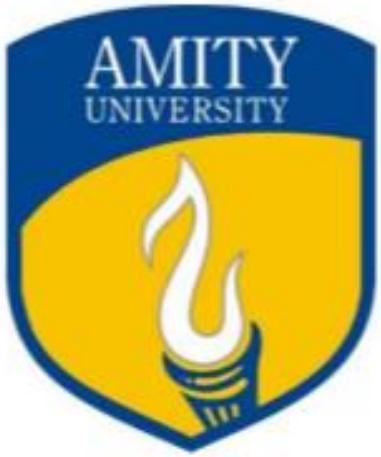


AMITY UNIVERSITY

UTTAR PRADESH

Project Report on

Data Analysis For Agriculture Production Engine

Submitted To

Dr. Rashmi Vashisth

AIIT, Amity University

Noida, U.P

Submitted By

Yatin Sengar

Enroll No.:- A1004820014

Batch:- BCA-6A(2020-23)

DECLARATION BY STUDENT

I –Yatin Sengar student of BCA hereby declare that the Major Project titled "Data Analysis For Agriculture Production Engine" which is submitted by me to Dr. Rashmi Vashisth, Amity Institute of Information Technology, Amity University Uttar Pradesh, Noida, in partial fulfillment of requirement for the award of the degree of BCA, has not been previously formed the basis for the award of any degree, diploma or other similar title or recognition.

The Author attests that permission has been obtained for the use of any copy righted material appearing in the Dissertation / Project report other than brief excerpts requiring only proper acknowledgement in scholarly writing and all such use is acknowledged.

Place: - Noida

Name: Yatin Sengar

Enrollment No: A1004820014

Semester: 6 Singnature

AMITY UNIVERSITY

UTTAR PRADESH

GUIDE CERTIFICATE

I hereby certify that the Major Project by –Yatin Sengar, student of BCA 6th Semester Enrolment No -A1004820014 with title "Data Analysis for Agriculture Production Engine" which is submitted to Amity Institute of Information Technology, Amity University Uttar Pradesh, Noida in partial fulfillment of requirement for the award of the degree of BCA is an original contribution with existing knowledge and faithful record of work carried out by him/her under my guidance and supervision and to the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: - Noida

Date:-

Dr. Rashmi Vashisth

AIIT, Amity University

Noida, Uttarpradesh Signature

ACKNOWLEDGEMENT

It is high privilege for me to express my deep sense of gratitude to those entire faculty Members who helped me in the completion of the project, I would like to thank Prof (Dr)Rekha Aggrawal, Head of Department AIIT, and Amity University for giving me the opportunity to undertake this project.

I would like to thank my faculty guide Dr. Rashmi Vashisth who is the biggest driving force behind my successful completion of the project. who was always there at hour of need.

My special thanks to all other faculty members, Batch mate & Seniors of Amity Institute of Information Technology, Amity University, Uttar Pradesh for helping me in the completion of project work and its report submission

SYNOPSIS

AMITY UNIVERSITY

-----UTTAR PRADESH-----

Amity Institute of Information Technology

Major Project

Student Name	YATIN SENGAR
Enrollment No	A1004820014
Programme	Bachelor of Computer Applications
Company's Name and Address	AIIT, Amity university Noida Amity Institute of Information Technology (AIIT), I-1 Block, Amity University Uttar Pradesh, Sector 125, NOIDA, 201303 110096
Industry Guide	
Name	Dr. Rashmi Vashisth
Designation	Assistant Proffesor
Contact Number	
	Ph.(O) : 9958810676 (R) : 9958810676 **Mobile : 9958810676** **Fax : rvashisth@amity.edu** **E-mail : aiit@amity.edu**

Project Information

1) Project Duration : **(112 Days)**
   a) Date of Summer Internship commencement **(16/01/2023)**
   a) Date of Summer Internship Completion **(08/05/2023)**

2) Topic

Data analysis for agriculture production engine

3) Project Objective

Analysis the the data of the agricultural crops and find the optimizing growth and output of the crops in specific environment.

4) Methodology to be adopted

Methodology adopted are data analysis, data visualization & data science technology using python and using python libraries like numpy and pandas etc.

5) **Brief Summery of project(*to be duly certified by the industry guide*)**

Agriculture is vast & emerging sector in the world. But problem like crops wastage and atmosphere uncertainty is the major problem in this sector so solving this type of problem we create a data set from past agriculture reports and data and find the optimal solution for growing more crops and protect crops from uncertain atmosphere by taking a wise decision using data analysis, data science, data visualization technologies.

Signature (Student)	Signature (Industry Guide)	Signature (Faculty Guide)

ABSTRACT

Agriculture is vast and emerging sector in the world. But problems like crops wastage and atmosphere uncertainty is the major problem in this sector. So solving this type of problem we create a data set and problem set from past agriculture reports and now we find the optimal solution for optimizing the growth of crops and protect crops from uncertain atmosphere by taking wise decision.

Key Consideration:- Then there are seven critical criteria that I have considered that will assist us in choosing exactly which crop should be produced and when, namely the amount of nitrogen, phosphorus, and potassium in the soil, temperature in degrees Celsius, moisture, pH, and rainfall in millimetres.

1) INTRODUCTION

Problem Statement:- As we all know, husbandry is heavily reliant on the nature of the soil and the climatic conditions, and we frequently face changeable changes in climate such as non-seasonal downfall or heat swells or oscillations in moisture situations, and all similar events result in significant losses to our growers and husbandry, preventing them from fully utilising their agrarian land.

Solution:- So, in order to solve all similar problems, I created a Machine Learning Model through which we can assist growers in optimising agrarian products, because this predictive model will help them understand that for a specific soil and given climatic conditions, which crop will be stylish suitable for the crop.

Aim/ Constraint- This approach focuses a machine learning predictive model on recommending the best crops to grow based on the type of the soil and the given climatic circumstances.

2) OVERVIEW OF THE AGRICULTURE INDUSTRY

The agriculture industry involves the cultivation, harvesting, and processing of crops and livestock for food, fuel, fiber, and other products. It is a vital sector of the global economy, providing employment, income, and food security to billions of people around the world.

Agriculture has been practiced for thousands of years, with early civilizations developing systems for domesticating plants and animals to meet their basic needs. Over time, agriculture has become more sophisticated, incorporating new technologies and scientific advancements to increase yields, improve quality, and reduce environmental impact.

Today, the agriculture industry encompasses a wide range of activities, from small-scale subsistence farming to large-scale commercial operations. It includes the production of crops such as grains, vegetables, fruits, and nuts, as well as livestock such as cattle, pigs, and poultry. It also involves the processing and distribution of these products, including transportation, storage, and marketing.

The agriculture industry faces a number of challenges, including changing weather patterns, soil degradation, water scarcity, and a growing demand for food from a growing population. As a result, there is a need for continued innovation and investment in new technologies and sustainable practices to ensure the long-term viability of agriculture and food production.

3) ROLE OF DATA & ML IN AGRICULTURE PDRODUCTION ENGINE

Data and machine learning are increasingly being used in agriculture production to improve efficiency, yield, and sustainability. Here are some methods by which they are helping:

3.1. Precision Agriculture: Data and machine learning algorithms are used to analyze various factors such as soil moisture, nutrient levels, and weather patterns to create precise maps of farm fields. These maps can then be used to optimize irrigation, fertilization, and pesticide application, which can increase yield and reduce waste.

3.2. Crop monitoring and prediction: Machine learning algorithms can analyze satellite and drone imagery to monitor crop growth, identify problems, and predict yield. This can help farmers to take corrective action before crop losses occur and optimize their harvest planning.

3.3. Livestock monitoring: Data and machine learning can also be used to monitor the health and behavior of livestock, allowing farmers to detect potential health problems earlier and take preventive measures to avoid losses.

3.4. Machinery optimization: Machine learning can analyze data from farm machinery sensors to optimize usage and predict maintenance.

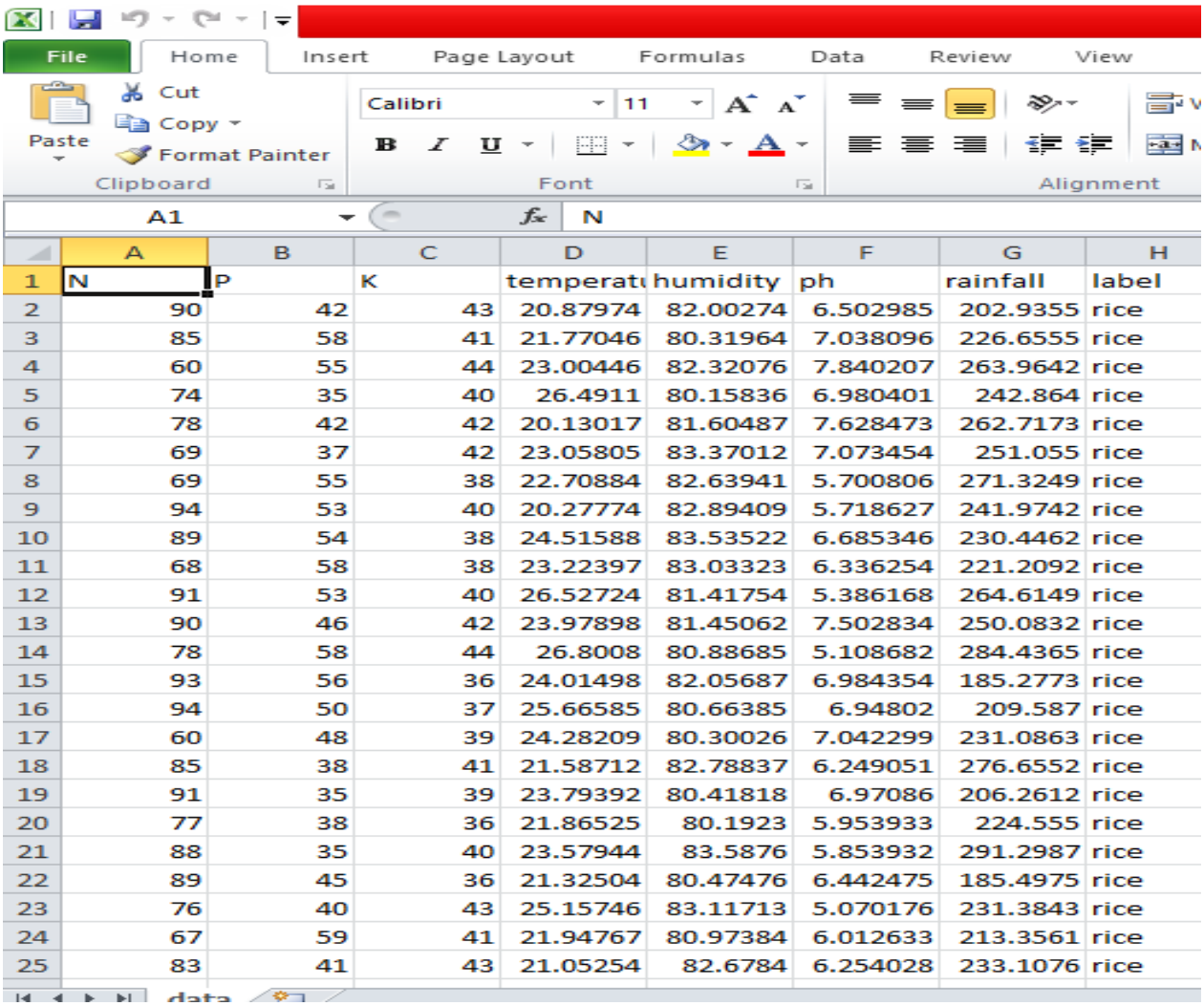
3.5. Market analysis: Machine learning can be used to analyze market trends and predict demand for different crops, helping.

4) PROBLEM SET AND DATA SET REALTED TO AGRICULTURE

I read the data sets and problem sets and reached the following conclusion:-

Then there are seven critical criteria that I have considered that will assist us in choosing exactly which crop should be produced and when, namely the amount of nitrogen, phosphorus, and potassium in the soil, temperature in degrees Celsius, moisture, pH, and rainfall in millimetres.

