

**SCTR's Pune Institute of Computer Technology
Dhankawadi, Pune**

**DSBDAL MINI PROJECT ON
Rice Yield Prediction**

SUBMITTED BY

Neeraj Bukane 33114
Sanyog Kalantri 33137
Samyak Jain 33169
Gauri Takalikar 33178

**Under the guidance of
Dr. Anant Bagade**



**DEPARTMENT OF INFORMATION TECHNOLOGY
ACADEMIC YEAR 2022-23**

Contents

1	Abstract	2
2	Introduction	3
3	Problem Statement	4
4	Objectives and Scope	5
4.1	Objectives	5
4.2	Scope	6
5	Feature Description	7
6	Implementation Details	8
6.1	Data Loading:	8
6.2	Data Cleaning:	8
6.3	Data Preprocessing:	8
6.4	Data Visualization:	8
6.5	Data Preparation:	9
6.6	Modeling:	9
7	Output	10
8	Conclusion	13

1 Abstract

Crop yield prediction plays a crucial role in agricultural planning and decision-making processes. Accurate predictions of crop yields enable farmers, policymakers, and stakeholders to optimize resource allocation, mitigate risks, and enhance agricultural productivity. This project aims to develop a crop yield prediction model using machine learning techniques. The model utilizes various environmental and agricultural factors such as temperature, wind speed, pressure, humidity, and nutrient levels (N, P, K) to predict crop yields. Different regression models, including Linear Regression, Random Forest Regression, Decision Tree Regression, and Support Vector Regression, are employed and evaluated for their performance in predicting crop yields.

2 Introduction

Agriculture is the backbone of many economies, and predicting crop yields accurately is of paramount importance for sustainable agricultural practices and food security. Traditional methods of crop yield estimation often rely on historical data and manual calculations, which are time-consuming and prone to errors. With the advancements in data science and machine learning, it is now possible to leverage historical and real-time agricultural data to develop predictive models for crop yield estimation.

The objective of this project is to build a crop yield prediction model that can provide accurate estimates of crop yields based on various factors that influence agricultural productivity. By analyzing historical data and incorporating relevant environmental and agricultural features, the model aims to forecast crop yields for a specific region. This information can assist farmers in making informed decisions regarding crop selection, resource allocation, and planning for optimal yields.

3 Problem Statement

The main problem addressed in this project is the accurate prediction of crop yields based on various environmental and agricultural factors. The challenge lies in identifying the significant factors that influence crop productivity and developing a robust predictive model that can generalize well to unseen data. The model should be capable of handling the complexities and nonlinear relationships present in agricultural data, enabling accurate and reliable crop yield predictions.

The proposed solution involves data cleaning and preprocessing, exploratory data analysis to understand the relationships between variables, feature selection, and the implementation and evaluation of different regression models. The performance of each model will be assessed using metrics such as Mean Squared Error (MSE) and R-squared (R^2) score to determine the most effective model for crop yield prediction.

By addressing this problem, we aim to provide farmers, policymakers, and stakeholders with a valuable tool for enhancing agricultural planning, optimizing resource allocation, and improving overall crop productivity.

4 Objectives and Scope

4.1 Objectives

We aim to achieve the following objectives:

1. To develop a crop yield prediction model using machine learning techniques. .
2. To identify and analyze the environmental and agricultural factors that significantly impact crop yields. .
3. To evaluate and compare the performance of different regression models for crop yield prediction.
4. To provide accurate and reliable crop yield predictions for better agricultural planning and decision-making.
5. To assist farmers, policymakers, and stakeholders in optimizing resource allocation and enhancing agricultural productivity.

