our project. Our whole system can be divided into 2 modules i.e., one model predicts the optimal yield and the other model analyses the patterns in the dataset. The operation of these models is specified clearly in the above diagram.

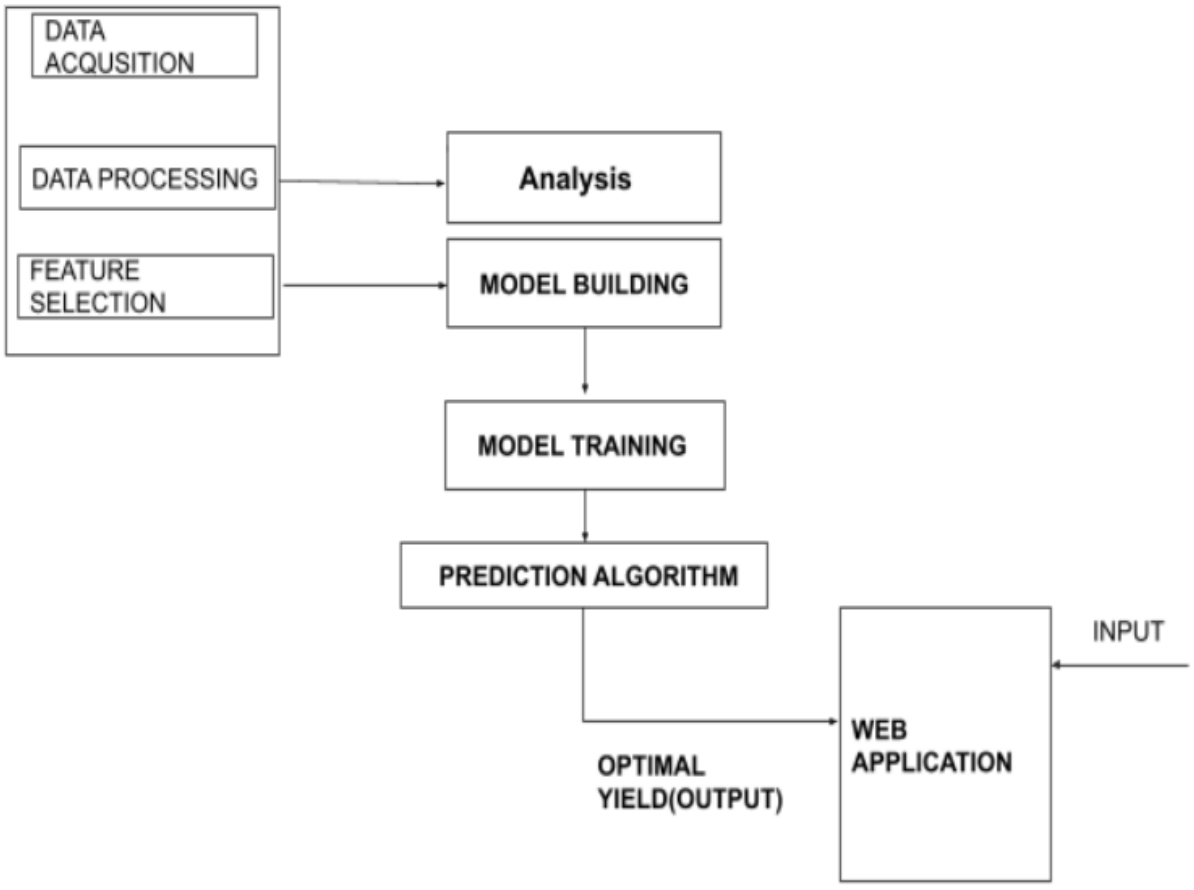


Fig-3.1 The blueprint of proposed system

3.2 Module Division

The major modules of the proposed system are:

3.2.1 Data Acquisition

3.2.2 Data Pre-processing

3.2.3 Feature Selection

3.2.4 Training Methods

3.2.5 Testing Data

3.2.1 Data Acquisition

The goal of this step is to identify and obtain all data-related problems. In this step, we need to identify the different data sources, as data can be collected from various sources such as files and databases. The quantity and quality of the collected data will determine the efficiency of the output. The more data, the more accurate the prediction will be.

```
t = pd.read_excel("soilh.xls")
t.head()
```

	Sl no	Date	Farmer No	Macro/ Micro nutrient	Farmer Name	District	Mandal	Village	Latitude	Longitude	Survey No.
0	1	2015-01-01	1910	RK2276	P.Krishna Naik	Anantapur	Penukonda	Gonipeta	14.0819	77.6922	114
1	2	2015-01-01	1911	RK2277	Kallu Thippe Naik	Anantapur	Penukonda	Gonipeta	14.0928	77.6858	184-4
2	3	2015-01-01	1912	RK2278	P.Duble Bai	Anantapur	Penukonda	Gonipeta	14.0933	77.6939	185
3	4	2015-01-01	1913	RK2279	H.Marekka (Kamma)	Anantapur	Penukonda	Gonipeta	14.0953	77.6961	163-3A
4	5	2015-01-01	1914	RK2280	M.Alevelamma	Anantapur	Penukonda	Gonipeta	14.0856	77.6892	241-1

Fig-3.2 Sample of raw data

Data Preparation:

Data preparation is a step where we put our data into a suitable place and prepare it to use in our machine learning training. It is used to understand the nature of data that we must work with. We need to understand the characteristics, format, and quality of data.

Data wrangling is the process of cleaning and converting raw data into a usable format. It is the process of cleaning the data, selecting the variable to use, and transforming the data in a proper format to make it more suitable for analysis in the next step. It is one of the most important steps of the complete process. Cleaning of data is required to address the quality issues.

3.2.2 Data Pre-processing

The goal of this step is to study and understand the nature of data that was acquired in the previous step and to know the quality of data. In this step, we will check for any null values and remove them as they may affect the efficiency. Identifying duplicates in the dataset and removing them is also done in this step. In the pre-processing, we used a method called data wrangling that is used to select the features to use, converting the acquired data in the dataset to a format that would be suitable for next steps and cleaning of data points.

```
[ ]  t = pd.read_csv("crop_production.csv")
     t.head()
```

	State_Name	District_Name	Crop_Year	Season	Crop	Area	Production
0	Andaman and Nicobar Islands	NICOBARS	2000.0	Kharif	Arecanut	1254.0	2000.0
1	Andaman and Nicobar Islands	NICOBARS	2000.0	Kharif	Other Kharif pulses	2.0	1.0
2	Andaman and Nicobar Islands	NICOBARS	2000.0	Kharif	Rice	102.0	321.0
3	Andaman and Nicobar Islands	NICOBARS	2000.0	Whole Year	Banana	176.0	641.0
4	Andaman and Nicobar Islands	NICOBARS	2000.0	Whole Year	Cashewnut	720.0	165.0

Fig-3.3 Sample data after Data preprocessing

3.2.2.1 Data Encoding:

Categorical data is data which has some categories such as, in our dataset; there are three categorical variables Season, Crop and District. Since the machine learning model completely works on mathematics and numbers, but if our dataset would have a categorical variable, then it

may create trouble while building the model. While encoding if we encode these values, we will have many unique patterns which may be difficult in proceeding further. Hence, we found out unique types of values in each categorical variable first and then encoded each of the variables using Label Encoder.

The below diagram sample of data after using Label Encoding.

Encoding the data using Label Encoder:

```
[ ]  from sklearn.preprocessing import LabelEncoder
     lb_make = LabelEncoder()
     for col in X.columns[0:4]:
          X[col] = lb_make.fit_transform(X[col])
     X.head()
```

	Crop_Year	Season	Crop	District	pH	Avail-P	Exch-K	N	Rainfall	Area
0	0	0	1	0	6.19	7.13	41	8.89	928.5	21400
1	0	0	2	0	8.40	10.34	102	3.24	928.5	1400
2	0	0	10	0	7.10	8.46	46	5.54	928.5	1000
3	0	0	14	0	8.30	2.31	35	1.79	928.5	7300
4	0	0	17	0	6.40	6.08	76	22.26	928.5	3700

Fig-3.4 sample data after encoding

Splitting the Dataset into the Training set and Test set:

In machine learning data preprocessing, we divide our dataset into a training set and test set. This is one of the crucial steps of data pre-processing as by doing this, we can enhance the performance of our machine learning model. Suppose if we have given training to our machine learning model by a dataset and we test it by a completely different dataset. Then, it will create difficulties for our model to understand the correlations between the models.

3.2.3 Feature Selection

After pre-processing the acquired data, the next step is to identify the best features. The identified best features should be able to give high efficiency. Most of the features we have in our dataset are not numerical. So, the data needs to be encoded initially. For encoding the data, the author has used binary encoding for each feature. After encoding, correlation between the

various features should be tested. The author has validated the correlations between the features using the Pearson correlation test. All the correlated features can be identified as the best features.

In this step, we processed massive data of 112 features and selected the final parameters that are influential in nature. The list of final parameters is tabulated as below:

S no	Feature	Description
1	Year	The year in which the crop will be cultivated. Generally, the upcoming year
2	Season	One among Kharif, Rabi and Whole Year.
3	Crop	Name of the crop
4	District	Name of the district
5	pH Level	This describes the nature of the soil
6	Nitrogen	Amount of nitrogen present
7	Potassium	Amount of potassium present
8	Phosphorus	Amount of phosphorus present
9	Rainfall	Expected rainfall in millimeters
10	Area	Area of field in hectares

Table-3.1 Description of Input Data

3.3 Models

Training Data:

Here, in this step we select the machine learning techniques such as Classification, Regression, Cluster analysis, etc. then build the model using prepared data, and evaluate the model. We use datasets to train the model using various machine learning algorithms. Training a model is required so that it can understand the various patterns, rules, and features.

Testing Data:

Once our machine learning model has been trained on a given dataset, then we test the model. In this step, we check for the accuracy of our model by providing a test dataset to it. Testing the model determines the percentage accuracy of the model as per the requirement of project or problem.

3.3.1 Model Building:

In this phase the data science team needs to develop data sets for training, testing, and production purposes. These data sets enable data scientists to develop analytical methods and train it, while holding aside some data for testing the model. Team develops datasets for testing, training, and production purposes. In addition, in this phase, the team builds and executes models based on work done in the model planning phase.

3.3.1.1 Clustering:

In our project we have done analysis using clustering using K-means, DBSCAN, PAM for identifying the spread of the data and finding the patterns in the dataset. We have also used KNN and Elbow method for finding the optimal K value to be used in other clustering techniques.

3.3.1.1.1 DBSCAN (Density Based Spatial Clustering of Applications with Noise):

It is a well-known data clustering algorithm based on density that is commonly used in data mining and machine learning. DBSCAN has two parameters namely Eps and MinPts. However, traditional DBSCAN cannot produce optimal Eps value. Determination of the optimal Eps value automatically is the one of the most necessary modifications for the DBSCAN.

Optimal Parameters: For two-dimensional data: use default value of minPts=4 (Ester et al., 1996) For more than 2 dimensions: minPts=2*dim (Sander et al., 1998) Once you know which MinPts to choose, you can determine

Epsilon: Plot the k-distances with k=minPts (Ester et al., 1996) Find the 'elbow' in the graph--> The k-distance value is your Epsilon value.

Elbow Method:

In cluster analysis, the Elbow method is a heuristic used in determining the number of clusters in a data set. The method consists of plotting the explained variation as a function of the number of clusters, and picking the elbow of the curve as the number of clusters to use. The same method can be used to choose the number of parameters in other data-driven models, such as the number of principal components to describe a data set.

The below figure 3.4 shows the elbow method implementation for our training data.

```
from scipy.spatial.distance import cdist, pdist
# Choosing the optimal k
k_range = range(1,10)
# Try clustering the data for k values ranging 1 to 10
k_means_var = [KMeans(n_clusters = k).fit(train) for k in k_range]
centroids = [X.cluster_centers_ for X in k_means_var]

k_euclid = [cdist(train, cent, 'euclidean') for cent in centroids]
dist = [np.min(ke, axis=1) for ke in k_euclid]

# Calculate within-cluster sum of squares
wcss = [sum(d**2) for d in dist]

# Visualize the elbow method for determining k
import matplotlib.pyplot as plt
plt.plot(k_range, wcss)
plt.show()
```

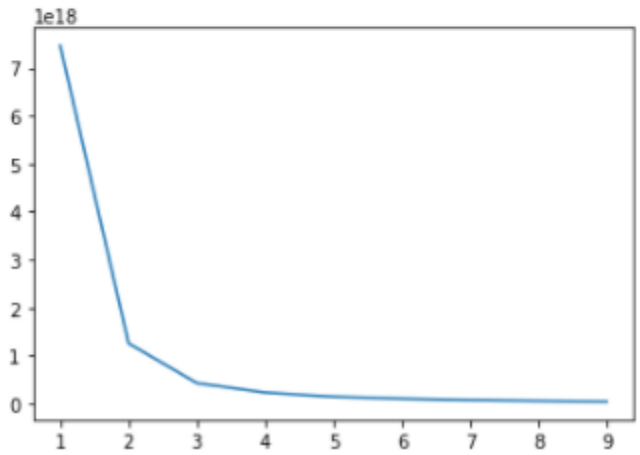


Fig -3.5: The result of Elbow method

DBSCAN groups the points together based on distance measurement (usually Euclidean distance) and a minimum number of points. It also marks outliers as low-density regions. This is used to find associations and structures in the data that are hard to find manually but that can be relevant and useful to find patterns and predict trends. DBSCAN will be provided by sklearn's cluster module, this belongs to AgglomerativeClustering class and the parameters to be mentioned are Eps value and min_samples i.e., minpts. These are obtained from elbow method and KNN graph respectively. In our we fitted the data and given the eps value of 0.4 and min_samples of 200.Then printed the labels of the clusters using dbscan_model.labels_.

The implementation of DBSCAN is shown below figure 3.5.

```
from sklearn.cluster import AgglomerativeClustering
from sklearn.decomposition import PCA
from sklearn.cluster import DBSCAN
dbscan_model = DBSCAN(eps =0.4, min_samples = 200).fit(X_scaled)
labels = dbscan_model.labels_
print(np.unique(labels, return_counts=True))
#minpts = n_neighbors = 10 * 2
#eps = 0.4
```

```
(array([-1,  0,  1,  2]), array([2271, 2689, 3305, 1158]))
```

Fig-3.6 Clusters formed based on using DBSCAN

3.3.1.1.2 PAM (Partition around medoids):

It is a partitioning based algorithm. It breaks the input data into several groups. It finds a set of objects called medoids that are centrally located. The Steps involved are:

1. Initialize: select k random points out of n data points as the medoids.
2. Associate each data point to the closest medoid by using any distance metric methods.
3. While the cost decreases: For each medoid m, for data point p which is not a medoid:
    a. Swap m and p, associate each data point to the closest medoid, recompute the cost.
    b. If the total cost is more than that in the previous step, undo the swap.

3.3.1.1.3 PCA (Principal Component Analysis):

Principal Component Analysis, or PCA, is a dimensionality-reduction method that is often used to reduce the dimensionality of large data sets, by transforming a large set of variables into a smaller one that still contains most of the information in the large set.

Reducing the number of variables of a data set naturally comes at the expense of accuracy, but the trick in dimensionality reduction is to trade a little accuracy for simplicity. Because smaller data sets are easier to explore and visualize and make analyzing data much easier and faster for machine learning algorithms without extraneous variables to process.

Although PCA in its standard form is a widely used and adaptive descriptive data analysis tool, it also has many adaptations of its own that make it useful to a wide variety of situations and data types in numerous disciplines. Adaptations of PCA have been proposed, among others, for binary data, ordinal data, compositional data, discrete data, symbolic data, or data with special structure, such as time series or datasets with common covariance matrices.