Glimpse of the dataset:-

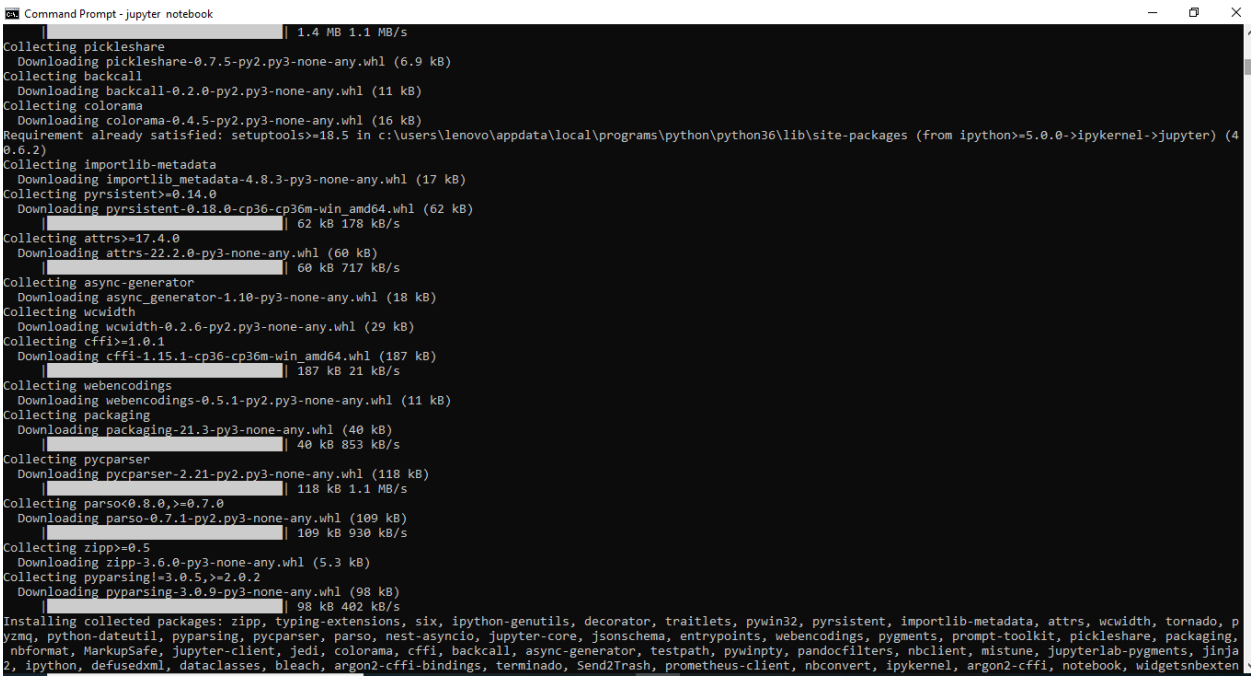


	A	B	C	D	E	F	G	H
1	N	P	K	temperatu	humidity	ph	rainfall	label
2	90	42	43	20.87974	82.00274	6.502985	202.9355	rice
3	85	58	41	21.77046	80.31964	7.038096	226.6555	rice
4	60	55	44	23.00446	82.32076	7.840207	263.9642	rice
5	74	35	40	26.4911	80.15836	6.980401	242.864	rice
6	78	42	42	20.13017	81.60487	7.628473	262.7173	rice
7	69	37	42	23.05805	83.37012	7.073454	251.055	rice
8	69	55	38	22.70884	82.63941	5.700806	271.3249	rice
9	94	53	40	20.27774	82.89409	5.718627	241.9742	rice
10	89	54	38	24.51588	83.53522	6.685346	230.4462	rice
11	68	58	38	23.22397	83.03323	6.336254	221.2092	rice
12	91	53	40	26.52724	81.41754	5.386168	264.6149	rice
13	90	46	42	23.97898	81.45062	7.502834	250.0832	rice
14	78	58	44	26.8008	80.88685	5.108682	284.4365	rice
15	93	56	36	24.01498	82.05687	6.984354	185.2773	rice
16	94	50	37	25.66585	80.66385	6.94802	209.587	rice
17	60	48	39	24.28209	80.30026	7.042299	231.0863	rice
18	85	38	41	21.58712	82.78837	6.249051	276.6552	rice
19	91	35	39	23.79392	80.41818	6.97086	206.2612	rice
20	77	38	36	21.86525	80.1923	5.953933	224.555	rice
21	88	35	40	23.57944	83.5876	5.853932	291.2987	rice
22	89	45	36	21.32504	80.47476	6.442475	185.4975	rice
23	76	40	43	25.15746	83.11713	5.070176	231.3843	rice
24	67	59	41	21.94767	80.97384	6.012633	213.3561	rice
25	83	41	43	21.05254	82.6784	6.254028	233.1076	rice

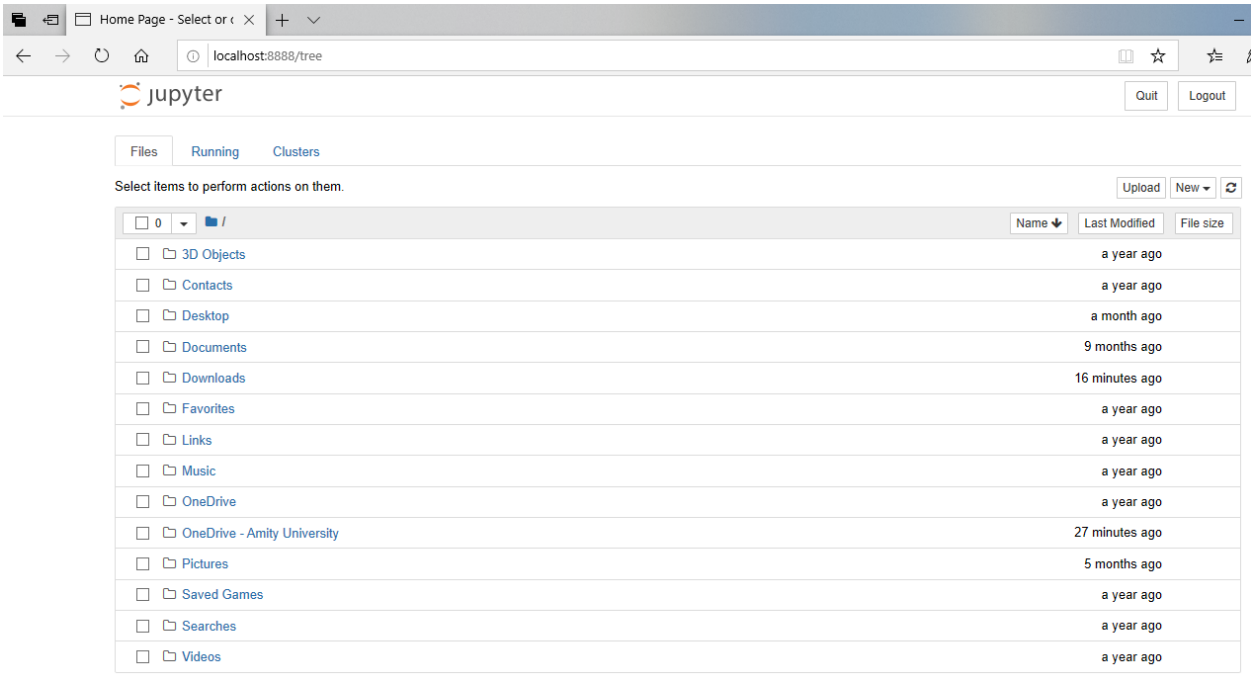
5) SYSTEM SETUP FOR DATA ANALYSIS & TRAIN OUR PREDECTIVE ML MODEL

5.1) Download python 3 version

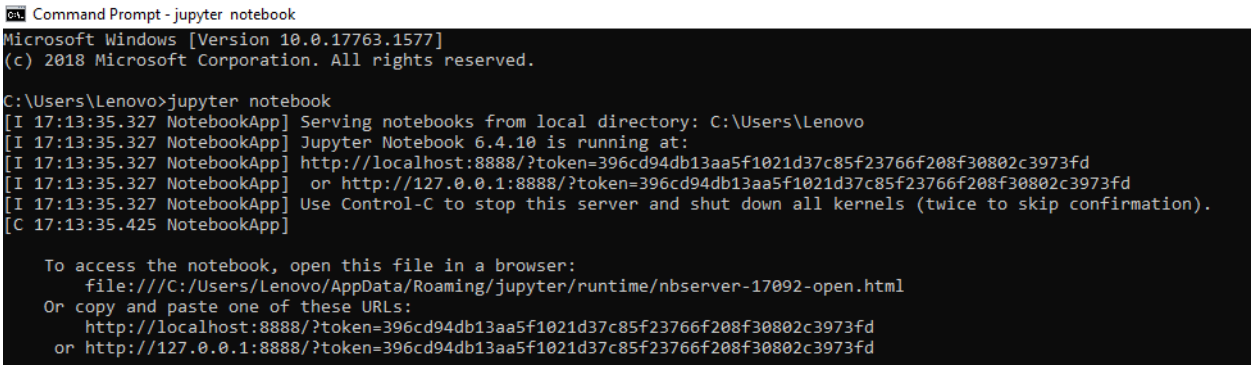
5.2) Download dependencies and import libraries through command prompt

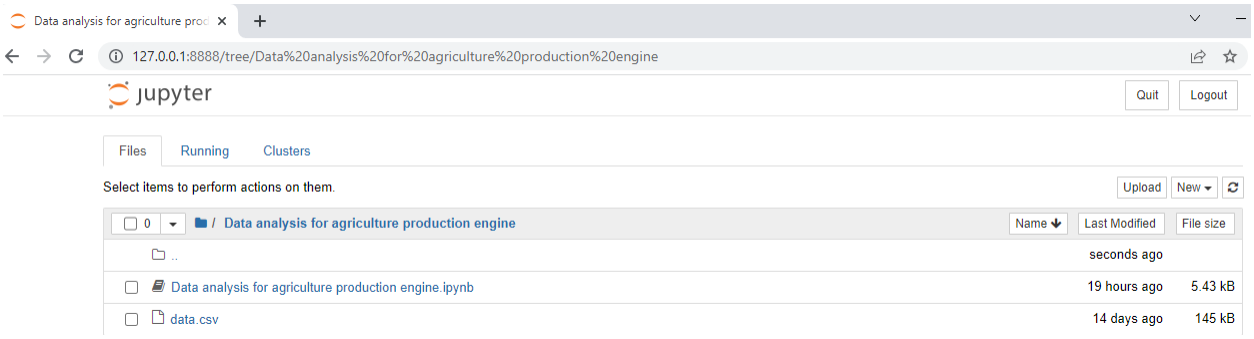


5.3) Open jupyter in server upload the dataset and creating a folder for data analysis and visualization.



5.4) Create proper channel for data pipeline.





6) TOOLS & TECHNOLOGY USED

6.1) Python & Jupyter Notebook:- Python is a high-level programming language used in a wide range of applications such as data analysis, machine learning, scientific computing, and web development.

6.2) Data Analysis:- Python is widely used for data analysis, and there are several popular libraries and tools available for this purpose.

6.3) Data Visualization:- Python is also commonly used for data visualisation, which is the process of constructing graphical representations of data to assist users in understanding patterns, trends, and correlations.

6.4) Data Science Technology:- Data Science technology refers to the tools and techniques used to process, analyze, and derive insights from large and complex datasets.

6.5) Machine Learning Algorithms used:- Clustering Analysis, Logistic Regression & Linear Regression of matrices

7) LIBRARIES USED

7.1) Numpy:- NumPy is a Python numerical computing package. It supports multidimensional arrays, mathematical operations, and tools for manipulating these arrays.

7.2) Pandas:- Is a Python data manipulation and analysis library. It has strong data structures for dealing with structured data, such as data frames and series.

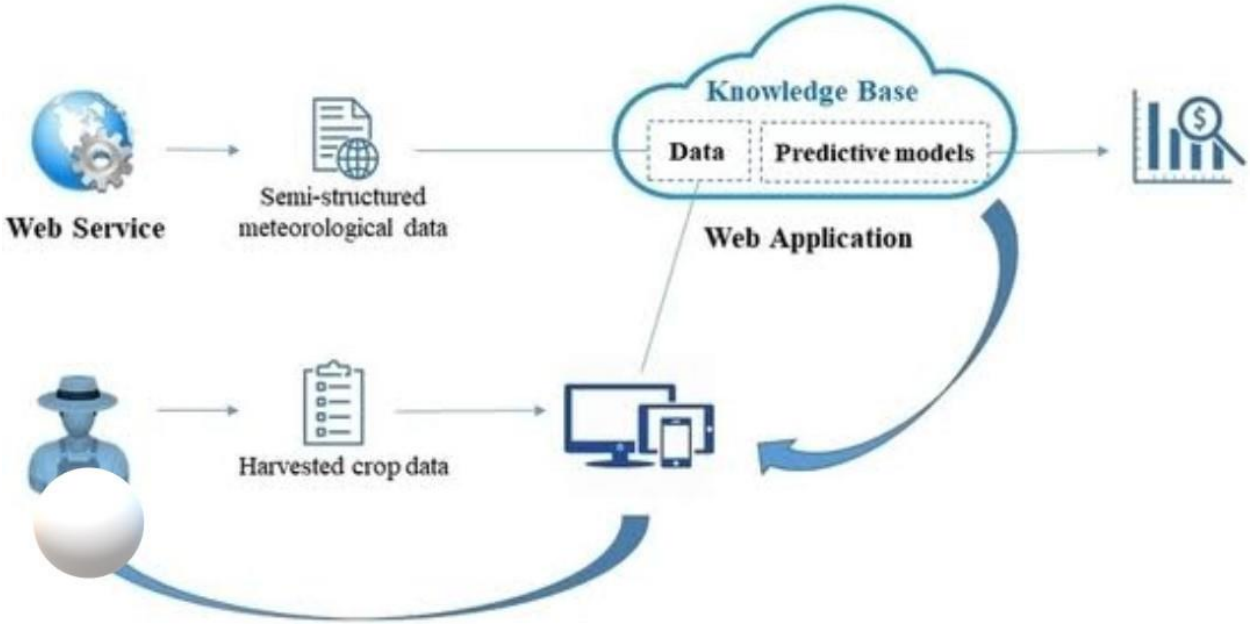
7.3) Seaborn:- Seaborn is a Python data visualization library grounded on Matplotlib. It provides a high- position interface for creating seductive and instructional statistical plates.

7.4) Matplotlib:- Matplotlib is a Python charting package that offers static, animated, and interactive visualisations. It is commonly used in scientific and technical computers to generate high-quality graphs and charts.

7.5) ipywidgets and sklearn:- ipywidgets is a Python library for creating interactive widgets in Jupyter notebooks. It allows users to create sliders, dropdowns, text boxes, and other user interface elements to interact with code and data in real-time.

Scikit-learn, sometimes known as sklearn, is a Python machine learning library. It has tools for data preprocessing, classification, regression, clustering, and dimensionality reduction, among other things.