4.2 Scope

1. The project focuses on crop yield prediction for the specific region of Maharashtra, India.
2. Historical data related to crop production, environmental factors (temperature, wind speed, pressure, humidity), and nutrient levels (N, P, K) will be considered.
3. The project includes data cleaning, preprocessing, and feature selection to ensure the quality and relevance of the input data.
4. Different regression models, including Linear Regression, Random Forest Regression, Decision Tree Regression, and Support Vector Regression, will be implemented and evaluated.
5. The performance of each model will be assessed using standard evaluation metrics such as Mean Squared Error (MSE) and R-squared (R²) score.
6. The project aims to provide insights and predictions for crop yields, enabling farmers to make informed decisions regarding crop selection and resource allocation.
7. The results and findings of the project are specific to crop yield prediction in the given region and may not be directly applicable to other regions or crops.

5 Feature Description

1. **state names** The name of the state where the crop yield data was recorded.
2. **district names** The name of the district within the state where the crop yield data was recorded.
3. **crop year** The year in which the crop yield data was recorded.
4. **season names:** The name of the season during which the crop was cultivated (e.g., Kharif, Rabi).
5. **crop names:** The name of the crop for which the yield data is provided.
6. **area:** The area of land (in some unit of measurement) used for cultivating the crop.
7. **temperature:** The temperature (in some unit of measurement) during the crop cultivation period.
8. **wind speed:** The speed of wind (in some unit of measurement) during the crop cultivation period.
9. **pressure:** The atmospheric pressure (in some unit of measurement) during the crop cultivation period.
10. **humidity:** The humidity level (in some unit of measurement) during the crop cultivation period.
11. **soil type:** The type of soil in which the crop was cultivated.
12. **N:** The amount of nitrogen (N) used as fertilizer for the crop.
13. **P:** The amount of phosphorus (P) used as fertilizer for the crop.
14. **K:** The amount of potassium (K) used as fertilizer for the crop.
production: The total production of the crop (in some unit of measurement).
15. **Yield:** The yield of the crop, calculated as the production divided by the area.

6 Implementation Details

The steps followed are:

6.1 Data Loading:

1. The dataset is loaded using the pandas library's `read_csv()` function.
2. The dataset is stored in a pandas DataFrame named `df`.
3. The shape of the dataset is checked using the `shape` attribute of the DataFrame.

6.2 Data Cleaning:

1. The data is reduced to one state, i.e., "Maharashtra", for ease of analysis.
2. Missing values are checked using the `isnull()` function, and the count of missing values in each column is obtained using the `sum()` function.
3. Correlation heatmap is generated using the `corr()` function from pandas and the `heatmap()` function from seaborn.

6.3 Data Preprocessing:

1. Data is filtered to include only records after the year 2004.
2. A new column called "Yield" is created by dividing the production by the area.
3. Unnecessary columns like "district names", "season names", "state names", "soil type", and "Unnamed: 0" are dropped.
4. Missing values are filled with the mean value of each column using the `fillna()` function.

6.4 Data Visualization:

1. Bar plot: A bar plot is created to visualize the production of different crops in Maharashtra using the `bar()` function from `matplotlib.pyplot`.

2. Scatter plot: A scatter plot is created to visualize the relationship between the area and yield of different crops using the `bar()` function from `matplotlib.pyplot`.

6.5 Data Preparation:

1. Train and test split: The dataset is split into training and testing sets using the `train test split()` function from `sklearn.model selection`.
2. Relevant features and the target variable are selected for modeling.
3. The features are stored in a variable named `features`, and the target variable is stored in a variable named `target` and identify the most important features contributing to rating prediction.

6.6 Modeling:

1. Linear Regression: A linear regression model is created using the `LinearRegression()` class from `sklearn.linear model`.
2. The model is trained on the training data using the `fit()` method.
3. Predictions are made on the test data using the `predict()` method.
4. Mean Squared Error (MSE) and R-squared (R2) score are calculated using the `mean squared error()` and `r2 score()` functions from `sklearn.metrics`, respectively.