PCA or PCA-related approaches have also played an important direct role in other statistical methods, such as linear regression (with principal component regression). Then PCA is used to view the obtained clusters in two-dimensional view. The implementation of PCA can be seen in figure 3.6.

```
[ ]  pca = PCA(n_components = 2)
     data_principal = pca.fit_transform(X_scaled)
     data_principal = pd.DataFrame(data_principal)
     data_principal.columns = ['P1', 'P2']
     print(data_principal.head())
```

```
          P1        P2
0 -0.360360  0.319105
1 -0.363627  0.278764
2 -0.351537  0.291070
3 -0.349506  0.264803
4 -0.341562  0.298084
```

Fig-3.7 Applying PCA and reducing dimensionality to two features.

```
[ ]  fg = sns.FacetGrid(data=data_principal, hue='clusters', aspect=1.61)
     fg.map(plt.scatter, 'P1', 'P2').add_legend()
```

```
<seaborn.axisgrid.FacetGrid at 0x7f4ac52c9d50>
```

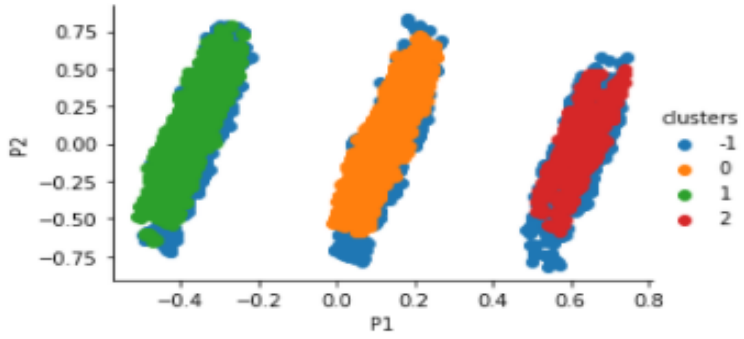


Fig-3.8 Graphical representation of clusters formed.

3.3.1.2 Regression Analysis:

Regression models contribute to the second module of our project i.e., prediction of yield that can be produced for the given list of parameters. The types of regression techniques include Linear Regression, Ridge Regression, Lasso Regression, Decision Tree Regression and Random Forest Regression.

3.3.1.2.1 Random Forest Regression

A Random Forest is an ensemble technique capable of performing both regression and classification tasks with the utilization of multiple decision trees and a way called Bootstrap and Aggregation.

The basic idea behind this is often to mix multiple decision trees in determining the ultimate output instead of counting on individual decision trees. Generally, it has multiple decision trees as base learning models. Randomly perform row sampling and have sampling from the dataset forming sample datasets for each model. The basic steps involved in Random Forest algorithm is as follows:

Step 1: Start selecting the random samples from the given training dataset.

Step 2: Next, this algorithm will construct a decision tree for each sample using the decision tree algorithm. Then for each decision tree an outcome is obtained.

Step 3: Next voting will be performed for every result that is predicted.

Step 4: Now select the most voted result as the final prediction result.

In scikit-learn python library, sklearn.ensemble.RandomForestRegressor module is used for carrying out Random Forest regression. We must specify the maximum decision trees as a parameter for this function. We will use our training dataset to fit the model. Fig 3.8 shows the sample code for the training model using Random Forest regressor.

```
[ ]    from sklearn.ensemble import RandomForestRegressor
       rf_obj =RandomForestRegressor(n_estimators = 10,random_state=0)
       rf_obj.fit(X_train,y_train)
       print('Train score RF:',rf_obj.score(X_train,y_train))
       print('Test score RF:',rf_obj.score(X_test,y_test))

Train score RF: 0.9859482658364138
Test score RF: 0.8994486718857367
```

Fig 3.9 Implementation of Random Forest Regression.

3.3.1.2.2 Decision Tree Regression

Decision Tree is one of the most used, practical approaches for supervised learning. It can be used to solve both Regression and Classification tasks with the latter being put more into practical application. It is a tree-structured classifier with three types of nodes. The Root Node is the initial node which represents the entire sample and may get split further into further nodes. The Interior Nodes represent the features of a data set and the branches represent the decision rules. Finally, the Leaf Nodes represent the outcome. This algorithm is very useful for solving decision-related problems.

Decision tree regression observes features of an object and trains a model in the structure of a tree to predict data in the future to produce meaningful continuous output. Continuous output means that the output/result is not discrete, i.e., it is not represented just by a discrete, known set of numbers or values.

In scikit-learn python library, sklearn.tree.DecisionTreeRegressor module is used for carrying out Decision Tree regression. We will use our training dataset to fit the model. Fig 3.9shows the sample code for the training model using Decision Tree regressor.

```
from sklearn.tree import DecisionTreeRegressor
dt_obj =DecisionTreeRegressor(random_state=1)
dt_obj.fit(X_train,y_train)
print('Train Score DT:',dt_obj.score(X_train,y_train))
print('Test Score DT:',dt_obj.score(X_test,y_test))
```

```
Train Score DT: 1.0
Test Score DT: 0.8162150580308982
```

Fig 3.10 Implementation of Decision Tree Regression

CHAPTER 4-REQUIREMENTS AND ANALYSIS

4.1 Software Requirements:

1. Software:
    - Python Version 3.0 or above
    - Django Framework
    - Jupyter Notebook
2. Operating System: Windows 10
3. Tools: Microsoft Visual Studio, Google Collab, Web Browser(Google Chrome/Firefox)
4. Python Libraries: NumPy, pandas, sklearn, matplotlib, seaborn, pickle

4.1.1 Introduction to Python

Python is a popular programming language. It was created by Guido van Rossum and released in 1991. It's often used as a "scripting language" for web applications. This means that it can automate a specific series of tasks, making it more efficient. Consequently, Python (and languages like it) is often used in software applications, pages within a web browser, the shells of operating systems and some games. It is used for:

- Web development (server-side)
- Software development
- Mathematics
- System scripting.

Advantages of using python are:

- Python works on different platforms (Windows, Mac, Linux etc.).
- Python has a simple syntax like the English language.
- Python has syntax that allows developers to write fewer lines of code.
- Python runs on an interpreter system, so prototyping can be very quick.
- Python can be treated in a procedural, an object-orientated way.

4.1.2 Introduction to Machine Learning

Machine learning is a branch of artificial intelligence (AI) focused on building applications that learn from data and improve their accuracy over time without being programmed to do so. Machine learning is learning based on experience. As an example, it is like a person who learns

to play chess through observation as others play. In this way, computers can be programmed through the provision of information in which they are trained, acquiring the ability to identify elements or their characteristics with high probability. Machine learning algorithms are divided into two groups:

- Unsupervised learning
- Supervised learning

With Unsupervised learning, your machine receives only a set of input data. Thereafter, the machine is up to determine the relationship between the entered data and any other hypothetical data. Unlike supervised learning, where the machine is provided with some verification data for learning, independent Unsupervised learning implies that the computer itself will find patterns and relationships between different data sets. Unsupervised learning can be further divided into clustering and association. Supervised learning implies the computer ability to recognize elements based on the provided samples. The computer studies it and develops the ability to recognize new data based on this data.

Some Supervised learning algorithms include:

- Decision trees
- Support-vector machine
- Naive Bayes classifier
- K-Nearest Neighbors
- linear regression

Some Unsupervised learning algorithms include:

- K-means clustering
- Hierarchical clustering
- Neural Networks
- Principal Component Analysis
- Independent Component Analysis
- Apriori algorithm

4.1.2.1 NumPy

NumPy is the fundamental package needed for scientific computing with Python. NumPy, which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. NumPy is an open-source numerical Python library. NumPy array is a powerful N-dimensional array object which is in the form of rows and columns. We can initialize NumPy arrays from nested Python lists and access it elements. This package contains:

- A powerful N-dimensional array object
- Sophisticated (broadcasting) functions
- Basic linear algebra functions
- Basic Fourier transforms
- Sophisticated random number capabilities
- Tools for integrating C/C++ cod

NumPy is a successor for two earlier scientific Python libraries: Numeric and Numarray.

4.1.2.2 Pandas

Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data. The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis". Pandas allows us to analyze big data and make conclusions based on statistical theories. Pandas can clean messy data sets, and make them readable and relevant. Relevant data is very important in data science.

Firstly, the DataFrame can contain data that is:

- A Pandas DataFrame.
- A Pandas Series: a one-dimensional labeled array capable of holding any data type with labels or index. An example of a Series object is one column from a DataFrame.
- A NumPy ndarray, which can be a record or structured.
- A two-dimensional ndarray.
- Dictionaries of one-dimensional ndarray, lists, dictionaries, or Series.

4.1.2.3 Sk-Learn

Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering, and dimensionality reduction via a consistent interface in Python. Scikit-learn is a library in Python that provides many unsupervised and supervised learning algorithms. It's built upon some of the technology you might already be familiar with, like NumPy, pandas, and Matplotlib. The functionality that scikit-learn provides include:

- Regression, including Linear and Logistic Regression
- Classification, including K-Nearest Neighbors
- Clustering, including K-Means and K-Means++
- Model selection
- Preprocessing, including Min-Max Normalization

Some popular groups of models provided by scikit-learn include:

- **Clustering:** for grouping unlabeled data such as K-Means.
- **Cross Validation:** for estimating the performance of supervised models on unseen data.
- **Datasets:** for test datasets and for training datasets with specific properties for investigating model behavior.
- **Dimensionality Reduction:** for reducing the number of attributes in data for summarization, visualization, and feature selection such as Principal component analysis
- **Ensemble methods:** for combining the predictions of multiple supervised models.
- **Feature extraction:** for defining attributes in image and text data.
- **Feature selection:** for identifying meaningful attributes from which to create supervised models
- **Parameter Tuning:** for getting the most out of supervised models.
- **Manifold Learning:** For summarizing and depicting complex multi-dimensional data.
- **Supervised Models:** a vast array not limited to generalized linear models, discriminant analysis, naive bayes, lazy methods, neural networks, support vector machines and decision trees.

4.1.2.4 Matplotlib

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in 2002. One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc. Matplotlib pyplot is a collection of functions that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc. In matplotlib. Matplotlib is a cross-platform, data visualization and graphical plotting library for Python and its numerical extension NumPy. As such, it offers a viable open source alternative to MATLAB. Developers can also use matplotlib's APIs (Application Programming Interfaces) to embed plots in GUI applications.

4.1.2.5 Seaborn

Seaborn is a library mostly used for statistical plotting in Python. It is built on top of Matplotlib and provides beautiful default styles and color palettes to make statistical plots more attractive.

Seaborn helps resolve the two major problems faced by Matplotlib; the problems are −

- Default Matplotlib parameters
- Working with data frames

As Seaborn compliments and extends Matplotlib, the learning curve is quite gradual. If you know Matplotlib, you are already halfway through Seaborn. Seaborn comes with some very important features. The features help in −

- Built in themes for styling matplotlib graphics
- Visualizing univariate and bivariate data
- Fitting in and visualizing linear regression models
- Plotting statistical time series data
- Seaborn works well with NumPy and Pandas data structures
- It comes with built in themes for styling Matplotlib graphics

4.1.2.6 Pickle

Python pickle module is used for serializing and de-serializing a Python object structure. Pickling is a way to convert a python object (list, dict, etc.) into a character stream. The idea is that this character stream contains all the information necessary to reconstruct the object in another python script. Pickling is useful for applications where you need some degree of persistence in your data. Your program's state data can be saved to disk, so you can continue working on it later.

4.1.3 Introduction to Flask Framework

Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it. Flask is based on the Werkzeug WSGI toolkit and Jinja2 template engine.

Web Server Gateway Interface (WSGI)

WSGI has been adopted as a standard for Python web application development. WSGI is a specification for a universal interface between the web server and the web applications.

Werkzeug

Werkzeug is a WSGI toolkit, which implements requests, response objects, and other utility functions. This enables building a web framework on top of it. The Flask framework uses Werkzeug as one of its bases.

Jinja2

Jinja2 is a popular templating engine for Python. A web templating system combines a template with a certain data source to render dynamic web pages.

Flask is often referred to as a micro framework. It aims to keep the core of an application simple yet extensible. Flask does not have a built-in abstraction layer for database handling, nor does it have form validation support. Instead, Flask supports the extensions to add such functionality to the application. Some of the popular Flask extensions are discussed later in the tutorial. Features of Flask:

- Development server and debugger.
- Integrated support for unit testing.
- RESTful request dispatching.
- Uses Jinja templating.

- Support for secure cookies (client-side sessions)
- 100% WSGI 1.0 compliant.
- Unicode-based.
- Extensive documentation.

4.2 Hardware requirements:

1. CPU - 8 to 16 with each octa core processor in a distributed network.
2. RAM - 128 to 256 GB
3. Storage – 30 to 50 GB

CHAPTER 5 SAMPLE CODE

5.1 Sample Code

5.1.1 Yield Prediction.ipynb:

```
import numpy as np
import pandas as pd
import os

from matplotlib import pyplot as plt

# %matplotlib inline
import matplotlib
import seaborn as sns

from sklearn.preprocessing import LabelEncoder

train = pd.read_excel("Dataset.xlsx")
train = train.dropna()

X = (train.drop(['Production.1'], 1))
#input variables

y = (train['Production.1'])
#target variable
t = train['District'].unique()

labelEncoder = LabelEncoder()

mapping_dict ={}
for col in X.columns[1:4]:
    X[col] = labelEncoder.fit_transform(X[col])
    le_name_mapping = dict(zip(labelEncoder.classes_,
                        labelEncoder.transform(labelEncoder.classes_)))
    mapping_dict[col]= le_name_mapping

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 1)

from sklearn.tree import DecisionTreeRegressor
  dt_obj =DecisionTreeRegressor(random_state=1)
  dt_obj.fit(X_train,y_train)

  print('Train Score DT:',dt_obj.score(X_train,y_train))
```

```
  print('Test Score DT:',dt_obj.score(X_test,y_test))

def return_keys(test):
    l=[]
    for key, value in Season.items():
        if key==test[0][1]:
            test[0][1] = value

    for key, value in Crop.items():
        if key==test[0][2]:
            test[0][2] = value

    for key, value in District.items():
        if key==test[0][3]:
            test[0][3] = value
    return test

# return_keys(test[0][1] ,test[0][2] ,test[0][3] )
return_keys(test)

y = dt_obj.predict(test)

print(y)
```

5.1.2 app.py:

```
from logging import DEBUG, debug
from flask import Flask, render_template , request

import pickle

from sklearn.preprocessing import LabelEncoder
from sklearn.tree import DecisionTreeRegressor

app = Flask(__name__)

@app.route('/')
def home():
    return render_template('home.html')

@app.route('/crop_prediction/')
def crop_prediction():
     return render_template('crop_prediction.html')

@app.route('/predict',methods=['POST'])
def predict():
```

```
    if request.method == 'POST':
        year     = request.form['year']
        season   = request.form['season']
        crop     = request.form['crop']
        district = request.form['district']
        ph       =  request.form['ph']
        phos     =  request.form['phosphorous']
        potash   = request.form['potassium']
        nitrogen = request.form['nitrogen']
        rainfall = request.form['rainfall']
        area     = request.form['area']


    test = [[year,season,crop,district,ph,phos,potash,nitrogen,rainfall,area]]

    Season = {'Kharif':0, 'Rabi':1, 'Whole Year':2}

    District = {'Anantapur': 0, 'Chittoor': 1, 'East Godavari': 2, 'Guntur': 3, 'Kadapa': 4, 'Krishna': 5,
'Kurnool': 6, 'Nellore': 7, 'Prakasam': 8, 'Srikakulam': 9, 'Visakhapatnam': 10, 'Vizianagaram': 11,
'West Godavari': 12}

    for key, value in Season.items():
        if key==test[0][1]:
            test[0][1] = value

 for key, value in Crop.items():
        if key==test[0][2]:
            test[0][2] = value

    for key, value in District.items():
        if key==test[0][3]:
            test[0][3] = value

    model = pickle.load(open('model.pkl','rb'))
    prediction = model.predict(test)

    return render_template('crop_prediction.html',prediction = prediction[0])

if __name__ == '__main__':
    app.run(debug=True)
```