8) ARCHITECTURE



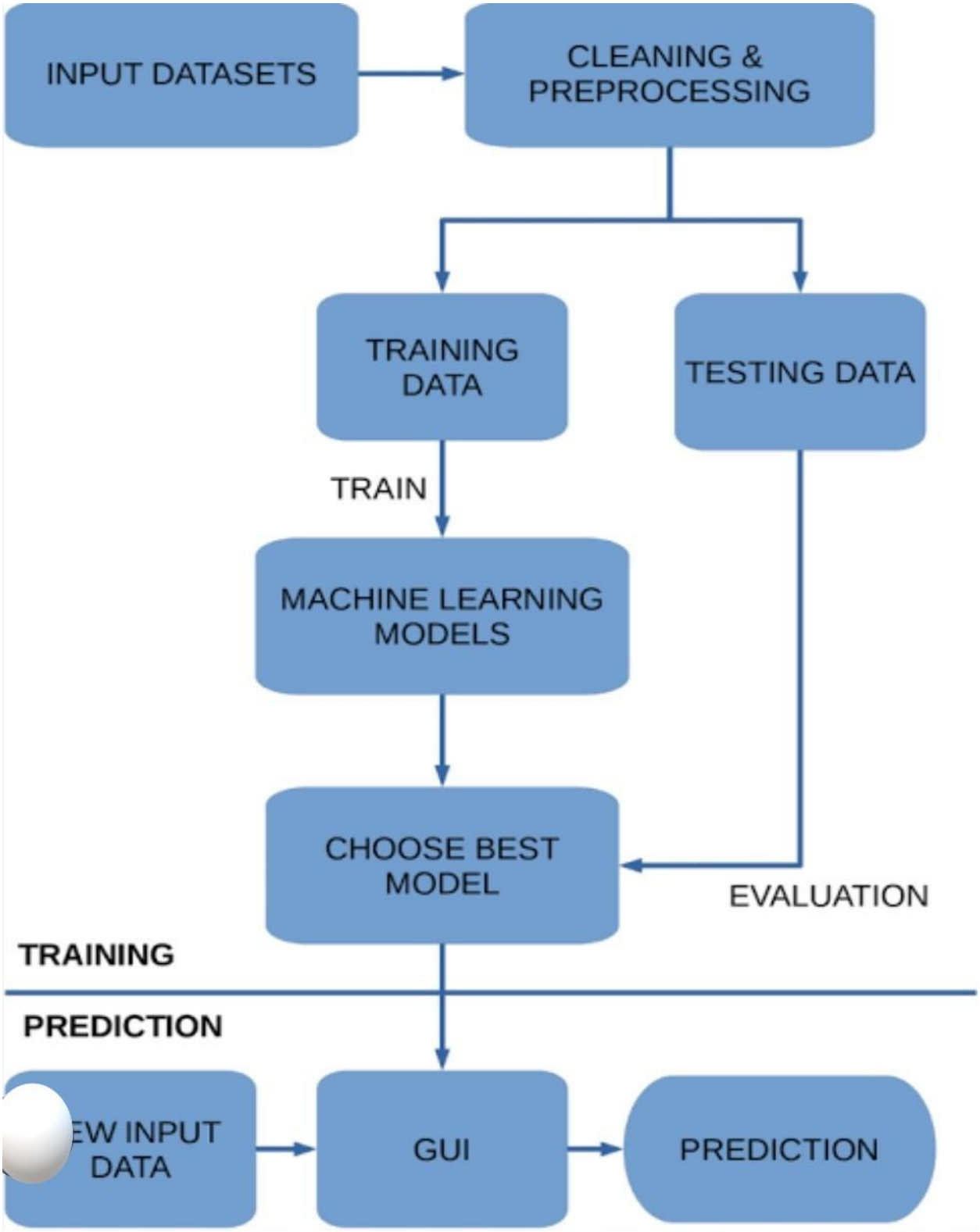
#) From Web service and farmer we can collect semi-structured meteorlogical data & Harvested crop data respectively.

#) And by the help of cloud computing we store this data in our virtual database.

#) Where we can analysis the data, Clean the data & train the data for predictive model

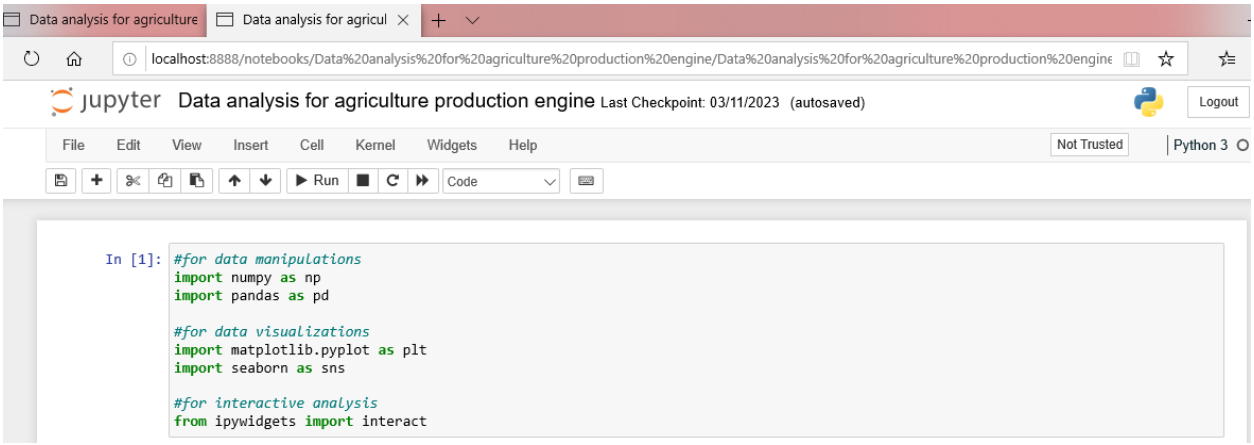
#) All this step make a wise decision and we know in which climatic and atmospheric condition which type of crops should be grown this help in optimize crop growth and also benefits in monetary terms.

9) FLOWCHART


INPUT DATASETS
CLEANING & PREPROCESSING
TRAINING DATA
TESTING DATA
TRAIN
MACHINE LEARNING MODELS
CHOOSE BEST MODEL
EVALUATION
TRAINING
PREDICTION
EW INPUT DATA
GUI
PREDICTION

10) DEPLOYMENT OF THE SYSTEM

10.1) Import libraries for data manipulation, Visualization and for interactive analysis



10.2) Reading the data set and check shape of our data set

```
In [2]: #Reading the dataset
        data = pd.read_csv('data.csv')

In [3]: #Shape of the dataset
        print("Shape of the dataset :", data.shape)

        Shape of the dataset : (2200, 8)
```

10.3) Clean our data set and remove duplicate data, Check missing value & head count. Overall we need to make our data in structured form.

```
In [4]: #Checking the head of the dataset
        data.head()
```

Out[4]:

	N	P	K	temperature	humidity	ph	rainfall	label
0	90	42	43	20.879744	82.002744	6.502985	202.935536	rice
1	85	58	41	21.770462	80.319644	7.038096	226.655537	rice
2	60	55	44	23.004459	82.320763	7.840207	263.964248	rice
3	74	35	40	26.491096	80.158363	6.980401	242.864034	rice
4	78	42	42	20.130175	81.604873	7.628473	262.717340	rice

```
In [5]: #Checking the missing values
        data.isnull().sum()

Out[5]: N              0
        P              0
        K              0
        temperature    0
        humidity       0
        ph             0
        rainfall       0
        label          0
        dtype: int64
```

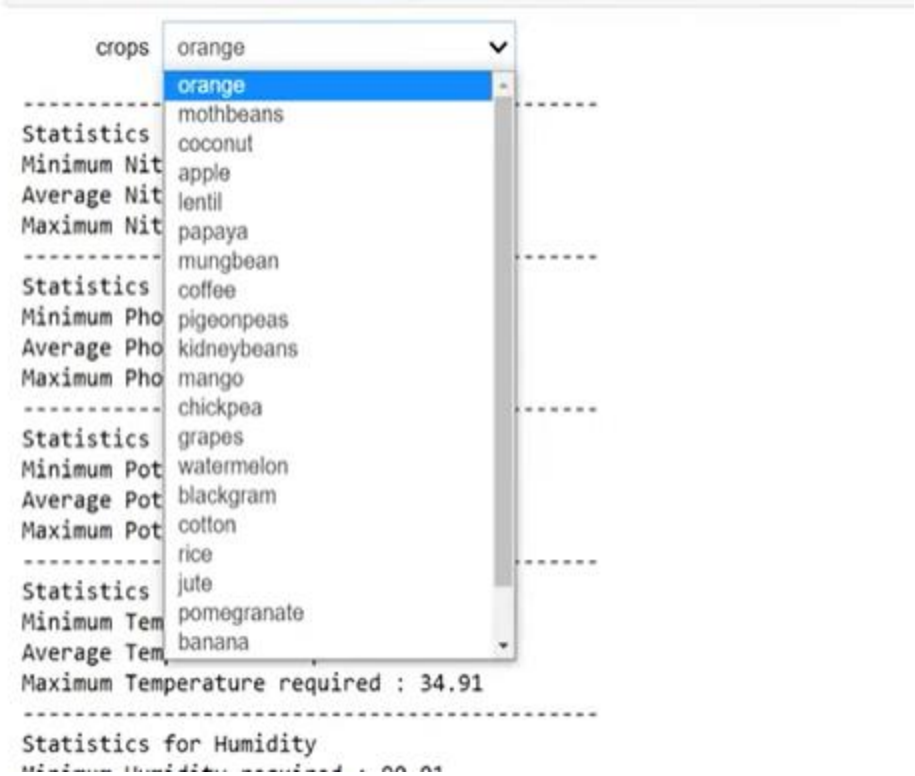
In [6]:
```
#Checking the crops present
data['label'].value_counts()
```

Out[6]:
```
coconut        100
pomegranate    100
kidneybeans    100
pigeonpeas     100
mungbean       100
mango          100
maize          100
cotton         100
banana         100
rice           100
orange         100
coffee         100
lentil         100
mothbeans      100
apple          100
watermelon     100
jute           100
muskmelon      100
grapes         100
blackgram      100
papaya         100
chickpea       100
Name: label, dtype: int64
```

10.4) Checking the detailed statistics data for each crop

In [8]:
```
#Checking the detailed statistics for each crop

@interact
def summary(crops = list(data['label'].value_counts().index)):
    x = data[data['label'] == crops]
    print(".....................................")
    print("Statistics for Nitrogen")
    print("Minimum Nitrogen required:", x['N'].min())
    print("Average Nitrogen required:", x['N'].mean())
    print("Maximum Nitrogen required:", x['N'].max())
    print(".....................................")
    print("Statistics for Phosphorous")
    print("Minimum Phosphorous required:", x['P'].min())
    print("Average Phosphorous required:", x['P'].mean())
    print("Maximum Phosphorous required:", x['P'].max())
    print(".....................................")
    print("Statistics for Pottasium")
    print("Minimum Pottasium required:", x['K'].min())
    print("Average Pottasium required:", x['K'].mean())
    print("Maximum Pottasium required:", x['K'].max())
    print(".....................................")
    print("Statistics for Temperature")
    print("Minimum Temperature required: {0:.2f}".format(x['temperature'].min()))
    print("Average Temperature required: {0:.2f}".format(x['temperature'].mean()))
    print("Maximum Temperature required: {0:.2f}".format(x['temperature'].max()))
    print(".....................................")
    print("Statistics for Humidity")
    print("Minimum Humidity required: {0:.2f}".format(x['humidity'].min()))
    print("Average Humidity required: {0:.2f}".format(x['humidity'].mean()))
    print("Maximum Humidity required: {0:.2f}".format(x['humidity'].max()))
```

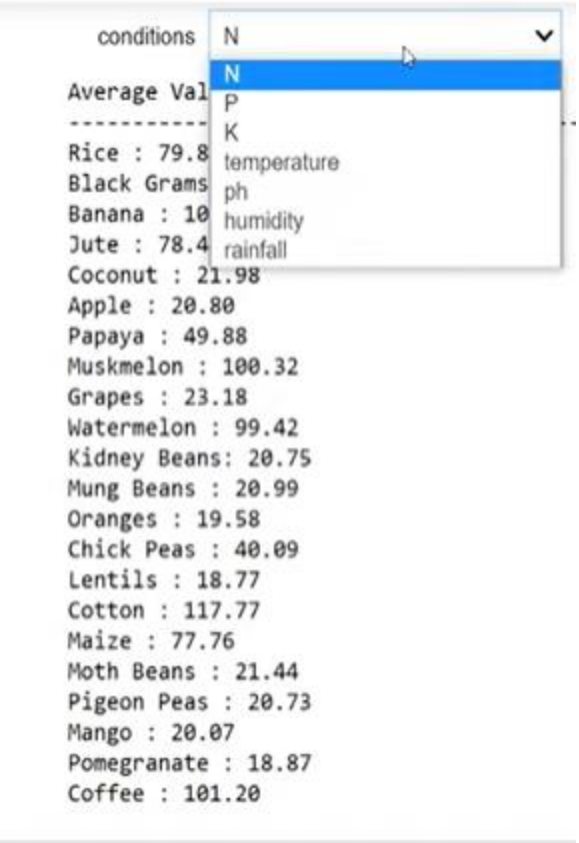
```
print(".....................................")
print("Statistics for PH")
print("Minimum PH required: {0:.2f}".format(x['ph'].min()))
print("Average PH required: {0:.2f}".format(x['ph'].mean()))
print("Maximum PH required: {0:.2f}".format(x['ph'].max()))
print(".....................................")
print("Statistics for Rainfall")
print("Minimum Rainfall required: {0:.2f}".format(x['rainfall'].min()))
print("Average Rainfall required: {0:.2f}".format(x['rainfall'].mean()))
print("Maximum Rainfall required: {0:.2f}".format(x['rainfall'].max()))
```