7 Output

The screenshot shows a Jupyter Notebook interface with the title 'finalised_code_on_final_dataset-checkpoint'. The first cell, titled '1.Importing the libraries', contains the following code:

```
In [1]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sb
```

The second cell, titled '1.1 Loading the dataset', contains the following code:

```
In [2]: df = pd.read_csv('finalised_dataset_x.csv', na_values='')
df
```

The output of the second cell is a DataFrame with 17 columns: 'Unnamed: 0', 'state_names', 'district_names', 'crop_year', 'season_names', 'crop_names', 'area', 'temperature', 'wind_speed', 'pressure', 'humidity', and 'soil_type'. The DataFrame contains 12628 rows of data. A preview of the first few rows is shown:

	Unnamed: 0	state_names	district_names	crop_year	season_names	crop_names	area	temperature	wind_speed	pressure	humidity	soil_type
0	125191	Maharashtra	AHMEDNAGAR	1997	Autumn	Maize	1.0	20.770894	2.068280	1014.863796	21.947147	loamy
1	125192	Maharashtra	AHMEDNAGAR	1997	Kharif	Arhar/Tur	17600.0	20.160426	1.976480	1015.193528	20.843237	sandy
2	125193	Maharashtra	AHMEDNAGAR	1997	Kharif	Bajra	274100.0	21.998299	2.000524	1014.184607	21.422312	clay
3	125194	Maharashtra	AHMEDNAGAR	1997	Kharif	Gram	40800.0	21.776377	2.019750	1015.053118	21.810567	chalky
4	125195	Maharashtra	AHMEDNAGAR	1997	Kharif	Jowar	900.0	20.075734	1.974351	1015.170238	21.830206	clay

The output also indicates that the DataFrame has 12628 rows and 17 columns.

Figure 1: Importing the libraries

The screenshot shows the third cell of the Jupyter Notebook, titled '2. Data Cleaning'. The cell contains the following code:

```
In [7]: df = df[df['state_names'] == "Maharashtra"]
df.info()
```

The output of the code is the following information:

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 12628 entries, 0 to 12627
Data columns (total 16 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Unnamed: 0   12628 non-null  int64
1   state_names  12628 non-null  object
2   district_names 12628 non-null  object
3   crop_year    12628 non-null  int64
4   season_names 12628 non-null  object
5   crop_names   12628 non-null  object
6   area         12628 non-null  float64
7   temperature  12628 non-null  float64
8   wind_speed   12628 non-null  float64
9   pressure     12628 non-null  float64
10  humidity     12628 non-null  float64
11  soil_type    12628 non-null  object
12  N            12628 non-null  float64
13  P            12628 non-null  float64
14  K            12628 non-null  float64
15  production   12496 non-null  float64
dtypes: float64(9), int64(2), object(5)
memory usage: 1.6+ MB
```

The output also shows the result of the `df.info()` command in the next cell:

