CROP RECOMMENDATION USING MACHINE LEARNING

By

RUPIKA K	**713521AM036**
SOORIYA R	**713521AM045**
SUKANTH K	**713521AM050**
PRENESH V K	**713521AM504**

Submitted to the

DEPARTMENT OF INFORMATION TECHNOLOGY

SNS COLLEGE OF TECHNOLOGY
(AN AUTONOMOUS INSTITUTION)

DEPARTMENT OF INFORMATION TECHNOLOGY

DECEMBER 2022

BONAFIDE CERTIFICATE

Certified that this Mini Project Report titled "**CROP RECOMMENDATION USING MACHINE LEARNING**" is the Bonafide record of work done by the students "**RUPIKA K(713521AM036), SOORYA R(713521AM045), SUKANTH K(713521AM050), PRENESH V K (713521AM504)**" had carried out the project work under my supervision during the academic year 2022-2023.

PROJECT GUIDE

Mr. K.V.S MOHAN
Assistant Professor,
Department of IT,
SNS College of Technology,
Coimbatore – 641 035.

DEAN(CSE & IT)

Dr.L.M NITHYA
Dean,
Department of IT,
SNS College of Technology,
Coimbatore – 641 035.

Submitted for the Mini Project -II examination held on..................

Internal Examiner

External Examiner

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

Agriculture and its allied sectors are undoubtedly the largest providers of livelihoods in rural India. The agriculture sector is also a significant contributor factor to the country's Gross Domestic Product (GDP). Blessing to the country is the overwhelming size of the agricultural sector.

However, regrettable is the yield per hectare of crops in comparison to international standards. This is one of the possible causes for a higher suicide rate among marginal farmers in India.

The user provides the temperature and humidity of the area & soil type as input. Machine learning algorithms allow choosing the most profitable crop list or predicting the crop yield for a user-selected crop. To predict the crop yield, selected Machine Learning algorithms such as Support Vector Machine (SVM), Decision Tree Classifier, Guassian Naïve Bayes, Random Forest (RF), Logistic Regression (LR) and XGBoost Classifier are used. Among them, the Random Forest showed the best results with 99.7% accuracy. Additionally, the system also suggests the best time to use the fertilizers to boost up the yield.

CHAPTER 1
EMPATHY

Agricultural Universities, Research Institutes have been generating ample technologies to improve the productivity and profitability of the farmers. However, knowledge gap is prevailing among farmers and those who have access to knowledge harvest better profits.

The main consideration in selecting the crops that are most suitable for smallholder production is of course the demands of the market - there is no point in producing something unless some one wants to buy it. However, among crops for which there is a sure demand some require agronomic practices or environmental controls which make them particularly suitable, or particularly unsuitable, for smallholder producers.

Export crops which are produced using agronomic techniques with which small farmers are already familiar (e.g. green beans, baby corn) are less likely to present difficulties than crops which are completely new to an area;

Crops which make full use of family labour and do not require large amounts of purchased inputs are more manageable for smallholders than crops which are best suited to mechanical cultivation and which need heavy applications of agro-chemicals;

Crops whose planting, weeding and harvesting dates come at periods of peak labour demand for other activities in the smallholder year (e.g. food crop planting, weeding or harvesting) are more difficult to fit into the smallholder farming system than crops whose peak labour needs come at an 'off season period' in the smallholder year.

1.1 CLIMATIC FACTORS

Climatic factors influence all aspects and stages of plant growth and hence affect agricultural productivity and stability of production. Their influence extends from the upper reaches of the atmosphere, in which spores and pollen are encountered, to the soil depth penetrated by the roots. Although water supply plays a dominant role in dryland agriculture, other climatic factors either enhance or reduce problems of water deficit, or have beneficial or negative effects on crop production.

Enormous amounts of climatic data are collected daily, in practically every country. These efforts involve considerable investment in personnel and capital; it would be an exercise in futility if these data were not put to use for increasing agricultural productivity.

Agricultural meteorology has been defined as the discipline that studies the relationship between agricultural production and climatic factors and attempts to enhance their favourable effects and/or attenuate their adverse effects.

Much information has accrued in this discipline in recent years as the result of a considerable research effort in the laboratory, in growth chambers and in the field, and practical application of this know-how is being developed.

Each climate factor has a range of optimum intensity which differs with species, and within species, with different growth stages. Beyond the limits of the optimum intensity, plant development is adversely affected, either by excess or deficiency intensity of the factor involved.

The resulting stress may cause physical or chemical changes, such as structural damage or destruction of enzymes that are vital to essential metabolic processes, or they may simply cause a slowing down of these functions, that return to normal when the stress ceases. According to the degree of intensity, the effects of a climate factor may therefore be favourable, neutral, harmful or lethal.

Temperature is a measure of intensity of heat energy. Most of the agricultural plants require temperature between 15 and 400C for growth. The minimum, maximum (above which crop growth ceases) and optimum temperature of individual's plant is called as cardinal temperature. The temperature of a place is largely determined by its distance from the equator (latitude) and altitude.

Germination, growth and development of crops are highly influenced by temperature. It affects leaf production, expansion and flowering.

- Physical and chemical processes within the plants are governed by air temperature.
- Diffusion rates of gases and liquids changes with temperature.
- Solubility of different substances in plant is dependent on temperature.

It is clear that agriculture sustains and defines our modern lives, but it is often disruptive of natural ecosystems. This is especially true for plant communities, animal populations, soil systems, and water resources. Understanding, evaluating, and balancing detrimental and beneficial agricultural disturbances of soil and water resources are essential tasks in human efforts to sustain and improve human well-being. Such knowledge influences our emerging ethics of sustainability and responsibility to human populations and ecosystems of the future.

Although agriculture is essential for human food and the stability of complex societies, almost all of our evolution has taken place in small, mobile, kin-based social groups, such as bands and tribes (Diamond 1999, Johanson & Edgar 2006). Before we became sedentary people dependent on agriculture, we were largely dependent on wild plant and animal foods, without managing soil and water resources for food production. Our social evolution has accelerated since the Agricultural Revolution and taken place synergistically with human biological evolution, as we have become dependent on domesticated plants and animals grown purposefully in highly managed, soil-water systems. Unsecured loans are monetary loans that are not secured against the borrower's assets. These may be available from financial institutions under many different guises or marketing packages:

The early use of fire to flush out wild game and to clear forested land provided the first major anthropogenic influence on the environment. By burning native vegetation, early humans were able to gain access to herbivores grazing on the savanna and in nearby woodlands, and to suppress the growth of less desirable plant species for those easier to forage and eat (Pyne 2001, Wrangham 2009). These and other factors (e.g., population pressures, climate change, encouraging/protecting desirable plants), help to lay the groundwork for the Agricultural Revolution and caused a dramatic shift in the interactions between humans and the earth. The shift from hunter-gatherer societies to an agrarian way of life drastically changed the course of human history and irreversibly altered natural nutrient cycling within soils. When humans sowed the first crop seeds at the dawn of the Neolithic Period, the soil provided plant-essential nutrients and served as the foundation for human agriculture.

As one might expect, contributions from each mineral size fraction help to provide the physical framework for a productive soil. Loamy-textured soils are commonly described as medium textured with functionally-equal contributions of sand, silt, and clay. These medium-textured soils are often considered ideal for agriculture as they are easily cultivated by farmers and can be highly productive for crop growth.

1.2MAJOR PROBLEMS

By empathizing the crop production in agriculture, there are some noticeable problems in the desired field

- The climate is not constant over the year
- Farmers unable to predict which crop suits for that period of time correctly

CHAPTER 2
DEFINE

2.1 SOIL CONTENT

Soil is the loose surface material that covers most land. It consists of inorganic particles and organic matter. Soil provides the structural support to plants used in agriculture and is also their source of water and nutrients. Soils vary greatly in their chemical and physical properties. Processes such as leaching, weathering and microbial activity combine to make a whole range of different soil types. Each type has particular strengths and weaknesses for agricultural production.

2.2 SOIL FERTILITY

Soil fertility is the ability of soil to sustain plant growth and optimize crop yield. This can be enhanced through organic and inorganic fertilizers to the soil. Nuclear techniques provide data that enhances soil fertility and crop production while minimizing the environmental impact. Advancing food security and environmental sustainability in farming systems requires an integrated soil fertility management approach that maximizes crop production while minimizing the mining of soil nutrient reserves and the degradation of the physical and chemical properties of soil that can lead to land degradation, including soil erosion. Such soil fertility management practices include the use of fertilizers, organic inputs, crop rotation with legumes and the use of improved germplasm, combined with the knowledge on how to adapt these practices to local conditions. The Joint FAO/IAEA Division assists Member States in developing and adopting nuclear-based technologies for improving soil fertility practices, thereby supporting the intensification of crop production and the preservation of natural resources.

2.3 TEMPERATURE

Regardless of how favourable light and moisture conditions may be, plant growth ceases when the air and leaf temperature drops below a certain minimum or exceeds a certain maximum value. Between these limits, there is an optimum temperature at which growth proceeds with greatest rapidity. These three temperature points are the cardinal temperatures for a given plant; the cardinal temperatures are known for most plant species, at least approximately. Cool-season crops (oats, rye, wheat, and barley) have low cardinal temperatures: minimum 32° to 41° F (0° to 5° C), optimum 77° to 88° F (25° τo 31° C), and maximum 88° τo 99° F (31° to 37° C). For hot-season crops, such as melons and sorghum, the span of cardinal temperatures is much higher. The cardinal temperatures may vary with stage of development. For example, cold treatment near 32° F (0° C) of germinated seeds before sowing can transform winter rye into the spring type; such treatment, called vernalization, has practical application in cold-climate plants.

The range of diurnal temperature variation is also important; the best net photosynthesis is related to a large diurnal temperature range, or high daytime and low nighttime temperatures. Knowledge of the difference between leaf and air temperatures aids farmers in adopting protective measures. In middle and high latitudes, frost often occurs before the air temperature drops to freezing; in summer, heat injury to plants might be much more serious than that suggested by the air temperature alone. Because of this factor, farmers in Taiwan shade the pineapple fruit to prevent heat damage.