By using dropdown menu we can check statistic of each crop

10.5) Comparing Average requirement and conditions for each crop

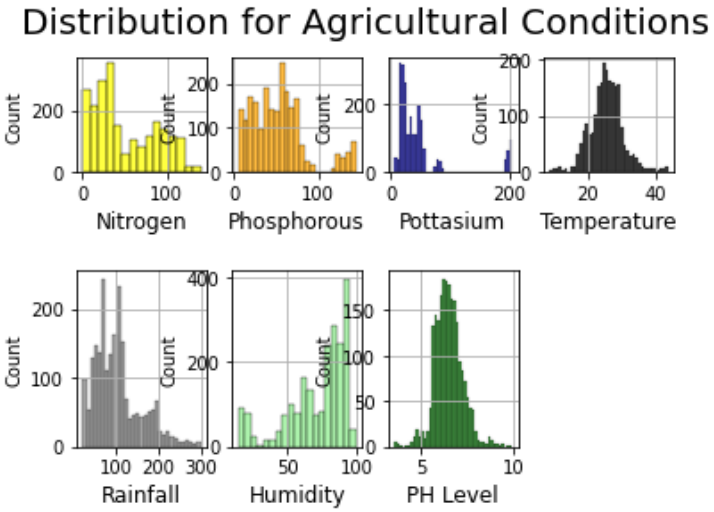
In [9]:
```
#Comparing Average requirement and conditions for each crop

@interact
def compare(conditions = ['N', 'P', 'K', 'temperature', 'ph', 'humidity', 'rainfall']):
    print("Average Value for", conditions, "is {0:.2f}".format(data[conditions].mean()))
    print("........................................")
    print("Rice : {0:.2f}".format(data[(data['label'] == 'rice')][conditions].mean()))
    print("Black grams : {0:.2f}".format(data[(data['label'] == 'blackgram')][conditions].mean()))
    print("Banana : {0:.2f}".format(data[(data['label'] == 'banana')][conditions].mean()))
    print("Jute : {0:.2f}".format(data[(data['label'] == 'jute')][conditions].mean()))
    print("Coconut : {0:.2f}".format(data[(data['label'] == 'coconut')][conditions].mean()))
    print("Apple : {0:.2f}".format(data[(data['label'] == 'apple')][conditions].mean()))
    print("Papaya : {0:.2f}".format(data[(data['label'] == 'papaya')][conditions].mean()))
    print("Muskmelon : {0:.2f}".format(data[(data['label'] == 'muskmelon')][conditions].mean()))
    print("Grapes : {0:.2f}".format(data[(data['label'] == 'grapes')][conditions].mean()))
    print("Watermelon : {0:.2f}".format(data[(data['label'] == 'watermelon')][conditions].mean()))
    print("Kidney Beans : {0:.2f}".format(data[(data['label'] == 'kidneybeans')][conditions].mean()))
    print("Mung Beans : {0:.2f}".format(data[(data['label'] == 'mungbean')][conditions].mean()))
    print("Oranges : {0:.2f}".format(data[(data['label'] == 'orange')][conditions].mean()))
    print("Chick Peas : {0:.2f}".format(data[(data['label'] == 'chickpea')][conditions].mean()))
    print("Lentils : {0:.2f}".format(data[(data['label'] == 'lentil')][conditions].mean()))
    print("Cotton : {0:.2f}".format(data[(data['label'] == 'cotton')][conditions].mean()))
    print("Maize : {0:.2f}".format(data[(data['label'] == 'maize')][conditions].mean()))
    print("Moth Beans : {0:.2f}".format(data[(data['label'] == 'mothbeans')][conditions].mean()))
    print("Pigeon Peas : {0:.2f}".format(data[(data['label'] == 'pigeonpeas')][conditions].mean()))
    print("Mango : {0:.2f}".format(data[(data['label'] == 'mango')][conditions].mean()))
    print("Pomegranate : {0:.2f}".format(data[(data['label'] == 'pomegranate')][conditions].mean()))
    print("Coffee : {0:.2f}".format(data[(data['label'] == 'coffee')][conditions].mean()))
```



10.6) Checking distribution for each crop using data visualization technology

In [11]:
```
#Checking distributiion for each crop

plt.subplot(3,4,1)
sns.histplot(data['N'], color="yellow")
plt.xlabel('Nitrogen', fontsize = 12)
plt.grid()

plt.subplot(3,4,2)
sns.histplot(data['P'], color="orange")
plt.xlabel('Phosphorous', fontsize = 12)
plt.grid()

plt.subplot(3,4,3)
sns.histplot(data['K'], color="darkblue")
plt.xlabel('Pottasium', fontsize = 12)
plt.grid()

plt.subplot(3,4,4)
sns.histplot(data['temperature'], color="black")
plt.xlabel('Temperature', fontsize = 12)
plt.grid()

plt.subplot(2,4,5)
sns.histplot(data['rainfall'], color="grey")
plt.xlabel('Rainfall', fontsize = 12)
plt.grid()

plt.subplot(2,4,6)
sns.histplot(data['humidity'], color="lightgreen")
plt.xlabel('Humidity', fontsize = 12)
plt.grid()
```

```
plt.subplot(3,4,4)
sns.histplot(data['temperature'], color="black")
plt.xlabel('Temperature', fontsize = 12)
plt.grid()

plt.subplot(2,4,5)
sns.histplot(data['rainfall'], color="grey")
plt.xlabel('Rainfall', fontsize = 12)
plt.grid()

plt.subplot(2,4,6)
sns.histplot(data['humidity'], color="lightgreen")
plt.xlabel('Humidity', fontsize = 12)
plt.grid()

plt.subplot(2,4,7)
sns.histplot(data['ph'], color="darkgreen")
plt.xlabel('PH Level', fontsize = 12)
plt.grid()

plt.suptitle('Distribution for Agricultural Conditions', fontsize = 20)
plt.show()
```



10.7) from sklearn.cluster import KMeans.

KMeans is a Machine learning algorithm which is used in clustering analysis.

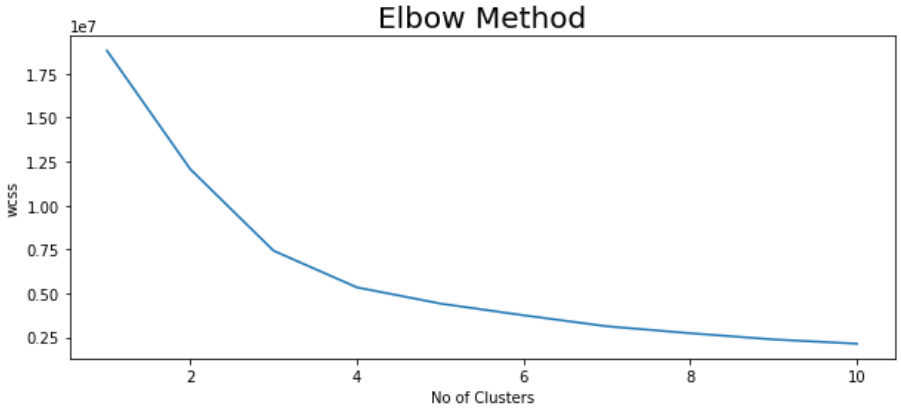
In [22]:
```
from sklearn.cluster import KMeans

#removing the labels column
x = data.drop(['label'], axis=1)

#selecting all the values of data
x = x.values

#checking the shape
print(x.shape)
```

```
(2200, 7)
```

10.8) Determining the optimum number of clusters within the Dataset

In [23]:
```
#Determining the optimum number of clusters within the Dataset

plt.rcParams['figure.figsize'] = (10,4)

wcss = []
for i in range(1,11):
    km = KMeans(n_clusters = i, init = 'k-means++', max_iter = 2000, n_init = 10, random_state = 0)
    km.fit(x)
    wcss.append(km.inertia_)

#Plotting the results

plt.plot(range(1,11), wcss)
plt.title('Elbow Method', fontsize = 20)
plt.xlabel('No of Clusters')
plt.ylabel('wcss')
plt.show
```

Out[23]:
```
<function matplotlib.pyplot.show(close=None, block=None)>
```



10.9) Implementation of K Means algorithm to perform Clustering analysis

In [24]:
```
#Implementation of K Means algorithm to perform Clustering analysis

km = KMeans(n_clusters = 4, init = 'k-means++',  max_iter = 2000, n_init = 10, random_state = 0)
y_means = km.fit_predict(x)

#Finding the results
a = data['label']
y_means = pd.DataFrame(y_means)
z = pd.concat([y_means, a], axis = 1)
z = z.rename(columns = {0: 'cluster'})

#Checking the clusters for each crop
print("Lets Check the results after applying K Means Clustering Analysis \n")
print("Crops in First Cluster:", z[z['cluster'] == 0]['label'].unique())
print("........................................")
print("Crops in Second Cluster:", z[z['cluster'] == 1]['label'].unique())
print("........................................")
print("Crops in Third Cluster:", z[z['cluster'] == 2]['label'].unique())
print("........................................")
print("Crops in Fourth Cluster:", z[z['cluster'] == 3]['label'].unique())
```

```
Lets Check the results after applying K Means Clustering Analysis

Crops in First Cluster: ['grapes' 'apple']
........................................
Crops in Second Cluster: ['maize' 'banana' 'watermelon' 'muskmelon' 'papaya' 'cotton' 'coffee']
........................................
Crops in Third Cluster: ['rice' 'pigeonpeas' 'papaya' 'coconut' 'jute' 'coffee']
........................................
Crops in Fourth Cluster: ['maize' 'chickpea' 'kidneybeans' 'pigeonpeas' 'mothbeans' 'mungbean'
 'blackgram' 'lentil' 'pomegranate' 'mango' 'orange' 'papaya' 'coconut']
```

10.10) Splitting the Dataset for predictive modeling

In [25]:
```
#Splitting the Dataset for predictive modelling

y = data['label']
x = data.drop(['label'], axis=1)

print("Shape of x:", x.shape)
print("Shape of y:", y.shape)
```

```
Shape of x: (2200, 7)
Shape of y: (2200,)
```

10.11) Creating training and testing sets for results validation

In [26]:
```
#Creating training and testing sets for results validation
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 = 0)

print("The Shape Of x train:", x_train.shape)
print("The Shape Of x test:", x_test.shape)
print("The Shape Of y train:", y_train.shape)
print("The Shape Of y test:", y_test.shape)
```

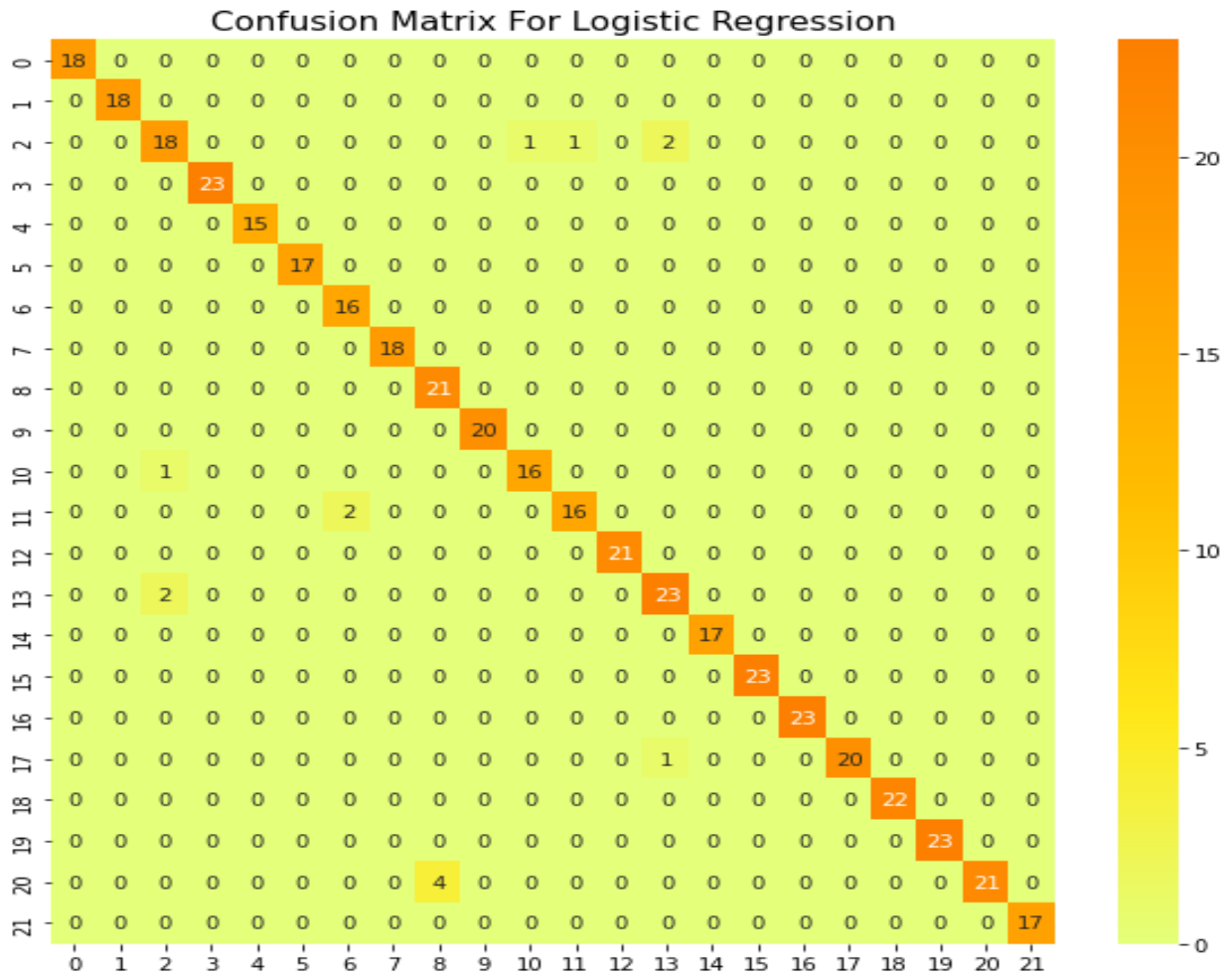
```
The Shape Of x train: (1760, 7)
The Shape Of x test: (440, 7)
The Shape Of y train: (1760,)
The Shape Of y test: (440,)
```