```
In [8]: df.info()
```

Figure 2: Data Cleaning

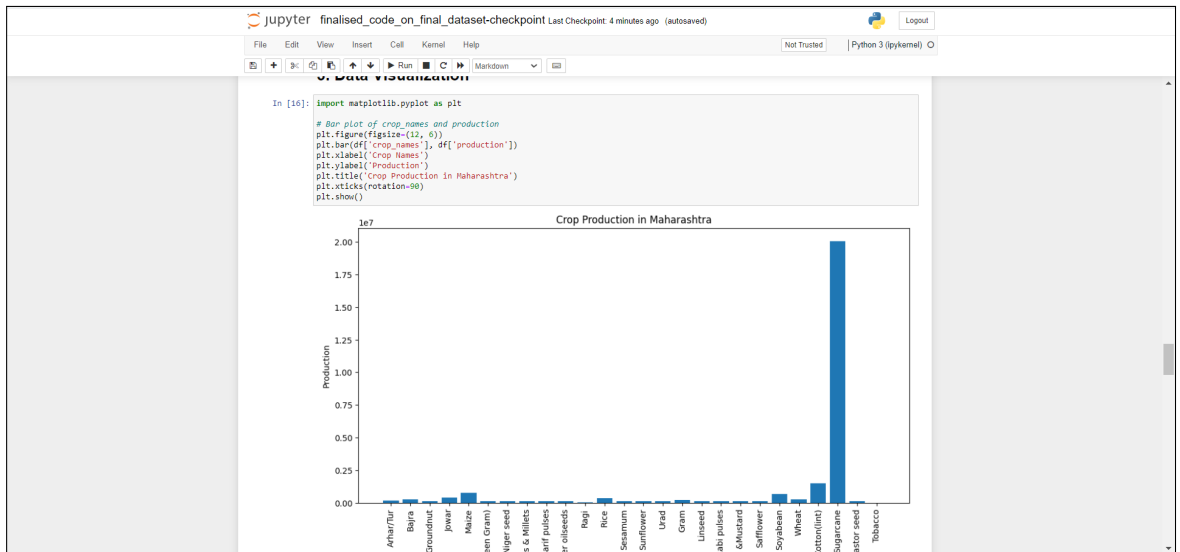


Figure 3: Data Visualization

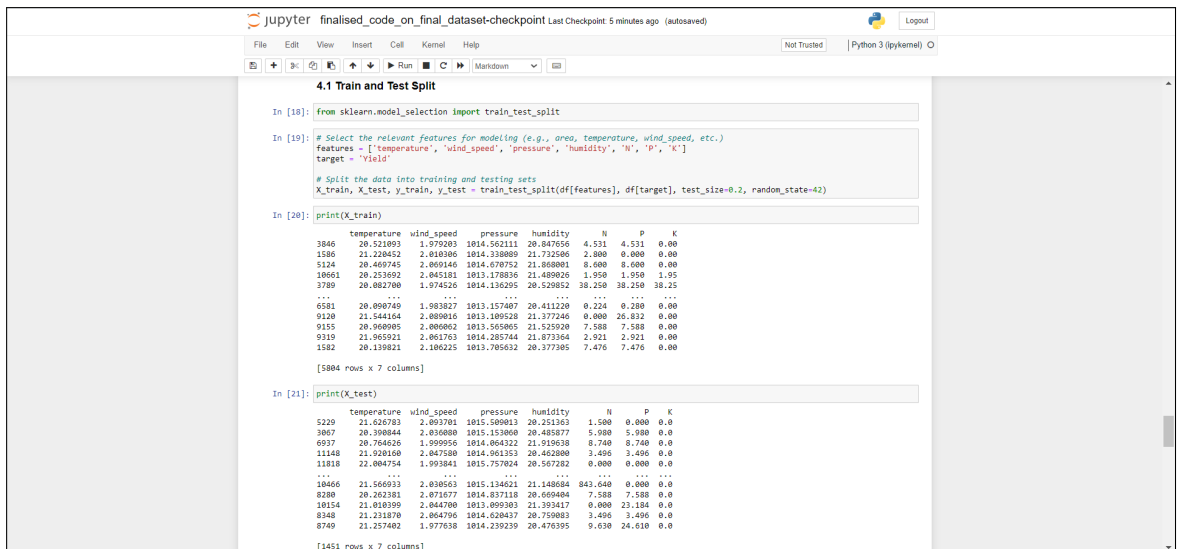
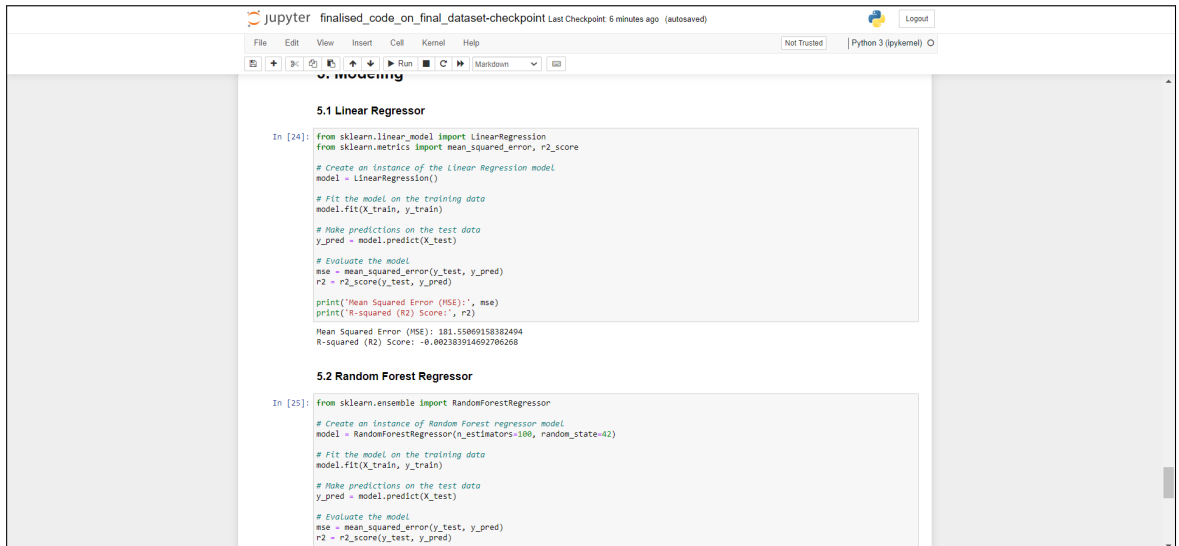


Figure 4: Data Preparing



The image shows a Jupyter Notebook interface with a menu bar (File, Edit, View, Insert, Cell, Kernel, Help) and a toolbar with icons for file operations, running, and saving. The notebook title is "finalised_code_on_final_dataset-checkpoint" with a status bar indicating "Last Checkpoint: 6 minutes ago (autosaved)" and a "Logout" button. The code is organized into two sections: "5.1 Linear Regressor" and "5.2 Random Forest Regressor".

5.1 Linear Regressor