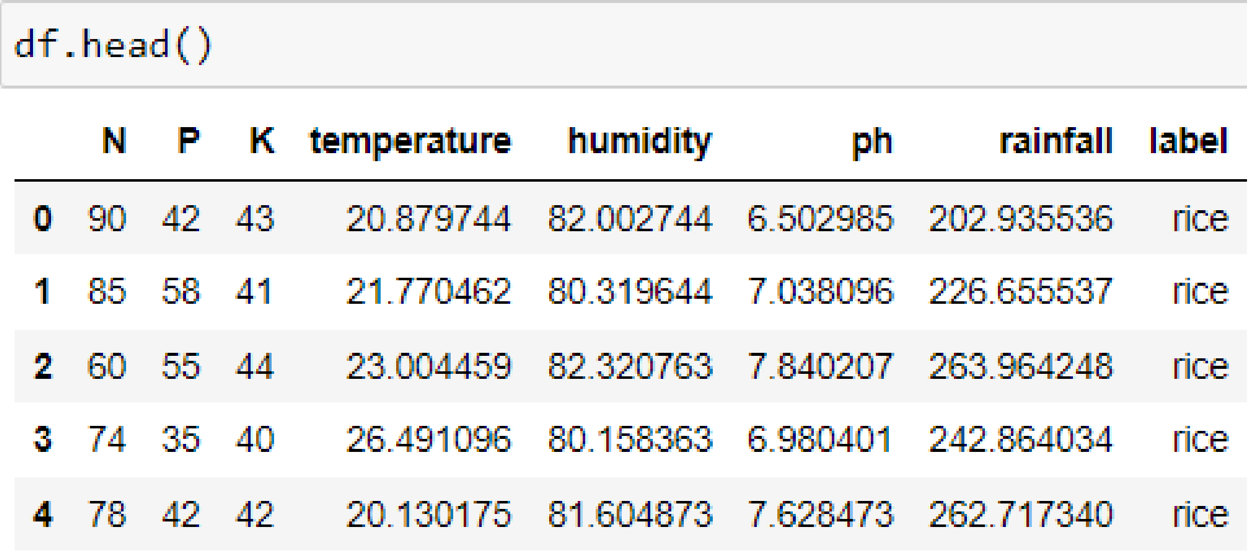
Soil temperature sometimes is of greater ecological significance to plant life than air temperature. Germination of seed, root function, rate of plant growth, and occurrence and severity of plant diseases all are affected by soil temperature. Since an unfavourable soil temperature during the growing season can retard or ruin a crop, techniques have been developed for modifying the temperature. The two most important methods are (1) regulation of the energy exchange and (2) altering the thermal properties of the ground. Incoming energy can be regulated by an insulation layer on or near the ground surface, such as paper, straw, plastic, or trees; the outgoing radiation can be reduced by insulation materials or by generating smoke or fog in the air. Thermal properties of the ground can be modified by cultivation or irrigation, increasing the soil’s ability to absorb radiation, or by varying the rate of evaporation. Mulching is a common technique for soil temperature control. Carbon black or white material can change the soil’s ability to absorb radiation. In the Soviet Union, for example, it was reported that 100 pounds of coal dust per acre (112 kilograms per hectare) caused a one-month advance in the maturity date of cotton.

2.4 RAINFALL

Rain is usually seen as a benefit to crops and fields, but there is an “ideal” amount of rainfall in any given growing season for most crops. If the average rainfall is much lower or higher than the ideal, it can lead to significant problems, from drowned crops to lower yields. If crops are too wet, they could also start to mold or catch a fungus. The soil can also start to collect bacteria, mold, and fungus, which can then be absorbed by the plant. While this isn’t as common in crops as it is in indoor plants, poor drainage and irrigation systems can lead to these types of growths taking control over your crops. Along with mold or a fungus, disease can also spread amongst your crops. Rainfall is also a good indicator of predicting common crop disease, as it can affect the spread of disease. Rain can spread pathogens, pests, and other diseases to plants, leading to massive diseased crops. This could affect its yield or cause the entire field to become unusable.

2.5 DATA SET WITH DESCRIPTION

Variable	**Description**
N	Nitrogen content in soil
P	Phosphorous content in soil
K	Potassium content in soil
temperature	Temperature of the agriculture area
humidity	Humidity of the agriculture area
ph	pH value of the water
rainfall	Rainfall data over that region
label	Crop names to suggest



```
df.head()
```

	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

2.5 DATASET

CHAPTER 3
IDEATE

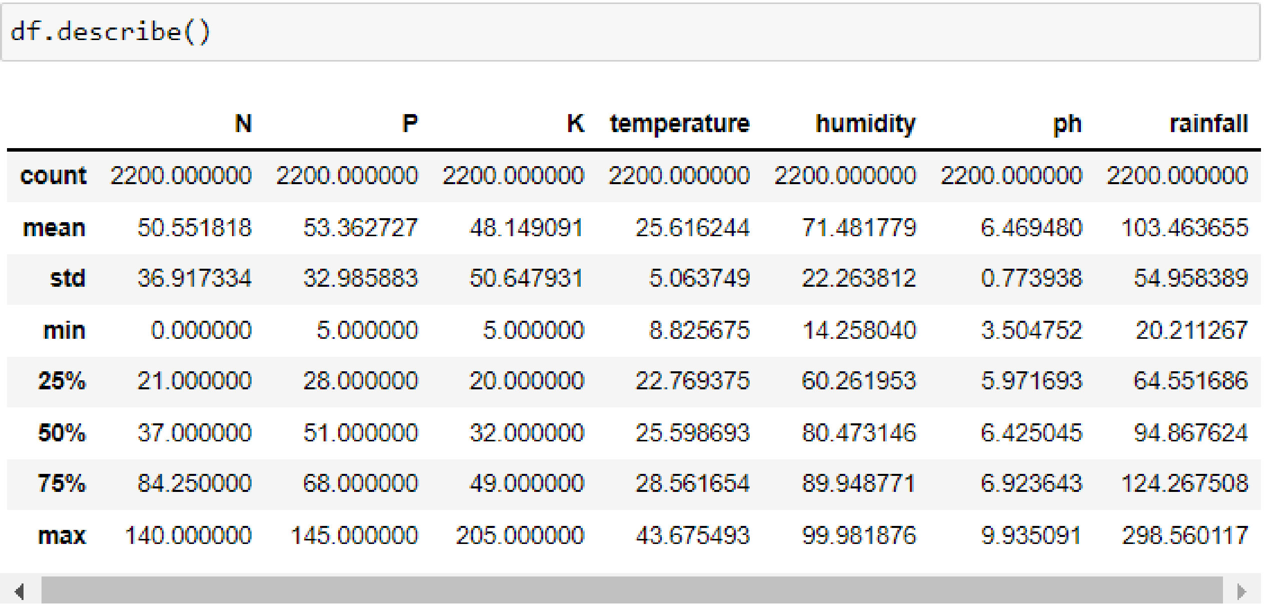
These problems can be resolved by developing a machine learning model to predict which crop to yield in that region

The dataset is taken from Kaggle.com and exploring it to understand what the data tells by exploratory data analysis.

Finally, by selecting an appropriate machine learning algorithm, the prediction is made with better accuracy providing algorithm.

3.1 EXPLORATORY DATA ANALYSIS

Exploratory Data Analysis (EDA) is an approach to analyze the data using visual techniques. It is used to discover trends, patterns, or to check assumptions with the help of statistical summary and graphical representations. It is a good practice to understand the data first and try to gather as many insights from it. EDA is all about making sense of data in hand, before getting them dirty with it.



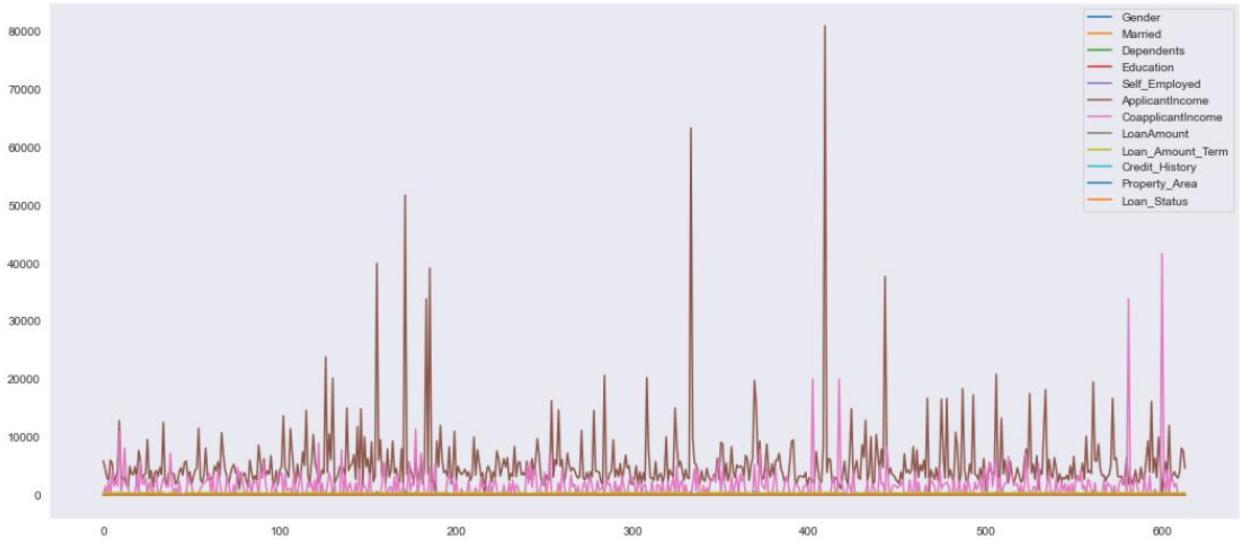
```
df.describe()
```

	N	P	K	temperature	humidity	ph	rainfall
count	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000
mean	50.551818	53.362727	48.149091	25.616244	71.481779	6.469480	103.463655
std	36.917334	32.985883	50.647931	5.063749	22.263812	0.773938	54.958389
min	0.000000	5.000000	5.000000	8.825675	14.258040	3.504752	20.211267
25%	21.000000	28.000000	20.000000	22.769375	60.261953	5.971693	64.551686
50%	37.000000	51.000000	32.000000	25.598693	80.473146	6.425045	94.867624
75%	84.250000	68.000000	49.000000	28.561654	89.948771	6.923643	124.267508
max	140.000000	145.000000	205.000000	43.675493	99.981876	9.935091	298.560117