5.1.3 Home.html:

```
<!DOCTYPE html>
<html lang="en">
<head>

<title>CropLabs</title>

<link rel="shortcut icon" href="{{ url_for('static', filename='images/favicon.ico') }}"/>

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"

integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
                crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
                integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
                crossorigin="anonymous"></script>

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
integrity="sha384-
DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
                crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"
        integrity="sha384-
Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo"
                crossorigin="anonymous"></script>

<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
        integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">

<link href="{{ url_for('static', filename='css/bootstrap.css') }}" rel='stylesheet' type='text/css' />

<link href="{{ url_for('static', filename='css/style.css') }}" rel='stylesheet' type='text/css' />

<link href="{{ url_for('static', filename='css/font-awesome.min.css') }}"rel="stylesheet">

<!-- google fonts -->
```

```
<link href="//fonts.googleapis.com/css?family=Thasadith:400,400i,700,700i&amp;subset=latin-ext,thai,vietnamese"
                    rel="stylesheet">

<!-- //google fonts -->

<style>
html,
body {
    margin: 0;
    font-size: 100%;
    background: rgba(250, 246, 222, 0.452);
          font-family: 'Thasadith', sans-serif;
}

html {
  scroll-behavior: smooth;
}
body a {
    text-decoration: none;
    transition: 0.5s all;
    -webkit-transition: 0.5s all;
    -moz-transition: 0.5s all;
    -o-transition: 0.5s all;
    -ms-transition: 0.5s all;
          font-family: 'Thasadith', sans-serif;
}

body img {
    max-width: 100%;
}

a:hover {
    text-decoration: none;
}

input[type="button"],
input[type="submit"],
input[type="text"],
input[type="email"],
input[type="search"] {
    transition: 0.5s all;
    -webkit-transition: 0.5s all;
    -moz-transition: 0.5s all;
    -o-transition: 0.5s all;
    -ms-transition: 0.5s all;
```

```
}

h1,
h2,
h3,
h4,
h5,
h6 {
   margin: 0;
        color: #323648;
}
li {
   list-style-type: none;
}

p {
   margin: 0;
   font-size: 17px;
   line-height: 2em;
   letter-spacing: 2px;
   color: #707579;
   font-weight: 600;
}

ul {
   margin: 0;
   padding: 0;
}


/*-- header --*/

header {
   position: absolute;
   z-index: 9;
   width: 100%;
}

.toggle,
[id^=drop] {
        display: none;
}

/* Giving a background-color to the nav container. */
nav {
        margin:0;
```

```
        padding: 0;
        /* position: relative; */
}


#logo a {
        float: left;
   font-size: .8em;
   display: initial;
   margin: 0;
   letter-spacing: 1px;
   color: #fff;
   font-weight: 600;
   padding: 3px 0;
   border: none;
}
#logo a span.fa {
   color: #e8cd30;
}


/* Since we'll have the "ul li" "float:left"
 * we need to add a clear after the container. */

nav:after {
        content:"";
        display:table;
        clear:both;
}

/* Removing padding, margin and "list-style" from the "ul",
 * and adding "position:relative" */
nav ul {
        float: right;
        padding:0;
        margin:0;
        list-style: none;
        position: relative;
        }

/* Positioning the navigation items inline */
nav ul li {
        margin: 0px;
        display:inline-block;
        }
```

```
/* Styling the links */
nav a {
   color: #ddd;
   text-transform: capitalize;
   letter-spacing: 1px;
   padding-left: 0;
   padding-right: 0;
        padding: 10px 0;
   font-weight: 700;
}

header {
                        background-color: rgba(30, 30, 30, 1);
                        margin-top: 0rem;
                        display: block;
}
/* Stylinf the toggle lable */
        .toggle {
                display: block;
                padding: 5px 15px;
                font-size: 20px;
                text-decoration: none;
                border: none;
                float: right;
                background-color: #009f4d;
                color: #fff;
        }
        .menu .toggle {
                float: none;
                text-align: center;
                margin: auto;
                width: 30%;
                padding: 5px;
                font-weight: normal;
                font-size: 15px;
                letter-spacing: 1px;
        }

        .toggle:hover {
                color:#333;
                background-color: #fff;
        }

        /* Display Dropdown when clicked on Parent Lable */
        [id^=drop]:checked + ul {
                display: block;
```

```
        background: #fff;
        padding: 15px 0;
        width:100%;
        text-align: center;
}

/* Change menu item's width to 100% */
nav ul li {
        display: block;
        width: 100%;
        padding: 7px 0;
        }
nav a{
        padding: 5px 0;
}
nav a:hover {
        color: #333;
}
.login-icon {
        text-align: center;
}
nav ul ul .toggle,
nav ul ul a {
        padding: 0 40px;
}

nav ul ul ul a {
        padding: 0 80px;
}

nav a:hover,
nav ul ul ul a {
        background-color: transparent;
}

nav ul li ul li .toggle,
nav ul ul a,
nav ul ul ul a{
        padding:14px 20px;
        color:#FFF;
        font-size:17px;
}


nav ul li ul li .toggle,
nav ul ul a {
```

```
		background-color: #fff;
	}
	nav ul ul li a {
		font-size: 15px;
	}
	ul.inner-ul{
		padding: 0!important;
	}
	/* Hide Dropdowns by Default */
	nav ul ul {
		float: none;
		position:static;
		color: #ffffff;
		/* has to be the same number as the "line-height" of "nav a" */
	}

	/* Hide menus on hover */
	nav ul ul li:hover > ul,
	nav ul li:hover > ul {
		display: none;
	}

	/* Fisrt Tier Dropdown */
	nav ul ul li {
		display: block;
		width: 100%;
		padding: 0;
	}

	nav ul ul ul li {
		position: static;
		/* has to be the same number as the "width" of "nav ul ul li" */

	}

}

@media all and (max-width : 330px) {

	nav ul li {
		display:block;
		width: 94%;
	}

}
.user span.fa {
```

```
    font-size: 25px;
    color: #fff;
}
/*-- //header --*/

/* banner style */
.banner_w3lspvt {
    position: relative;
    z-index: 1;
}

.banner-top {
    background: url(../images/3.jpg) no-repeat center;
    background-size: cover;
    -webkit-background-size: cover;
    -moz-background-size: cover;
    -o-background-size: cover;
    -moz-background-size: cover;
}

.banner-top1 {
    background: url(../images/1.jpg) no-repeat center;
    background-size: cover;
    -webkit-background-size: cover;
    -moz-background-size: cover;
    -o-background-size: cover;
    -moz-background-size: cover;
}

.banner-top2 {
    background: url(../images/5.jpg) no-repeat center;
    background-size: cover;
    -webkit-background-size: cover;
    -moz-background-size: cover;
    -o-background-size: cover;
    -moz-background-size: cover;
}

.banner-top3 {
    background: url(../images/2.jpg) no-repeat center;
    background-size: cover;
    -webkit-background-size: cover;
    -moz-background-size: cover;
    -o-background-size: cover;
    -moz-background-size: cover;
}
```

```
.w3layouts-banner-info {
	padding-top: 16em;
}

.w3layouts-banner-info h3 {
	font-size: 4em;
	text-shadow: 3px 4px 6px rgba(45, 45, 45, 0.15);
	font-weight: 600;
	color: #fff;
	letter-spacing: 10px;
	text-transform: uppercase;
}
.w3layouts-banner-info p {
	max-width: 650px;
	color: #fff;
}
.w3layouts-banner-info h4 {
	color: #eee;
	letter-spacing: 5px;
		line-height: 35px;
		text-transform: capitalize;
}

.w3layouts-banner-info i {
	vertical-align: middle;
}

.banner-top,
.banner-top1,
.banner-top2,
.banner-top3 {
	min-height: 770px;
}
.overlay {
	min-height: 770px;
	background: rgba(0, 0, 0, 0.4);
}
.overlay1 {
	min-height: 770px;
	background: rgba(0, 0, 0, 0.5);
}

.button-style {
	padding: 15px 40px;
	color: #fff;
```

```
    font-size: 16px;
    font-weight: 600;
    text-transform: uppercase;
    letter-spacing: 3px;
    border: 2px solid #ccc;
    background: none;
    display: inline-block;
}

.button-style:hover {
    color: #fff;
}

/*-- //banner style --*/

/*-- about --*/
h3.heading {
    font-size: 40px;
    letter-spacing: 2px;
    font-weight: 600;
}
p.about-text {
    width: 80%;
}
.feature-grids .f-icon {
    vertical-align: middle;
    background: #009f4d;
    width: 70px;
    height: 70px;
    line-height: 70px;
    margin: 0.5em auto 0;
    border-radius: 50%;
}
.feature-grids span.fa {
    color: #fff;
    font-size: 20px;
    line-height: 70px;
}
.feature-grids h3 {
    font-size: 22px;
    font-weight: 600;
    letter-spacing: 3px;
    line-height: 30px;
    text-transform: uppercase;
}
.feature-grids  p {
```

```
    letter-spacing: 1px;
}
/*-- //about --*/

/*-- core grids --*/
.core-grids p {
    letter-spacing: 1px;
}
.core-right h3 {
    font-size: 24px;
    line-height: 42px;
    letter-spacing: 2px;
    font-weight: 600;
    text-transform: uppercase;
}
/*-- //core grids --*/

/*-- works --*/
.serives-agile {
    background: #009f4d;
}
.serives-agile h3.heading{
        color: #fff;
}
.welcome-grid {
    width: 20%;
    float: left;
}

.welcome-grid h4 {
    font-size: 22px;
    letter-spacing: 2px;
    color: #fff;
    font-weight: 600;
    text-transform: uppercase;
}
.welcome-grid span.fa {
    color: #5eca9f;
    color: #e8cd30;
    font-size: 50px;
    margin-bottom: 10px;
}

.welcome-grid p {
    color: #ccc;
    line-height: 1.8em;
```

```
    font-size: 16px;
}
/*-- //works --*/

/*-- bg --*/
.background-img {
    background: url(../images/5.jpg) no-repeat center;
    background-size: cover;
    -webkit-background-size: cover;
    -moz-background-size: cover;
    -o-background-size: cover;
    -moz-background-size: cover;
}
/*-- blog info --*/

.blog-grids {
    margin-bottom: 120px;
}

.blog-left,.blog-middle,.blog-right{
        position: relative;
}
.blog-info {
    background: #fff;
    padding: 30px;
    margin-top: -2em;
    position: absolute;
    left: 6%;
    right: 6%;
    top: 200px;
        box-shadow: 0 3px 5px -1px rgba(0, 0, 0, 0.08), 0 5px 8px 0 rgba(0, 0, 0, 0.12), 0 1px
14px 0 rgba(0, 0, 0, 0.06);
}
.blog-info p {
    letter-spacing: 1px;
    line-height: 28px;
}
.blog-info h4 {
    font-size: 22px;
    line-height: 42px;
    letter-spacing: 2px;
    font-weight: 600;
    text-transform: uppercase;
}
.blog-info h4 span.fa {
    color: #009f4d;
```

```
}
/*-- //blog info --*/


/*--  text --*/
.text {
    background: url(../images/2.jpg) no-repeat center;
    background-size: cover;
    position: relative;
}
.text:before {
    content: '';
    position: absolute;
    width: 100%;
    height: 100%;
    top: 0;
    left: 0;
    opacity: 0.6;
    background: #000;
}
.text h3.heading{
            color: #fff;
}
.text h3.heading span {
    color: #e8cd30;
}
.text p {
    color: #ccc;
    width: 80%;
    margin: auto;
    letter-spacing: 1px;
}
.text a.btn {
    font-size: 17px;
    letter-spacing: 2px;
    color: #333;
    font-weight: 700;
    padding: 12px 25px;
    margin-top: 30px;
    border-radius: 4px;
    background: #e8cd30;
    display: inline-block;
}
.text a.btn1 {
    font-size: 17px;
    letter-spacing: 2px;
```

```
    color: #fff;
    font-weight: 700;
    padding: 12px 25px;
    margin-top: 30px;
    border-radius: 4px;
    background: #009f4d;
    display: inline-block;
}
/*--  //text --*/

/*-- footer --*/

p.footer-para {
    max-width: 650px;
    font-size: 15px;
}

/*-- footer logo --*/
.logo2 {
    position: relative;
}

.logo2 a {
    font-size: 36px;
    font-weight: 600;
    color: #fff;
    letter-spacing: 1px;
}

.logo2 a span.fa {
        color: #e8cd30;
}

/*-- //footer logo --*/

/*-- footer home dashboard about  --*/
.homelogo {
    position: relative;
}

.homelogo a {
    font-size: 18px;
    font-weight: 300;
    color: #fff;
    letter-spacing: 1px;
}
```

```
.homelogo a span.fa {
        color: #e8cd30;
}

/*-- //footer logo --*/


/*-- social icons --*/
.footercopy-social ul li,
.contact-left-footer ul li {
   display: inline-block;
}
footer{
   background: #191818;
}
.footercopy-social ul li a span.fa {
   width: 20px;
   font-size: 20px;
   color: #666;
   transition: 0.5s all;
   -webkit-transition: 0.5s all;
   -moz-transition: 0.5s all;
   -o-transition: 0.5s all;
   -ms-transition: 0.5s all;
}

/*-- //social icons --*/

/*-- address --*/
.contact-left-footer ul li p span.fa {
   color: #aaa;
}

.contact-left-footer ul li p a,
.contact-left-footer ul li p {
   color: #707579;
   font-size: 16px;
   font-weight: 600;
}

/*-- //address --*/

/*-- copyright --*/
.w3l-copy p {
   letter-spacing: 1px;
```

```
}

.w3l-copy p a {
   color: #aaa;
}
/*-- //copyright --*/
/*-- //footer --*/

/*-- inner banner --*/
.inner-banner{
   background: url(../images/2.jpg) no-repeat center;
   background-size: cover;
         min-height: 250px;
         position: relative;
}
.inner-banner:before {
   content: '';
   position: absolute;
   width: 100%;
   height: 100%;
   top: 0;
   left: 0;
   opacity: 0.6;
   background: #000;
}
/*-- //inner banner --*/

/*-- about page --*/
.about-left h5 {
   color: #009f4d;
   font-weight: 600;
   letter-spacing: 1px;
   font-size: 24px;
}
.about-left h3 {
   font-size: 32px;
   line-height: 44px;
   letter-spacing: 2px;
   font-weight: 600;
   text-transform: uppercase;
}
.about-left h4 {
   line-height: 1.5;
   font-size: 25px;
   letter-spacing: 2px;
   font-weight: 600;
```

```
    text-transform: capitalize;
}
.about-right p{
        letter-spacing: 1px;
}

.about span.fa-quote-left {
    font-size: 20px;
    vertical-align: top;
    color: #009f4d;
}

.banner-bottom {
    background: #f8f9fa;
}
.wthree_banner_bottom_grid_left span {
    background: #ffc168;
    color: #fff;
    width: 80px;
    height: 80px;
    border-radius: 50%;
    text-align: center;
    font-size: 38px;
    line-height: 2;
}

.wthree_banner_bottom_grid_left.icons-w3pvt2 span {
    background: #ff4f81;
}

.wthree_banner_bottom_grid_left.icons-w3pvt3 span {
    background: #2dde98
}


/* about bottom */

h4.abt-text {
    font-size: 2.5em;
    letter-spacing: 2px;
    color: #fff;
    line-height: 1.4em;
}
.abt_bottom{
    background: #009f4d;
}
```

```
.abt_bottom a.serv_link {
   font-size: 17px;
   letter-spacing: 2px;
   color: #333;
   font-weight: 700;
   padding: 12px 25px;
   border-radius: 4px;
   background: #e8cd30;
   display: inline-block;
        margin-top:10px;
}
/* //about bottom */

/* stats */
section.w3_stats {
   background: url(../images/1.jpg) no-repeat center;
   background-size: cover;
   position: relative;
}
section.w3_stats h3.heading {
        color: #fff;
}
.counter span.fa {
   color: #fff;
   font-size: 3em;
}

.timer {
   font-size: 3em;
   font-weight: 300;
        color: #fff;
}

p.count-text {
   letter-spacing: 2px;
   font-weight: 600;
        color: #fff;
}

/* //stats */

/* news */
.news{
   background: #f8f9fa;
}
.feedback-info h4 {
```

```
    font-size: 22px;
    line-height: 34px;
    letter-spacing: 1px;
    font-weight: 600;
    text-transform: uppercase;
}
.feedback-info p {
    letter-spacing: 1px;
    line-height: 1.8em;
}

.feedback-info h4 a {
    letter-spacing: 1px;
    line-height: 1.4;
}


.feedback-img {
    float: left;
    width: 25%;
}

.feedback-img-info {
    float: right;
    width: 68%;
    margin: 1.5em 0 0 1em;
}

.feedback-img-info h5 {
    color: #504e4e;
    font-size: 17px;
    letter-spacing: 1px;
    font-weight: 600;
}
.feedback-info {
    background: #fff;
}
/* //news */

/*-- team --*/
.team-text h4 {
    font-size: 22px;
    letter-spacing: 2px;
    font-weight: 600;
    text-transform: uppercase;
    margin-top: 1em;
```

```
}
.caption ul li {
    display: inline-block;
    margin: 0 5px;
}
.caption ul li a {
    color: #aaa;
    font-size: 14px;
}
/*-- //team --*/

/*-- //about page --*/


/*-- services page --*/
/* home grid */

.home-grid {
    padding: 1.5em;
    border: 1px solid #555;
    position: relative;
        text-align: center;
}

.home-grid span {
    color: #009f4d;
    font-size: 1.5em;
    font-weight: 700;
    position: absolute;
    top: 0;
    left: 0px;
    padding: 2px 7px;
}


.wthree-bnr-btn {
    display: inline-block;
    border-top: 1px solid #1dc6bc;
    border-radius: 0;
    margin-top: 1em;
    padding: 10px 0;
    color: #5341b4;
    text-transform: capitalize;
    font-size: 14px;
    letter-spacing: 0.5px;
    font-weight: 800;
```

```
}


</style></head>

<body>
<!-- banner -->
   <section class="banner_w3lspvt" id="home">
            <div class="csslider infinity" id="slider1">
           <div class="banner-top"><div class="overlay">
         <div class="container"><div class="w3layouts-banner-info text-center">


                        <h3 class="text-wh">CropLabs</h3>
<h4 class="text-wh mx-auto my-4"><b>Get insights about your crop before you sow your seed.</b></h4><br></div></div></div></div></div>

   </section>

   <section class="core-value py-5">
            <div class="container py-md-4">

                  <div class="col-lg-6 core-right">

 <h3 class="mt-4">SUSTAINABILITY ALONG WITH PROFITABLE METHODS TO CULTIVATE CROPS</h3>


            <p class="mt-3">We use state-of-the-art machine learning and data mining technologies to help you  guide through the entire farming process. Make informed decisions to understand the demographics of your area,
understand the factors that affect your crop and keep them healthy for a super awesome successful yield.</p>

   <section class="blog py-5">
            <div class="container py-md-5">
                  <h3 class="heading mb-sm-5 mb-4 text-center"> Our Services</h3>
                              <img src="{{url_for('static', filename='images/ss3.jpg') }}" class="img-fluid" alt="" />

  <!--<a href="crop_prediction.html"> -->

<a href="{{ url_for('crop_prediction') }}">
                                    <div class="blog-info">
                                          <h4>Crop</h4>
```

```
<p class="mt-2"> Recommendation about the yield of the crops which is cultivated and best suited for the respective conditions</p>
        </div></a> <div class="col-lg-4 col-md-6 blog-right mb-lg-0 mb-sm-5 pb-lg-0 pb-md-5">
<a href="https://www.indiatoday.in/education-today/gk-current-affairs/story/10-important-government-schemes-agriculture-sector-divd-1593413-2019-08-30">

<div class="blog-info">
                <h4>More Information</h4>
                <p class="mt-2">Know before you grow.More information about the statistics of the Indian Agriculture and other details</p>
</div>
</a>
 </div>


<div class="col-lg-4 col-md-6 blog-right mb-lg-0 mb-sm-5 pb-lg-0 pb-md-5">

<img src="{{url_for('static', filename='images/soil.jpg') }}" class="img-fluid" alt="" style="width: 120%;" />

<a href="https://youngagrarians.org/soil-testing-new-farmers/">
<div class="blog-info">

<h4>Know your soil</h4>

<p class="mt-2">Soil tests are used to determine the soil's nutrient level and pH content. Farmers get to know the needs of soil</p>
</div>
</a> </div>
</div>
</div>
   </section>

<div class="container">
                {% block content %}
                {% endblock %}
          </div>
</body>
</html>
```