```
In [24]: from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Create an instance of the Linear Regression model
model = LinearRegression()

# Fit the model on the training data
model.fit(X_train, y_train)

# Make predictions on the test data
y_pred = model.predict(X_test)

# Evaluate the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print('Mean Squared Error (MSE):', mse)
print('R-squared (R2) Score:', r2)

Mean Squared Error (MSE): 181.55869158382494
R-squared (R2) Score: -0.002383914692706268
```

5.2 Random Forest Regressor

```
In [25]: from sklearn.ensemble import RandomForestRegressor

# Create an instance of Random Forest regressor model
model = RandomForestRegressor(n_estimators=100, random_state=42)

# Fit the model on the training data
model.fit(X_train, y_train)

# Make predictions on the test data
y_pred = model.predict(X_test)

# Evaluate the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
```

Figure 5: Modeling

8 Conclusion

In this project, we focused on predicting crop yield in Maharashtra using various agricultural and environmental features. The dataset was preprocessed and cleaned to remove any missing values and unnecessary columns. Data visualization was performed to gain insights into crop production and yield trends.

We split the data into training and testing sets and implemented four different regression models: Linear Regression, Random Forest Regressor, Decision Tree Regressor, and Support Vector Machine (SVM) Regressor. The models were evaluated using mean squared error (MSE) and R-squared (R²) score as performance metrics.

Based on our analysis, the Random Forest Regressor demonstrated the best performance in predicting crop yield, with the lowest MSE and highest R² score. This indicates that the Random Forest model was able to capture the non-linear relationships between the selected features and the target variable, resulting in more accurate predictions.

It is important to note that the predictive accuracy of the models can be further improved by considering additional factors such as crop-specific features, weather patterns, soil characteristics, and historical data. Furthermore, domain expertise and domain-specific feature engineering can also contribute to better prediction results.

The insights gained from this project can be valuable for farmers, agricultural policymakers, and other stakeholders in making informed decisions related to crop yield optimization, resource allocation, and risk management. By accurately predicting crop yields, we can enhance agricultural productivity, optimize resource usage, and contribute to sustainable agricultural practices.