3.1 EXPLORATORY DATA ANALYSIS

3.2 DATA VISUALIZATION

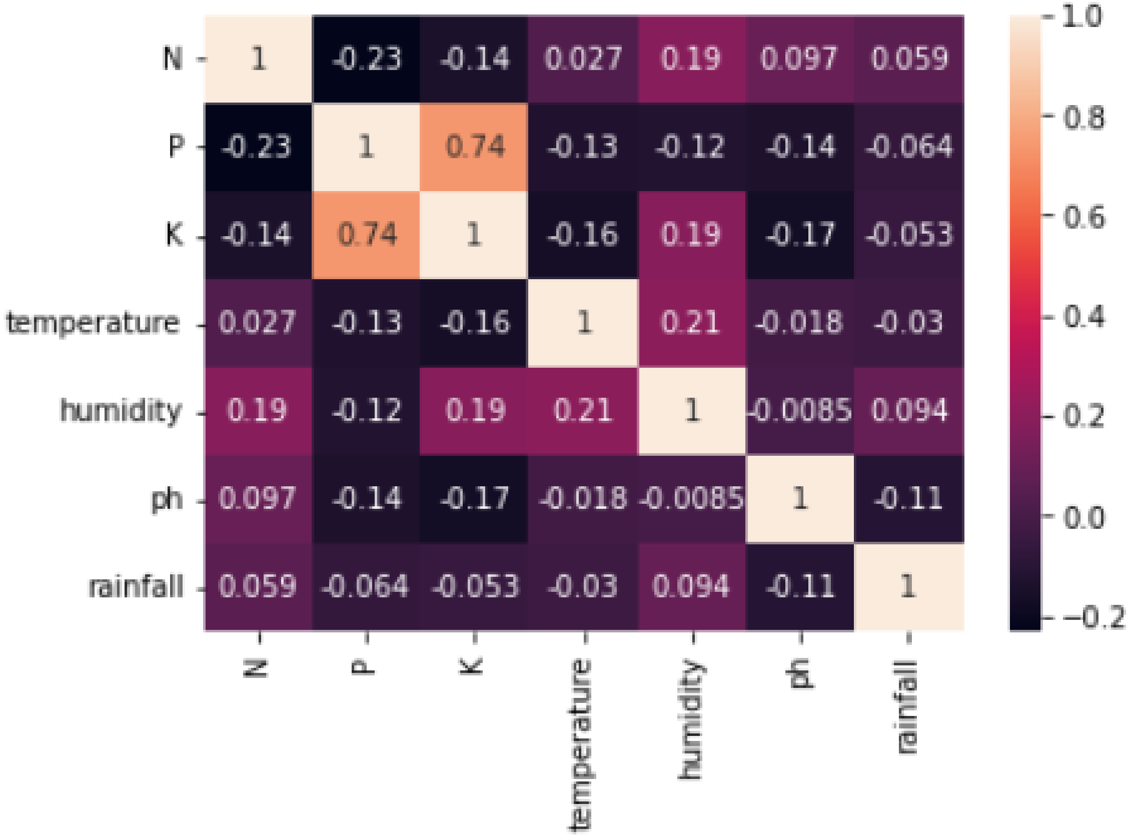
Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data. Additionally, it provides an excellent way for employees or business owners to present data to non-technical audiences without confusion.



3.2 DATA VISUALIZATION

3.3 CORRELATION ANALYSIS

Correlation analysis in research is a statistical method used to measure the strength of the linear relationship between two variables and compute their association. Simply put - correlation analysis calculates the level of change in one variable due to the change in the other. A high correlation points to a strong relationship between the two variables, while a low correlation means that the variables are weakly related. When it comes to market research, researchers use correlation analysis to analyze quantitative data collected through research methods like surveys and live polls. They try to identify the relationship, patterns, significant connections, and trends between two variables or datasets. There is a positive correlation between two variables when an increase in one variable leads to the increase in the other. On the other hand, a negative correlation means that when one variable increases, the other decreases and vice-versa.



3.3 CORRELATION ANALYSIS

3.4 SOFTWARE & LIBRARIES

Normally, a library is a collection of books or is a room or place where many books are stored to be used later. Similarly, in the programming world, a library is a collection of precompiled codes that can be used later on in a program for some specific well-defined operations. Other than pre-compiled codes, a library may contain documentation, configuration data, message templates, classes, and values, etc.

A Python library is a collection of related modules. It contains bundles of code that can be used repeatedly in different programs. It makes Python Programming simpler and convenient for the programmer. As we don’t need to write the same code again and again for different programs. Python libraries play a very vital role in fields of Machine Learning, Data Science, Data Visualization, etc.

SOFTWARE / LIBRARIES	**USAGE**
Python3 software	Programming language
Pandas	To work with data in data frames and data exploration
NumPy	For scaling and reshaping data
Sklearn	Pre-processing, model selection, metrics
Matplotlib, Seaborn	For data visualization
Pickle	To save the trained model
Streamlit	To deploy the ML model in localhost web app

3.4 SOFTWARE AND LIBRARIES

CHAPTER 4
PROTOTYPE

4.1 PYTHON3 SOFTWARE

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is a multi-paradigm programming language. Object-oriented programming and structured programming are fully supported, and many of their features support functional programming and aspect-oriented programming (including metaprogramming and metaobjects). Many other paradigms are supported via extensions, including design by contract and logic programming.

4.1 PYTHON SOFTWARE

4.2 PANDAS

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals. Its name is a play on the phrase "Python data analysis" itself. Wes McKinney started building what would become pandas at AQR Capital while he was a researcher there from 2007 to 2010.

4.2 PANDAS LIBRARY

4.3 NUMPY

NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. The ancestor of NumPy, Numeric, was originally created by Jim Hugunin with contributions from several other developers. In 2005, Travis Oliphant created NumPy by incorporating features of the competing Numarray into Numeric, with extensive modifications. NumPy is open-source software and has many contributors. NumPy is a NumFOCUS fiscally sponsored project.

4.3 NUMPY LIBRARY

4.4 SK LEARN

Scikit-learn (formerly **scikits.learn** and also known as **sklearn**) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support-vector machines, random forests, gradient boosting, *k*-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy. Scikit-learn is a NumFOCUS fiscally sponsored project. scikit-learn is an open-source Python library that implements a range of machine learning, pre-processing, cross-validation, and visualization algorithms using a unified interface. It is simple and efficient tools for data mining and data analysis. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means, etc. It is accessible to everybody and reusable in various contexts. Built on the top of NumPy, SciPy, and matplotlib. It is open source, commercially usable – BSD license.

4.4 SCIKIT-LEARN LIBRARY

4.5 MATPLOTLIB

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged.[3] SciPy makes use of Matplotlib.

4.5 MATPLOTLIB LIBRARY

4.6 SEABORN

Seaborn is one of an amazing library for visualization of the graphical statistical plotting in Python. Seaborn provides many color palettes and defaults beautiful styles to make the creation of many statistical plots in Python more attractive. Seaborn library aims to make a more attractive visualization of the central part of understanding and exploring data. It is built on the core of the matplotlib library and also provides dataset-oriented APIs. Seaborn is also closely integrated with the Panda's data structures, and with this, we can easily jump between the various different visual representations for a given variable to better understand the provided dataset.

4.6 SEABORN LIBRARY

4.7 STREAMLIT

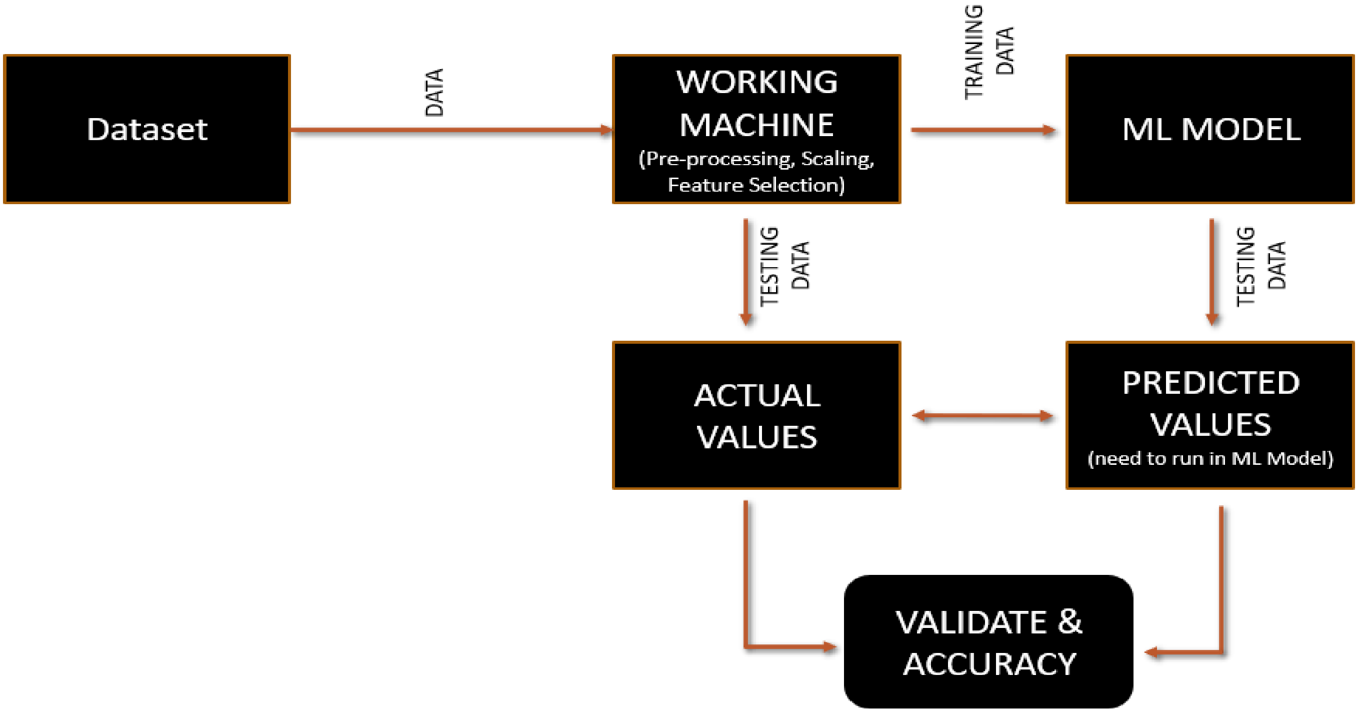
Streamlit is an open source python based framework for developing and deploying interactive data science dashboards and machine learning models. This means that you do not have to rely on a team of front end developers or spend large amounts of time learning web design languages such as HTML, CSS or JavaScript in order to deploy your dashboard or model.

Streamlit was founded in 2018 by ex Google engineers who had gained first hand experience of the challenges faced when developing and deploying machine learning models and dashboards.

It is built on top of Python and supports many of the mainstream Python libraries such as matplotlib, plotly and pandas.