CHAPTER 6 ANALYSIS AND USER INTERFACE

6.1 Experimental Analysis

Experimental data in science and engineering is data produced by a measurement, test method, experimental design, or quasi-experimental design. Experimental data can be reproduced by a variety of different investigators and mathematical analysis may be performed on these data.

6.1.1 Cross Validation Score

Cross-validation may be a statistical procedure that is used to estimate the skill of machine learning models. It is commonly utilized in applied machine learning to match and choose a model for a given predictive modelling problem because it is easy to know, easy to implement, and leads to skill estimates that generally have a lower bias than other methods. It is also known as a resampling procedure used to evaluate machine learning models on a limited data sample.

Cross-validation gives a more accurate measure of model quality, which is especially important if you are making a lot of modeling decisions. Sometimes it takes longer to run because it estimates multiple models. It is a popular method because it is simple to understand and it generally results in a less biased or less optimistic estimate of the model skill than other methods, such as a simple train/test split.

Cross-validation results of Random forest and Decision Tree Regression can be seen in below figure 6.1. The number of splits are 5 and test size is 0.2 which means 20 records out of every 100 records are taken for the test.

Cross Validation for RandomForest Regression:

```
from sklearn.model_selection import ShuffleSplit
from sklearn.model_selection import cross_val_score
from sklearn.ensemble import RandomForestRegressor

cv = ShuffleSplit(n_splits=5, test_size=0.2)

cross_val_score(RandomForestRegressor(n_estimators = 10), X, y, cv=cv)
```

```
array([0.89842273, 0.83105375, 0.96403624, 0.82111085, 0.98910053])
```

Cross Validation for DecisionTree Regression:

```
[ ] from sklearn.model_selection import ShuffleSplit
    from sklearn.model_selection import cross_val_score
    from sklearn.tree import DecisionTreeRegressor

    cv = ShuffleSplit(n_splits=5, test_size=0.2)

    cross_val_score( DecisionTreeRegressor(random_state=0), X, y, cv=cv)
```

```
array([0.85325866, 0.89433854, 0.90529817, 0.89356039, 0.76169771])
```

Fig-6.1 Cross Validation results for regression techniques used.

6.1.2 Performance Measures

The end users of prediction tools should be able to understand how evaluation is done and how to interpret the results. Six main performance evaluation measures are introduced. These include

- Sensitivity
- Specificity
- Positive predictive value
- Negative predictive value
- Accuracy and
- Matthews correlation coefficient.

Accuracy: Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations. One may think that if we have high accuracy then our model is best. Yes, accuracy is a great measure but only when you have symmetric datasets where values of false positives and false Negatives are almost the same. Therefore, you must look at other parameters to evaluate the performance of your model. For our model, we have got 0.803 which means our model is approx. 80% accurate.

Accuracy = TP+TN/TP+FP+FN+TN

True Positives (TP) - These are the correctly predicted positive values which means that the value of actual class is yes, and the value of predicted class is also yes. E.g. if the actual class value indicates that this passenger survived, and the predicted class tells you the same thing.

True Negatives (TN) - These are the correctly predicted negative values which means that the value of actual class is no, and value of predicted class is also no. E.g. If the actual class says this passenger did not survive and the predicted class tells you the same thing.

False Positives (FP) – When actual class is no and predicted class is yes. E.g. if the actual class says this passenger did not survive but the predicted class tells you that this passenger will survive.

False Negatives (FN) – When actual class is yes but predicted class in no. E.g. if the actual class value indicates that this passenger survived, and the predicted class tells you that the passenger will die.

6.1.3 Performance Metrics

To evaluate how good our regression model is, we can use the following metrics:

- **R-squared:** indicate how many variables compared to the total variables the model predicted. R-squared does not take into consideration any biases that might be present in the data. Therefore, a good model might have a low R-squared value, or a model that does not fit the data might have a high R-squared value.
- **Average error:** the numerical difference between the predicted value and the actual value.
- **Mean Square Error (MSE)**: good to use if you have a lot of outliers in the data.
- **Median error:** the average of all differences between the predicted and the actual values.
- **Average absolute error:** like the average error, only you use the absolute value of the difference to balance out the outliers in the data.
- **Median absolute error** represents the average of the absolute differences between prediction and actual observation. All individual differences have equal weight, and big outliers can therefore affect the final evaluation of the model.

Mean Absolute Error

Mean Absolute Error is a model evaluation metric used with regression models. The mean absolute error of a model with respect to a test set is the mean of the absolute values of the individual prediction errors on all instances in the test set. Each prediction error is the difference between the true value and the predicted value for the instance.

Mean Square Error

The mean squared error (MSE) tells you how close a regression line is to a set of points. It does this by taking the distances from the points to the regression line (these distances are the "errors") and squaring them. The squaring is necessary to remove any negative signs. It also gives more weight to larger differences. It's called the mean squared error

Root Means Square Error

Root mean squared error (RMSE) is the square root of the mean of the square of all of the errors. The use of RMSE is very common, and it is considered an excellent general-purpose error metric for predictions. RMSE is a good measure of accuracy, but only to compare prediction errors of different models or model configurations for a particular variable and not between variables, as it is scale-dependent.

The models are tested using the above metrics and their results are compared manually. The below figure 6.2 shows the test results of various metrics.

```
] from sklearn import metrics
  print('Mean Absolute Error:', metrics.mean_absolute_error(y_test//1000000, prediction//10000000))
  print('Mean Square Error:', metrics.mean_squared_error(y_test//10000000, prediction//10000000))
  print('Root Mean Square Error:', np.sqrt(metrics.mean_squared_error(y_test//10000000, prediction//10000000)))

  Mean Absolute Error: 1.311431623931624
  Mean Square Error: 1.2414529914529915
  Root Mean Square Error: 1.1142050939809023

] prediction=rf_obj.predict(X_test)

] print("Mean Absolute Error:",metrics.mean_absolute_error(y_test//10000000, prediction//10000000))
  print('Mean Square Error:', metrics.mean_squared_error(y_test//10000000, prediction//10000000))
  print('Root Mean Square Error:', np.sqrt(metrics.mean_squared_error(y_test//10000000, prediction//10000000)))

  Mean Absolute Error: 0.0438034188034188
  Mean Square Error: 0.6773504273504274
  Root Mean Square Error: 0.823013017728412
```

Fig-6.2 Comparison of performance metrics on Random Forest and Decision Tree Regression

6.1.4 Experimental Analysis

The best fit model for our system has been found out through the above-mentioned model's comparison. So now let's take some sample data and analyze how our model is performing with respect to that data.

The below Table-6.1 shows a sample of data points along with sample id, it's actual rating and predicted rating by the model.

Test Case Number	Actual Value	Predicted Value
01	18000	18000
02	7100	7100
03	9400	9400
04	7100	7310
05	500	500

Table 6.1- Test Cases

In the above table, the performance of the system is compared for a sample of 5 data points. The predictions are made by the Random Forest Regressor Model as it is our best fit model. We can see for our sample that predictions production value do not deviate / vary much from the actual production value.