10.12) Creating a Predictive Machine Learning Model

```
In [28]: #Creating a Predictive Model

         from sklearn.linear_model import LogisticRegression

         model = LogisticRegression()
         model.fit(x_train, y_train)
         y_pred = model.predict(x_test)
```

10.13) Evaluating Model Performance

```
In [29]: #Evaluating the model performance
         from sklearn.metrics import confusion_matrix

         #Printing the Confusing Matrix
         plt.rcParams['figure.figsize'] = (10,10)
         cm = confusion_matrix(y_test, y_pred)
         sns.heatmap(cm, annot = True, cmap = 'Wistia')
         plt.title('Confusion Matrix For Logistic Regression', fontsize = 15)
         plt.show()
```



10.14) Defining the classification Report & Printing the Classification Report

In [30]:
```
#Defining the classification Report
from sklearn.metrics import classification_report

#Printing the Classification Report
cr = classification_report(y_test, y_pred)
print(cr)
```

```
              precision    recall  f1-score   support

       apple       1.00      1.00      1.00        18
      banana       1.00      1.00      1.00        18
   blackgram       0.86      0.82      0.84        22
    chickpea       1.00      1.00      1.00        23
     coconut       1.00      1.00      1.00        15
      coffee       1.00      1.00      1.00        17
      cotton       0.89      1.00      0.94        16
      grapes       1.00      1.00      1.00        18
        jute       0.84      1.00      0.91        21
 kidneybeans       1.00      1.00      1.00        20
      lentil       0.94      0.94      0.94        17
       maize       0.94      0.89      0.91        18
       mango       1.00      1.00      1.00        21
   mothbeans       0.88      0.92      0.90        25
    mungbean       1.00      1.00      1.00        17
   muskmelon       1.00      1.00      1.00        23
      orange       1.00      1.00      1.00        23
      papaya       1.00      0.95      0.98        21
  pigeonpeas       1.00      1.00      1.00        22
 pomegranate       1.00      1.00      1.00        23
        rice       1.00      0.84      0.91        25
  watermelon       1.00      1.00      1.00        17

    accuracy                           0.97       440
   macro avg       0.97      0.97      0.97       440
weighted avg       0.97      0.97      0.97       440
```

10.15) Now our predictive ML model is trained perfectly and now our model suggested crops for given climatic and soil conditions.

Example:-

```
In [32]: prediction = model.predict((np.array([[90, 40, 40, 20, 80, 7, 200]])))
         print("The Suggested Crop for given climatic condition is :",prediction)

         The Suggested Crop for given climatic condition is : ['rice']
```

11) FUTURE SCOPE & CONCLUSION

Data analysis & ML predictive model has already begun to play an increasingly important role in agriculture production, and there is a significant potential for its future applications. Some of the key areas where data analysis can be useful include:

1. Precision agriculture: Data analysis can help farmers identify areas of their fields that need more attention, such as those with poor soil quality, disease, or pest infestations. This allows them to target their efforts more efficiently and reduce waste.

2. Weather forecasting: Agriculture production is heavily influenced by weather patterns, and data analysis can be used to forecast weather conditions and help farmers plan their operations accordingly. This can help farmers optimize their crop yields and minimize crop damage due to weather events.

3. Crop monitoring: Data analysis can help farmers monitor crop growth and identify any issues that may arise, such as nutrient deficiencies or diseases. This allows farmers to take corrective action quickly and avoid yield losses.

4. Predictive analytics: Data analysis can be used to predict future crop yields based on historical data and current conditions. This can help farmers plan their operations and make informed decisions about planting, harvesting, and marketing their crops.

5. Supply chain optimization: Data analysis can be used to optimize the supply chain for agricultural products, from farm to table. This includes tracking inventory levels, forecasting demand, and optimizing transportation and logistics.

But we don't stop our self only using technology like machine learning & data analysis in the field of agriculture production engine. Because there are more technology which is also very useful in agriculture production engine these technology are Sensors, IOT & Cloud Computing.

Let's how these technology helps in agriculture production engine:-

1. Sensors: Sensors are being used to collect data on various aspects of agriculture production, such as soil moisture, temperature, humidity, and light levels. This data can be used to optimize irrigation, fertilization, and other inputs, which can increase yields and reduce waste. Sensors can also be used to monitor plant health and detect diseases or pests early, allowing farmers to take corrective action before significant damage occurs.

2. IoT: The IoT allows farmers to connect sensors, cameras, and other devices to the internet, enabling real-time monitoring and control of agriculture production processes. IoT devices can be used to automate irrigation, lighting, and other processes, reducing labor costs and increasing efficiency.

3. Cloud computing: Cloud computing technologies are being used to store and analyze the vast amounts of data generated by agriculture production. Cloud-based analytics platforms can be used to analyze data from sensors and other devices to provide insights into crop yields, soil health, and other factors that affect agriculture production. Cloud computing can also be used to enable remote monitoring and control of agriculture production processes, improving overall efficiency and reducing costs.

Conclusion:-

In conclusion, data analysis and machine learning predictive models have the potential to revolutionize agriculture production by providing valuable insights into crop yields, plant health, weather patterns, and supply chain optimization.

Data analysis can help farmers identify areas of their fields that need more attention, such as those with poor soil quality, disease, or pest infestations. This allows them to target their efforts more efficiently and reduce waste. Machine learning predictive models can help predict crop yields, identify diseases, analyze soil health, and predict pest infestations, enabling farmers to take corrective action quickly and avoid yield losses.

Furthermore, machine learning predictive models can help farmers adapt to changing weather patterns and reduce their environmental impact. These technologies can also be used to optimize the supply chain for agricultural products, from farm to table. This includes tracking inventory levels, forecasting demand, and optimizing transportation and logistics.

12) REFERENCE

1) Kross, S., Peng, R.D., Caffo, B.S., Gooding, I. and Leek, J.T., 2020. The democratization of data science education. *The American Statistician*, *74*(1), pp.1-7.

2) Batt, Steven, Tara Grealis, Oskar Harmon, and Paul Tomolonis. "Learning Tableau: A data visualization tool." *The Journal of Economic Education* 51, no. 3-4 (2020): 317-328.

3) Hartmann, U. and von Both, P., 2009. A declarative approach to cross-domain model analysis. *Managing It in Construction/Managing Construction for Tomorrow*, *26*, pp.45-51.

4) Van Rossum, G., Warsaw, B. and Coghlan, N., 2001. PEP 8–style guide for python code. *Python. org*, *1565*, p.28.

5) Ahmad, W.U., Tushar, M.G.R., Chakraborty, S. and Chang, K.W., 2021. Avatar: A parallel corpus for java-python program translation. *arXiv preprint arXiv:2108.11590*.

6) Ludwig, H., Baracaldo, N., Thomas, G., Zhou, Y., Anwar, A., Rajamoni, S., Ong, Y., Radhakrishnan, J., Verma, A., Sinn, M. and Purcell, M., 2020. Ibm federated learning: an enterprise framework white paper v0. 1. *arXiv preprint arXiv:2007.10987*.

7) Mosavi, A., Ozturk, P. and Chau, K.W., 2018. Flood prediction using machine learning models: Literature review. *Water*, *10*(11), p.1536.

8) Nguyen, A.T., Dzator, J. and Nadolny, A., 2015. Does contract farming improve productivity and income of farmers?: A review of theory and evidence. *The Journal of Developing Areas*, *49*(6), pp.531-538.

9) Saiz-Rubio, V. and Rovira-Más, F., 2020. From smart farming towards agriculture 5.0: A review on crop data management. *Agronomy*, *10*(2), p.207.

10) Mahendra Dev, S., 2014. Small farmers in India: Challenges and opportunities.

qw

by Er We

Submission date: 19-May-2023 09:24AM (UTC+0530)
Submission ID: 2096779061
File name: NTCC,_Major_Project,_For_Plag_Check_1.pdf (1.46M)
Word count: 2500
Character count: 13792

qw

ORIGINALITY REPORT

5%	2%	0%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Oakton Community College Student Paper	1%
2	Submitted to Zambia Centre for Accountancy Studies Student Paper	1%
3	Submitted to Cardinal Newman College Student Paper	1%
4	Submitted to University of West London Student Paper	1%
5	articles.connectnigeria.com Internet Source	1%
6	www.editions-eni.fr Internet Source	1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		

AMITY UNIVERSITY
UTTAR PRADESH
AMITY INSTITUE OF INFORMATION TECHNOLOGY

WEEKLY PROGRESS REPORT (WPR) – 1

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

Faculty Guide- Dr. Rashmi Vasishth

Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Find a problem in agriculture sector and in production engine and work on how to find a optimal solution and read and analyze some previous data set and problem sets related to agriculture sector.

Progress For The Week:-

Learn data analysis tools and clear my hands on data science, Visualization technologies. Revise numpy, matplotlib and pandas libraries in python environment.

Regards

Yatin Sengar

AMITY UNIVERSITY

UTTAR PRADESH

AMITY INSTITUE OF INFORMATION TECHNOLOGY

WPR-2

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

Faculty Guide- Dr. Rashmi Vasishth

Project Title:-

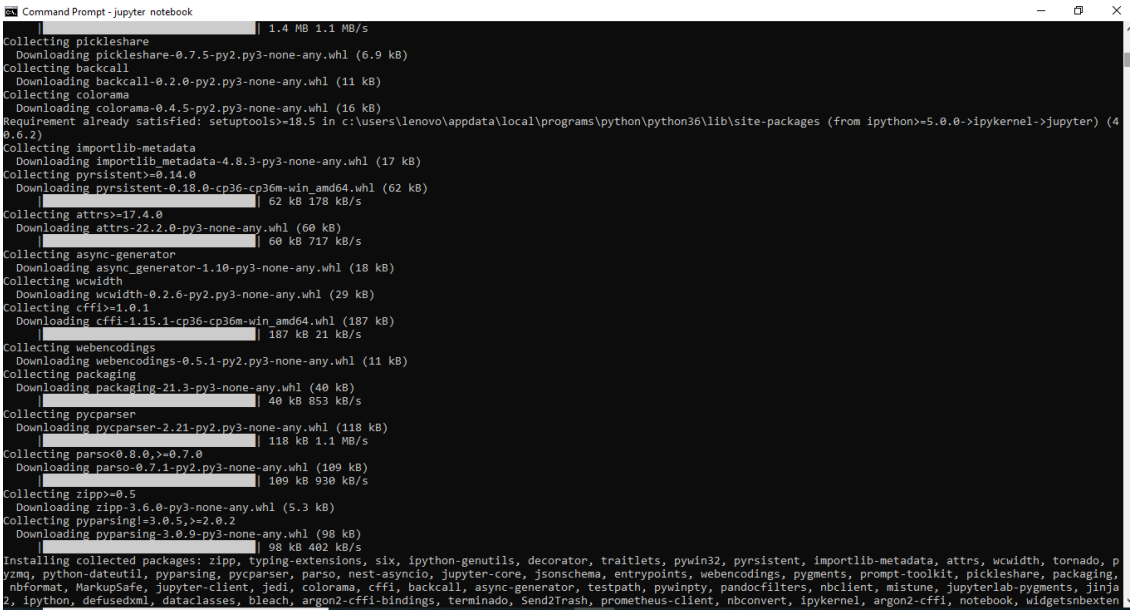
Data analysis for agriculture production engine

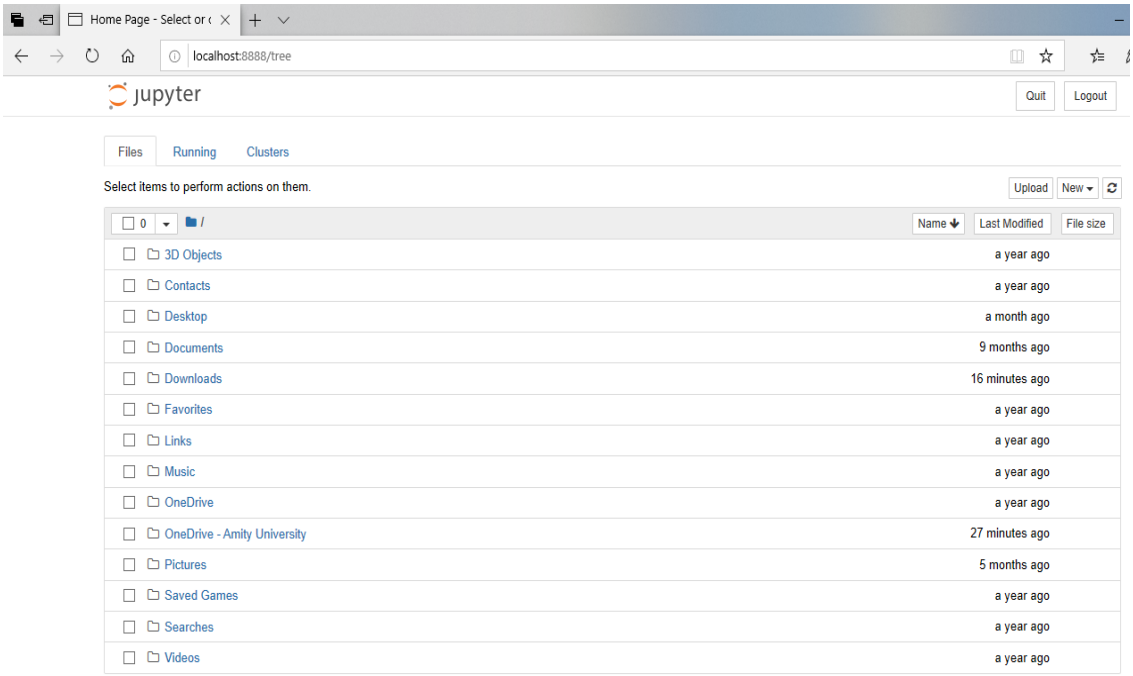
Summary of This Week:-

Learn the concept of precious farming and download and analysis the data set and problem set related to farming.