4.7 STREAMLIT LIBRARY

4.8 ARCHITECTURAL DIAGRAM



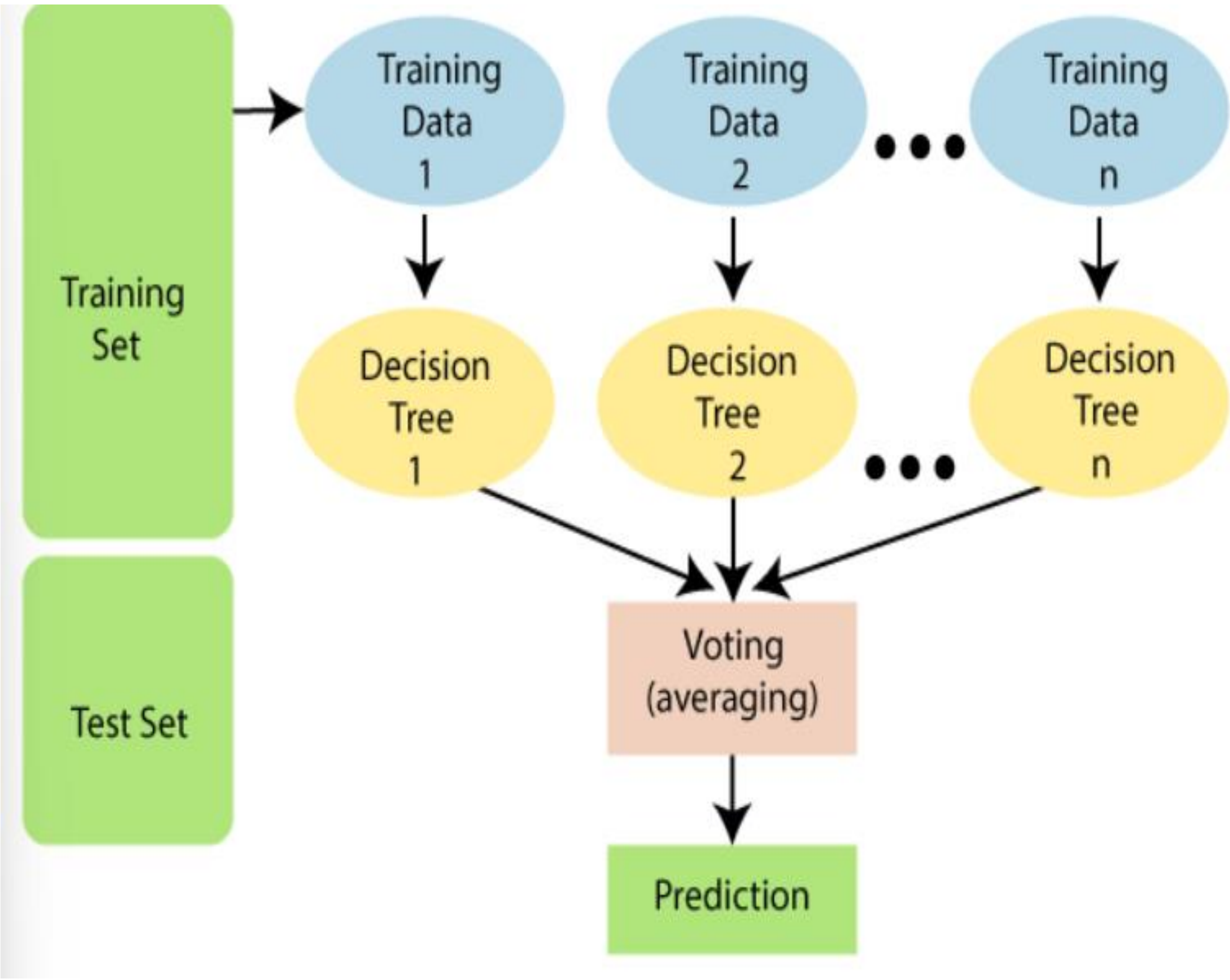
4.8 ARCHITECTURAL DIAGRAM

4.9 RANDOM FOREST CLASSIFIER ALGORITHM

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

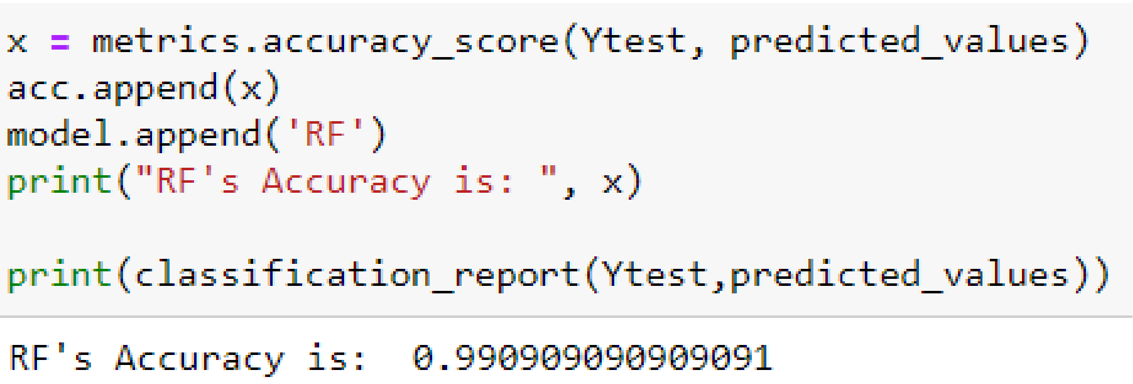
As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.



4.9 RANDOM FOREST CLASSIFIER WORKING

4.10 PREDICTION ACCURACY

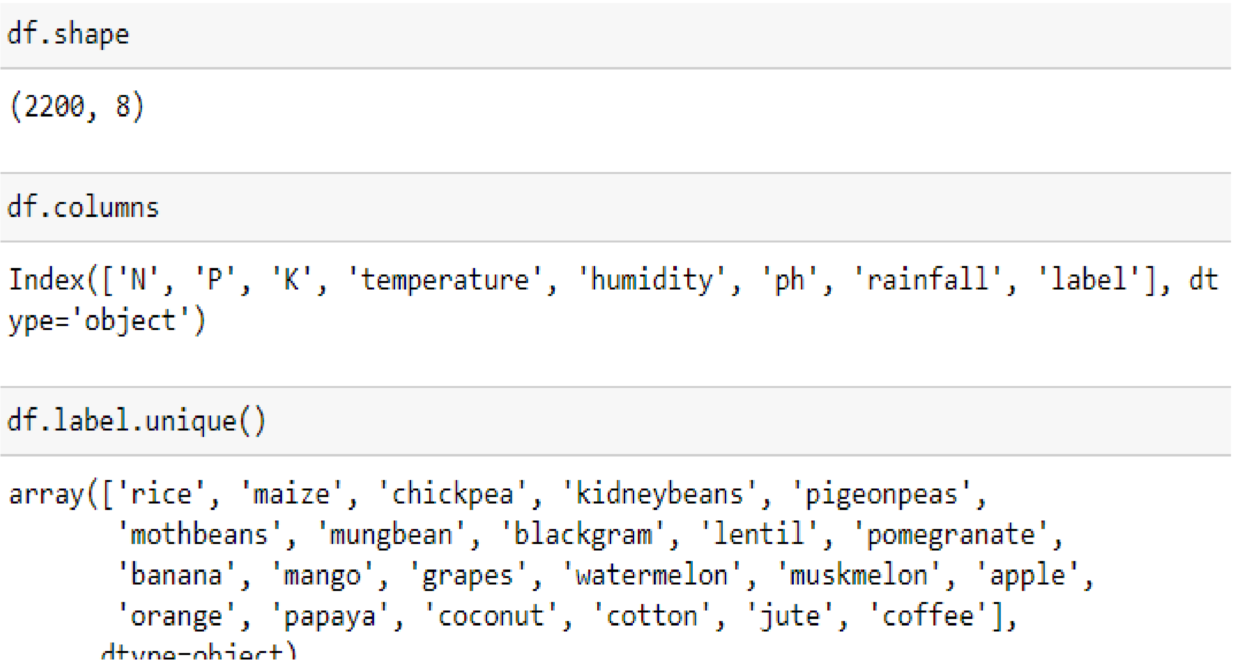


```
x = metrics.accuracy_score(Ytest, predicted_values)
acc.append(x)
model.append('RF')
print("RF's Accuracy is: ", x)

print(classification_report(Ytest,predicted_values))
```

```
RF's Accuracy is:  0.990909090909091
```

4.10 PREDICTION ACCURACY

CHAPTER 5
TESTING



```
df.shape
```

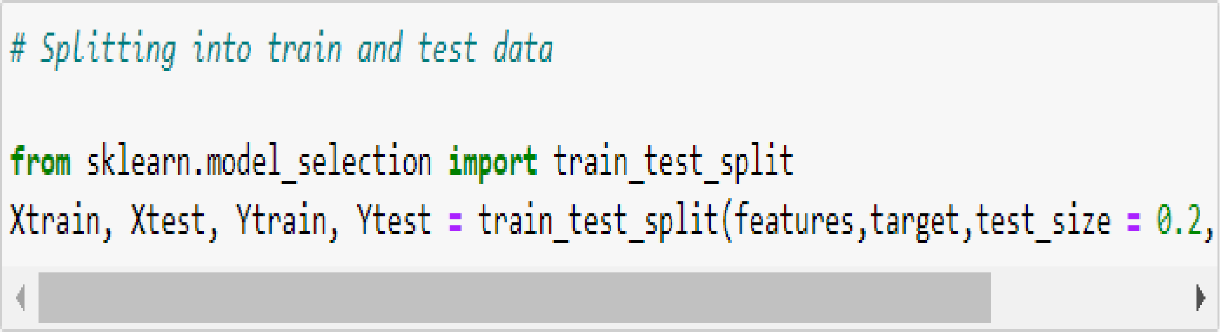
```
(2200, 8)
```

```
df.columns
```

```
Index(['N', 'P', 'K', 'temperature', 'humidity', 'ph', 'rainfall', 'label'], dt
ype='object')
```

```
df.label.unique()
```

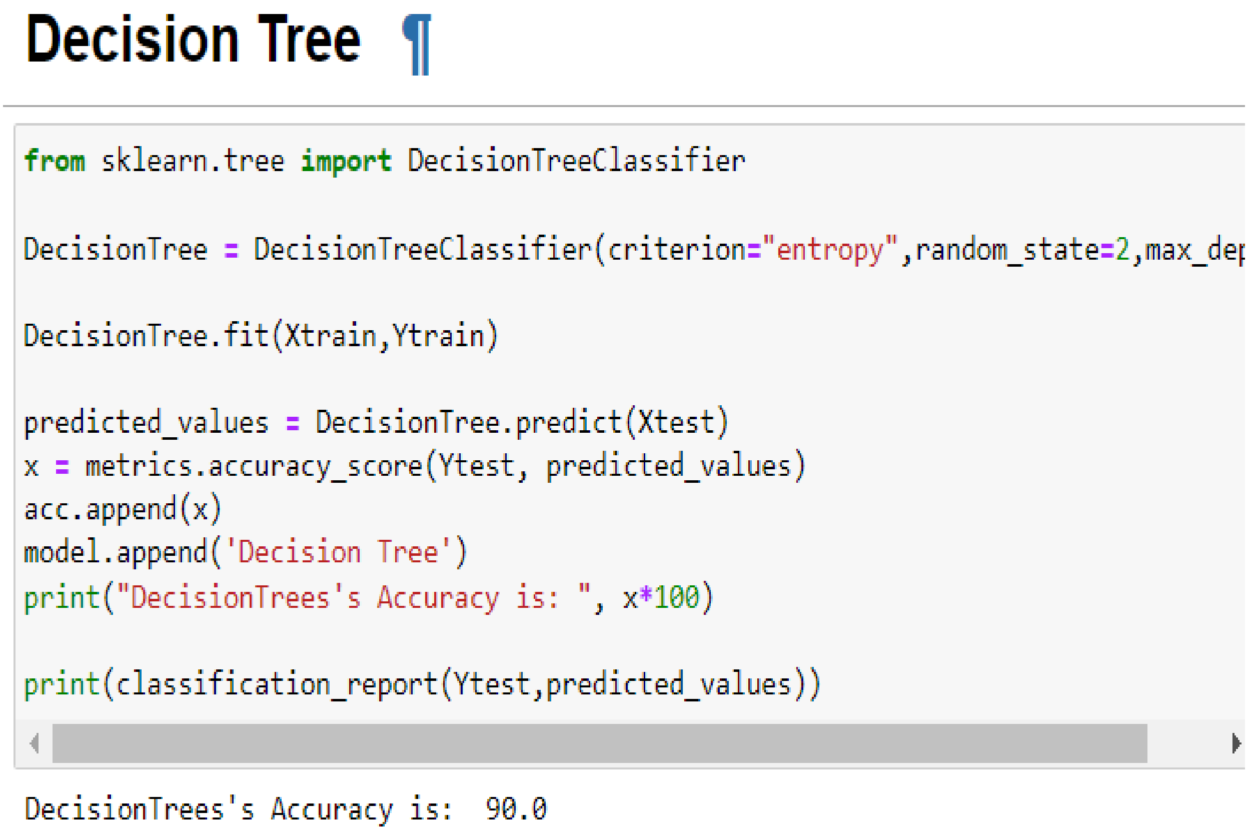
```
array(['rice', 'maize', 'chickpea', 'kidneybeans', 'pigeonpeas',
       'mothbeans', 'mungbean', 'blackgram', 'lentil', 'pomegranate',
       'banana', 'mango', 'grapes', 'watermelon', 'muskmelon', 'apple',
       'orange', 'papaya', 'coconut', 'cotton', 'jute', 'coffee'],
      dtype=object)
```



```
# Splitting into train and test data

from sklearn.model_selection import train_test_split
Xtrain, Xtest, Ytrain, Ytest = train_test_split(features,target,test_size = 0.2,
```

```
df.label.value_counts()
```