6.2 User Interface

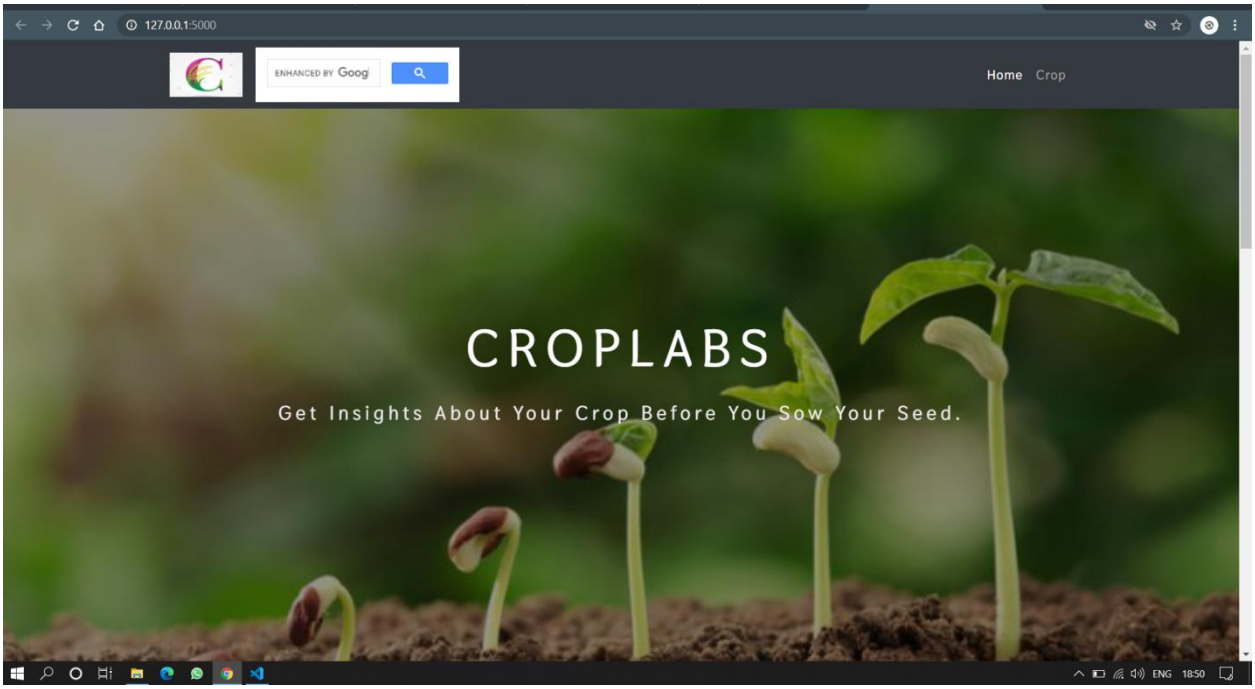


Fig 6.3 Home page

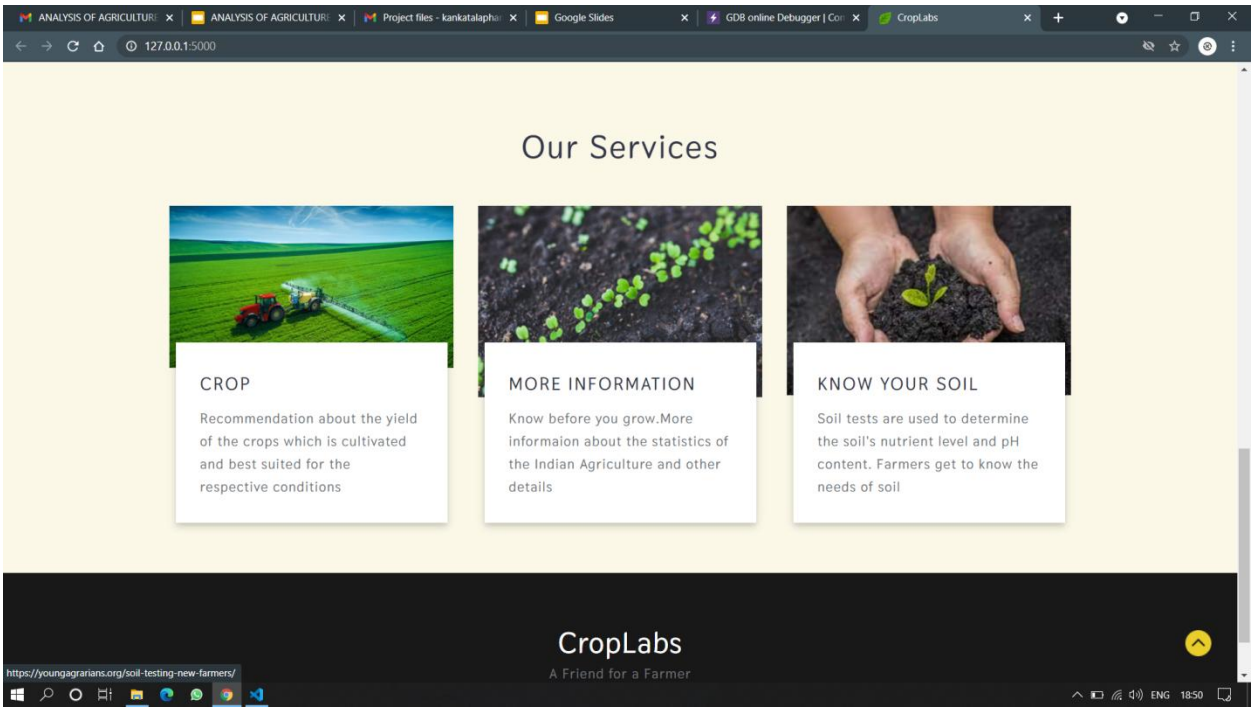


Fig 6.4 Our Services

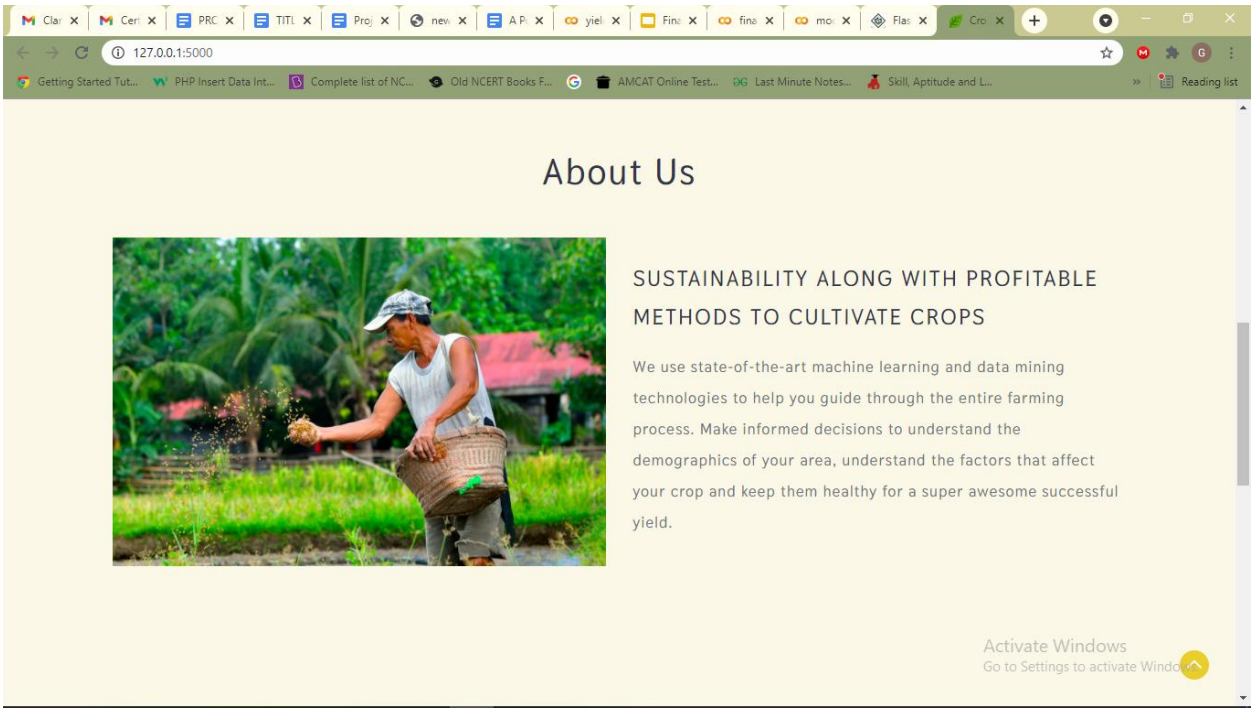


Fig 6.5 About Us

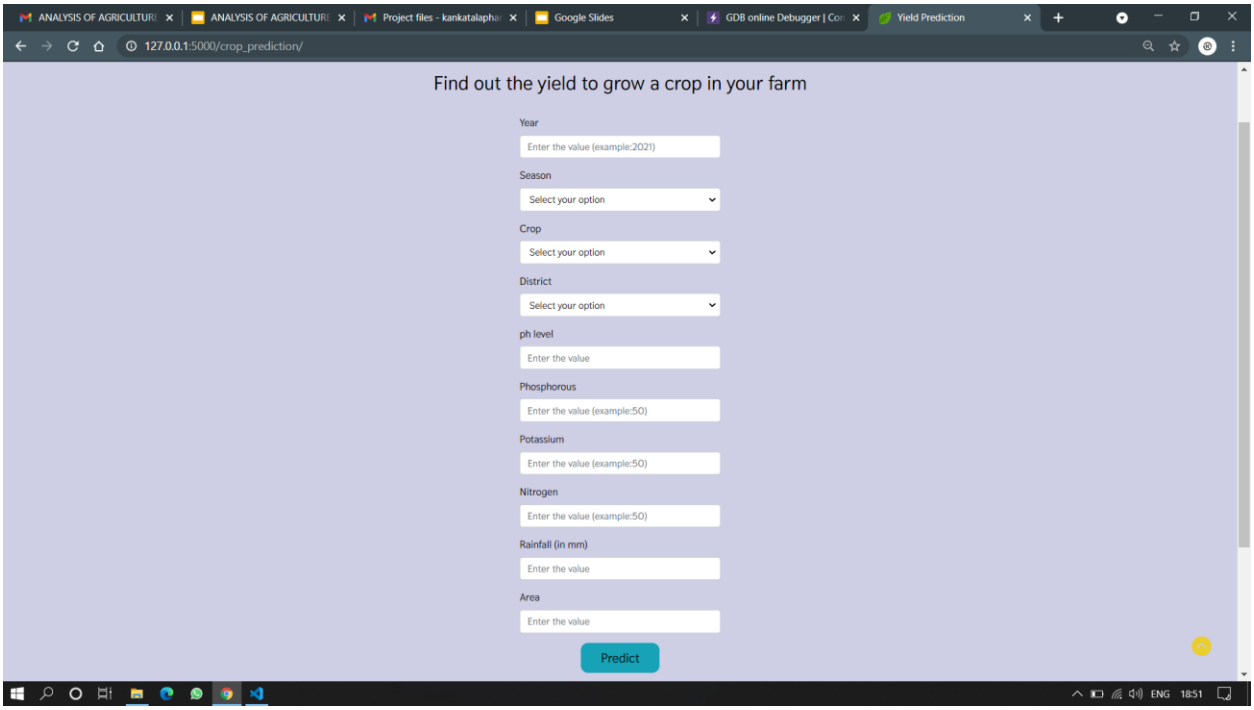


Fig 6.6 Crop yield prediction

CHAPTER 7 CONCLUSION AND FUTURE WORK

7.1 Conclusion

Both Decision tree regression and Random Forest regression techniques are implemented on the input data to assess the best performance yielding method. These methods are compared using performance metrics. According to the analyses of metrics both the algorithms work well, but Random Forest regression gives a better accuracy score on test data than Decision tree regression. The proposed work can also be extended to analyze the climatic conditions and other factors for the crop and to increase the crop production.

7.2 Future Work

In the future, our model can be trained further with more data points of different states. This system can be extended to different climatic conditions also. The proposed model can be used not only for our state but also for other states if we provide more accurate data to it using satellite and sensor data.

APPENDIX

Google Colab

Google Colab is free to use like other G Suite products. Google Colaboratory more commonly referred to as “Google Colab” or just simply “Colab” is a research project for prototyping machine learning models on powerful hardware options such as GPUs and TPUs. ... Google Colab is free to use like other G Suite products. With Colab you can import an image dataset, train an image classifier on it, and evaluate the model, all in just a few lines of code. Colab notebooks execute code on Google's cloud servers, meaning you can leverage the power of Google hardware, including GPUs and TPUs, regardless of the power of your machine.

Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms and can be freely distributed.

Machine Learning

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it to learn for themselves.

Supervised Learning

Supervised learning is the type of machine learning in which machines are trained using well "labelled" training data, and based on that data, machines predict the output. The labelled data means some input data is already tagged with the correct output. In supervised learning, the training data provided to the machines work as the supervisor that teaches the machines to predict the output correctly.

Unsupervised Learning

Unsupervised learning is a type of machine learning in which models are trained using an unlabeled dataset and can act on that data without any supervision. Instead, models itself find the hidden patterns and insights from the given data. It can be compared to learning which takes place in the human brain while learning new things.

Regression Analysis

Regression analysis consists of a set of machine learning methods that allow us to predict a continuous outcome variable (y) based on the value of one or multiple predictor variables (x). Briefly, the goal of a regression model is to build a mathematical equation that defines y as a function of the x variables. Predicting prices of a house given the features of house like size, price etc. is one of the common examples of Regression.

NumPy

NumPy is a Python library used for working with arrays. It also has functions for working in the domain of linear algebra, Fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely. NumPy stands for Numerical Python.

Pandas

Pandas is an open-source library that allows you to perform data manipulation and analysis in Python. Pandas Python library offers data manipulation and data operations for numerical tables and time series. Pandas provide an easy way to create, manipulate, and wrangle the data

Sk-Learn

Scikit-learn is a free machine learning library for Python. It features various algorithms like support vector machine, random forests, and k-neighbors, and it also supports Python numerical and scientific libraries like NumPy and SciPy Scikit-learn is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering, and dimensionality reduction.

Matplotlib

Matplotlib is a cross-platform, data visualization and graphical plotting library for Python and its numerical extension NumPy. As such, it offers a viable open source alternative to MATLAB. Developers can also use matplotlib's APIs (Application Programming Interfaces) to embed plots in GUI applications.

Seaborn

Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures. Its plotting functions operate on data frames and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots. Its dataset-oriented, declarative API lets you focus on what the different elements of your plots mean, rather than on the details of how to draw them.

Pickle

Python pickle module is used for serializing and de-serializing python object structures. The process to convert any kind of python objects (list, dict, etc.) into byte streams (0s and 1s) is called pickling or serialization or flattening or marshalling.

Flask

Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it. Flask is based on the Werkzeug WSGI toolkit and Jinja2 template engine.

REFERENCES

Concept:

- DATA MINING-Concepts and techniques,3rd edition by The Morgan Kaufmann Series in Data Management Systems).

Algorithms:

- https://towardsdatascience.com/how-dbscan-works-and-why-should-i-use-it-443b4a191c80
- https://link.springer.com/chapter/10.1007/978-3-319-14142-8_1
- https://link.springer.com/chapter/10.1007/978-3-319-14142-8_6
- https://scikit-learn.org/stable/modules/clustering.html#dbscan
- http://troindia.in/journal/ijcesr/vol4iss3/66-70.pdf

Data:

- http://www.apagrisnet.gov.in/
- http://www.apagrisnet.gov.in/2018/weekly/October/weekly_report_(Rabi)_06_21-11-18.pdf
- https://desap.in/jsp/social/AGRICULTURALSTATISTICSATAGLANCE201819.pdf
- https://desap.in/jsp/social/SEASONANDCROPREPORT201819.pdf

Others:

- https://stackoverflow.com/questions/58983528/how-to-find-optimal-parametrs-for-dbscan
- https://medium.com/@tarammullin/dbscan-parameter-estimation-ff8330e3a3bd
- https://blog.exploratory.io/visualizing-k-means-clustering-results-to-understand-the-characteristics-of-clusters-better-b0226fb3d
- https://scikit-learn.org/stable/modules/model_evaluation.html
- https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3303716/#:~:text=The%20end%20users%20of%20prediction,accuracy%20and%20Matthews%20correlation%20coefficient.
- https://www.investopedia.com/articles/investing/100615/4-countries-produce-most-food.asp
- http://www.fao.org/fileadmin/templates/ess/documents/meetings_and_workshops/GS_SAC_2013/Improving_methods_for_crops_estimates/Crop_Yield_Forecasting_Methods_and_Early_Warning_Systems_Lit_review.pdf

Majumdar *et al. J Big Data (2017) 4:20*
DOI 10.1186/s40537-017-0077-4

Journal of Big Data

RESEARCH Open Access

Analysis of agriculture data using data mining techniques: application of big data

Jharna Majumdar*, Sneha Naraseeyappa and Shilpa Ankalaki

*Correspondence:
jharna.majumdar@gmail.com
Department of M.Tech CSE,
NMIT, Bangalore 560064,
India

Abstract

In agriculture sector where farmers and agribusinesses have to make innumerable decisions every day and intricate complexities involves the various factors influencing them. An essential issue for agricultural planning intention is the accurate yield estimation for the numerous crops involved in the planning. Data mining techniques are necessary approach for accomplishing practical and effective solutions for this problem. Agriculture has been an obvious target for big data. Environmental conditions, variability in soil, input levels, combinations and commodity prices have made it all the more relevant for farmers to use information and get help to make critical farming decisions. This paper focuses on the analysis of the agriculture data and finding optimal parameters to maximize the crop production using data mining techniques like PAM, CLARA, DBSCAN and Multiple Linear Regression. Mining the large amount of existing crop, soil and climatic data, and analysing new, non-experimental data optimizes the production and makes agriculture more resilient to climatic change.

Keywords: Big Data, PAM, CLARA and DBSCAN

Background

Today, India ranks second worldwide in the farm output. Agriculture is demographically the broadest economic sector and plays a significant role in the overall socio-economic fabric of India. Agriculture is a unique business crop production which is dependent on many climate and economy factors. Some of the factors on which agriculture is dependent are soil, climate, cultivation, irrigation, fertilizers, temperature, rainfall, harvesting, pesticide weeds and other factors. Historical crop yield information is also important for supply chain operation of companies engaged in industries. These industries use agricultural products as raw material, livestock, food, animal feed, chemical, poultry, fertilizer, pesticides, seed and paper. An accurate estimate of crop production and risk helps these companies in planning supply chain decision like production scheduling. Business such as seed, fertilizer, agrochemical and agricultural machinery industries plan production and marketing activities based on crop production estimates [1, 2]. There are 2 factors which are helpful for the farmers and the government in decision making namely:

a. It helps farmers in providing the historical crop yield record with a forecast reducing the risk management.

b. It helps the government in making crop insurance policies and policies for supply chain operation.

Data mining technique plays a vital role in the analysis of data. Data mining is the computing process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics, and database system. Unsupervised (clustering) and supervised (classifications) are two different types of learning methods in the data mining. Clustering is the process of examining a collection of "data points," and grouping the data points into "clusters" according to some distance measure. The goal is that data points in the same cluster have a small distance from one another, while data points in different clusters are at a large distance from one another. Cluster analysis divides data into well-formed groups. Well-formed clusters should capture the "natural" structure of the data [3]. This paper focuses on PAM, CLARA and DBSCAN clustering methods. These methods are used to categorize the different districts of Karnataka which are having similar crop production.

Literature survey

Clustering is considered as an unsupervised classification process [4]. A large number of clustering algorithms have been developed for different purposes [4–6]. Clustering techniques can be categorised into Partitioning clustering, Hierarchical clustering, Density-based methods, Grid-based methods and Model based clustering methods.

Partitioning clustering algorithms, such as K-means, K-medoids PAM, CLARA and CLARANS assign objects into k (predefined cluster number) clusters, and iteratively reallocate objects to improve the quality of clustering results. Hierarchical clustering algorithms assign objects in tree structured clusters, i.e., a cluster can have data point's representatives of low level clusters [7]. The idea of Density-based clustering methods is that for each point of a cluster the neighbourhood of a given unit distance contains at least a minimum number of points, i.e. the density in the neighbourhood should reach some threshold. The idea of the density-based clustering algorithm is that, for each point of a cluster, the neighbourhood of a given unit distance has to contain at least a minimum number of points [8].

There are different forecasting methodologies developed and evaluated by the researchers all over the world in the field of agriculture. Some of such studies are: Researchers like Ramesh and Vishnu Vardhan are analysed the agriculture data for the years 1965–2009 in the district East Godavari of Andhra Pradesh, India. Rain fall data is clustered into 4 clusters by adopting the K means clustering method. Multiple linear regression (MLR) is the method used to model the linear relationship between a dependent variable and one or more independent variables. The dependent variable is rainfall and independent variables are year, area of sowing, production. Purpose of this work is to find suitable data models that achieve high accuracy and a high generality in terms of yield prediction capabilities [9].

Bangladesh offers several varieties of rice which has different cropping season [10]. For this a prior study of climate (effect on temperature and rainfall) in Bangladesh and its effect on agricultural production of rice has been done. Then this study was being taken into regression analysis with temperature and rainfall. Temperature puts an adverse

consequence on the crop production. The data has been taken from the "Bangladesh Agricultural Research Council (BARC)" for past 20 years with 7 attributes: "rainfall", "max and min temperature", "sunlight", "speed of wind", "humidity" and "cloud-coverage". In Pre-processing, the whole dataset was divided in 3 month duration phases (March to June, July to October, November to February). For this duration, the average for every attribute has been taken and associated with it. This pre-processing has been done for each kind of rice variety. In clustering, the different pre-processed table has been analysed to find the sharable group of region based on similar weather attribute.

Soil characteristics are studied and analysed using data mining techniques. As an example, the k-means clustering is used for clustering soils in combination with GPS-based technologies [11]. Authors like Alberto Gonzalez-Sanchez, Juan Frausto-Solis and Waldo Ojeda-Bustamante have done extensive study on predictive ability of machine learning techniques such as multiple linear regression, regression trees, artificial neural network, support vector regression and k-nearest neighbour for crop yield production [12]. Wheat yield prediction using machine learning and advanced sensing techniques has done by Pantazi, DimitriosMoshou, Thomas Alexandridis and Abdul Mounem-Mouazen [13]. The aim of their work is to predict within field variation in wheat yield, based on on-line multi-layer soil data, and satellite imagery crop growth characteristics. Supervised self-organizing maps capable of handling existent information from different soil and crop sensors by utilizing an unsupervised learning algorithm were used. The software tool 'Crop Advisor' has been developed by S. Veenadhari, B. Misra and CD Singh [14] is an user friendly web page for predicting the influence of climatic parameters on the crop yields. C4.5 algorithm is used to find out the most influencing climatic parameter on the crop yields of selected crops in selected districts of Madhya Pradesh.