Set-up the python environment and download the jupyter notebook and other dependencies using command prompt

Progress For The Week:-





Regards

Yatin Sengar

AMITY UNIVERSITY

UTTAR PRADESH

AMITY INSTITUE OF INFORMATION TECHNOLOGY

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

Faculty Guide- Dr. Rashmi Vasishth

Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Import other dependencies through pip command in command prompt like numpy, pandas, matplotlib, etc.

And create a file in our python3 jupyter environment. And also upload our data set in the environment.

Regards

Yatin Sengar

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

Faculty Guide- Dr. Rashmi Vasishth

Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

I read the data sets and problem sets and reached the following conclusion:-

There are 7 key factors that I've taken into account which will help us in determining, exactly which crop should be grown and at what period of time, viz. Amount of Nitrogen, Phosphorus and Potassium in soil, Temperature in degree celcius, Humidity, pH and Rainfall in mm.

Tools used: Python & Jupyter Notebook Libraries used: Numpy, Pandas, Seaborn, Matplotlib, ipywidgets and sklearn. Machine Learning Algorithms used: Clustering Analysis and Logistic Regression.

Regards

Yatin Sengar

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

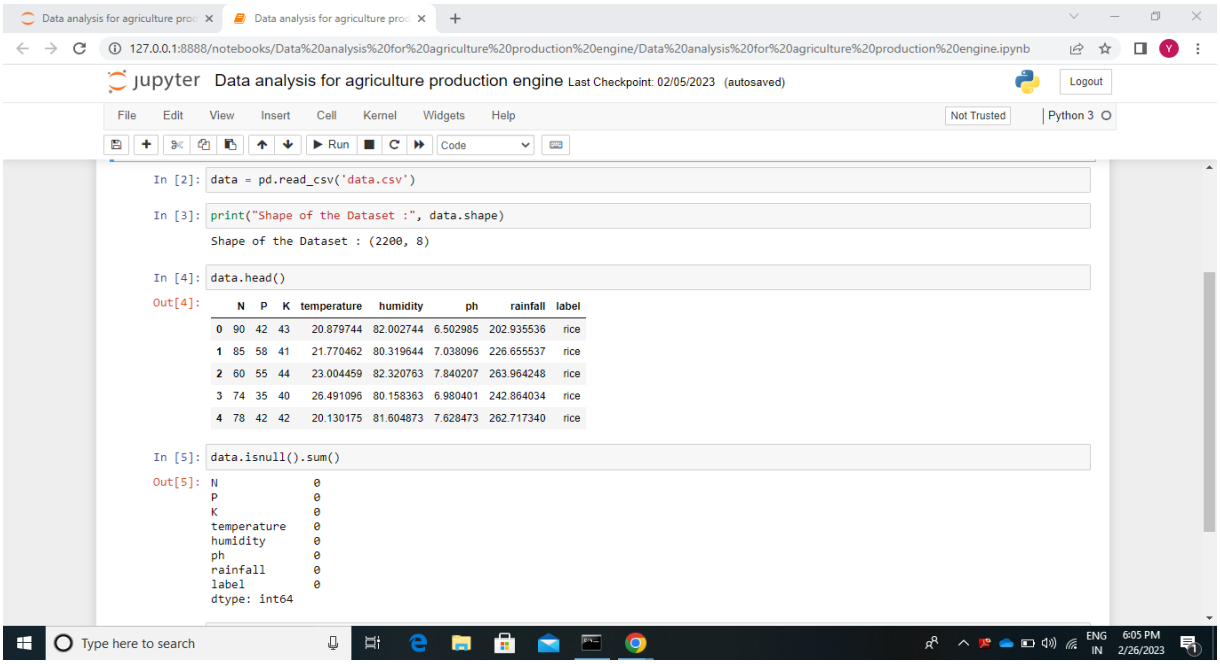
Faculty Guide- Dr. Rashmi Vasishth

Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Check the validity, heads and null values of the dataset



Regards

Yatin Sengar

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

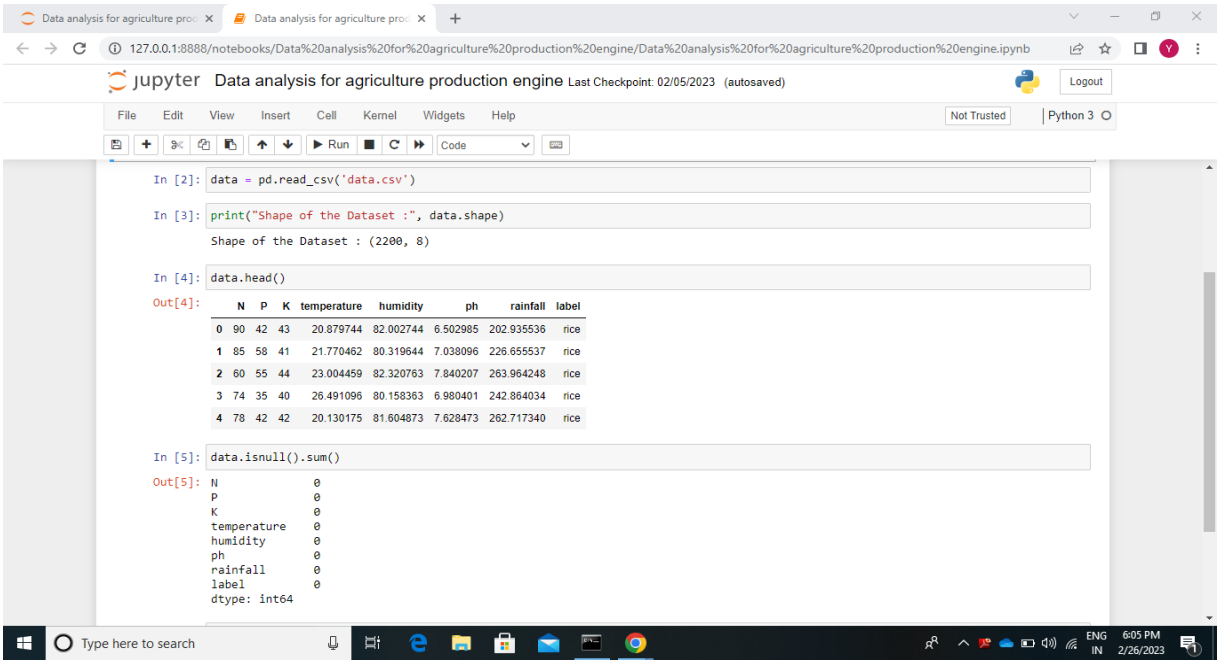
Faculty Guide- Dr. Rashmi Vasishth

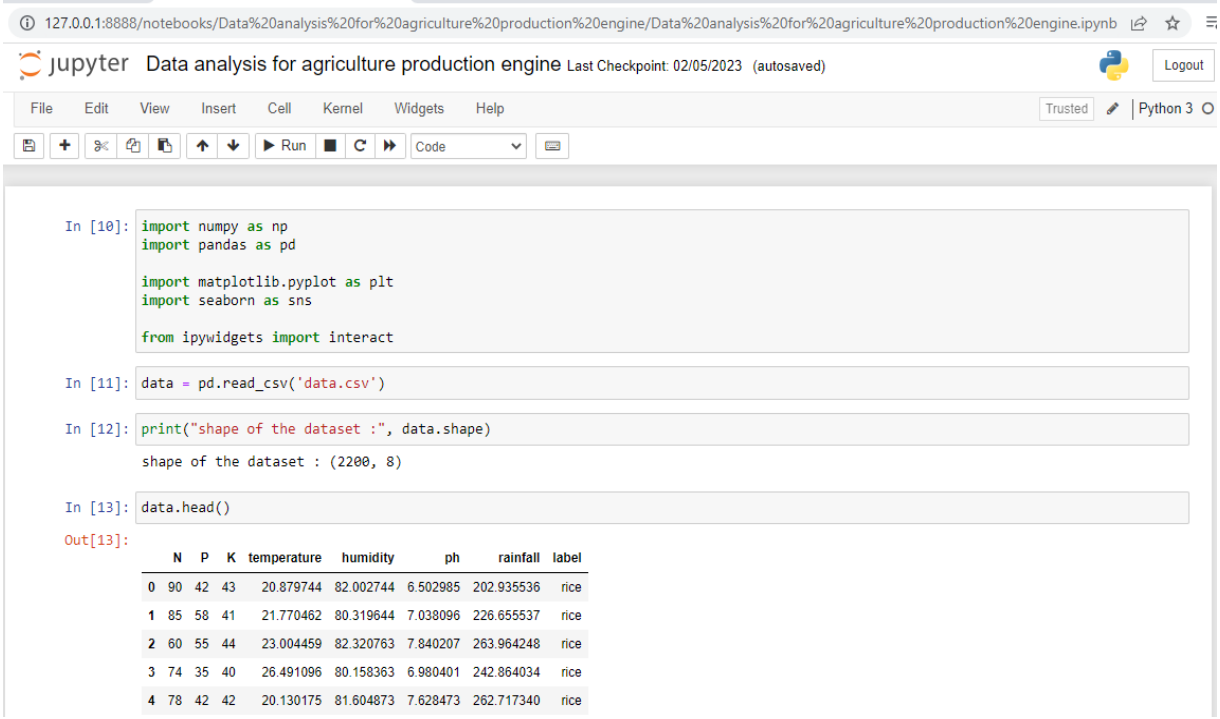
Project Title:-

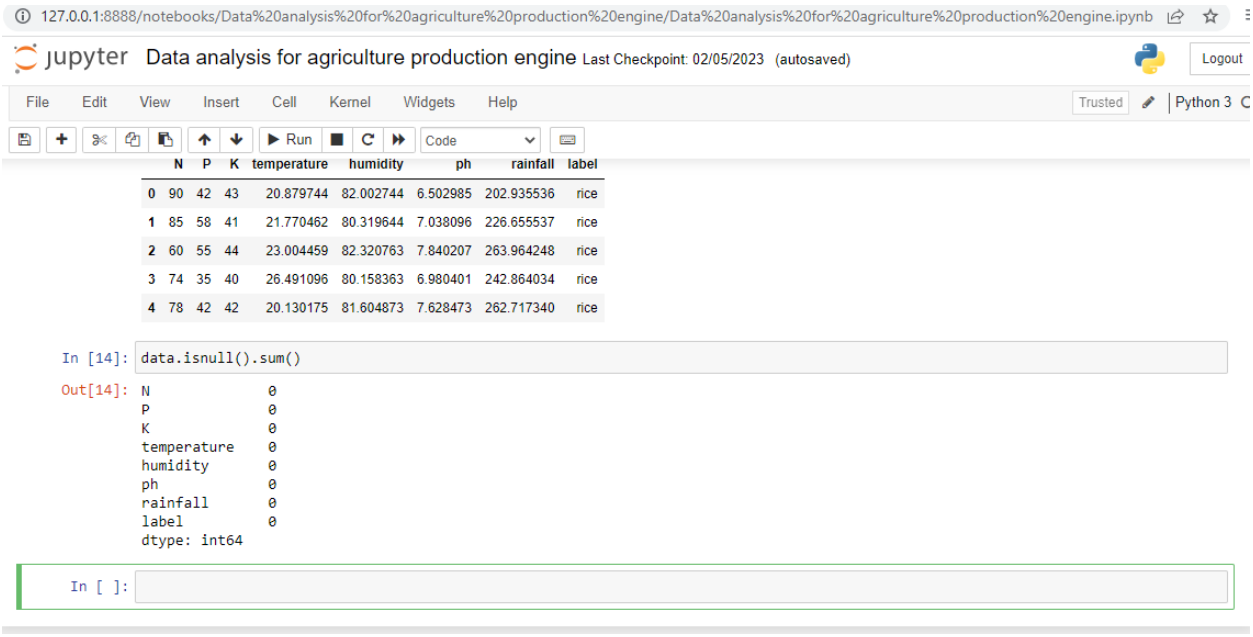
Data analysis for agriculture production engine

Summary of This Week:-

Data cleaning, Check duplicate entries


Data analysis for agriculture production engine
Last Checkpoint: 02/05/2023 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [2]: data = pd.read_csv('data.csv')
In [3]: print("Shape of the Dataset :", data.shape)
Shape of the Dataset : (2200, 8)
In [4]: data.head()
Out[4]:
N P K temperature humidity ph rainfall label
0 90 42 43 20.879744 82.002744 6.502985 202.935536 rice
1 85 58 41 21.770462 80.319644 7.038096 226.655537 rice
2 60 55 44 23.004459 82.320763 7.840207 263.964248 rice
3 74 35 40 26.491096 80.158363 6.980401 242.864034 rice
4 78 42 42 20.130175 81.604873 7.628473 262.717340 rice
In [5]: data.isnull().sum()
Out[5]:
N 0
P 0
K 0
temperature 0
humidity 0
ph 0
rainfall 0
label 0
dtype: int64
Type here to search
ENG IN
6:05 PM
2/26/2023


127.0.0.1:8888/notebooks/Data%20analysis%20for%20agriculture%20production%20engine/Data%20analysis%20for%20agriculture%20production%20engine.ipynb
Data analysis for agriculture production engine
Last Checkpoint: 02/05/2023 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [10]:
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from ipywidgets import interact
In [11]: data = pd.read_csv('data.csv')
In [12]: print("shape of the dataset :", data.shape)
shape of the dataset : (2200, 8)
In [13]: data.head()
Out[13]:
N P K temperature humidity ph rainfall label
0 90 42 43 20.879744 82.002744 6.502985 202.935536 rice
1 85 58 41 21.770462 80.319644 7.038096 226.655537 rice
2 60 55 44 23.004459 82.320763 7.840207 263.964248 rice
3 74 35 40 26.491096 80.158363 6.980401 242.864034 rice
4 78 42 42 20.130175 81.604873 7.628473 262.717340 rice



Regards

Yatin Sengar

AMITY UNIVERSITY
UTTAR PRADESH
AMITY INSTITUE OF INFORMATION TECHNOLOGY

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

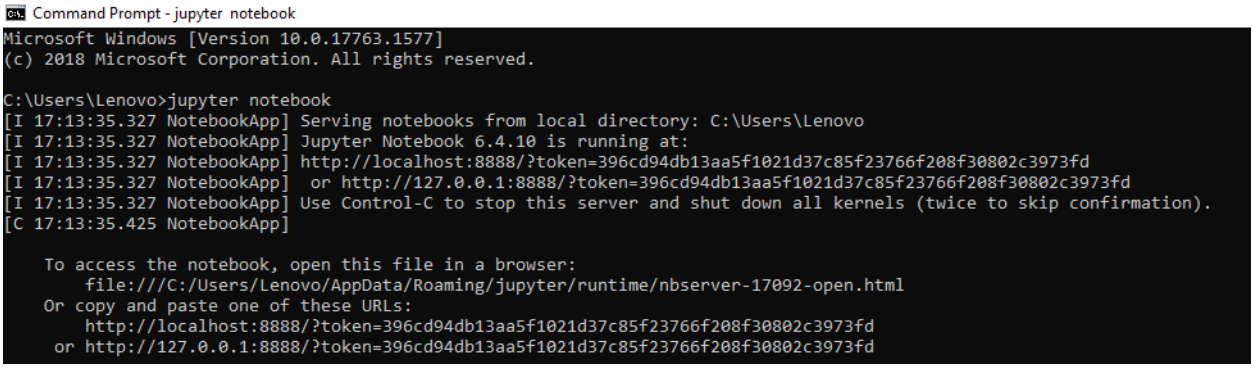
Faculty Guide- Dr. Rashmi Vasishth

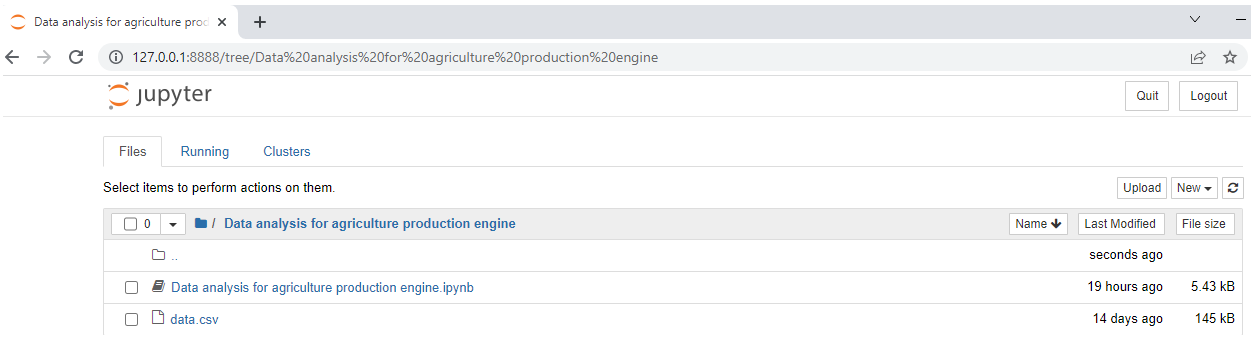
Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Create proper channel for data pipeline





Regards

Yatin Sengar

AMITY UNIVERSITY
UTTAR PRADESH
AMITY INSTITUE OF INFORMATION TECHNOLOGY

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

Faculty Guide- Dr. Rashmi Vasishth

Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Use different technology for data analyzing like:-

ML:- Machine learning is a branch of artificial intelligence that uses algorithms to extract data and then predict future trends. Software is programmed with models that allow engineers to conduct statistical analysis to understand patterns in the data

Logistic Regression:- Logistic regression is a data analysis technique that uses mathematics to find the relationships between two data factors. It then uses this relationship to predict the value of one of those factors based on the other. The prediction usually has a finite number of outcomes, like yes or no.

Data Visualization:- Data visualization is the representation of data through use of common graphics, such as charts, plots, infographics, and even animations. These visual displays of information communicate complex data relationships and data-driven insights in a way that is easy to understand

AMITY UNIVERSITY
UTTAR PRADESH
AMITY INSTITUE OF INFORMATION TECHNOLOGY

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

Faculty Guide- Dr. Rashmi Vasishth

Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Check summary of the crops

```
In [19]: # Lets check the Summary for all the crops

         print("Average Ratio of Nitrogen in the Soil : {0:.2f}".format(data['N'].mean()))
         print("Average Ratio of Phosphorous in the Soil : {0:.2f}".format(data['P'].mean()))
         print("Average Ratio of Potassium in the Soil : {0:.2f}".format(data['K'].mean()))
         print("Average Tempature in Celsius : {0:.2f}".format(data['temperature'].mean()))
         print("Average Relative Humidity in % : {0:.2f}".format(data['humidity'].mean()))
         print("Average PH Value of the soil : {0:.2f}".format(data['ph'].mean()))
         print("Average Rainfall in mm : {0:.2f}".format(data['rainfall'].mean()))

         Average Ratio of Nitrogen in the Soil : 50.55
         Average Ratio of Phosphorous in the Soil : 53.36
         Average Ratio of Potassium in the Soil : 48.15
         Average Tempature in Celsius : 25.62
         Average Relative Humidity in % : 71.48
         Average PH Value of the soil : 6.47
         Average Rainfall in mm : 103.46
```

Regards

AMITY UNIVERSITY
UTTAR PRADESH
AMITY INSTITUE OF INFORMATION TECHNOLOGY

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

Faculty Guide- Dr. Rashmi Vasishth

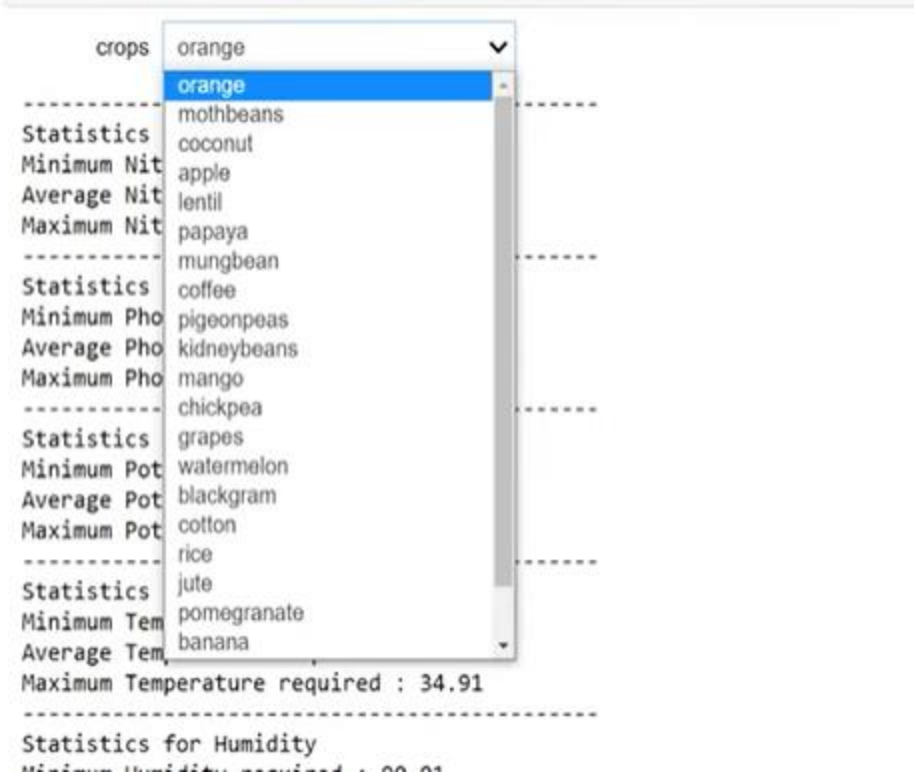
Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Check summary Statistics of the crops by ipywidgets

```
In [20]: # Lets check the Summary Statistics for each of the Crops

         @interact
         def summary(crops = list(data['label'].value_counts().index)):
             x = data[data['label'] == crops]
             print("--------------------------------------------")
             print("Statistics for Nitrogen")
             print("Minimum Nitrigen required :", x['N'].min())
             print("Average Nitrogen required :", x['N'].mean())
             print("Maximum Nitrogen required :", x['N'].max())
             print("--------------------------------------------")
             print("Statistics for Phosphorous")
             print("Minimum Phosphorous required :", x['P'].min())
             print("Average Phosphorous required :", x['P'].mean())
             print("Maximum Phosphorous required :", x['P'].max())
             print("--------------------------------------------")
             print("Statistics for Potassium")
             print("Minimum Potassium required :", x['K'].min())
             print("Average Potassium required :", x['K'].mean())
             print("Maximum Potassium required :", x['K'].max())
             print("--------------------------------------------")
             print("Statistics for Temperature")
```



Regards

Yatin Sengar

AMITY UNIVERSITY
UTTAR PRADESH
AMITY INSTITUE OF INFORMATION TECHNOLOGY

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

Faculty Guide- Dr. Rashmi Vasishth

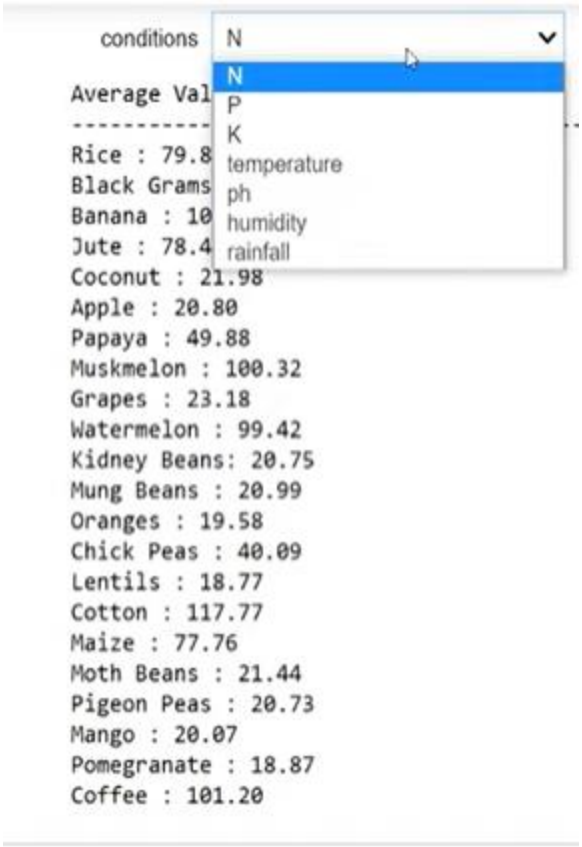
Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Compare the average requirement for each crops with average condition

```
@interact
def compare(conditions = ['N','P','K','temperature','ph','humidity','rainfall']):
    print("Average Value for", conditions,"is {0:.2f}".format(data[conditions].mean()))
    print("--------------------------------------------")
    print("Rice : {0:.2f}".format(data[(data['label'] == 'rice')][conditions].mean()))
    print("Black Grams : {0:.2f}".format(data[data['label'] == 'blackgram'][conditions].mean()))
    print("Banana : {0:.2f}".format(data[(data['label'] == 'banana')][conditions].mean()))
    print("Jute : {0:.2f}".format(data[data['label'] == 'jute'][conditions].mean()))
    print("Coconut : {0:.2f}".format(data[(data['label'] == 'coconut')][conditions].mean()))
    print("Apple : {0:.2f}".format(data[data['label'] == 'apple'][conditions].mean()))
    print("Papaya : {0:.2f}".format(data[(data['label'] == 'papaya')][conditions].mean()))
    print("Muskmelon : {0:.2f}".format(data[data['label'] == 'muskmelon'][conditions].mean()))
    print("Grapes : {0:.2f}".format(data[(data['label'] == 'grapes')][conditions].mean()))
    print("Watermelon : {0:.2f}".format(data[data['label'] == 'watermelon'][conditions].mean()))
    print("Kidney Beans: {0:.2f}".format(data[(data['label'] == 'kidneybeans')][conditions].mean()))
    print("Mung Beans : {0:.2f}".format(data[data['label'] == 'mungbean'][conditions].mean()))
    print("Oranges : {0:.2f}".format(data[(data['label'] == 'orange')][conditions].mean()))
    print("Chick Peas : {0:.2f}".format(data[data['label'] == 'chickpea'][conditions].mean()))
    print("Lentils : {0:.2f}".format(data[(data['label'] == 'lentil')][conditions].mean()))
    print("Cotton : {0:.2f}".format(data[data['label'] == 'cotton'][conditions].mean()))
    print("Maize : {0:.2f}".format(data[(data['label'] == 'maize')][conditions].mean()))
    print("Moth Beans : {0:.2f}".format(data[data['label'] == 'mothbeans'][conditions].mean()))
    print("Pigeon Peas : {0:.2f}".format(data[(data['label'] == 'pigeonpeas')][conditions].mean()))
    print("Mango : {0:.2f}".format(data[data['label'] == 'mango'][conditions].mean()))
    print("Pomegranate : {0:.2f}".format(data[(data['label'] == 'pomegranate')][conditions].mean()))
```



Regards

Yatin Sengar

AMITY UNIVERSITY
UTTAR PRADESH
AMITY INSTITUE OF INFORMATION TECHNOLOGY

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

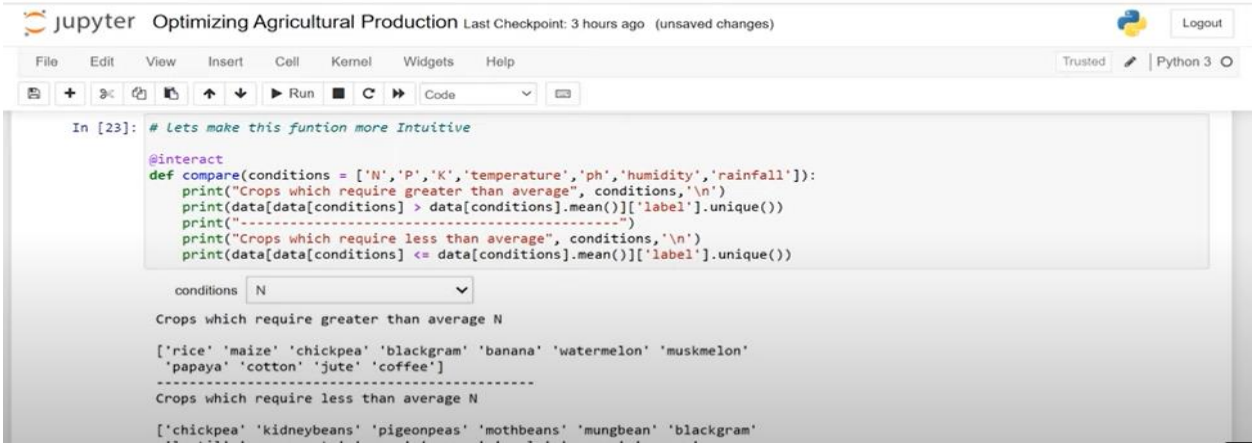
Faculty Guide- Dr. Rashmi Vasishth

Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Check crops which is grown in average or Above average conditions take N as greater than



Regards

Yatin Sengar

AMITY UNIVERSITY
UTTAR PRADESH
AMITY INSTITUE OF INFORMATION TECHNOLOGY

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

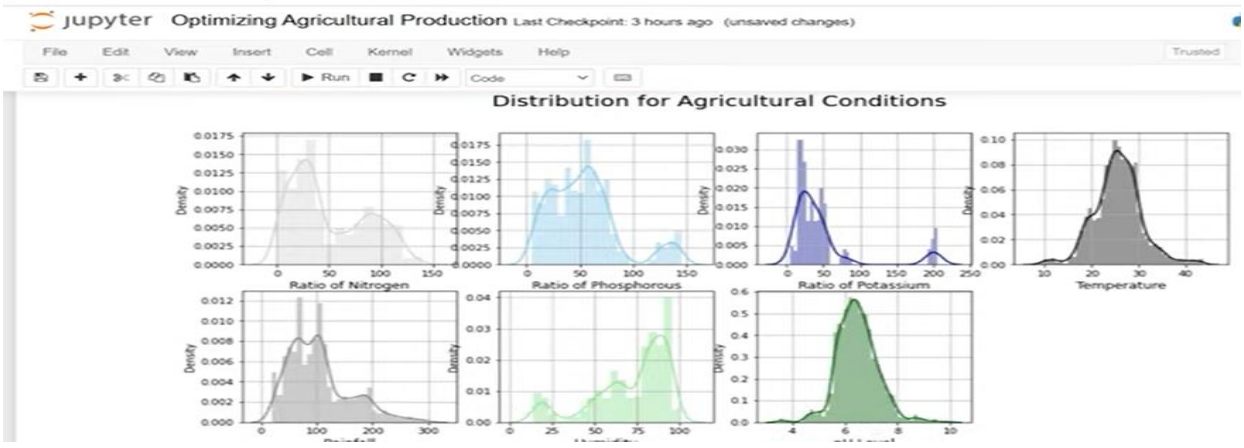
Faculty Guide- Dr. Rashmi Vasishth

Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Find out the pattern of potassium, nitrogen & temperature needed for crop grown



Regards

Yatin Sengar

AMITY UNIVERSITY
UTTAR PRADESH
AMITY INSTITUE OF INFORMATION TECHNOLOGY

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

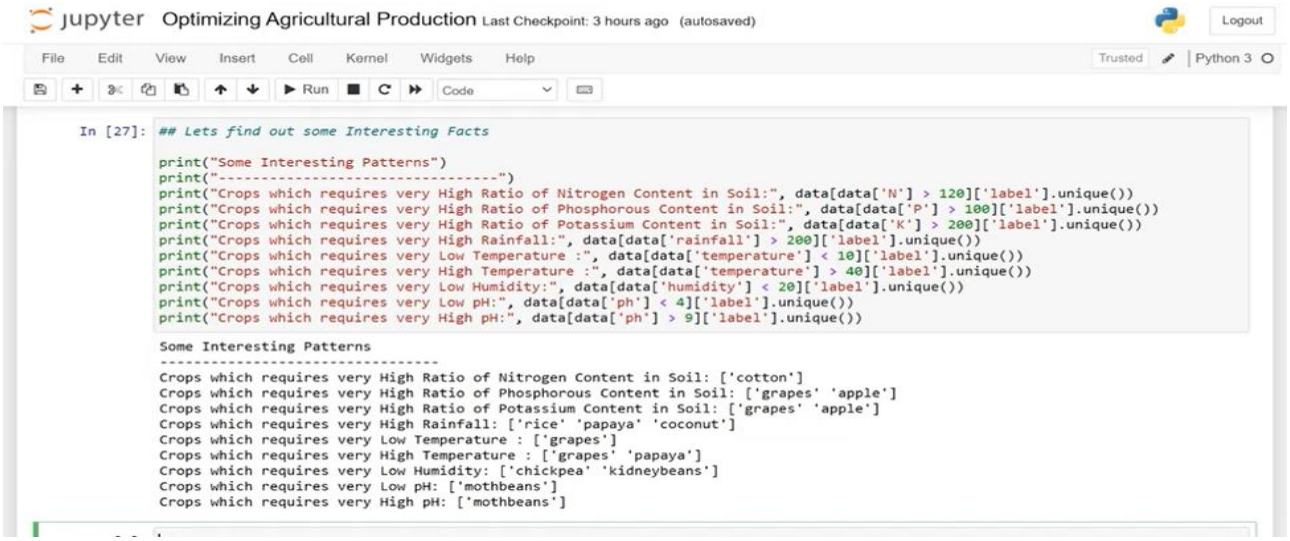
Faculty Guide- Dr. Rashmi Vasishth

Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Interesting facts:-



AMITY UNIVERSITY
UTTAR PRADESH
AMITY INSTITUE OF INFORMATION TECHNOLOGY

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

Faculty Guide- Dr. Rashmi Vasishth

Project Title:-

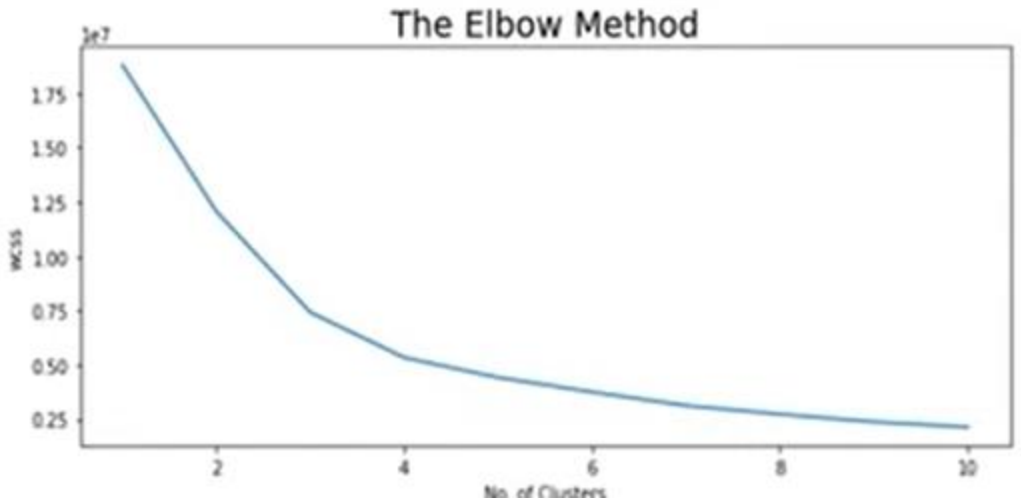
Data analysis for agriculture production engine

Summary of This Week:-

Plot no. of clusters with error using elbow method.

```
In [30]: from sklearn.cluster import KMeans

         # removing the labels column
         x = data.drop(['label'], axis=1)

         # selecting all the values of the data
         x = x.values

         # checking the shape
         print(x.shape)

         (2200, 7)
```

```
# lets plot the results
plt.plot(range(1, 11), wcss)
plt.title('The Elbow Method', fontsize = 20)
plt.xlabel('No. of Clusters')
plt.ylabel('wcss')
plt.show()
```



Regards

Yatin Sengar

AMITY UNIVERSITY
UTTAR PRADESH
AMITY INSTITUE OF INFORMATION TECHNOLOGY

Program – BCA

Batch – 2020-2023

Enrollment No.- A1004820014

Name- Yatin Sengar

Faculty Guide- Dr. Rashmi Vasishth

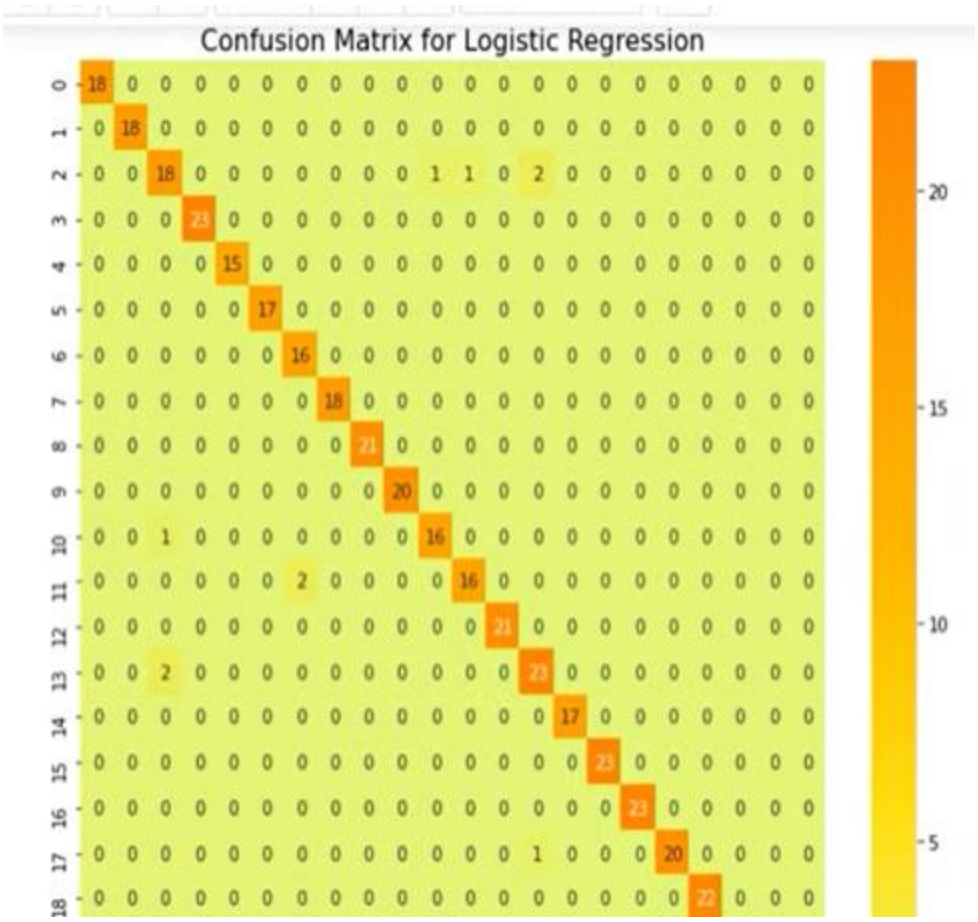
Project Title:-

Data analysis for agriculture production engine

Summary of This Week:-

Create predictive model using logistic regression.

In [37]:
```
# lets evaluate the Model Performance
from sklearn.metrics import confusion_matrix

# lets print the Confusion matrix first
plt.rcParams['figure.figsize'] = (10, 10)
cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot = True, cmap = 'Wistia')
plt.title('Confusion Matrix for Logistic Regression', fontsize = 15)
plt.show()
```



Regards

Yatin Sengar