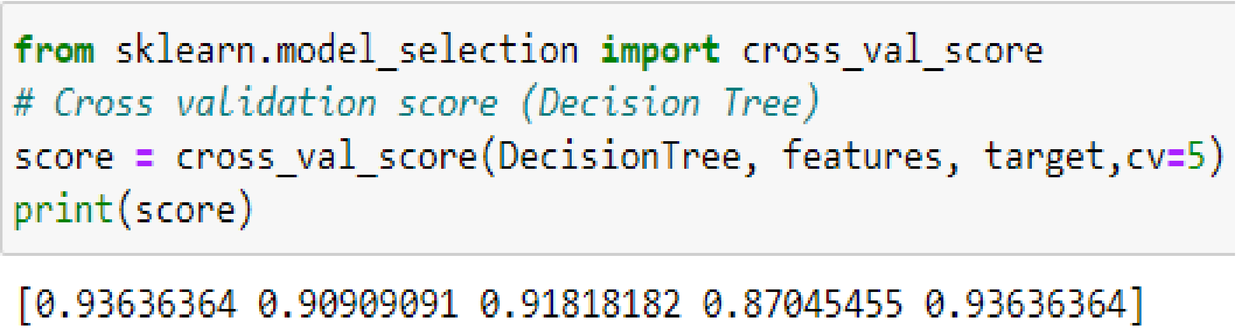
```
rice           100
maize          100
jute           100
cotton         100
coconut        100
papaya         100
orange         100
apple          100
muskmelon      100
watermelon     100
grapes         100
mango          100
banana         100
pomegranate    100
lentil         100
blackgram  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



Decision Tree ¶

```
from sklearn.tree import DecisionTreeClassifier

DecisionTree = DecisionTreeClassifier(criterion="entropy",random_state=2,max_dep

DecisionTree.fit(Xtrain,Ytrain)

predicted_values = DecisionTree.predict(Xtest)
x = metrics.accuracy_score(Ytest, predicted_values)
acc.append(x)
model.append('Decision Tree')
print("DecisionTrees's Accuracy is: ", x*100)

print(classification_report(Ytest,predicted_values))
```

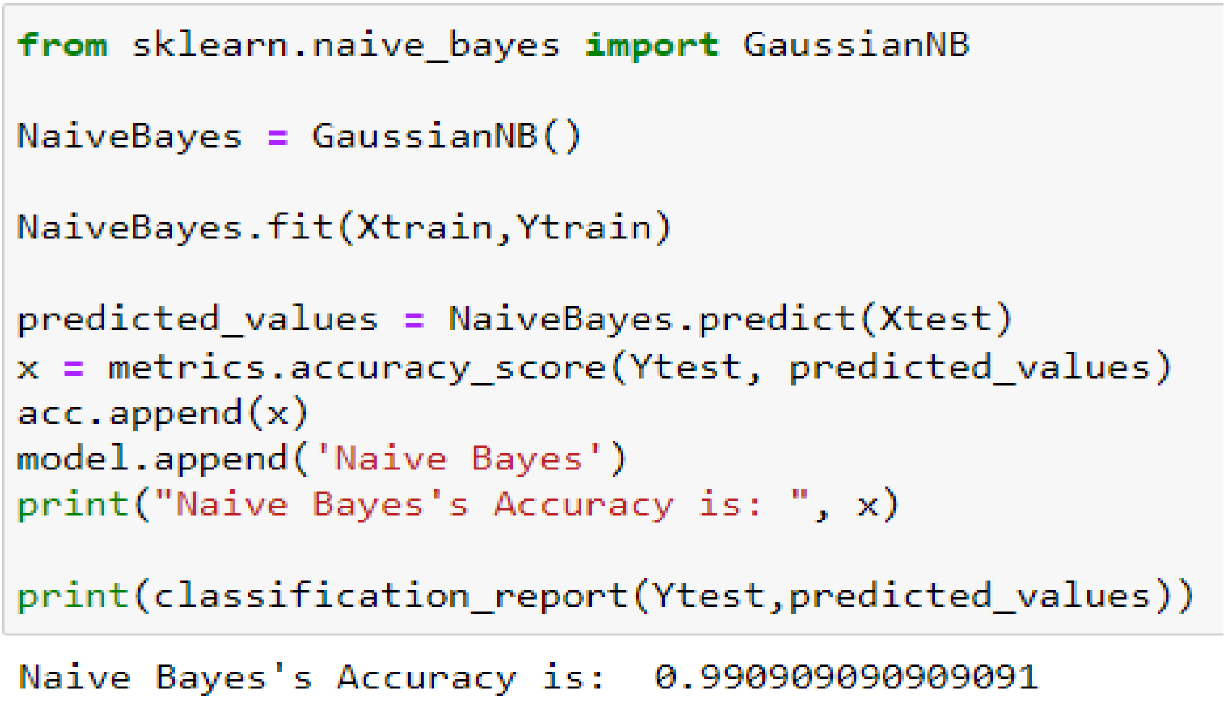
```
DecisionTrees's Accuracy is:  90.0
```



```
from sklearn.model_selection import cross_val_score
# Cross validation score (Decision Tree)
score = cross_val_score(DecisionTree, features, target,cv=5)
print(score)
```

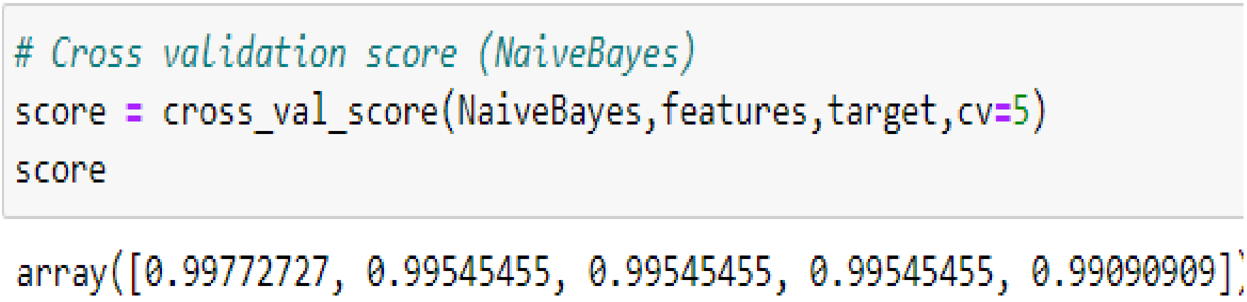
```
[0.93636364 0.90909091 0.91818182 0.87045455 0.93636364]
```

Guassian Naive Bayes



```
from sklearn.naive_bayes import GaussianNB

NaiveBayes = GaussianNB()

NaiveBayes.fit(Xtrain,Ytrain)

predicted_values = NaiveBayes.predict(Xtest)
x = metrics.accuracy_score(Ytest, predicted_values)
acc.append(x)
model.append('Naive Bayes')
print("Naive Bayes's Accuracy is: ", x)

print(classification_report(Ytest,predicted_values))
```

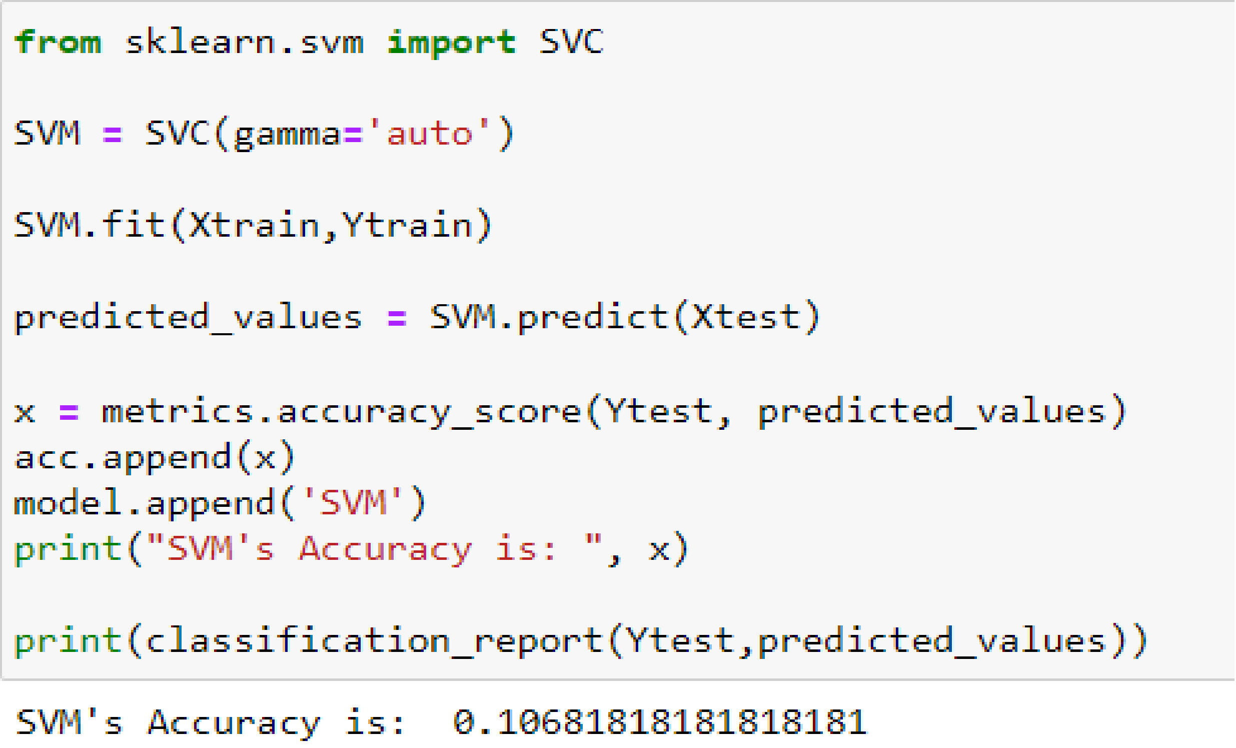
```
Naive Bayes's Accuracy is:  0.990909090909091
```