Methods

The objective of proposed work is to analyse the agriculture data using data mining techniques. In proposed work, agriculture data has been collected from following sources:

Dataset in agricultural sector [https://data.gov.in/, http://raitamitra.kar.nic.in/statistics],

Crop wise agriculture data [html://CROPWISE_NORMAL_AREA],

Agriculture data of different districts [http://14.139.94.101/fertimeter/Distkar.aspx], http://raitamitra.kar.nic.in/ENG/statistics.asp],

Agriculture data based on weather, temperature, and relative humidity [http://dmc.kar.nic.in/trg.pdf].

Input dataset consist of 6 year data with following parameters namely: year, State-Karnataka (28 districts), District, crop (cotton, groundnut, jowar, rice and wheat.), season (kharif, rabi, summer), area (in hectares), production (in tonnes), average temperature (°C), average rainfall (mm), soil, PH value, soil type, major fertilizers, nitrogen (kg/Ha), phosphorus (Kg/Ha),Potassium(Kg/Ha), minimum rainfall required, minimum temperature required.

In proposed work, modified approach of DBSCAN method is used to cluster the data based on districts which are having similar temperature, rain fall and soil type. PAM and CLARA are used to cluster the data based on the districts which are producing maximum crop production (In proposed work wheat crop is considered as example). Based

on these analyses we are obtaining the optimal parameters to produce the maximum crop production. Multiple linear regression method is used to forecast the annual crop yield.

Modified approach of DBSCAN

DBSCAN is a base algorithm for density based clustering containing large amount of data which has noise and outliers. DBSCAN has two parameters namely Eps and MinPts. However, traditional DBSCAN cannot produce optimal Eps value [15]. Determination of the optimal Eps value automatically is the one of the most necessary modification for the DBSCAN. Figure 1 briefs the modified approach of the DBSCAN method.

Modified DBSCAN proposes the method to find the minimum points and Epsilon (radius value) automatically. KNN plot is used to find out the epsilon value where input to the KNN plot (K value) is user defined. To avoid the user define K value as input to the KNN plot, Batchelor Wilkins clustering algorithm is applied to the database and obtain the K value along with its respective cluster centres. This K value is given as input to the KNN Plot.

Determination of Eps and Minpts

The Epsilon (Eps) value can be found by drawing a "K-distance graph" for entire data-points in dataset for a given 'K', obtained by the Batchelor Wilkins Algorithm [16]. Initially, the distance of a point to every 'K' of its nearest-neighbours is calculated. KNN plot is plotted by taking the sorted values of average distance values. When the graph is plotted, a knee point is determined in order to find the optimal Eps value [15].

Partition around medoids (PAM)

It is a partitioning based algorithm. It breaks the input data into number of groups. It finds a set of objects called medoids that are centrally located. With the medoids, nearest data points can be calculated and made it as clusters. The algorithm has two phases:

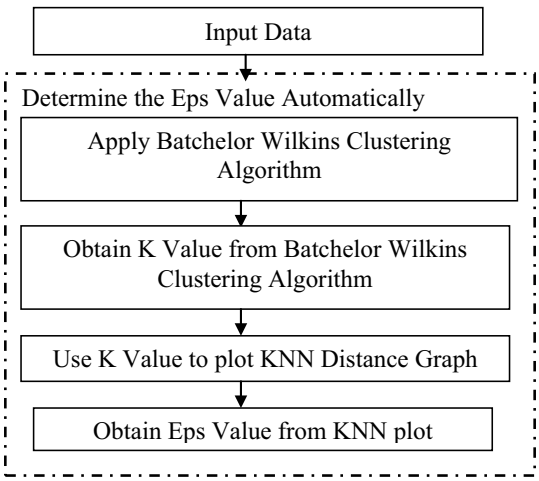


Fig. 1 Determine the Eps value automatically

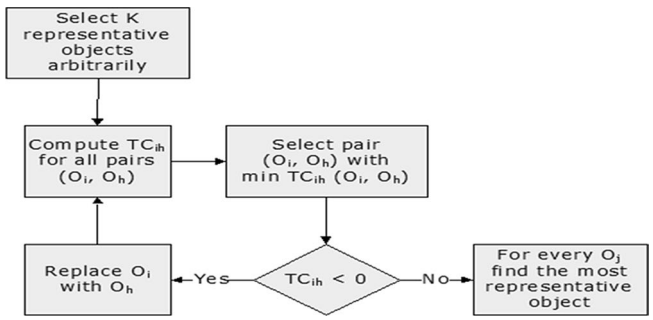


Fig. 2 PAM Algorithm steps

1. BUILD phase, a collection of k objects are selected for an initial set S.

- Arbitrarily choose k objects as the initial medoids.
- Until no change, do.

  – (Re) assign each object to the cluster with the nearest medoid.
  – Improve the quality of the k-medoids (randomly select a non medoid object, O random, compute the total cost of swapping a medoid with O random).

2. SWAP phase, one tries to improve the quality of the clustering by exchanging selected objects with unselected objects. Choose the minimum swapping cost.

Example: For each medoid m1, for each non-medoid data point d; Swap m1 and d, recompute the cost (sum of distances of points to their medoid), if total cost of the configuration increased in the previous step, undo the swap Fig. 2 depicts the steps involved the PAM algorithms.

CLARA (clustering large applications)

It is designed by Kaufman and Rousseeuw to handle large datasets, CLARA (clustering large applications) relies on sampling [17, 18]. Instead of finding representative objects for the entire data set, CLARA draws a sample of the data set, applies PAM on the sample, and finds the medoids of the sample. To come up with better approximations, CLARA draws multiple samples and gives the best clustering as the output. Here, for accuracy, the quality of the clustering is measured based on the average dissimilarity of all objects in the entire data set. Figure 3 briefs about the steps involved in the CLARA Algorithm.

Multiple linear regression to forecast the crop yield

Multiple linear regression is a variant of "linear regression" analysis. This model is built to establish the relationship that exists between one dependent variable and two or more independent variables [19].For a given dataset where $x_1 \ldots x_k$ are independent variables and Y is a dependent variable, the multiple linear regression fits the dataset to the model:

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_k x_{ki} + \varepsilon_i$$

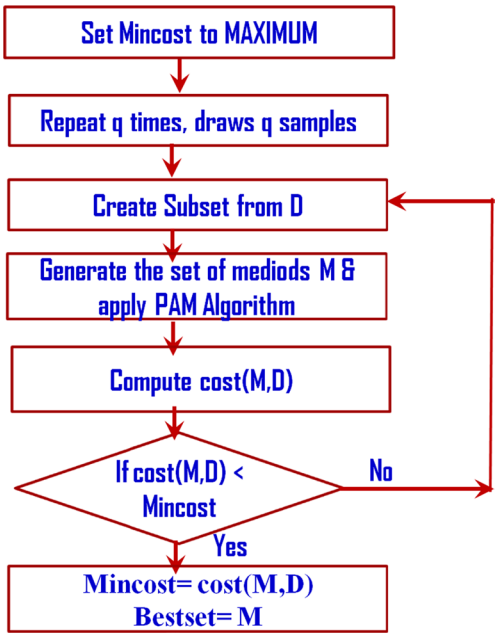


Fig. 3 Steps involved in CLARA algorithm

where β_0 is the y-intercept and $\beta_1, \beta_2, \ldots, \beta_k$ parameters are called the partial coefficients. In matrix form

$$\mathrm{Y} = \mathrm{XB} + \mathrm{E}$$

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ . \\ . \\ . \\ y_n \end{bmatrix} X = \begin{bmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1k} \\ 1 & x_{21} & x_{22} & \ldots & x_{2k} \\ . & . & . & \ldots & . \\ . & . & . & \ldots & . \\ . & . & . & \ldots & . \\ 1 & x_{n1} & x_{n2} & \ldots & x_{nk} \end{bmatrix} B = \begin{bmatrix} \beta_0 \\ \beta_1 \\ . \\ . \\ . \\ \beta_k \end{bmatrix} E = \begin{bmatrix} \varepsilon_0 \\ \varepsilon_1 \\ . \\ . \\ . \\ \varepsilon_n \end{bmatrix}$$

Before applying the multiple linear regression to forecast the crop yield, it's necessary to know the significant attributes from the database. All the attributes used in the database will not be significant or changing the value of these attributes will not affect anything on the dependent variables. Such attributes can be neglected. P value test is performed on the database to find the significant attributes and multiple linear regression is applied only on the significant values to forecast the crop yield.

Evaluation methods

Data mining algorithms work with different principles, being able to be influenced by different kinds of associations on data. To ensure fairer conditions in evaluation, this work finds the optimal clustering method for agriculture data analysis. Proposed work adopts the external quality metrics [3] like Purity, Homogeneity, Completeness, V Measure, Rand Index, Precision, Recall and F measure to compare the PAM, CLARA and DBSCAN clustering methods.

```
.....................................
Total number of clusters : 7
.....................................
cluster point 1 is : BIJAPUR
.....................................
cluster point 2 is : DAKSHINAKANNADA
.....................................
cluster point 3 is : KOPPAL
.....................................
cluster point 4 is : SHIMOGA
.....................................
cluster point 5 is : UTTARAKANNADA
.....................................
cluster point 6 is : BAGALKOT
.....................................
cluster point 7 is : CHIKMAGALUR
```

Fig. 4 Cluster centres obtained from the Batchelor Wilkins algorithm

Purity of the clustering is computed by assigning each cluster to the class which is most frequent in the cluster. Homogeneity represents the each cluster contains only members of a single class. Completeness represents the all members of a given class are assigned to the same cluster. V-measure is computed as the harmonic mean of distinct homogeneity and completeness scores. Rand Index measures the percentage of decisions that are correct. Precision is calculated as the fraction of pairs correctly put in the same cluster. Recall represents the fraction of actual pairs that were identified. F measure indicates the harmonic mean of precision and recall. Higher quality metrics value represents the better cluster quality.

Experimental results

Modified approach of DBSCAN

Before applying DBSCAN algorithm on the dataset user needs to determine the Minpts and Eps values. The Batchelor Wilkins algorithm is applied on the dataset in order to determine the K value (Number of clusters) automatically. For the dataset used in the proposed work, K value obtained from the Batchelor Wilkins is 7 with following districts as cluster centres. Results of Batchelor Wilkins algorithm are shown in Fig. 4.

KNN plot is plotted using K value obtained from the Batchelor & Wilkins' Algorithm to determine the epsilon value and the min points for the DBSCAN.

Figure 5 depicts the result of KNN plot. The KNN plot is plotted using K value obtained from the Batchelor & Wilkins' Algorithm (i.e. here $K = 7$). Eps value is calculated by taking the slope of the line from any point and sought-after pair of points that have the greatest slope to locate the point. The slope of the line is located at the point of 0.4, a point which is the optimal value Eps [20].

DBSCAN clustering algorithm is applied on the dataset to cluster the different districts of Karnataka which are having similar rain fall, temperature and soil type using optimal Eps value.

Figure 6 depicts the different districts of Karnataka which are considered for the purpose of analysis.

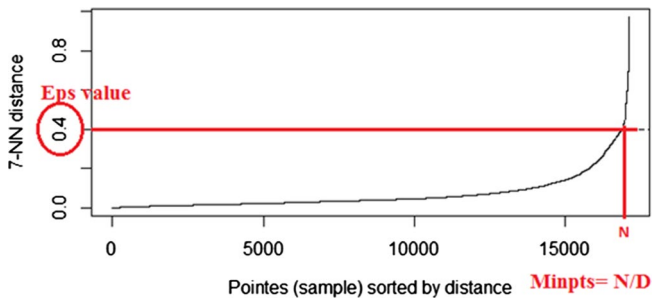


Fig. 5 KNN plot for given dataset

DISTRICT	Symbol	District	Symbol
BAGALKOT		GULBARGA	
BANGALORE (RURAL)		HASSAN	
BANGALORE (URBAN)		HAVERI	
BELGAUM		KODAGU(COORG)	
BELLARY		KOLAR	
BIDAR		KOPPAL	
BIJAPUR		MANDYA	
CHAMARAJANNAGAR		MYSORE	
CHIKMAGALUR		RAICHUR	
CHITRADURGA		SHIMOGA	
DAKSHINAKANNADA		TUMKUR	
DAVANGERE		UDUPI	
DHARWAD		UTTARAKANNADA	
GADAG			

Fig. 6 Districts of Karnataka considered for the analysis

Figures 7, 8 and 9 depicts the different districts of Karnataka which are having similar temperature range, rain fall range and soil types respectively.

PAM

To apply the PAM algorithm on the dataset, initially user need to give k (Number of clusters), where k is given as 3 in current experiment. Crop yield is categorised into LOW, MODERATE and HIGH production. Total districts are clustered into 3 clusters using PAM clustering method. Resultant clusters are shown in the Table 1.

- Study and analysis of wheat crop production in different districts of Karnataka as shown in Fig. 10.

As a result of the analysis, North Karnataka districts such as Bijapur, Dharwad, Bagalkot, Belgaum, Raichur, Bellary, Chitradurga and Davangere are the districts which have maximum wheat crop production.

CLARA

Districts in the dataset are clustered into 3 clusters using CLARA algorithm. Clusters are shown in the Fig. 11. It represents the districts which are having similar factors like

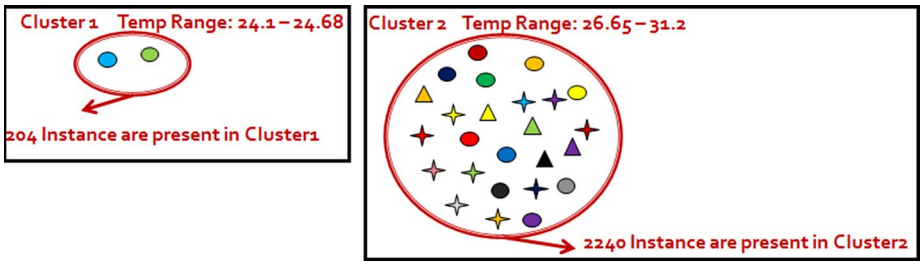


Fig. 7 Districts which are having similar temperature range

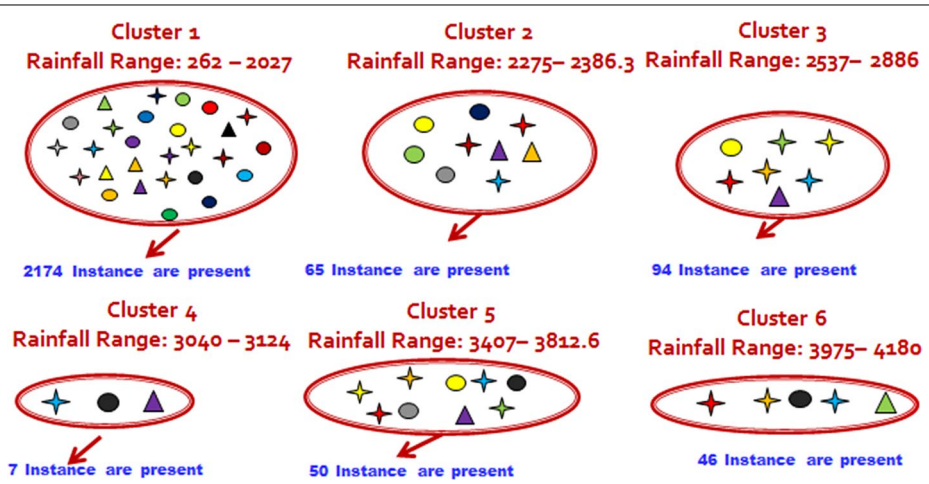


Fig. 8 Districts which are having similar rain fall during 6 year duration

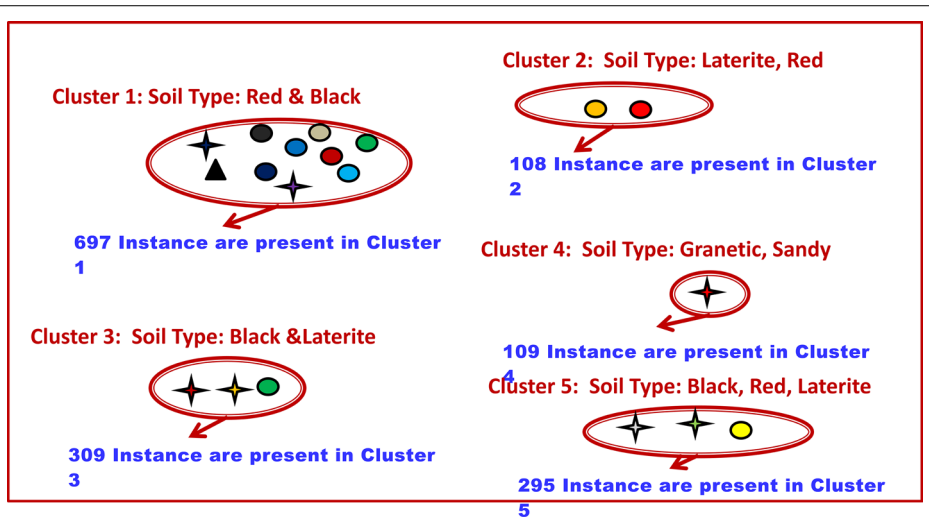


Fig. 9 Districts which are having similar soil type

Table 1 Results of PAM algorithm

Low-moderate production	High production	Moderate-high production
Mandya, Raichur, Gadag, Gulbarga, Bellary	Koppal, Dharwad, Haveri, Bijapur, Bidar, Chamarajannagar, Belgaum, Tumkur	Davangere, Shimoga, Chikmagalur, Bangalore

