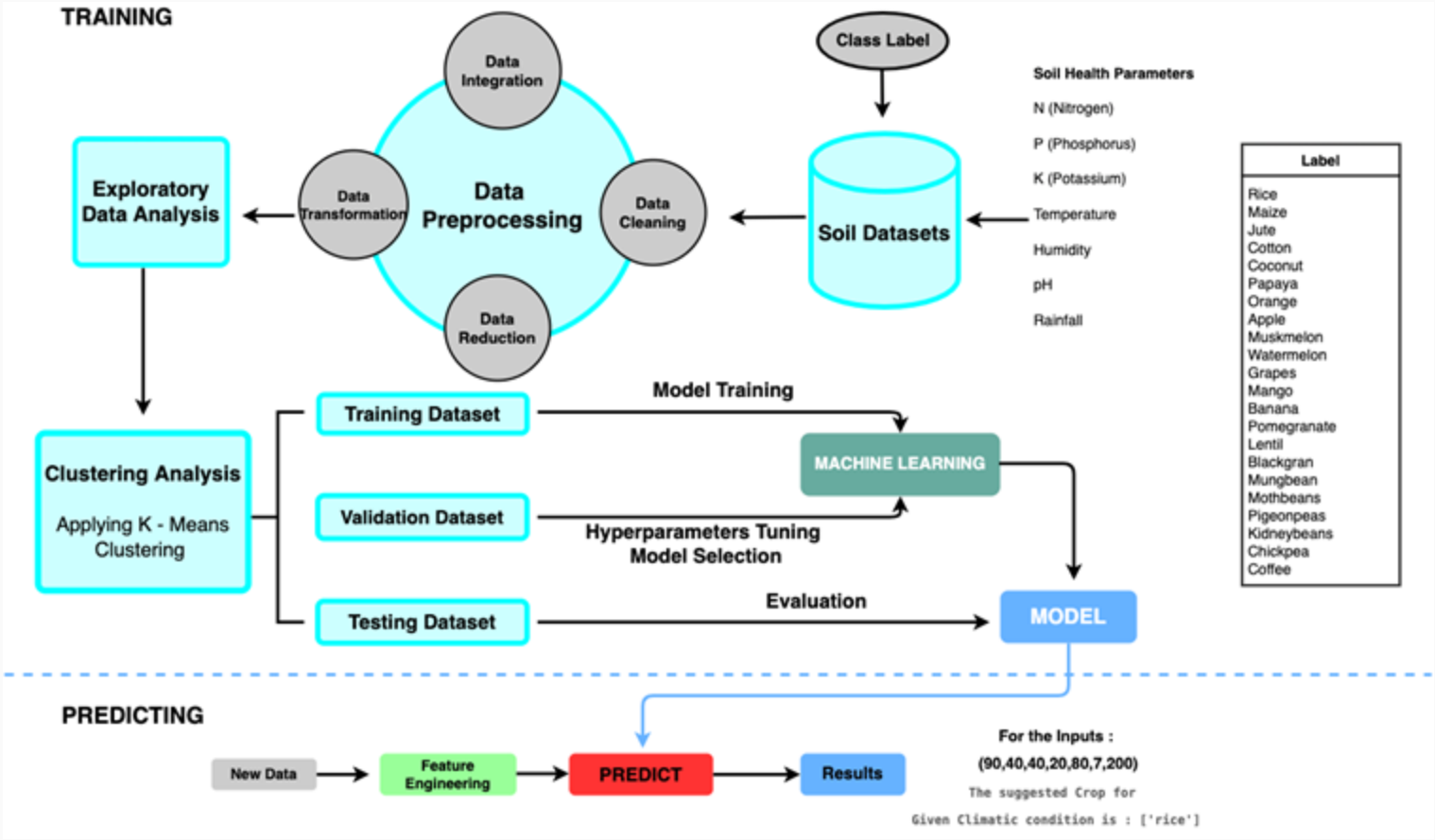
Crop Prediction Based on Soil Health Parameters using Machine Learning Techniques

Meity Project

CONTENTS

PROPOSED ARCHITECTURE



INTRODUCTION

- In general, agriculture is the backbone of India and also plays an important role in Indian economy by providing a certain percentage of domestic product to ensure the food security.
- But now-a-days, food production and prediction is getting depleted due to unnatural climatic changes, which will adversely affect the economy of farmers by getting a poor yield and also help the farmers to remain less familiar in forecasting the future crops.
- This project work helps the beginner farmer in such a way to guide them for sowing the reasonable crops by deploying machine learning, one of the advanced technologies in crop prediction.
- A web-based platform has been developed and deployed on the AWS server. The users are encouraged to enter parameters like N, P, K, Temperature, Humidity, pH, and rainfall will be taken automatically in this application in order to start the prediction process.

SOIL HEALTH

Soil health, also referred to as soil quality, is defined as the continued capacity of soil to function as a vital living ecosystem that sustains plants, animals, and humans.

Soil…

Covers the earth's surface can support plant life is unconsolidated material containing water, air space, organic matter, plant roots and myriad living organisms. Parent materials, climate, related organisms. Relief act together through time space to produce a continuum of different soils within the soil.

Health…

A condition in which vital functions are performed normally or properly.

Soil Functions…

Produce biomass (food, fiber, energy) Regulate water flow and storage Filter, biffer, and transform matter storehouse minerals, organic matter and myriad organisms. Supports for plants and structures. Provide raw materials(sand, silt and clay) Provide clues to past climates, vegetation, ecology and civilizations.

SOIL HEALTH PARAMETERS

Soil Health Testing involves identifying indicators of soil functions that can be measured.

- **Chemical Parameters :** The chemical parameters of soil are the levels and availability of nutritional minerals elements for the plants and the chemical parameters of soil in connection with there restoration or availability.
- **Physical Parameters :** The physical parameters of soil include texture, structure, porosity, density, consistence, aggregate stability, and temperature.
- **Land Use Parameters :** Th Land Use Parameters help devices strategies for monitoring microbial diversity and soil functioning for end users and policy makers.
- **Soil Color Parameters :** The Soil Color is influenced by it's minerals composition as well as water and organic contents.

DATA SET COLLECTION

- Data is composed from a different source and optimized for data sets. And the data is used to evaluate descriptively.
- Several abstract online outlets, like Kaggle, Google weather forestation and data government, provide the data for up to 10 years in series.
- The data sets such as soil nature, climatic conditions and seed data are used for the crop prediction and better crop yields.
- For this particular project the data set has been collected from Kaggle which consists of more than two thousands row of data having different attributes such has N, P, K, temperature, humidity, pH, rainfall, label.

ANALYZING THE DATA

- Analyzing the data is considered as a significant step machine learning phase.
- Data Analysis is the process of systematically applying statistical and/or logical techniques to describe and illustrate, condense and recap, and evaluate data. According to Shamoo and Resnik (2003) various analytic procedures “provide a way of drawing inductive inferences from data and distinguishing the signal (the phenomenon of interest) from the noise (statistical fluctuations) present in the data”..
- An essential component of ensuring data integrity is the accurate and appropriate analysis of research findings. Improper statistical analyses distort scientific findings, mislead casual readers (Shepard, 2002), and may negatively influence the public perception of research. Integrity issues are just as relevant to analysis of non-statistical data as well.

DATA WRANGLING

Data wrangling is the process of cleaning and unifying messy and complex data sets for easy access and analysis.

With the amount of data and data sources rapidly growing and expanding, it is getting increasingly essential for large amounts of available data to be organized for analysis. This process typically includes manually converting and mapping data from one raw form into another format to allow for more convenient consumption and organization of the data.

Goals of Data Wrangling

- Reveal a “deeper intelligence” by gathering data from multiple sources.
- Provide accurate, actionable data on the hands of business analysis in a timely matter
- Reduce the time spent collecting and organizing unruly data before it can be utilized
- Enable data scientists and analysts to focus on the analysis of data, rather than the wrangling.
- Derive better decision-making skills leaders in an organization.

EXPLORATORY DATA ANALYSIS

	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
...	...	...	...	...	...	...	...	...
2195	107	34	32	26.774637	66.413269	6.780064	177.774507	coffee
2196	99	15	27	27.417112	56.636362	6.086922	127.924610	coffee
2197	118	33	30	24.131797	67.225123	6.362608	173.322839	coffee
2198	117	32	34	26.272418	52.127394	6.758793	127.175293	coffee
2199	104	18	30	23.603016	60.396475	6.779833	140.937041	coffee

2200 rows × 8 columns

EXPLORATORY DATA ANALYSIS

```
#Shape of dataset
print("Shape of the dataset :",data.shape)
```

✓ 0.7s

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

```
#Checking missing values
data.isnull().sum()
```

✓ 0.8s

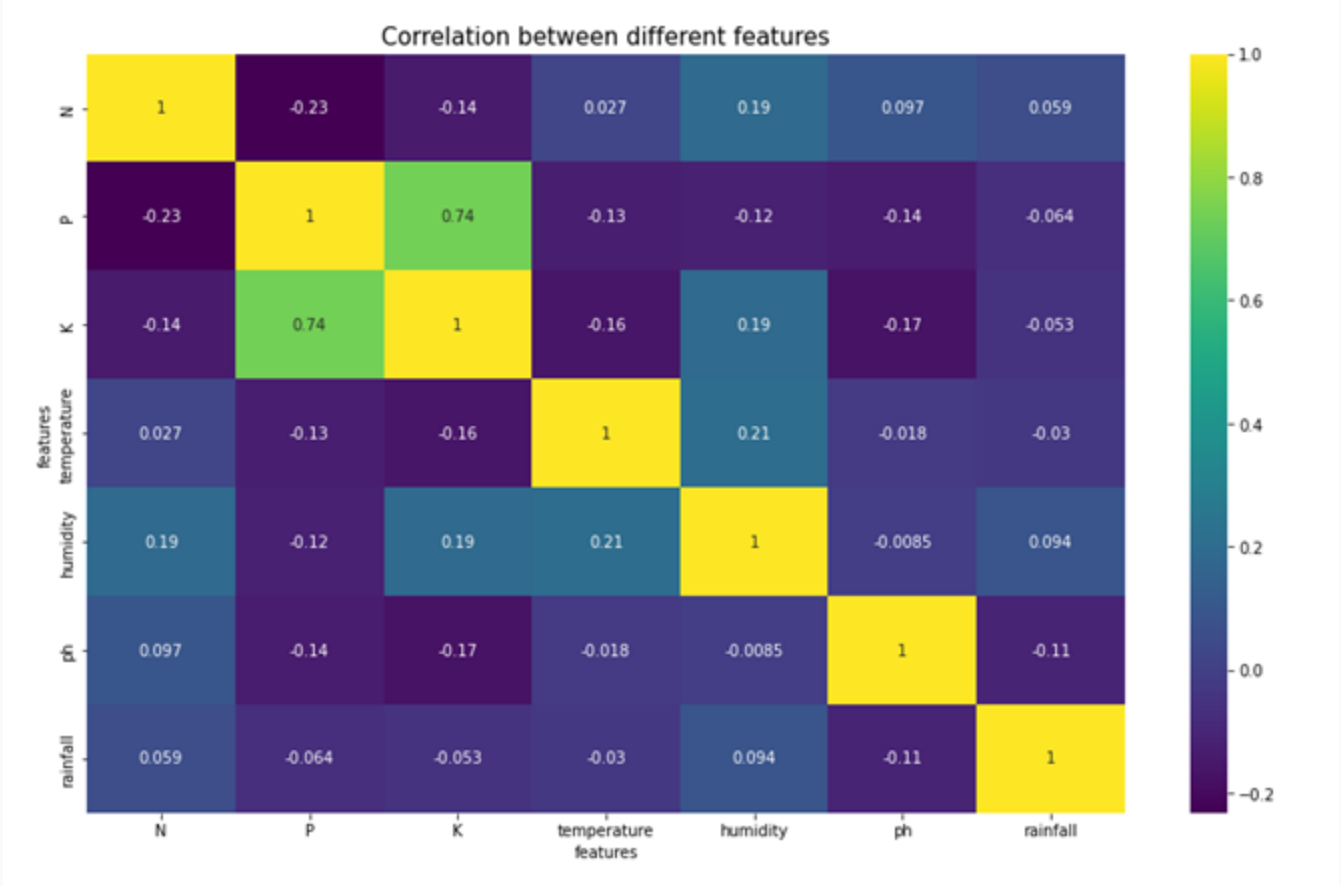
```
N              0
P              0
K              0
temperature    0
humidity       0
ph             0
rainfall       0
label          0
dtype: int64
```

```
#Checking Crops present in Dataset
data['label'].value_counts()
```

✓ 0.3s

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

EXPLORATORY DATA ANALYSIS



EXPLORATORY DATA ANALYSIS

Checking the Statistics of 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 temperature in Celsius : {0: .2f}".format(data['temperature'].mean()))
print("Average Relative Humidity in % is : {0: .2f}".format(data['humidity'].mean()))
print("Average pH value of the soil : {0: .2f}".format(data['ph'].mean()))
print("Average Rain fall in mm : {0: .2f}".format(data['rainfall'].mean()))
```

[11] ✓ 0.4s

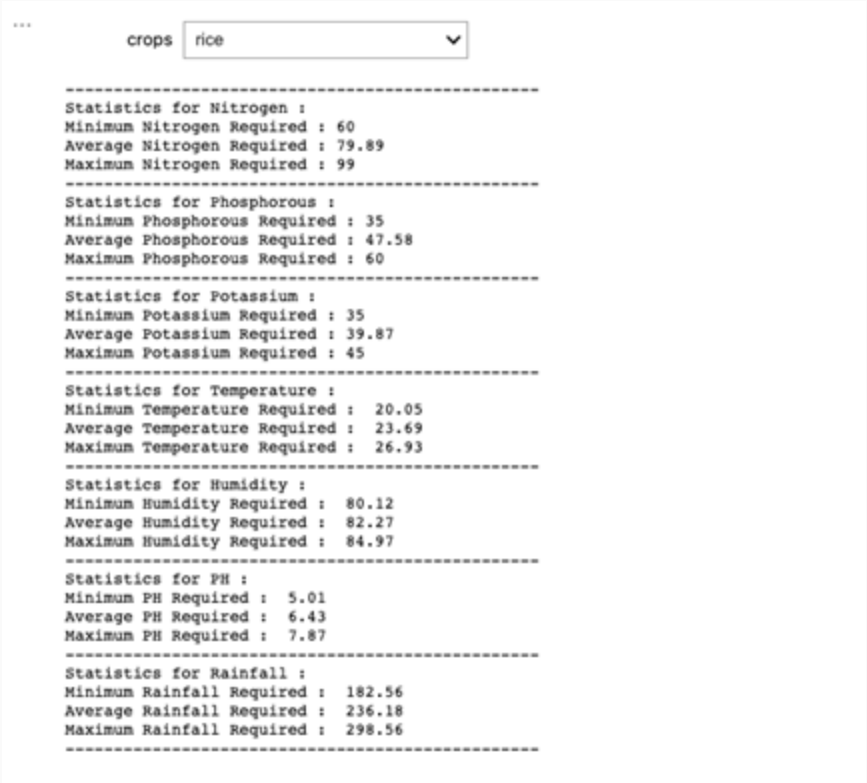
```
Average Ratio of nitrogen in the soil :  50.55
Average Ratio of Phosphorous in the soil :  53.36
Average Ratio of Potassium in the soil :  48.15
Average temperature in Celsius :  25.62
Average Relative Humidity in % is :  71.48
Average pH value of the soil :  6.47
Average Rain fall in mm :  103.46
```

1. Above shows that nitrogen,Phosphorous and potassium should be around 50%
2. Temperature should be around 25°C and Humidity around 70%
3. Rain fall should be around 100mm and PH should be arount 7

EXPLORATORY DATA ANALYSIS

Checking the Statistics For Each Crops :

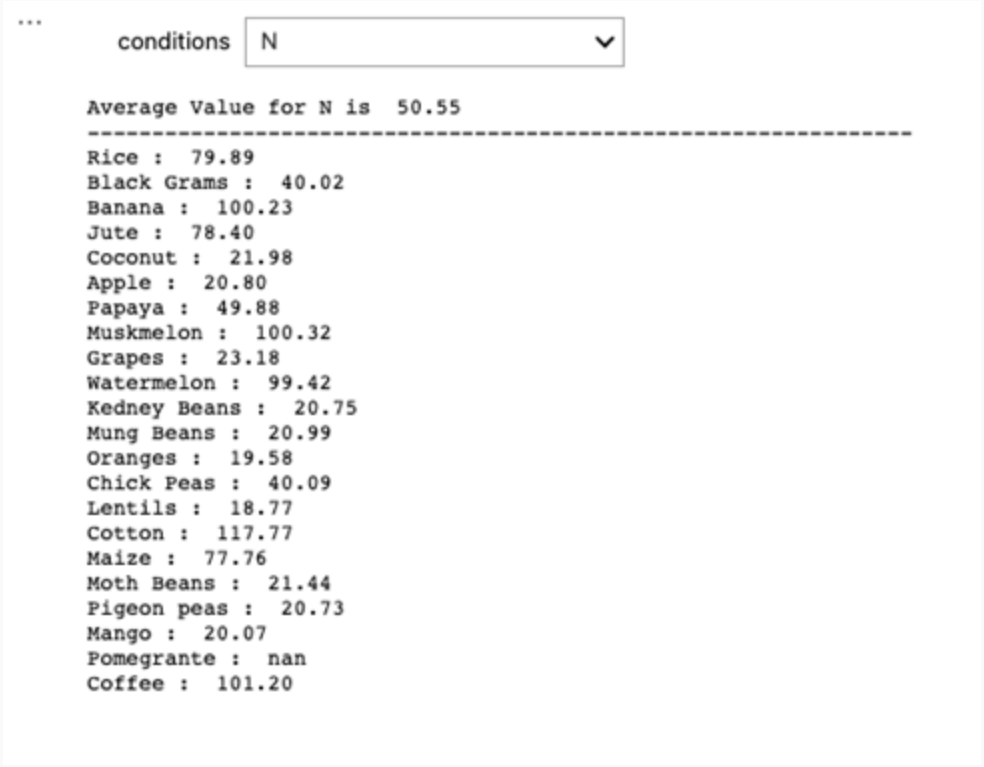
We can find statistics of any other crop by selecting on drop down list.



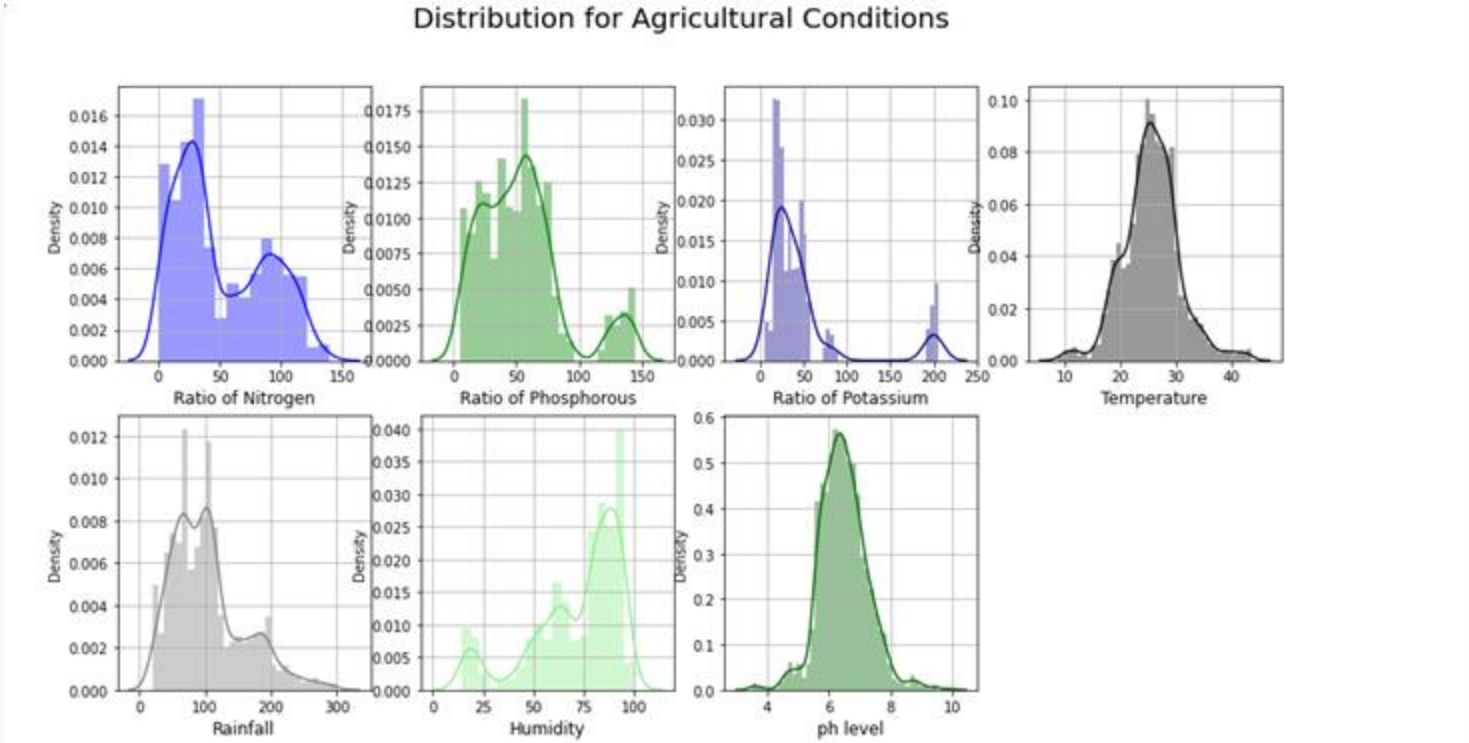
EXPLORATORY DATA ANALYSIS

Average requirements For Each Crops With Average Conditions :

We can find rainfall, temperature, humidity, Nitrogen, Phosphorus and pH value of every individual crop by selecting on the drop down menu.



EXPLORATORY DATA ANALYSIS



Above Graph shows us many hidden patterns like many crops need Phosphorous and Potassium at very high level.

EXPLORATORY DATA ANALYSIS : HIDDEN PATTERNS

Crops requires more Rainfall, Temperature, Humidity, Nitrogen, Potassium, Phosphorus and ph value VS Crops requires Less.

```
@interact
def compare(conditions = ['N','P','K','temperature','ph','humidity','rainfall']):
    print("Crops which require greater than average",conditions,'\n')
    print(data[data[conditions] > data[conditions].mean()]['label'].unique())
    print("----------------------------------------------------")
    print("Crops which require less than average",conditions,'\n')
    print(data[data[conditions] <= data[conditions].mean()]['label'].unique())
```

[15] ✓ 0.1s

conditions N

```
Crops which require greater than average N

['rice' 'maize' 'chickpea' 'blackgram' 'banana' 'watermelon' 'muskmelon'
 'papaya' 'cotton' 'jute' 'coffee']
----------------------------------------------------
Crops which require less than average N

['chickpea' 'kidneybeans' 'pigeonpeas' 'mothbeans' 'mungbean' 'blackgram'
 'lentil' 'pomegranate' 'mango' 'grapes' 'apple' 'orange' 'papaya'
 'coconut']
```

Crops which Require more Rainfall vs Crops which Require Less Rainfall

```
print("Crops which requires very High rainfall:",data[data['rainfall'] > 200]['label'].unique())
print("Crops which requires very Low rainfall:",data[data['rainfall'] < 40]['label'].unique())
```

[16] ✓ 0.4s

```
Crops which requires very High rainfall: ['rice' 'papaya' 'coconut']
Crops which requires very Low rainfall: ['mothbeans' 'mungbean' 'lentil' 'muskmelon']
```

[17] ✓ 0.8s

```
Crops which requires very High ratio of Nitrogen Content in soil : ['cotton']
Crops which requires very High ratio of Phosphorous Content in soil : ['grapes' 'apple']
Crops which requires very High ratio of Potassium Content in soil : ['grapes' 'apple']
Crops which requires very High Rainfall : ['rice' 'papaya' 'coconut']
Crops which requires very Low Rainfall: ['mothbeans' 'mungbean' 'lentil' 'muskmelon']
Crops which requires very Low Temperature : ['grapes']
Crops which requires very High Temperature : ['grapes' 'papaya']
Crops which requires very Low Humidity : ['chickpea' 'kidneybeans']
Crops which requires very Low pH : ['mothbeans']
Crops which requires very High pH : ['chickpea' 'mothbeans']
```

[18] ✓ 0.4s

```
Summer Crops
['pigeonpeas' 'mothbeans' 'blackgram' 'mango' 'grapes' 'orange' 'papaya']
----------------------------------------------------------------------------
Winter Crops
['maize' 'pigeonpeas' 'lentil' 'pomegranate' 'grapes' 'orange']
----------------------------------------------------------------------------
Rainy Crops
['rice' 'papaya' 'coconut']
```

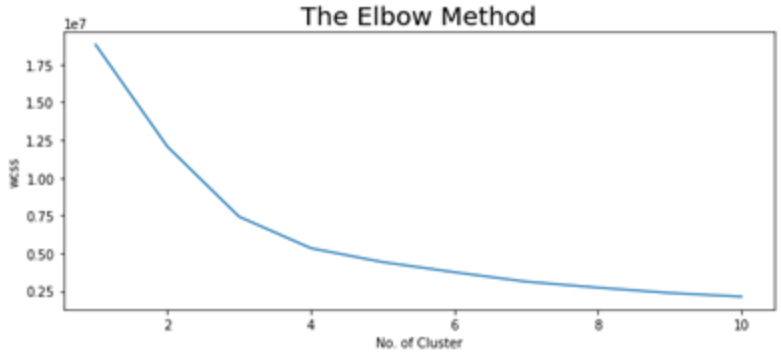
CLUSTERING ANALYSIS

```
#Determining Optimum number of Clusters within Dataset by using K-means Clustering
plt.rcParams['figure.figsize'] = (10,4)

wcss = []
for i in range(1,11):
    km = KMeans(n_clusters = i,init = 'k-means++',max_iter = 300, n_init = 10, random_state = 0)
    km.fit(x)
    wcss.append(km.inertia_)

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

[20] ✓ 3.1s



By using Elbow method,we concluded that dataset consists of four clusters.

CLUSTERING ANALYSIS

```
#Implementing K-means Algorithm to perform Clustering Analysis
km = KMeans(n_clusters = 4,init = 'k-means++',max_iter = 300, n_init = 10, random_state = 0)
y_means = km.fit_predict(x)

#Lets find out 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 Clusters of Each crop
print("Checking 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 Forth Cluster:", z[z['cluster'] == 3]['label'].unique())
```

[21] ✓ 0.1s

```
Checking results after applying K-means Clustering Analysis

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

PREDICTIVE ANALYSIS

```
#Splitting 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)
```

[22] ✓ 0.4s

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

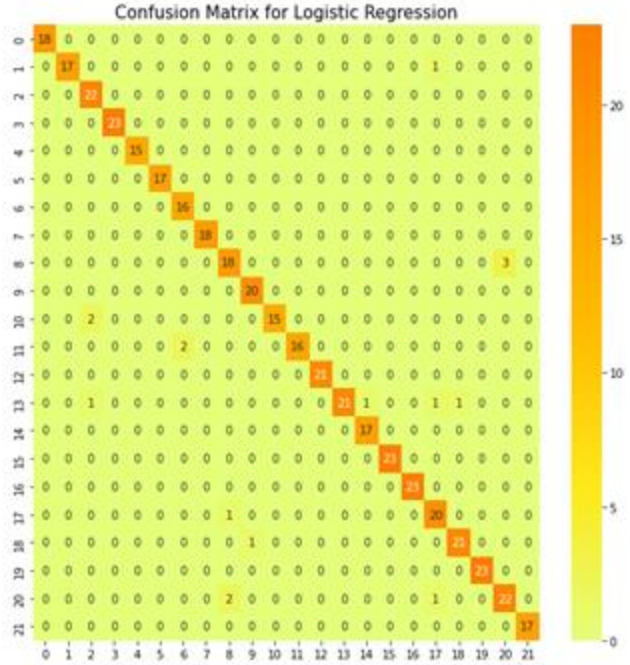
```
#Training and Testing Sets for Validation of Results
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.2,random_state = 0)

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)
```

[23] ✓ 0.1s

```
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,)
```

[25] ✓ 2.9s



ACCURACY CHECK

	precision	recall	f1-score	support
apple	1.00	1.00	1.00	18
banana	1.00	0.94	0.97	18
blackgram	0.88	1.00	0.94	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	0.86	0.86	0.86	21
kidneybeans	0.95	1.00	0.98	20
lentil	1.00	0.88	0.94	17
maize	1.00	0.89	0.94	18
mango	1.00	1.00	1.00	21
mothbeans	1.00	0.84	0.91	25
mungbean	0.94	1.00	0.97	17
muskmelon	1.00	1.00	1.00	23
orange	1.00	1.00	1.00	23
papaya	0.87	0.95	0.91	21
pigeonpeas	0.95	0.95	0.95	22
pomegranate	1.00	1.00	1.00	23
rice	0.88	0.88	0.88	25
watermelon	1.00	1.00	1.00	17
...				
accuracy			0.96	440
macro avg	0.96	0.96	0.96	440
weighted avg	0.96	0.96	0.96	440

As we can see that for all 22 classes we have high precision and recall values which shows that our Predictive model works very well.

OUR MODEL IS READY, N WE CAN USE MODEL TO PREDICT CROP NAME

```
data.head()
```

[27] ✓ 0.1s

	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

```
prediction = model.predict((np.array([[90,
                                       40,
                                       40,
                                       20,
                                       80,
                                       7,
                                       200]])))
print("The suggested Crop for Given Climatic condition is :", prediction)
```

[28] ✓ 0.2s

```
The suggested Crop for Given Climatic condition is : ['rice']
```

Performance of our Predictive model is excellent, Now we are going to change our input values.

OUR MODEL IS READY, NOW WE CAN USE MODEL TO PREDICT CROP NAME

Performance of our Predictive model is excellent, Now we are going to change our input values.

```
data.tail()
```

[29] ✓ 0.1s

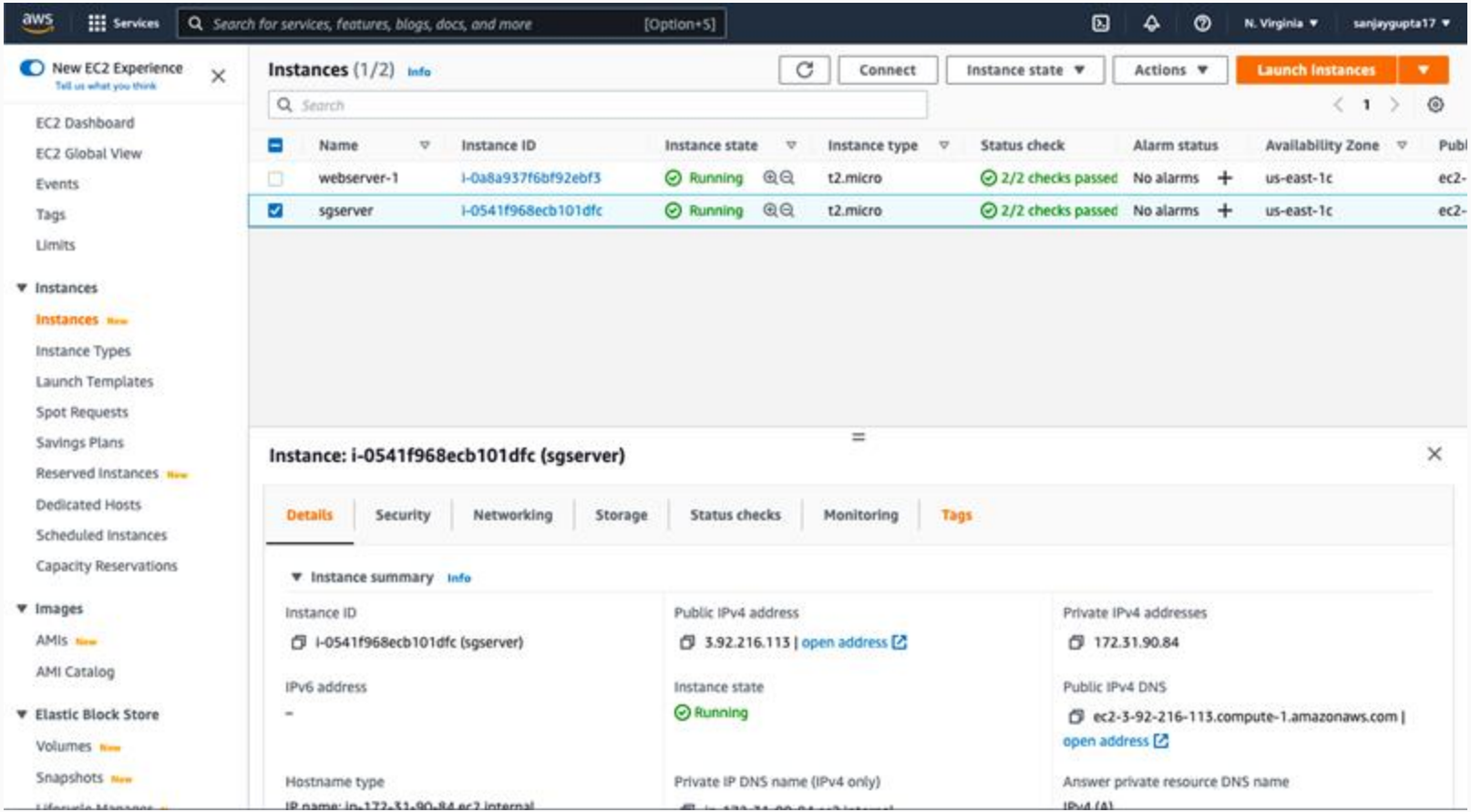
	N	P	K	temperature	humidity	ph	rainfall	label
2195	107	34	32	26.774637	66.413269	6.780064	177.774507	coffee
2196	99	15	27	27.417112	56.636362	6.086922	127.924610	coffee
2197	118	33	30	24.131797	67.225123	6.362608	173.322839	coffee
2198	117	32	34	26.272418	52.127394	6.758793	127.175293	coffee
2199	104	18	30	23.603016	60.396475	6.779833	140.937041	coffee

```
prediction = model.predict((np.array([[107,
                                       34,
                                       32,
                                       26,
                                       66,
                                       7,
                                       180]])))
print("The suggested Crop for Given Climatic condition is :", prediction)
```

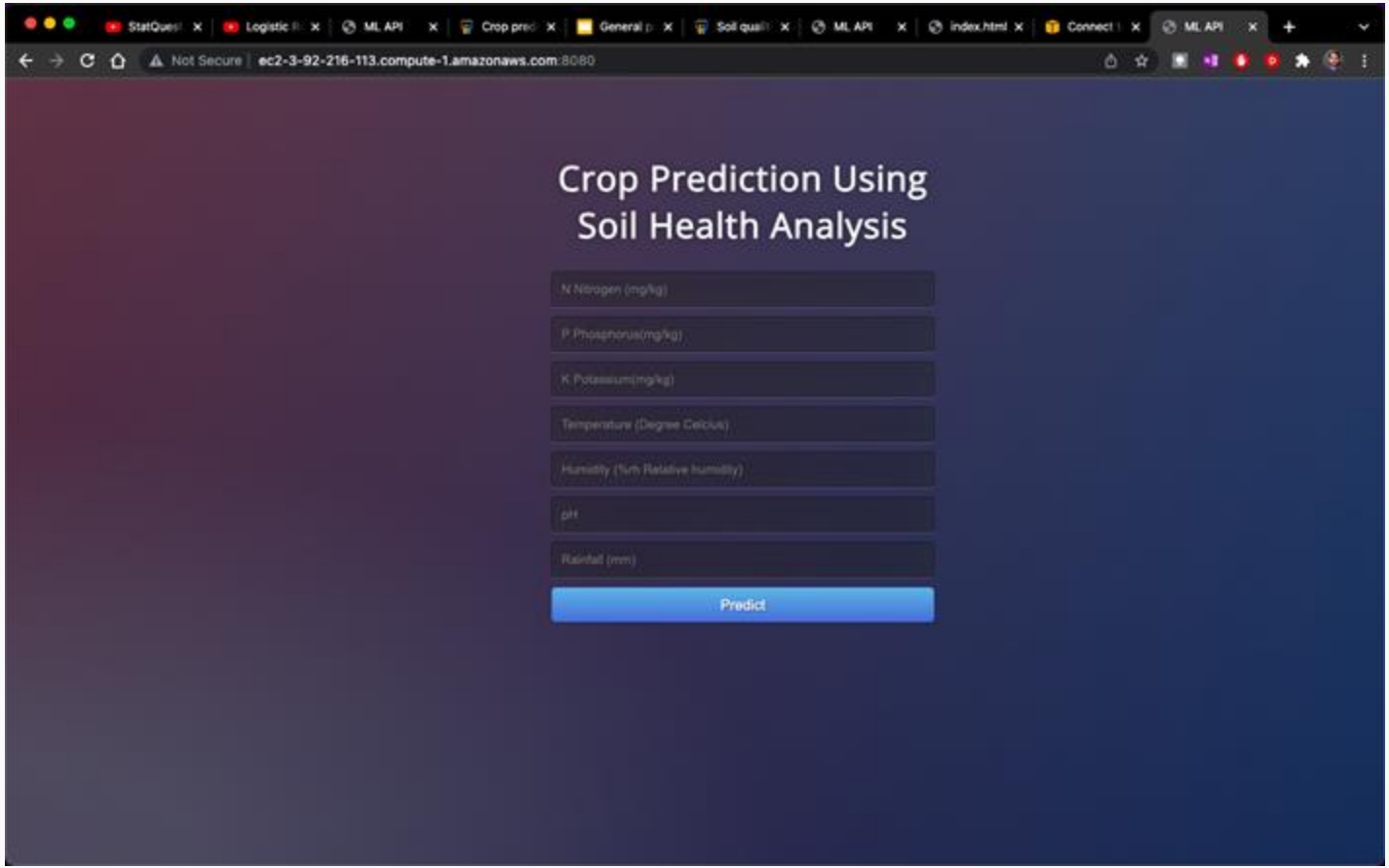
[30] ✓ 0.1s

```
The suggested Crop for Given Climatic condition is : ['coffee']
```

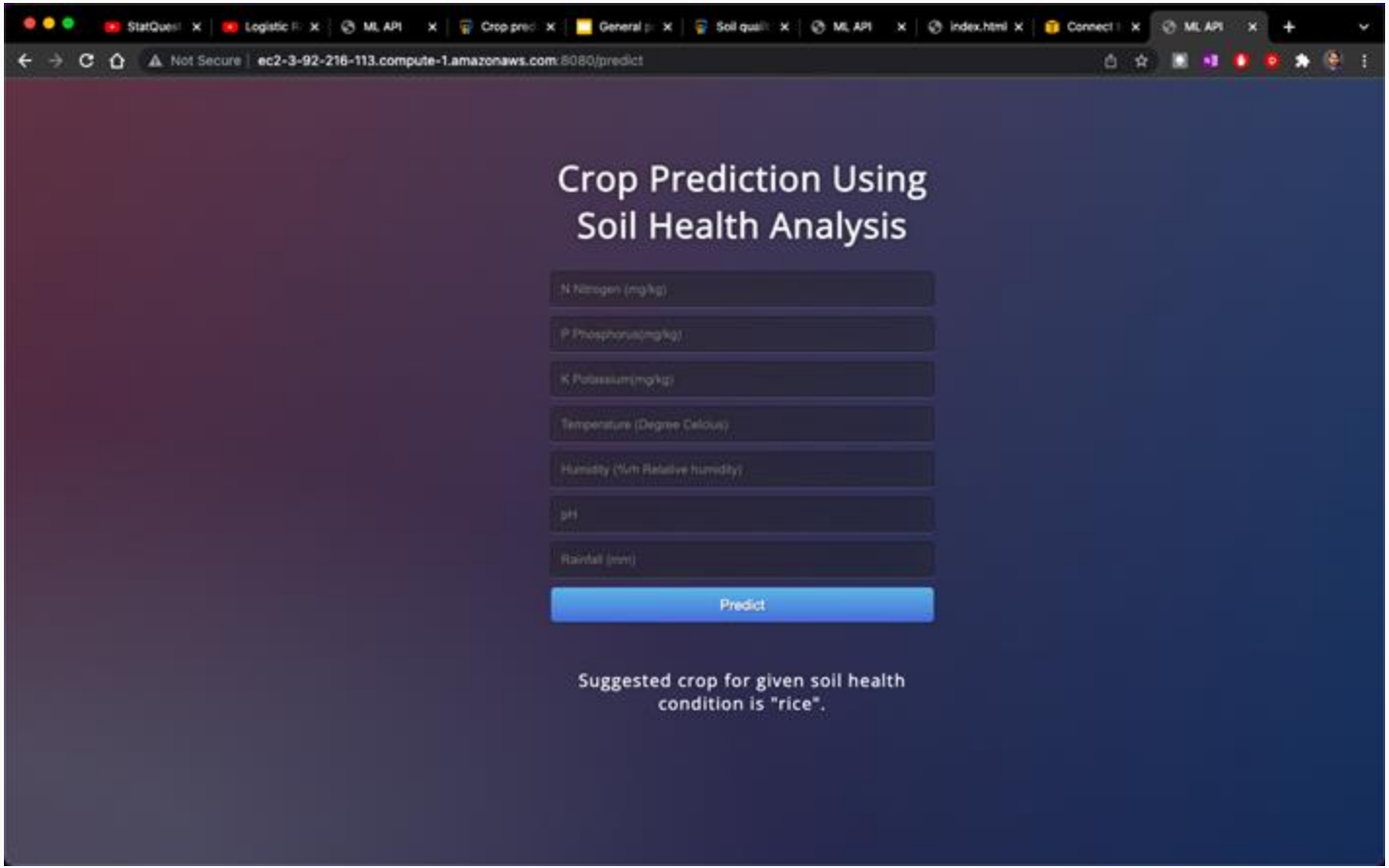
DEPLOYMENT OF MODEL ON EC2 SERVER



MODEL WEBSITE

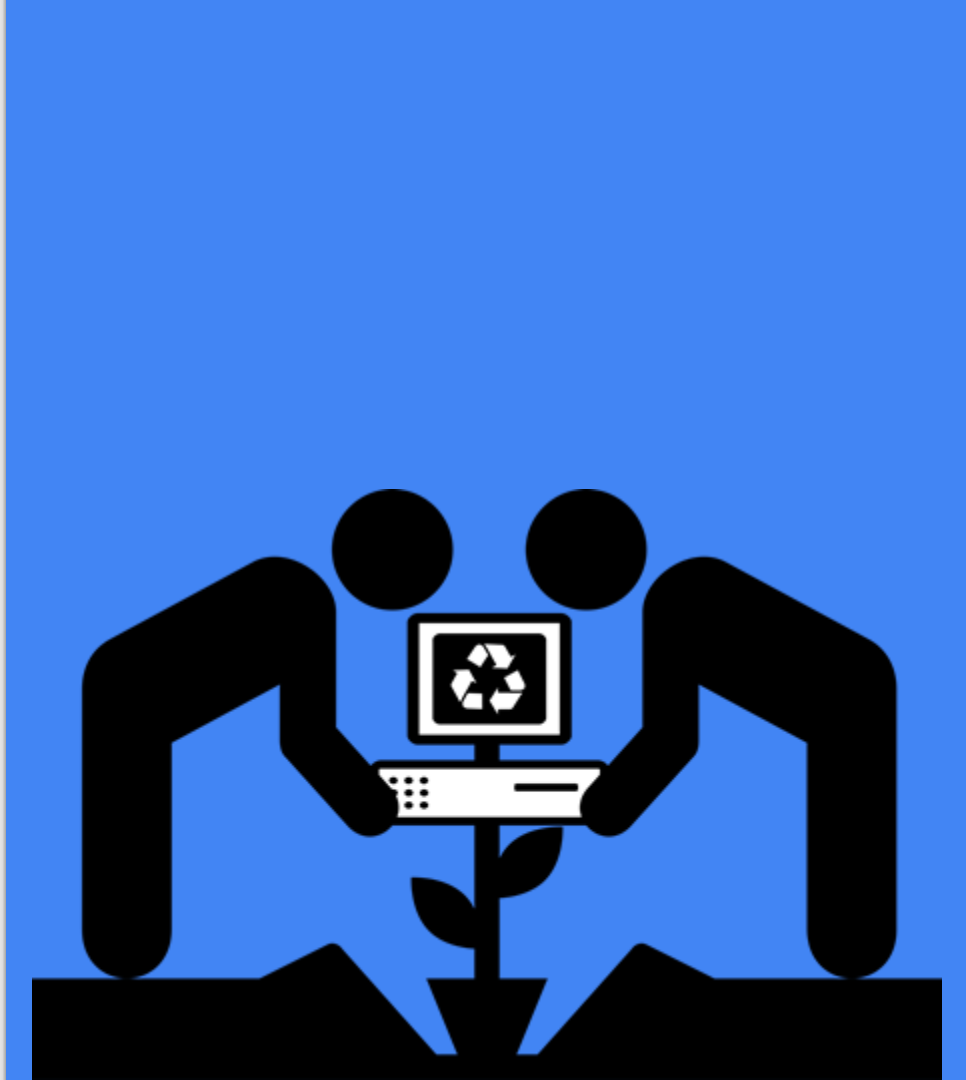


RESULTS


Not Secure
ec2-3-92-216-113.compute-1.amazonaws.com:8080/predict
Crop Prediction Using Soil Health Analysis
N Nitrogen (mg/kg)
P Phosphorus(mg/kg)
K Potassium(mg/kg)
Temperature (Degree Celcius)
Humidity (%rh Relative humidity)
pH
Rainfall (mm)
Predict
Suggested crop for given soil health condition is "rice".

CONCLUSION

- The proposed supervised machine learning using Logistic Regression algorithm is developed to predict the crop at high accuracy. Thus, the crop is predicted as an output for the given input parameter.
- This work may greatly help the needy farmers who have less knowledge in predicting the crops for developing a sustainable future.
- In future it may also extend to suggest the fertilizer, suitable guidelines for cropland and crops for the given input. In addition, source of sunlight and crop health are monitored at regular intervals and it is also taken into the account for achieving a better crop yield.



Thanks!