```
# Cross validation score (NaiveBayes)
score = cross_val_score(NaiveBayes,features,target,cv=5)
score
```

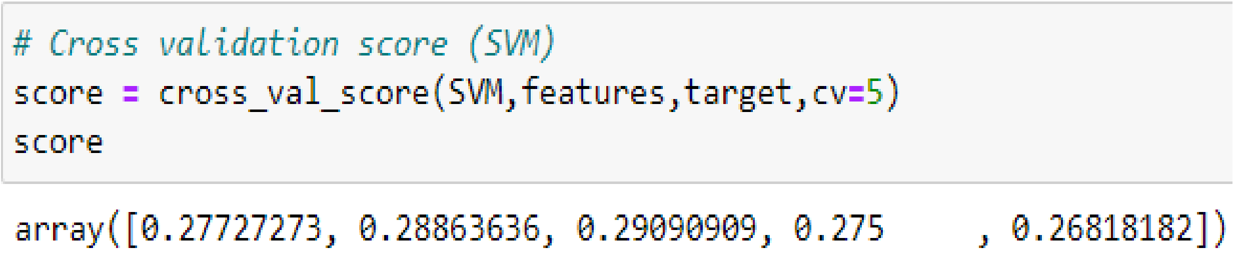
```
array([0.99772727, 0.99545455, 0.99545455, 0.99545455, 0.99090909])
```

Support Vector Machine (SVM)



```
from sklearn.svm import SVC

SVM = SVC(gamma='auto')

SVM.fit(Xtrain,Ytrain)

predicted_values = SVM.predict(Xtest)

x = metrics.accuracy_score(Ytest, predicted_values)
acc.append(x)
model.append('SVM')
print("SVM's Accuracy is: ", x)

print(classification_report(Ytest,predicted_values))
```

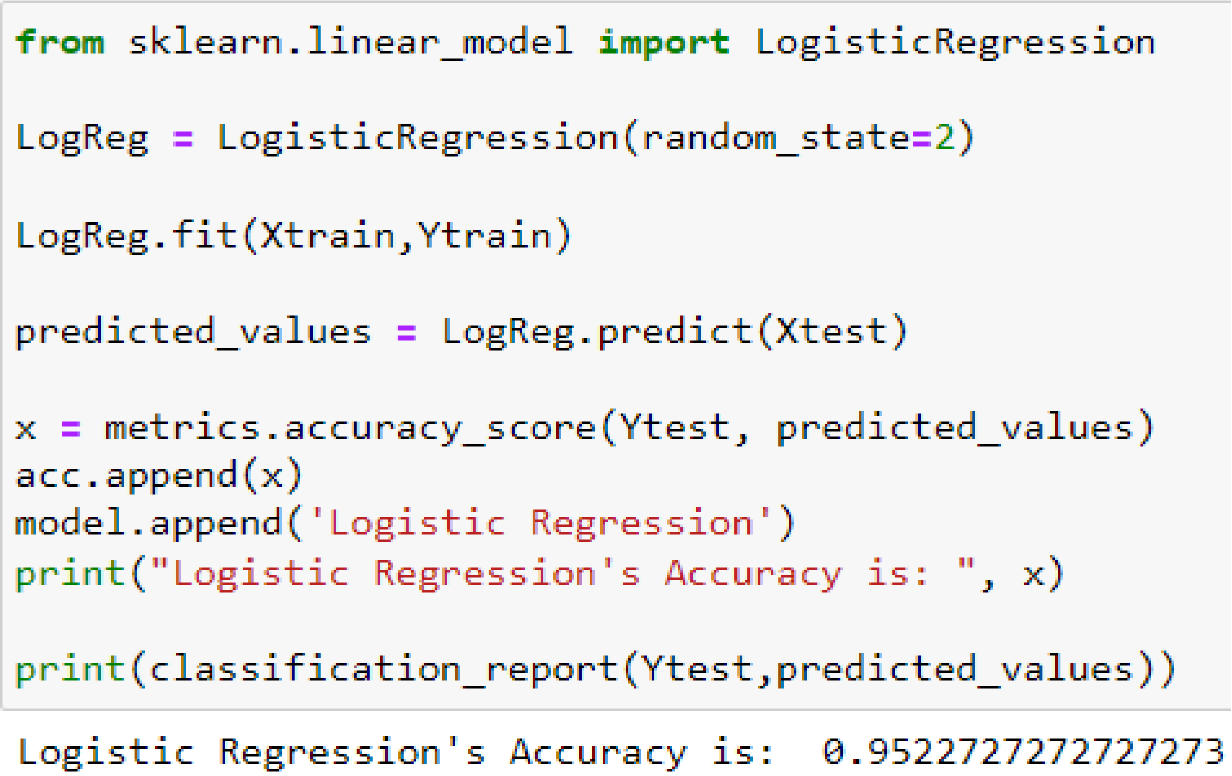
```
SVM's Accuracy is:  0.10681818181818181
```



```
# Cross validation score (SVM)
score = cross_val_score(SVM,features,target,cv=5)
score
```

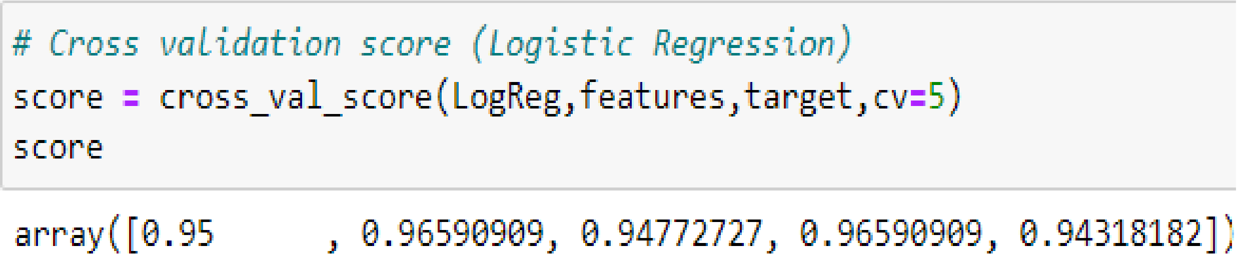
```
array([0.27727273, 0.28863636, 0.29090909, 0.275     , 0.26818182])
```

Logistic Regression



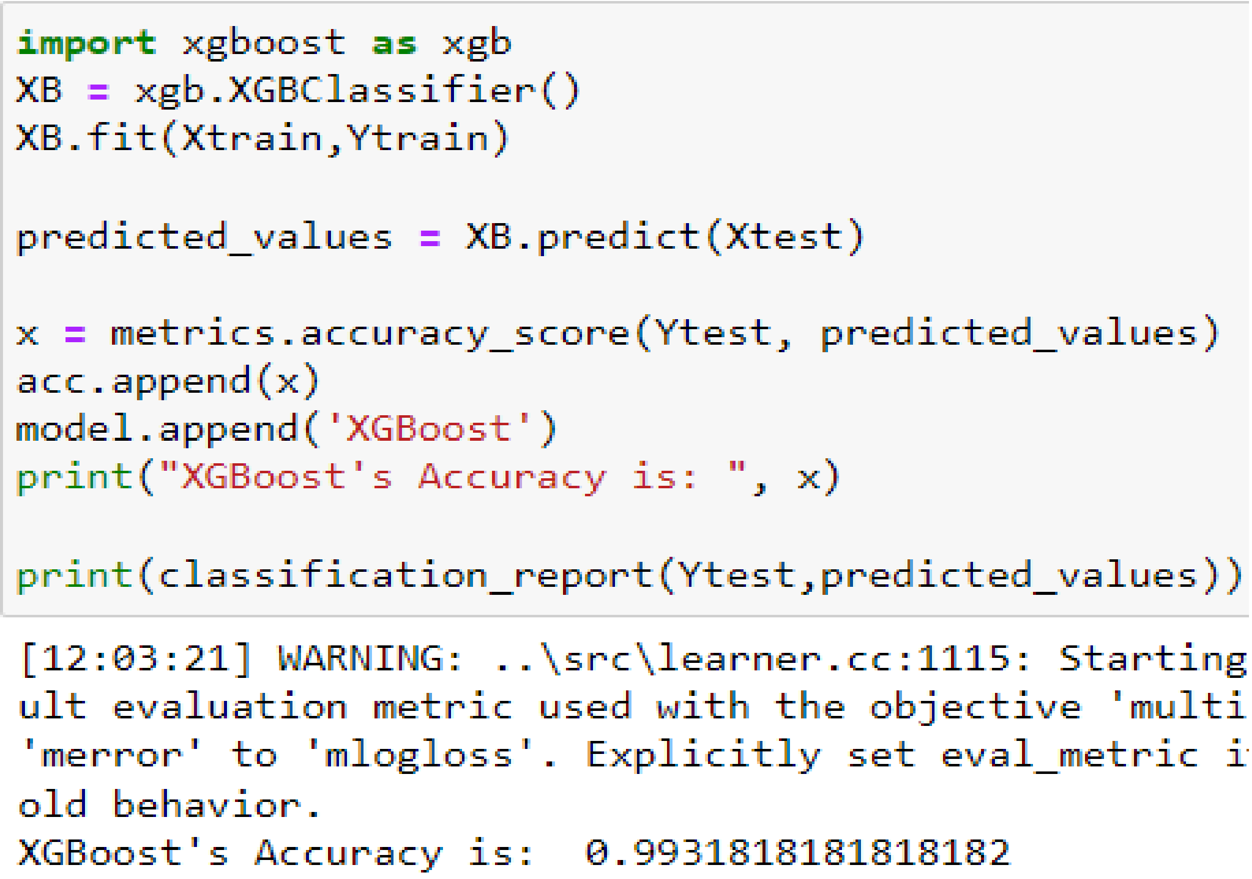
```
from sklearn.linear_model import LogisticRegression

LogReg = LogisticRegression(random_state=2)

LogReg.fit(Xtrain,Ytrain)

predicted_values = LogReg.predict(Xtest)

x = metrics.accuracy_score(Ytest, predicted_values)
acc.append(x)
model.append('Logistic Regression')
print("Logistic Regression's Accuracy is: ", x)

print(classification_report(Ytest,predicted_values))
```

```
Logistic Regression's Accuracy is:  0.9522727272727273
```



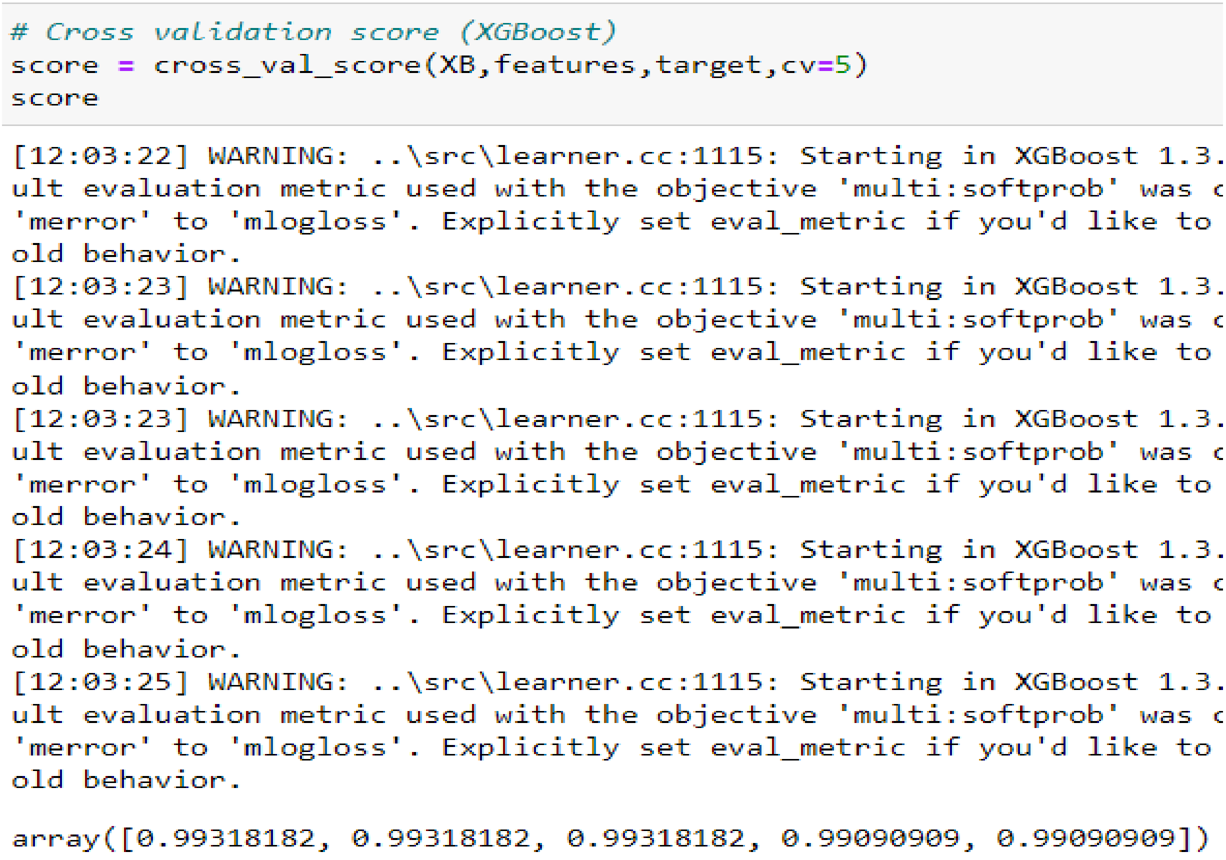
```
# Cross validation score (Logistic Regression)
score = cross_val_score(LogReg,features,target,cv=5)
score
```

```
array([0.95      , 0.96590909, 0.94772727, 0.96590909, 0.94318182])
```



```
import xgboost as xgb
XB = xgb.XGBClassifier()
XB.fit(Xtrain,Ytrain)

predicted_values = XB.predict(Xtest)

x = metrics.accuracy_score(Ytest, predicted_values)
acc.append(x)
model.append('XGBoost')
print("XGBoost's Accuracy is: ", x)

print(classification_report(Ytest,predicted_values))
```

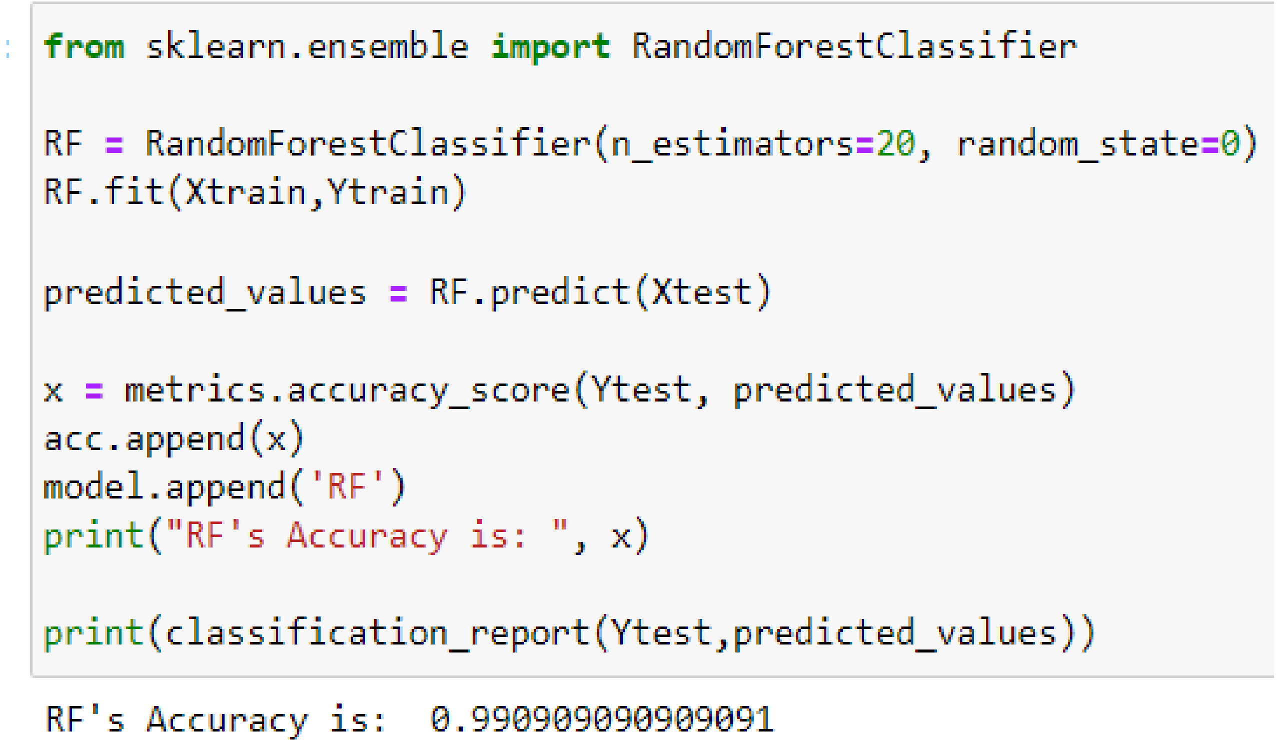
```
[12:03:21] WARNING: ..\src\learner.cc:1115: Starting
ult evaluation metric used with the objective 'multi
'merror' to 'mlogloss'. Explicitly set eval_metric i
old behavior.
XGBoost's Accuracy is:  0.9931818181818182
```



```
# Cross validation score (XGBoost)
score = cross_val_score(XB,features,target,cv=5)
score
```

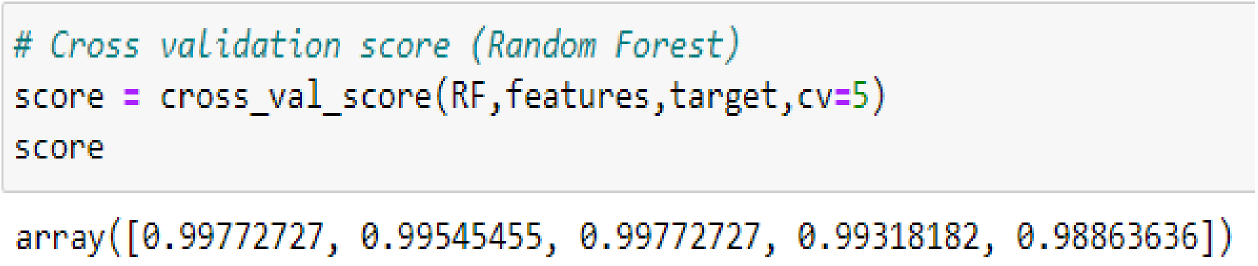
```
[12:03:22] WARNING: ..\src\learner.cc:1115: Starting in XGBoost 1.3.
ult evaluation metric used with the objective 'multi:softprob' was c
'merror' to 'mlogloss'. Explicitly set eval_metric if you'd like to
old behavior.
[12:03:23] WARNING: ..\src\learner.cc:1115: Starting in XGBoost 1.3.
ult evaluation metric used with the objective 'multi:softprob' was c
'merror' to 'mlogloss'. Explicitly set eval_metric if you'd like to
old behavior.
[12:03:23] WARNING: ..\src\learner.cc:1115: Starting in XGBoost 1.3.
ult evaluation metric used with the objective 'multi:softprob' was c
'merror' to 'mlogloss'. Explicitly set eval_metric if you'd like to
old behavior.
[12:03:24] WARNING: ..\src\learner.cc:1115: Starting in XGBoost 1.3.
ult evaluation metric used with the objective 'multi:softprob' was c
'merror' to 'mlogloss'. Explicitly set eval_metric if you'd like to
old behavior.
[12:03:25] WARNING: ..\src\learner.cc:1115: Starting in XGBoost 1.3.
ult evaluation metric used with the objective 'multi:softprob' was c
'merror' to 'mlogloss'. Explicitly set eval_metric if you'd like to
old behavior.

array([0.99318182, 0.99318182, 0.99318182, 0.99090909, 0.99090909])
```

Random Forest Classifier



```
from sklearn.ensemble import RandomForestClassifier

RF = RandomForestClassifier(n_estimators=20, random_state=0)
RF.fit(Xtrain,Ytrain)

predicted_values = RF.predict(Xtest)

x = metrics.accuracy_score(Ytest, predicted_values)
acc.append(x)
model.append('RF')
print("RF's Accuracy is: ", x)

print(classification_report(Ytest,predicted_values))
```

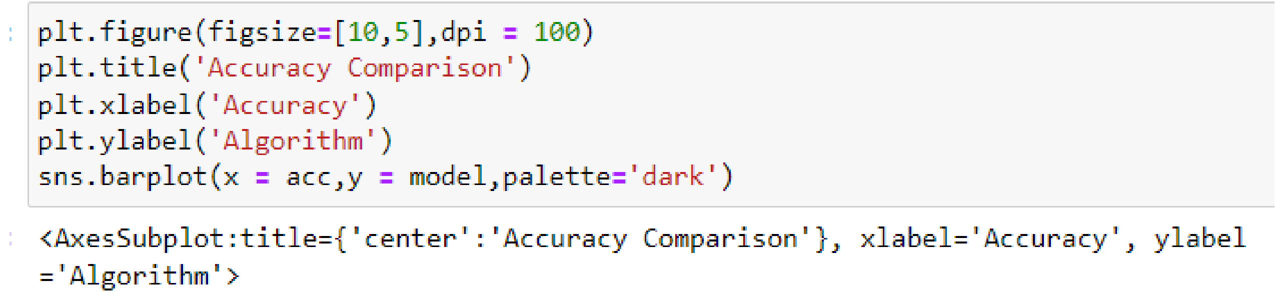
```
RF's Accuracy is:  0.990909090909091
```



```
# Cross validation score (Random Forest)
score = cross_val_score(RF,features,target,cv=5)
score
```

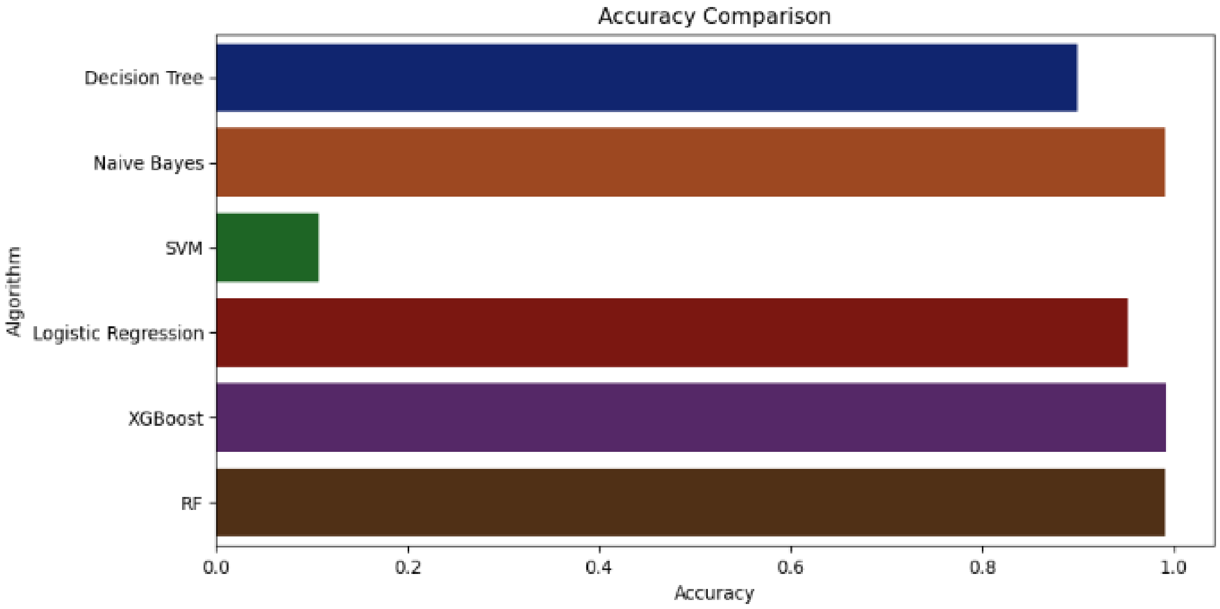
```
array([0.99772727, 0.99545455, 0.99772727, 0.99318182, 0.98863636])
```

Accuracy Comparison

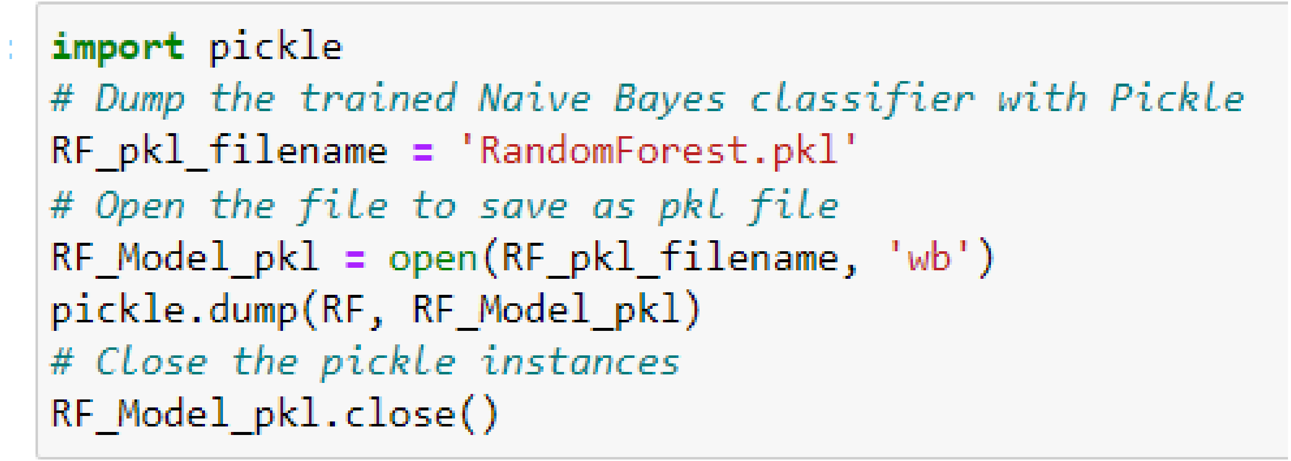


```
plt.figure(figsize=[10,5],dpi = 100)
plt.title('Accuracy Comparison')
plt.xlabel('Accuracy')
plt.ylabel('Algorithm')
sns.barplot(x = acc,y = model,palette='dark')
```

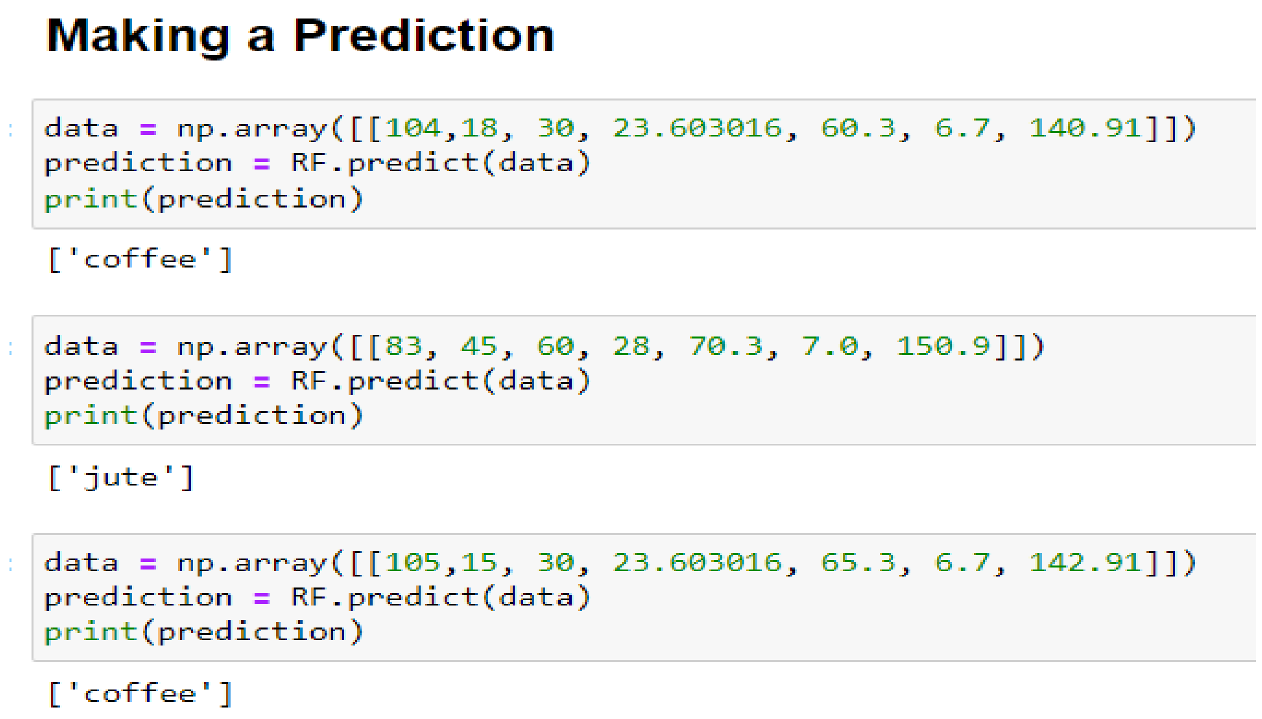
```
<AxesSubplot:title={'center':'Accuracy Comparison'}, xlabel='Accuracy', ylabel
='Algorithm'>
```



Saving the trained Random Forest Classifier model



```
import pickle
# Dump the trained Naive Bayes classifier with Pickle
RF_pkl_filename = 'RandomForest.pkl'
# Open the file to save as pkl file
RF_Model_pkl = open(RF_pkl_filename, 'wb')
pickle.dump(RF, RF_Model_pkl)
# Close the pickle instances
RF_Model_pkl.close()
```



Making a Prediction

```
data = np.array([[104,18, 30, 23.603016, 60.3, 6.7, 140.91]])
prediction = RF.predict(data)
print(prediction)
```

```
['coffee']
```

```
data = np.array([[83, 45, 60, 28, 70.3, 7.0, 150.9]])
prediction = RF.predict(data)
print(prediction)
```

```
['jute']
```

```
data = np.array([[105,15, 30, 23.603016, 65.3, 6.7, 142.91]])
prediction = RF.predict(data)
print(prediction)
```

```
['coffee']
```

CHAPTER 6
CONCLUSION

6.1 CONCLUSION

With more amount of data, the model can be well trained in hard cases. Fertilizer recommendation, disease detection in crops and other ML applications can be integrated. Finally, it is deployed in a user-friendly way to the farmers and customers.

CHAPTER 7
FUTURE WORKS

7.1 FUTURE WORKS

This type of AI technologies can change the world into massive automation. To make their work more automotive and easier, the model can be deployed in the app or website. By integrating this ML model to the backend of the app, when user provides the required data, the ML model process the output easily.

REFERENCE

- https://en.wikipedia.org/wiki/Crop#Important_food_crops
- https://www.fao.org/documents/card/en/c/cb4477en
- https://www.cropsreview.com/crop-selection.html
- https://wonderopolis.org/wonder/why-is-crop-rotation-important
- https://www.researchgate.net/publication/346627389_Crop_Recommendation_System
- https://towardsdatascience.com/farmeasy-crop-recommendation-portal-for-farmers-48a8809b421c
- https://www.javatpoint.com/machine-learning-random-forest-algorithm