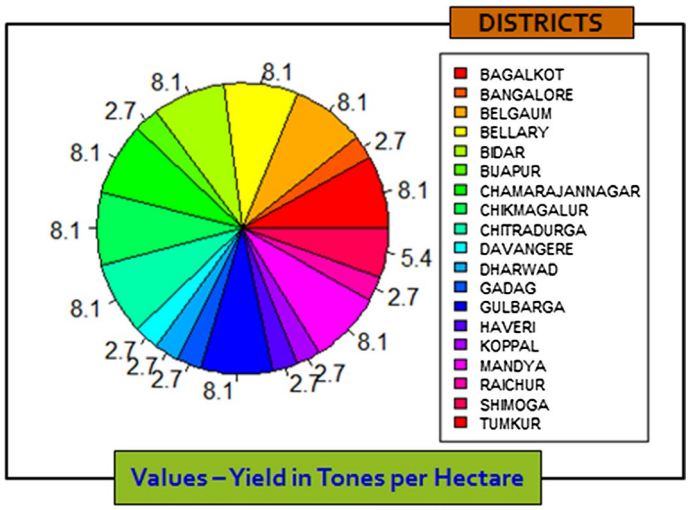


Fig. 10 Production of yield in tonnes per hectare of different districts

area, production, rainfall and temperature. Result of the CLARA algorithm is shown in the Table 2.

- Study and analysis of temperature and wheat crop production in different districts of Karnataka as shown in Fig. 12. From the Fig. 12, we can analyze that the optimal temperature for Wheat crop production is 29.9 °C.

Fig. 11 Results of CLARA in R language

Table 2 Results CLARA algorithm

Large area, production and moderate rainfall, temperature (24–26)	Moderate area, production and high rainfall, temperature (27–29)	Low area, production moderate rainfall, temperature (29–30)
Bijapur, Belgaum	Gadag, Gulbarga, Dharwad, Bangalore, Bagalkote	Koppal, Davangere, Shimoga, Haveri, Chikmagalur, Bidar, Chamarajannagar, Tumkur, Mandya, Raichur, Bellary

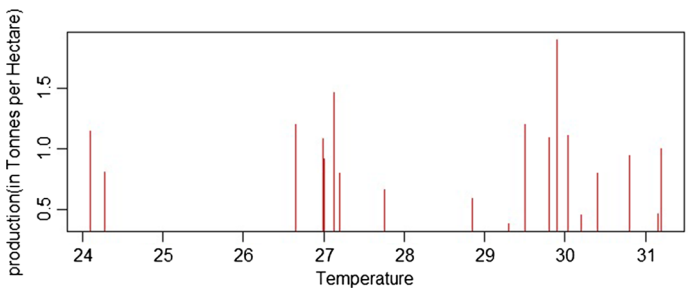


Fig. 12 Plot temperature vs. production

Multiple linear regression

Before applying the multiple linear regression, the "p value test" is performed on the dataset to determine the significant attributes. Table 3 depicts the significant values. An independent variable which has a "p value" of less than 0.05, specifies that the "null-hypothesis" can be rejected means it will have effect on regression analysis. So these independent values can be added to the model. Whereas if the p value is more than common alpha level i.e. 0.05, the variable will said to be not significant to the model.

Table 4 shows the multiple linear regression equation for different crop yield. For example, for Wheat crop, if all the independent variables are zero, the yield becomes 112. 1 unit increase in temperature level reduces the yield by 4.14e−02 units, 1 unit increase in rainfall will increase yield by 1.34e−04 units, 1 unit increase in pH will increase the yield by 0.079153 units, 1 unit increase in Nitrogen reduces the yield by 1.31e−03 units, 1 unit increase in potassium level decreases the yield by 0.00167 units and 1 unit increase in water requirement decreases the yield by 0.28125 unit.

Table 3 P value test: significant attributes

	Cotton	Groundnut	Jowar	Rice	Wheat
Temperature	*0.547536*	3.41E−07	3.86E−07	0.003139	0.001137
Rainfall	*0.784625*	1.86E−06	*0.653187*	*0.105878*	0.018042
Ph	0.011752	2.55E−05	0.029733	5.08E−07	0.01834
Nitrogen	5.85E−05	*0.071873*	*0.349257*	0.000841	8.6E−06
Phosphorus	*0.071843*	0.043345	*0.464847*	0.025816	*0.209524*
Potassium	2.82E−07	*0.643528*	0.050831	1.43E−05	0.021422
Water	4.95E−05	4.92E−49	1.2E−102	1.22E−26	NA

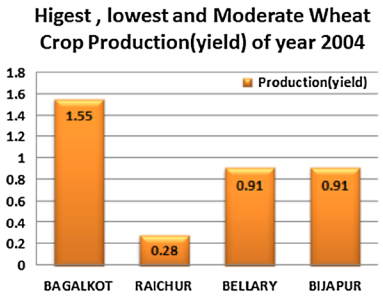
The italicized cells are representing the insignificant independent attributes for each crop as the values are more than 0.05. Regression Equation is formed using the independent variables

Table 4 Multiple linear regression equation for different crop yield

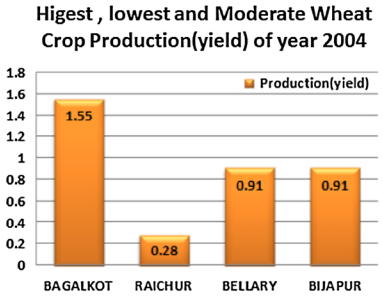
Crop	Yield forecast equation
Cotton	Yield = (7.149372) + (−0.14468)pH + (−0.00131) Nitrogen + (−0.00405) Potassium + (−0.00405) Water Required
Groundnut	Yield = (2.79115) + (0.029217) Temperature + (5.78e−05) Rainfall + (−0.05681) pH + (−0.00127) Phosphorus + (−0.00492) Water Required
Jowar	Yield = (−1.62694) + (−5.35e−02) Temperature + (0.051512) pH + (−0.00113) Potassium + (0.01685436) Water Required
Rice	Yield = (−0.18503) + (0.041593) Temperature + (0.172042) pH + (−8.27e−04) Nitrogen + (−4.28e−03) Phosphorus + (−0.00264) Potassium + (9.15e−04) Water Required
Wheat	Yield = (112) + (−4.14e−02) Temperature + (1.34e−04) Rainfall + (0.079153) pH + (−1.31e−03) Nitrogen + (−0.00167) Potassium + (−0.28125) Water Required

For 1 unit increase in pH, the crops like Jowar, Rice, and Wheat yield will increase but Groundnut and Cotton yield will decrease.

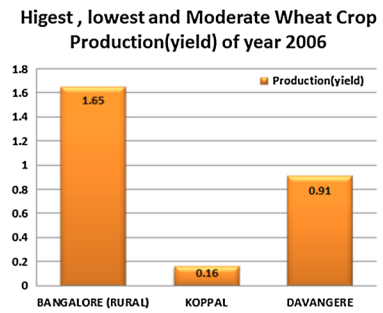
Results for optimal temperature and rainfall for wheat—Table 5



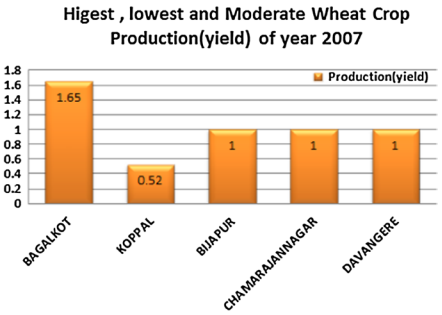
I .Yield plot-2004



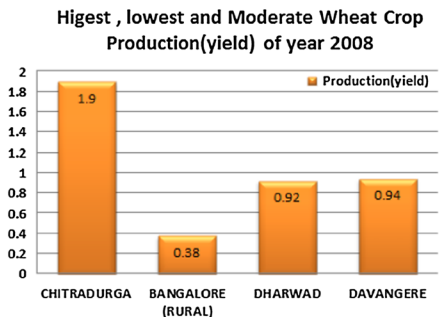
II. Yield plot-2005



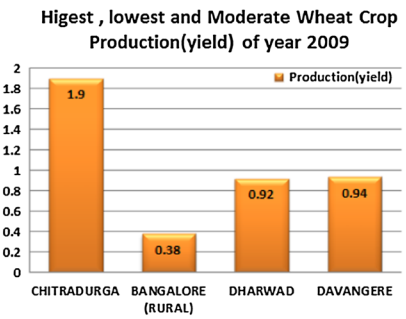
III. Yield plot-2006



IV. Yield plot-2007



V. Yield plot-2008



VI. Yield plot-2009

Table 5 shows the optimal parameters to achieve the higher wheat production.

Table 5 Optimal parameters to achieve higher production

Optimal parameters to achieve higher production	
Optimal temp	25.4–29.9
Worst temp	30.2–31.15
Rainfall	548–580

Comparison of clustering methods

As mentioned earlier, clustering comparison has done using four performance quality metrics. Table 6 shows the comparison of PAM, CLARA and DBSCAN methods for clustering the districts which are having similar crop productivity.

Table 6 and Fig. 13 depicts the comparison of PAM, CLARA and DBSCAN clustering methods. Higher quality metric values indicates better clustering quality. Analysis of the quality metrics parameters for different clustering methods is shown in the Fig. 13. From Fig. 13, DBSCAN has higher value for most of the quality metrics parameter. DBSCAN gives the better clustering quality than PAM and CLARA, CLARA gives the better clustering quality than the PAM.

Discussion

The crops are usually selected by its economic importance. However, the agricultural planning process requires a yield estimation of several crops. In this sense, five crops were selected for this work using the data availability as the key measure. Thus, a crop

Table 6 Comparison of clustering methods

Number of clusters k = 3			
	PAM	**CLARA**	**DBSCAN**
Purity	0.578947	0.631578	0.708512
Homogeneity	0.853526	0.879624	0.895275
Completeness	0.758264	0.782356	0.786854
V-measure	0.814447	0.805181	0.83757
Precision	0.40369	0.415365	0.42152
Recall	0.24856	0.25634	0.25655
F-measure	0.307677	0.317028	0.318966
Rand index	0.785364	0.796352	0.814561

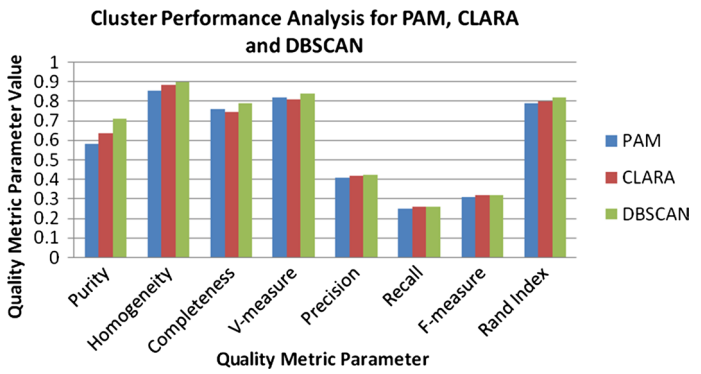


Fig. 13 Comparison of clustering methods

was selected when enough data samples appeared in the range of 6 years under analysis. In presents works, research is commonly limited to the 5 crops those are cotton, wheat, ground nut, jowar and rice. Example wheat crop analysis is discussed in this paper.

The present work covers the PAM, CLARA, Modified DBSCAN clustering methods and multiple linear regression method. PAM and CLARA are the traditional clustering methods where as DBSCAN method is modified by introducing the Batchelor Wilkins clustering method to determine the 'k' value and KNN method to determine the minimum points and radius value automatically. Using these methods crop data set is analysed and determined the optimal parameters for the wheat crop production. Multiple linear regression is used to find the significant attributes and form the equation for the yield prediction.

Some works measure the quality of the clustering methods using internal quality metrics [21], some other uses the external quality metrics. However, in these works, research is limited to the external quality metrics which are combination of several metrics those are [22]: set matching metrics, metrics based on counting pairs and metrics based on Entropy. The quality metrics were ranked, from the best to the worst, according to purity, homogeneity, completeness, v measure, precision, recall and rand index results, in the following order: DBSCAN, CLARA and PAM.

Conclusion

Various data mining techniques are implemented on the input data to assess the best performance yielding method. The present work used data mining techniques PAM, CLARA and DBSCAN to obtain the optimal climate requirement of wheat like optimal range of best temperature, worst temperature and rain fall to achieve higher production of wheat crop. Clustering methods are compared using quality metrics. According to the analyses of clustering quality metrics, DBSCAN gives the better clustering quality than PAM and CLARA, CLARA gives the better clustering quality than the PAM. The proposed work can also be extended to analyse the soil and other factors for the crop and to increase the crop production under the different climatic conditions.

Authors' contributions

JM, Dean R&D, Prof & HOD of Dept of M.Tech CSE at NMIT, has 40 years of experience in India and abroad has guided and given extensive help to develop the data mining algorithms. SN, Assistant Professor of Dept of M.Tech CSE at NMIT has developed the PAM and CLARA algorithms with the help of Dr. Jharna Majumdar. SA Assistant Professor of Dept of M.Tech CSE at NMIT has developed Modified approach of DBSCAN, Multiple Linear Regression and quality metrics for cluster comparison with the guidance and help of Dr. Jharna Majumdar. All authors together analysed the crop data set to determine the optimal parameters to maximise the crop yield. All authors read and approved the final manuscript.

Acknowledgements

The authors express their sincere gratitude to Prof N.R Shetty, Advisor and Dr H.C Nagaraj, Principal, Nitte Meenakshi Institute of Technology for giving constant encouragement and support to carry out research at NMIT.

The authors extend their thanks to Vision Group on Science and Technology (VGST), Government of Karnataka to acknowledge our research and providing financial support to setup the infrastructure required to carry out the research.

Competing interests

The authors declare that they have no competing interests.

Funding

This work was supported by the Research Department of Computer science, Nitte Meenakshi Institute of Technology.

Publisher's Note

Received: 25 February 2017 Accepted: 31 May 2017
Published online: 05 July 2017

References

1. Veenadhari S, Misra B, Singh CD. Data mining techniques for predicting crop productivity—A review article. In: IJCST. 2011; 2(1).
2. Gleaso CP. Large area yield estimation/forecasting using plant process models.paper presentation at the winter meeting American society of agricultural engineers palmer house, Chicago, Illinois. 1982; 14–17
3. Majumdar J, Ankalaki S. Comparison of clustering algorithms using quality metrics with invariant features extracted from plant leaves. In: Paper presented at international conference on computational science and engineering. 2016.
4. Jain A, Murty MN, Flynn PJ. Data clustering: a review. ACM Comput Surv. 1999;31(3):264–323.
5. Jain AK, Dubes RC. Algorithms for clustering data. New Jersey: Prentice Hall; 1988.
6. Berkhin P. A survey of clustering data mining technique. In: Kogan J, Nicholas C, Teboulle M, editors. Grouping multi-dimensional data. Berlin: Springer; 2006. p. 25–72.
7. Han J, Kamber M. Data mining: concepts and techniques. Massachusetts: Morgan Kaufmann Publishers; 2001.
8. Ester M, Kriegel HP, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Paper presented at International conference on knowledge discovery and data mining. 1996
9. Ramesh D, Vishnu Vardhan B. Data mining techniques and applications to agricultural yield data. In: International journal of advanced research in computer and communication engineering. 2013; 2(9).
10. MotiurRahman M, Haq N, Rahman RM. Application of data mining tools for rice yield prediction on clustered regions of Bangladesh. IEEE. 2014;2014:8–13.
11. Verheyen K, Adrianens M, Hermy S Deckers. High resolution continuous soil classification using morphological soil profile descriptions. Geoderma. 2001;101:31–48.
12. Gonzalez-Sanchez Alberto, Frausto-Solis Juan, Ojeda-Bustamante W. Predictive ability of machine learning methods for massive crop yield prediction. Span J Agric Res. 2014;12(2):313–28.
13. Pantazi XE, Moshou D, Alexandridis T, Mouazen AM. Wheat yield prediction using machine learning and advanced sensing techniques. Comput Electron Agric. 2016;121:57–65.
14. Veenadhari S, Misra B, Singh D. Machine learning approach for forecasting crop yield based on climatic parameters. In: Paper presented at international conference on computer communication and informatics (ICCCI-2014), Coimbatore. 2014.
15. Rahmah N, Sitanggang IS. Determination of optimal epsilon (Eps) value on DBSCAN algorithm to clustering data on peatland hotspots in Sumatra. IOP conference series: earth and environmental. Science. 2016;31:012012.
16. Forbes G. The automatic detection of patterns in people's movements. Dissertation, University of Cape Town. 2002.
17. Ng RT, Han J. CLARANS: A Method for Clustering Objects for Spatial Data Mining. In: IEEE Transactions on Knowledge and Data Engineering. 2002; 14(5).
18. Kaufman L, Rousseeuw PJ. Finding groups in data: an introduction to cluster analysis. Wiley. 1990. doi:10.1002/9780470316801.
19. Multiple linear regression-http://www.originlab.com/doc/Origin-Help/Multi-Regression-Algorithm. Accessed 3 July 2017.
20. Elbatta MNT. An improvement for DBSCAN algorithm for best results in varied densities. Dissertation, Gaza (PS): Islamic University of Gaza. 2012
21. Kirkl O, De La Iglesia B. Experimental evaluation of cluster quality measures. 2013. 978-1-4799-1568-2/13. IEEE.
22. Meila M (2003) Comparing clustering. In: Proceedings of COLT 2003.

iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** VI **Month of publication:** June 2021

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

A Potential Solution for Crop Yield Prediction by using Data Mining Techniques

Mrs. S. S. N. L. Priyanka[1], G. Sarmistanjali[2], M. Sudeepthi Swathi[3], K. Phanindra Kumar[4], S. Gowtham Kumar[5]
[1]*Assistant Professor,* [2, 3, 4, 5]*UG Student, Computer Science Engineering, Anil Neerukonda Institute of Technology and Sciences, Affiliated to Andhra University, Accredited by NBA and NAAC, India*

Abstract: ***Agriculture is undoubtedly the largest livelihood provider in India and also contributes a significant figure to the economy of our Country. The technological factors affecting the crop production includes practices used and also managerial decisions. So, predicting the crop yield prior to its harvest would help farmers to take appropriate steps. We attempt to resolve the issue by building a user-friendly prediction system. The results of the prediction are suggested to the farmer such that suitable changes can be made in order to improve the produce. There are different techniques or algorithms which help to predict crop yield. By analyzing all the parameters like location, soil nutrients, pH value, rainfall, moisture a potential solution can be obtained to overcome the situation faced by farmers. This paper focuses on the analysis of the agriculture data and finding optimal yield to provide an insight before the actual crop production using data mining techniques and Machine Learning algorithms.***

Keywords: Yield, Random forest regress or, Decision Tree regress or, GDP, Digitalisation.

I. INTRODUCTION

Today, India is one of the leading producers across the world in the agriculture sector[1]. Agriculture is the broadest economic sector and plays an outstanding role in the socio-economic part of India. Agriculture is an eccentric business crop production which is influenced by many climate and economic factors. Andhra Pradesh, basically being an agro-Based economy contributes more than 29% of the GDP as against 17% in the country's GDP. Periodical advice to the farmers either in terms of improved agricultural strategies or advancements in factors affecting the production of crops may strengthen the state in the agriculture sector. Yield prediction is one among the agricultural advancements. Due to these kinds of innovations agriculture is driving the interest of modern man. In the past farmers used to predict their yield from previous experiences[2]. Digitalisation in farming gives awareness about the cultivation of the crops at the right time and at the right place even to young farmers. These kinds of advancements need the use of data analytics. This is one such system that can be used to address yield prediction. The main objectives are:

1) To analyse different parameters (soil nutrients, rainfall, area etc)
2) To use machine learning techniques to predict crop yield.
3) To provide an easy to use User Interface.

II. HOW DATA MINING IS USED IN AGRICULTURE SECTOR

Data mining techniques are used in performing several activities in the agricultural sector such as pest identification, detection and classification and prediction of crop diseases. It can also be used in yield prediction, input management (planning of irrigation and pesticides), fertilizer suggestion and predicting soil. In a world full of data , data mining is the computational process for discovering new patterns[3]. Data mining techniques provide a major advantage in agriculture for detection and prediction for optimizing the pesticides. Techniques for agriculture related activities provide a lot of information. The yield of agriculture primarily depends on diseases, pests, weather conditions, planning of various crops for the harvest productivity are the results.

Crop production for reliable and timely requirements for various decisions for agriculture marketing. Predictions are very useful for agriculture data. For instance, by applying data mining techniques, the government can fully benefit from data about farmers' buying patterns and also to achieve a superior understanding of their land to achieve more profit on the farmer's part.

Data mining techniques followed in two ways[4]:

1) Descriptive data mining.
2) Predictive data mining.

Descriptive data mining tasks characterize the final properties of the info within the database while predictive data mining is employed to predict the direct values supported patterns determined from known results. Prediction involves using some variables or fields within the database to predict unknown or future values of other variables of interest.

As far as data mining techniques are concerned, in most cases predictive data mining approaches are employed. Predictive data mining techniques are employed to predict future crop, forecasting, pesticides and fertilizers to be used, revenue to be generated and so on. These techniques are used for pre-harvest forecasting for the agriculture field and are able to provide a lot of data on agricultural-related activities. Data of agriculture in data mining can be presented in the form of datasets.

III. PROPOSED SYSTEM

The main objectives of proposed work is to analyse the agricultural parameters using data mining algorithms and predict the yield.In our proposed work, agriculture data has been collected from various sources which include:
Dataset in agricultural sector[5] ,Crop wise agriculture data:[6], Soil data of different districts:[7]
In this proposed system , we mainly focussed on Andhra Pradesh State in India. As the state has two major rivers flowing , it has a diversity in factors useful for agriculture at district level. Periodical data about the crop , soil and water a particular region is the major focus of this study.The final dataset has been tabulated as in table-1:

Sno	Feature	Description
1	Year	The year in which the crop will be cultivated. Generally the upcoming year
2	Season	One among Kharif,Rabi and Whole Year.
3	Crop	Name of the crop
4	District	Name of the district
5	pH Level	This describes the nature of the soil
6	Nitrogen	Amount of nitrogen present
7	Potassium	Amount of potassium present
8	Phosphorus	Amount of phosphorus present
9	Rainfall	Expected rainfall in millimeters
10	Area	Area of field in hectares

Table-1: Description of Input data

The below diagram depicts the system architecture of our proposed system. Our whole system can be divided into 2 modules as a whole i.e., one model predicts the optimal yield and the other model analyses the patterns in the dataset. The operation of these models as a whole is specified clearly in the below diagram.

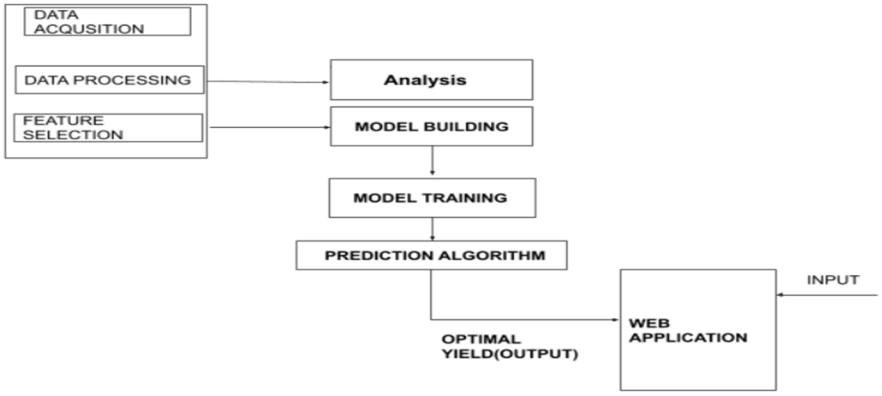


Fig.1 The blueprint of the proposed system

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

IV. METHODS

In the implementation of this yield prediction system Regression Analysis is used.Regression Analysis is considered as one of the oldest,and widely used multivariate analysis techniques in the social sciences. Unlike others regression stands as an example of dependence analysis in which the variables are treated asymmetrically. In regression analysis, the object is to obtain a prediction of one variable, based on given the values of the others[8].Random Forest and Decision Tree algorithms are generally used in classification problems but these can also be used in regression problems as well.

A. Decision Tree Regression

The Decision Tree algorithm comes under supervised machine learning techniques. A decision tree arrives at an outcome by asking a series of questions to the input data , each question narrows down the possible outcomes until the model gets enough potential to make a unique prediction[9]. The order of the questions as well as their contents are being determined by the model.All the questions that are raised have their answer as either true or false.

B. Random Forest Regression

Random Forest algorithm comes under the family of ensemble algorithms. This is also a supervised learning algorithm. This can be implemented in classification and regression as well. Random forest algorithm basically works on Decision Tree principle by constructing a number of decision trees having different sets of hyper-parameters for tuning and training on different subsets of data[10].

V. EXPERIMENTAL RESULT

A. Decision Tree Regression

Decision Tree algorithm on applying on the dataset resulted 100% on data and 82%(approx.) on test data.Fig-2 shows the accuracy of Decision tree algorithm on data:

DecisionTree Regression:

```
[ ]   from sklearn.tree import DecisionTreeRegressor
      dt_obj =DecisionTreeRegressor(random_state=1)
      dt_obj.fit(X_train,y_train)
      print('Train Score DT:',dt_obj.score(X_train,y_train))
      print('Test Score DT:',dt_obj.score(X_test,y_test))
```

```
Train Score DT: 1.0
Test Score DT: 0.8162150580308982
```

Fig-2:Result of Decision Tree algorithm

B. Random Forest Regression

Random Forest algorithm on applying on the dataset resulted 98% on data and 90%(approx.) on test data.Fig-3 shows the accuracy of Decision tree algorithm on data:

RandomForest Regression:

```
[ ]   from sklearn.ensemble import RandomForestRegressor
      rf_obj =RandomForestRegressor(n_estimators = 10,random_state=0)
      rf_obj.fit(X_train,y_train)
      print('Train score RF:',rf_obj.score(X_train,y_train))
      print('Test score RF:',rf_obj.score(X_test,y_test))
```

```
Train score RF: 0.9859482658364138
Test score RF: 0.8994486718857367
```

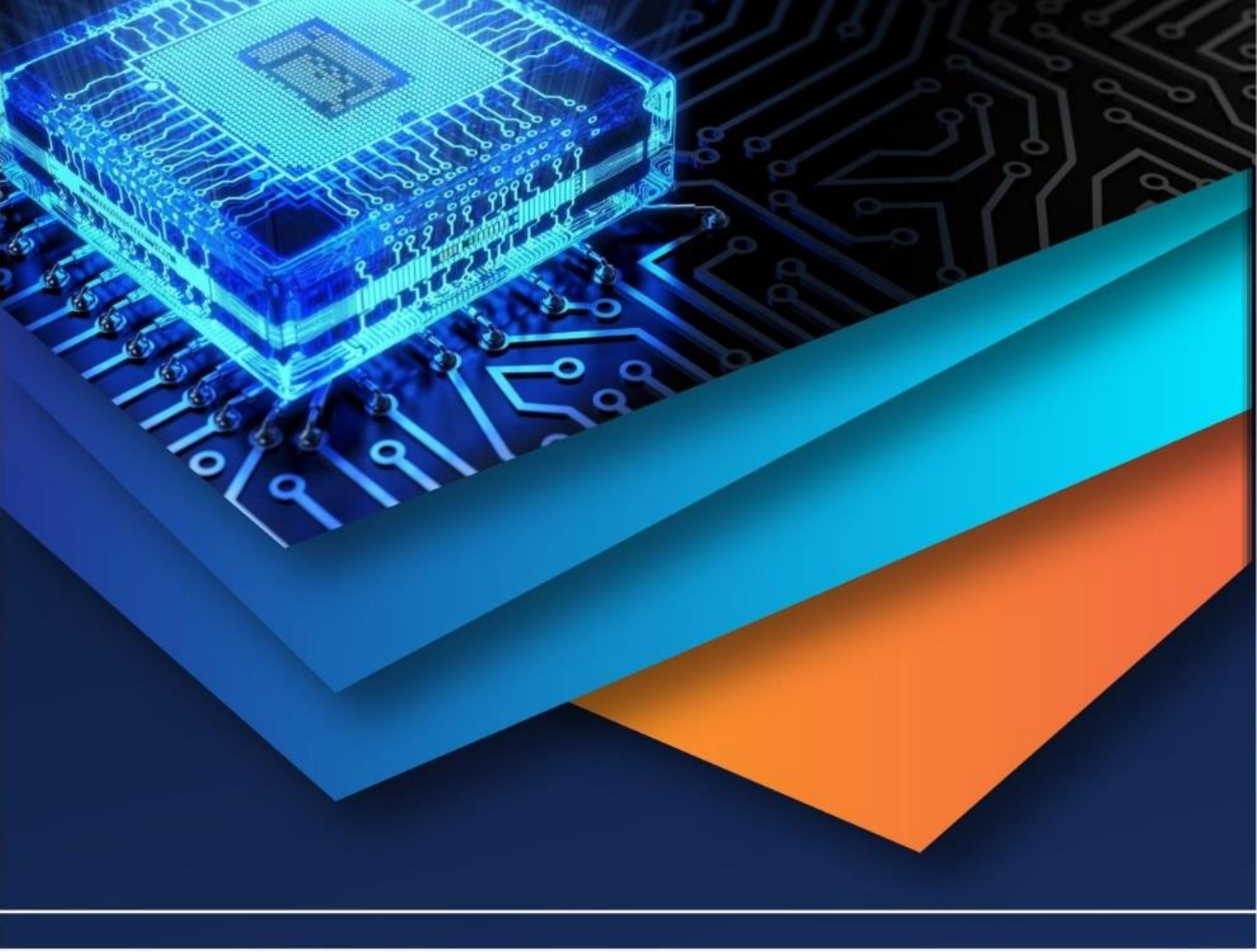
Fig-3:Result of Random Forest algorithm

VI. CONCLUSION

Both Decision tree regression and Random Forest regression techniques are implemented on the input data to assess the best performance yielding method. These methods are compared using performance metrics. According to the analyses of metrics both the algorithms work well , but Random Forest regression gives a better accuracy score on test data than Decision tree regression. The proposed work can also be extended to analyse the climatic conditions and other factors for the crop and to increase the crop production.

REFERENCES

[1] https://www.investopedia.com/articles/investing/100615/4-countries-produce-most-food.asp
[2] http://www.fao.org/fileadmin/templates/ess/documents/meetings_and_workshops/GS_SAC_2013/Improving_methods_for_crops_estimates/Crop_Yield_Forecasting_Methods_and_Early_Warning_Systems_Lit_review.pdf
[3] https://ijaers.com/uploads/issue_files/3%20IJAERS-MAY-2017-60-Different%20Types%20of%20Data%20Mining%20Techniques.pdf
[4] https://docs.oracle.com/cd/B12037_01/datamine.101/b10698/4descrip.htm
[5] https://data.gov.in/
[6] http://www.apagrisnet.gov.in/2018/weekly/October/weekly_report_(Rabi)_06_21-11-18.pdf
[7] http://dataverse.icrisat.org/file.xhtml?fileId=1185&version=RELEASED version=.3
[8] https://www.sciencedirect.com/topics/medicine-and-dentistry/regression-analysis
[9] https://gdcoder.com/decision-tree-regressor-explained-in-depth/
[10] https://www.analyticsvidhya.com/blog/2020/12/lets-open-the-black-box-of-random-forests/
[11] https://journalofbigdata.springeropen.com/articles/10.1186/s40537-017-0077-4



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY