Crop Yield Analysis

Project submitted to the

SRM University, AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology/Master of Technology

In

Computer Science and Engineering

School of Engineering and Sciences

Submitted by

Soumika | AP2210011228

Lohitha | AP22110011265

Sai Geetha |AP22110010764

Vijaya Lakshmi | AP22110011392

Under the Guidance of

Dr. Raju Imandi

SRM University–AP

Neerukonda, Mangalagiri, Guntur

Andhra Pradesh – 522 240

[December, 2025]

Certificate

Date: 2-Dec-25

This is to certify that the work in this project titled **"Crop Yield Analysis"** has been done **by Soumika (AP22110011228), Lohitha (AP22110011265), Geetha (AP22110010764), and Vijaya Lakshmi (AP22110011392)** under my supervision. The work is genuine, original, and appropriate for submission to **SRM University** for the award of **Bachelor of Technology** in the School of Engineering and Sciences.

Supervisor

(Signature)

Dr. Raju Imandi

Designation,

Affiliation.

Acknowledgements

I thank my instructor for providing the necessary tools and resources, as well as being a motivational source throughout the course of this Big Data Analytics Project. Their valuable support was critical to gaining a full understanding of how to use tools from the Hadoop Ecosystem to address real-world questions about data.

I would also like to thank the lab staff for providing the Cloudera Environment, which allowed for experimentation with Hive, HDFS, and preparing data for analysis in a hands-on manner. To my fellow classmates and colleagues: Thank you for giving me ideas during our conversations that improved my work.

Finally, I would like to give credit to the creators of the Online Crop Yield Data Set because they produced such a complete and usable public dataset, which permits insights from meaningful analyses so that the information can be learned and understood. Without all of these contributions, it would not have been possible to produce the result that is being presented here.

Table of Contents

ABSTRACT (Crop Yield Analysis Project)

India's increasing population requires a growing agricultural base, as well as an understanding of how climate impacts crop growth in order for people to produce enough food. The research outlined in this report looks at how combining many different data sources impacts how well farmers will be able to produce food in India's future.

The information used in this study falls into 3 categories: cropping systems information (crop yields by state), soil nutrient data (soil nutrients N, P, and K by state/region), and historical weather data (by region). All of these datasets were cleaned and merged together in a single server using the Hadoop/Hive framework.

Once the datasets were cleaned, formatted, and combined into one Hadoop dataset, a master dataset containing 16 key datasets was created.

We have utilized Microsoft Power BI to analyze and visualize the datasets presented in this report as reports and dashboards. We discussed the relationship of temperature, relative humidity, soil nutrients (N, P, K) and rainfall amounts to yield production; as well as the development of a machine learning predictive model based on the current conditions that will enable the author to estimate yield in the future. To correct for the skewed distribution of crop yield data, the author applied a logarithmic transformation to the training datasets.

INTRODUCTION

Currently, agriculture has the largest proportion of employment in India and contributes significantly to India's economy. The current pressure of Population, Climate Change and Resource challenges on Agricultural Sectors is impacting how Farmers make decisions and utilizes resources for maximizing their yield through increased efficiency. For this reason, Data is to be used to promote enhanced decision making by assisting Farmers in making Evidence Based Decisions.

With the proliferation of data emerging from agricultural sectors, Big data analytics technologies can be applied to the enormous volume of new data, creating new Insights. By combining data about environment / climatic conditions with these new datasets (including Soil / Climate / Crop Production / etc.), a comprehensive analysis can be done of how Agricultural Performance varies by State and by Season.

The goal of the **Crop Yield Analysis Project** is to use detailed datasets from many different sectors of the Agricultural Industry to Study and understand the correlations between Nutrient Content, Climate variables and Output of Agricultural Production. To complete this Study, we adopted a Hadoop architecture which allows us to disperse and store the datasets using HDFS (Hadoop Distributed File System) and integrate them into one dataset using Hive. This ultimately allowed us analyze the data using Power BI to detect patterns, correlations, and trends within the Agricultural Sector as a whole.

The project's goals are listed in these objectives:

- To collect and organize different types of data on crop yields, soil nutrient levels, and the weather patterns related to each crop.
- To understand the effect of temperature rainfall on crop production through the collection of the aforementioned datasets along with soil nutrient data.
- To perform data preprocessing, data transformation, and exploratory analysis of this data using the tools provided by Hadoop and Power BI.
- To develop a visual representation (dashboard) that provides insights into the trends of crop yields across different States/Season/Year combinations via the use of these datasets and by correlating the various values.

We used PySpark MLlib to develop predictive models from our cleaned and merged data set. Prior to training the Linear Regression, Random Forest, Gradient Boosting Trees (GBT), and Decision Tree models, we applied feature normalization/scaling, as well as log transformation. Each of the models evaluated using RMSE (Root Mean Squared Error), R^2 (Coefficient of Determination), based on both original yields and log-transformed yields.

Through this project, we will show how different technologies (Big Data) can assist the agriculture industry in making better decisions and predicting yield outcomes with higher accuracy.

PROBLEM SURVEY

Many environmental and soil conditions influence agriculture in India. Each state's different climate will influence their climate, and farmers will need help identifying the factors that have the greatest impact on crop yield.

Major challenges include:

1. Identifying the Factors that Influence Crop Yield:

Crop yield is influenced by multiple factors including rainfall, soil nutrients, temperature, humidity, fertilizers, and pesticides. Determining the most prominent influences on crop yield will be a challenge.

2. Variation Across States and Seasons:

The states of India will have different climatic conditions (such as rain, temperature, humidity, and soil conditions). The Kharif, Rabi, and Whole Year seasons add to the variability in crop yield.

3. Impact of Weather Conditions:

Differences in rainfall, temperature, and humidity influence crop growth. So, it is important to understand how these factors have played a role in the past.

4. Soil Nutrient Analysis:

Different areas of India will have different nutrient levels (as appropriate), including Nitrogen (N), Phosphorus (P), Potassium (K), and soil pH. Farmers will benefit from knowing how nutrient levels and other soil quality indicators contribute to their productivity.

5. Evaluating Fertilizer and Pesticide Usage:

Although excessive application is not an indication of excessive production, knowing how each input impacts crop production, helps toward a more sustainable farming system.

6. Lack of Combined Agricultural Data:

Soil Data, Crop Data and Weather Data are typically stored individually. The lack of an integrated database prohibits meaningful analysis.

7. Need for Visual Dashboards:

Farmers and policy-makers require easily interpretable visual trends representing comparative crop yield data and crop inputs - to facilitate decision making based on data.

DATASET DESCRIPTION

Analysis of Crop Yields in 20+ States of India Over Multiple Years Using Data from 3 Datasets The project used three datasets to perform crop yield analysis. These datasets contained different types of agricultural data, which were compiled into one comprehensive dataset.

1. Crop Yield Data (crop_yield.csv)

Crop yield data has information on crop yields, area harvested, and the total amount of fertilizer and pesticide that was applied. The crop yield dataset includes these columns:

- **Crop:** The specific name of the crop that was grown.
- **Year:** The year when the crop was grown.
- **Season:** Kharif, Rabi, and Whole Year cropping seasons for each crop.
- **State:** The Indian state where the crop was grown.
- **Area (hectares):** The total land area that was cultivated with this crop.
- **Production (metric tons):** The number of metric tons produced of this crop.
- **Annual Rainfall (mm):** The amount of rainfall in the area where the crop was produced.
- **Fertilizer applied (kg):** The total amount of fertilizer used.
- **Pesticide applied (kg):** The total amount of pesticide used.
- **Yield:** The amount of yield produced per unit area of cultivation.

2. State-wise Soil Data (state_soil_data.csv)

The state-wise soil data shows the nutrient content of soil in each Indian state, along with the soil pH level. The data includes the following columns:

- **State:** The name of the Indian State.
- **N**: Nitrogen content of soils (in kilos per hectare); this is important for plant leaf and stem growth.
- **P:** Phosphorous content of soils (in kilos per hectare); this is vital for root development and flowering.
- **K:** Potassium Content of Soils (in kilos per hectare), which supports crop health and disease resistance.
- **pH**: Soil acid/alkaline status, the optimal range for most crops is 6.0 to 7.5.

3. State-wise Weather Data (state_weather_data_1997_2020.csv)

Contains a weather data set compiled by the Indian Meteorological Department from each Indian state by tallying yearly records between 1997 and 2020. The following info is captured for every Indian state:

- **State -** Name of Indian state
- **Year -** The year of an individual state's weather records
- **avg_temp_c -** Average annual temperature for that state in degrees Celsius
- **total_rainfall_mm -** The state's total annual rainfall measured in millimeters
- **avg_humidity_percent -** The average annual % humidity for that state (this value will affect evapotranspiration and pest/disease conditions)

Final Combined Dataset

The process of combining records from **the Weather, Crop and Soil datasets** was done in a **Hadoop/Hive database** by using a left join based on State and Year to generate the complete dataset. This combined weather/crop/soil dataset contains **16 columns** and **19,689 individual records,** which contain all parameters needed to perform an effective analysis on Crop yields within a particular Indian state (based on weather records)

- **Columns:** crop, year, season, state, area, production, fertilizer, pesticide, yield, nitrogen, phosphorus, potassium, ph, avg_temp_c, total_rainfall_mm, avg_humidity_percent

The combined dataset was created to develop and create visualizations, analyze trends and provide insight in Power BI.

DATA PREPROCESSING

The pre-processing phase of big data analytics is the first step in creating a clean and usable data set to be analyzed (cleaned, transformed, integrated). All of the data was pre-processed in this project using the various technologies available in **Hadoop HDFS, Hive, Pyspark, and Power BI.**

The process of pre-processing data involved four key steps:

1. Loading data to HADOOP HDFS

2. Creating an EXTERNAL TABLE on HIVE

3. Joining the data into a single "final" table

4. Exporting the final data for visualization using Power BI.

5. Using the final table to build a model to predict yield through the application of algorithms.

1. Uploading the Datasets into Hadoop HDFS

Hadoop services must be started before starting the analysis, including:

```
start-dfs.sh
start-yarn.sh
hdfs dfsadmin -safemode leave
jps
```

Next, create directories in HDFS to store the crop, soil, and weather datasets:

```
hdfs dfs -mkdir -p /user/hadoop/cropdata/crop
hdfs dfs -mkdir -p /user/hadoop/cropdata/soil
hdfs dfs -mkdir -p /user/hadoop/cropdata/weather
```

Place CSV files in their respective folders:

```
hdfs dfs -mv /user/hadoop/cropdata/crop_yield.csv /user/hadoop/cropdata/crop/
hdfs dfs -mv /user/hadoop/cropdata/state_soil_data.csv /user/hadoop/cropdata/soil/
hdfs dfs -mv /user/hadoop/cropdata/state_weather_data_1997_2020.csv
/user/hadoop/cropdata/weather/
```

Now the datasets are properly organized in HDFS for processing by Hive.

2. Creating Hive External Tables

Hive was started using:

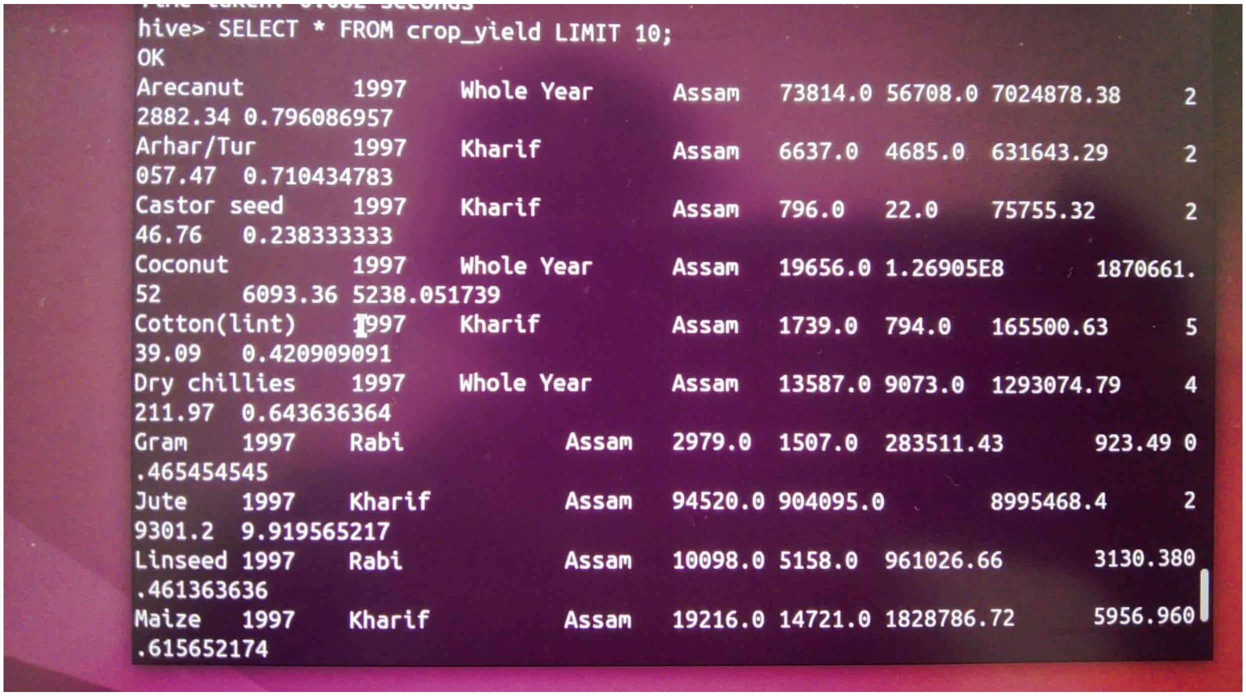
```
hive
```

Three external tables were created for crop data, soil data, and weather data. All tables used CSV format with comma delimiters, and the header row was ignored.

a. Crop Yield Table

```
CREATE EXTERNAL TABLE crop_yield (
    crop STRING,
    year INT,
    season STRING,
    state STRING,
    area DOUBLE,
    production DOUBLE,
    fertilizer DOUBLE,
    pesticide DOUBLE,
    yield DOUBLE
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
LOCATION '/user/hadoop/cropdata/crop'
TBLPROPERTIES ("skip.header.line.count"="1");


SELECT * FROM crop_yield LIMIT 10;
```

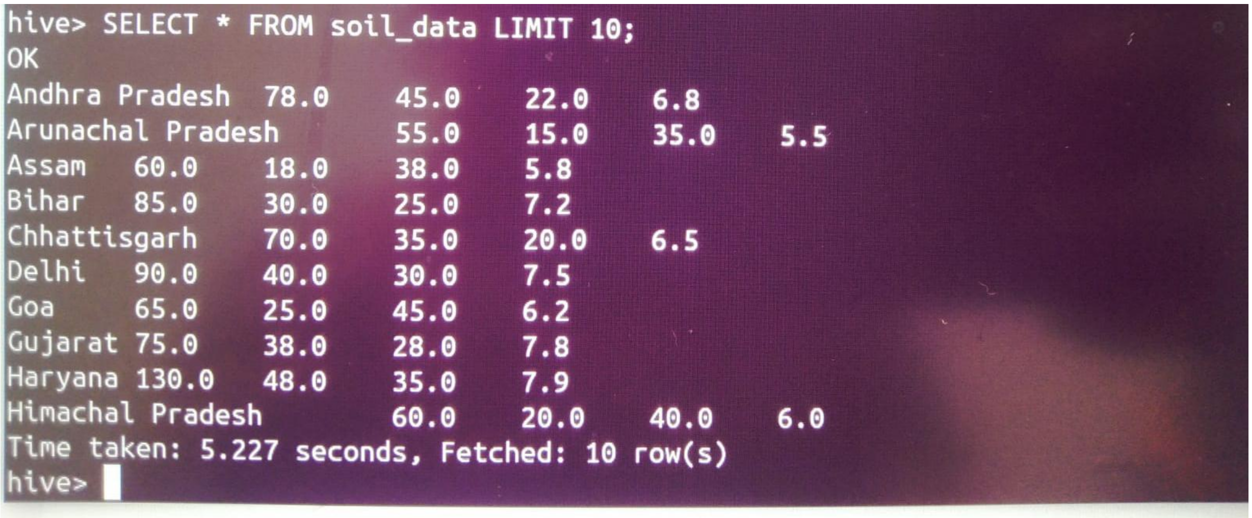


```
hive> SELECT * FROM crop_yield LIMIT 10;
OK
Arecanut        1997    Whole Year      Assam   73814.0 56708.0 7024878.38      2
2882.34 0.796086957
Arhar/Tur       1997    Kharif          Assam   6637.0  4685.0  631643.29       2
057.47  0.710434783
Castor seed     1997    Kharif          Assam   796.0   22.0    75755.32        2
46.76   0.238333333
Coconut         1997    Whole Year      Assam   19656.0 1.26905E8       1870661.
52      6093.36 5238.051739
Cotton(lint)    1997    Kharif          Assam   1739.0  794.0   165500.63       5
39.09   0.420909091
Dry chillies    1997    Whole Year      Assam   13587.0 9073.0  1293074.79      4
211.97  0.643636364
Gram    1997    Rabi            Assam   2979.0  1507.0  283511.43       923.49 0
.465454545
Jute    1997    Kharif          Assam   94520.0 904095.0        8995468.4       2
9301.2  9.919565217
Linseed 1997    Rabi            Assam   10098.0 5158.0  961026.66       3130.380
.461363636
Maize   1997    Kharif          Assam   19216.0 14721.0 1828786.72      5956.960
.615652174
```

b. Soil Data Table

```
CREATE EXTERNAL TABLE soil_data (
    state STRING,
    nitrogen DOUBLE,
    phosphorus DOUBLE,
    potassium DOUBLE,
    ph DOUBLE
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
LOCATION '/user/hadoop/cropdata/soil'
TBLPROPERTIES ("skip.header.line.count"="1");

SELECT * FROM soil_data LIMIT 10;
```

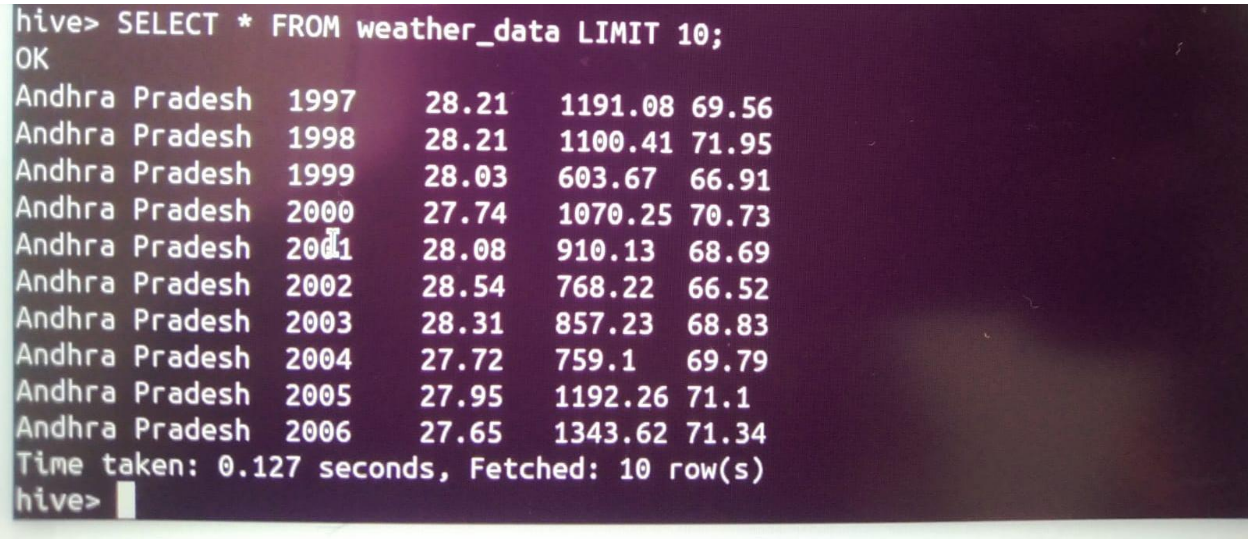


```
hive> SELECT * FROM soil_data LIMIT 10;
OK
Andhra Pradesh  78.0    45.0    22.0    6.8
Arunachal Pradesh       55.0    15.0    35.0    5.5
Assam   60.0    18.0    38.0    5.8
Bihar   85.0    30.0    25.0    7.2
Chhattisgarh    70.0    35.0    20.0    6.5
Delhi   90.0    40.0    30.0    7.5
Goa     65.0    25.0    45.0    6.2
Gujarat 75.0    38.0    28.0    7.8
Haryana 130.0   48.0    35.0    7.9
Himachal Pradesh        60.0    20.0    40.0    6.0
Time taken: 5.227 seconds, Fetched: 10 row(s)
hive> 
```

c. Weather Data Table

```
CREATE EXTERNAL TABLE weather_data (
    state STRING,
    year INT,
    avg_temp_c DOUBLE,
    total_rainfall_mm DOUBLE,
    avg_humidity_percent DOUBLE
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
LOCATION '/user/hadoop/cropdata/weather'
TBLPROPERTIES ("skip.header.line.count"="1");

SELECT * FROM weather_data LIMIT 10;
```



```
hive> SELECT * FROM weather_data LIMIT 10;
OK
Andhra Pradesh  1997    28.21   1191.08 69.56
Andhra Pradesh  1998    28.21   1100.41 71.95
Andhra Pradesh  1999    28.03   603.67  66.91
Andhra Pradesh  2000    27.74   1070.25 70.73
Andhra Pradesh  2001    28.08   910.13  68.69
Andhra Pradesh  2002    28.54   768.22  66.52
Andhra Pradesh  2003    28.31   857.23  68.83
Andhra Pradesh  2004    27.72   759.1   69.79
Andhra Pradesh  2005    27.95   1192.26 71.1
Andhra Pradesh  2006    27.65   1343.62 71.34
Time taken: 0.127 seconds, Fetched: 10 row(s)
hive>
```

3. Joining All Datasets to Create the Final Table

Using the Hive platform, the analysis of the 3 datasets was performed through a combination of the datasets using the following keys:

- **state** (for every dataset.)
- **year** (between crop data and weather data)

The creation of the final Schema was completed with **a Left Join** on crop records to ensure the inclusion of every crop record in the final dataset:

```
CREATE TABLE crop_model_data AS
SELECT
    c.crop,
    c.year,
    c.season,
    c.state,
    c.area,
    c.production,
    c.fertilizer,
    c.pesticide,
    c.yield,
    s.nitrogen,
    s.phosphorus,
    s.potassium,
    s.ph,
    w.avg_temp_c,
    w.total_rainfall_mm,
    w.avg_humidity_percent
FROM crop_yield c
LEFT JOIN soil_data s ON c.state = s.state
LEFT JOIN weather_data w ON c.state = w.state AND c.year = w.year;
```

The table structure was confirmed using:

```
DESC crop_model_data;
```

This table contained **16 columns** and **19,689 rows**.

4. Exporting the Final Dataset from Hive to Local System

The final dataset was exported to the local filesystem in order to be used within Power BI for visualisation purposes. The data was exported from Hive as follows:

```
INSERT OVERWRITE LOCAL DIRECTORY '/home/hadoop/export_crop_data'
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
SELECT * FROM crop_model_data;
```

All the part files that had been generated were subsequently added together into a single CSV file as follows:

```
cat /home/hadoop/export_crop_data/* > /home/hadoop/crop_model_data.csv
```

A header row was added:

```
echo "crop,year,season,state,area,production,fertilizer,pesticide,yield,nitrogen,phosphorus,potassium,ph,avg_temp_c,total_rainfall_mm,avg_humidity_percent" | cat - /home/hadoop/crop_model_data.csv > /home/hadoop/crop_model_data_with_header.csv
```

To verify that the newly created csv file was formatted correctly, the following command was run:

```
head /home/hadoop/crop_model_data_with_header.csv
```

Once verified, this csv file can now be imported into both **Power BI and Pyspark** for further data analysis.

5. Data Cleaning in Power BI

Once the dataset was imported:

- No NULL values were detected
- No duplicate rows existed
- Columns with crop, state and season information were converted to text/string data types
- Numerical column data types remained as whole numbers or decimals.

At this point the preprocessing phase was complete.

6. Preprocessing in PySpark

1. **Data Loading:**
   Load the dataset from a CSV file into PySpark via the use of automatic schema inference.
2. **Cleanup of Old Columns:**
   Delete any previously created index/vector columns as they will be recreated and will create duplicate entries when running the code again.
3. **Outlier Removal:**
   Remove extreme yield values (greater than 100) to help decrease the amount of noise in the dataset which may negatively affect the stability of the model.
4. **Categorical Encoding:**
   Encode the categorical variables of crop, season, and state using the following methods:
   - StringIndexer (for converting each variable to a numerical value)

- OneHotEncoder (for converting each variable to a row of vectors: crop_vec, season_vec, and state_vec).

5. **Feature Engineering:**

   New features are computed:

   - yield/area (for use in calculating yield).
   - Average humidity/Average temperature (for use in calculating humidity to temperature ratios).
   - Rainfall * pH (for use in calculating rain to pH Ratios).

   Also, create the log-transformed target variable (yield_log) to reduce skewness.

6. **Vector Assembly:**

   Assemble all numerical, engineered, and encoded numeric values into one vector of features

7. **Feature Scaling:**

   Scale the features to create scaled_features for model performance based on mean and standard deviation.

8. **Pipeline Creation:**

   All preprocessing steps (indexing → encoding → assembling → scaling) are chained into one Spark ML Pipeline and applied to the full dataset.

4. Implementation

The crop yield analysis application was executed in **Power BI** and processed the following unique datasets:

(a) crop production,

(b) crop area,

(c) weather conditions,

(d) soil nutrients,

(e) seasonal based.

During the execution phase, Power BI was able to analyze crops based on multiple perspectives by utilizing **DAX Measure, Creating Calculated Fields, Creating Interactive Visuals, and Creating Dashboards.**

4.1 Data Preparation

The power BI platform was loaded with the dataset in CSV format. The next preprocessing tasks were completed:

- Handling missing values for Rainfall, Pesticide, Fertilizer, Humidity.
- Changing the DataTypes of Year, Production and Area.
- Creating new fields (Calculated Yield).
- Filter out Duplicate Records.
- Verify the Categorical Fields of State, Crop, and Season.

4.2 DAX Measures Used

DAX (Data Analysis Expression) is responsible for the analytical capabilities of the dashboards.

The following is a listing of the areas of Major Measures developed during execution of the project.

4.2.1 Production-Related Measures

Total Production

```
Total Production = SUM(crop_model_data[production])
```

Total Area

```
Total Area = SUM(crop_model_data[area])
```

Average Yield / Calculated Yield

When yield was already present:

```
Average Yield = AVERAGE(crop_model_data[yield])
```

If yield needed to be calculated:

```
Calculated Yield =
DIVIDE(
    SUM(crop_model_data[production]),
    SUM(crop_model_data[area])
)
```

4.2.2 Yield Analysis Measures

Crop-wise Avg Yield

```
Crop Avg Yield =
AVERAGEX(
    VALUES(crop_model_data[crop]),
    [Calculated Yield]
)
```

State-wise Avg Yield

```
State Avg Yield =
AVERAGEX(
    VALUES(crop_model_data[state]),
    [Calculated Yield]
)
```

4.2.3 Soil Health Measures

Average Nutrients

```
Avg Nitrogen = AVERAGE(crop_model_data[nitrogen])
Avg Phosphorus = AVERAGE(crop_model_data[phosphorus])
Avg Potassium = AVERAGE(crop_model_data[potassium])
Avg PH = AVERAGE(crop_model_data[ph])
```

Soil Fertility Index

```
Soil Fertility Index =
AVERAGE(crop_model_data[nitrogen]) * 0.4 +
AVERAGE(crop_model_data[phosphorus]) * 0.3 +
AVERAGE(crop_model_data[potassium]) * 0.3
```

Nitrogen vs Yield Supportive Correlation

```
N-Yield =
SUMX(
    crop_model_data,
    crop_model_data[nitrogen] * crop_model_data[yield]
)
```

4.2.4 Weather Impact Measures

Avg Temperature

```
Avg Temp = AVERAGE(crop_model_data[avg_temp_c])
```

Total Rainfall

```
Total Rainfall = SUM(crop_model_data[total_rainfall_mm])
```

Avg Humidity

```
Avg Humidity = AVERAGE(crop_model_data[avg_humidity_percent])
```

Rainfall Efficiency

```
Rainfall Efficiency = DIVIDE([Total Production], [Total
Rainfall])
```

4.2.5 Input Efficiency Measures

Fertilizer Efficiency

```
Fertilizer Efficiency =
DIVIDE(
    [Total Production],
    SUM(crop_model_data[fertilizer])
)
```

Pesticide Efficiency

```
Pesticide Efficiency =
DIVIDE(
    [Total Production],
    SUM(crop_model_data[pesticide])
)
```

4.2.6 Ranking Measures

State Ranking

```
State Yield Rank =
RANKX(
    ALL(crop_model_data[state]),
    [State Avg Yield],
    ,
    DESC)
```

Crop Yield Ranking

```
Crop Yield Rank =
RANKX(
    ALL(crop_model_data[crop]),
    [Crop Avg Yield]
)
```

4.3 Dashboards Created

Three interactive dashboards have been created.

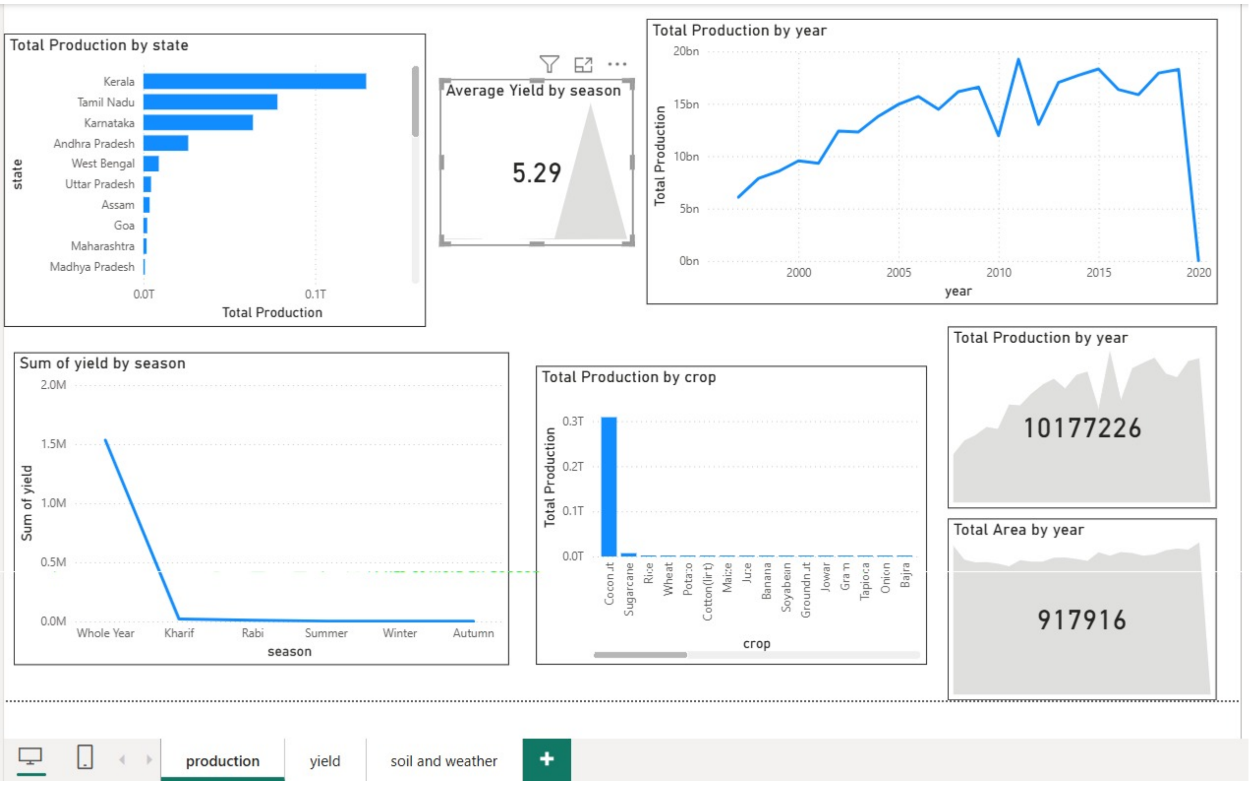
Dashboard 1: Production & Yield Overview

Contains:

- Total Production by state
- Total Production by year
- Average Yield by year
- Total Production by crop

- Season-wise yield
- KPI cards: Total Production, Total Area, Avg Yield

This dashboard allows understanding of long-term fluctuations in yield and production.



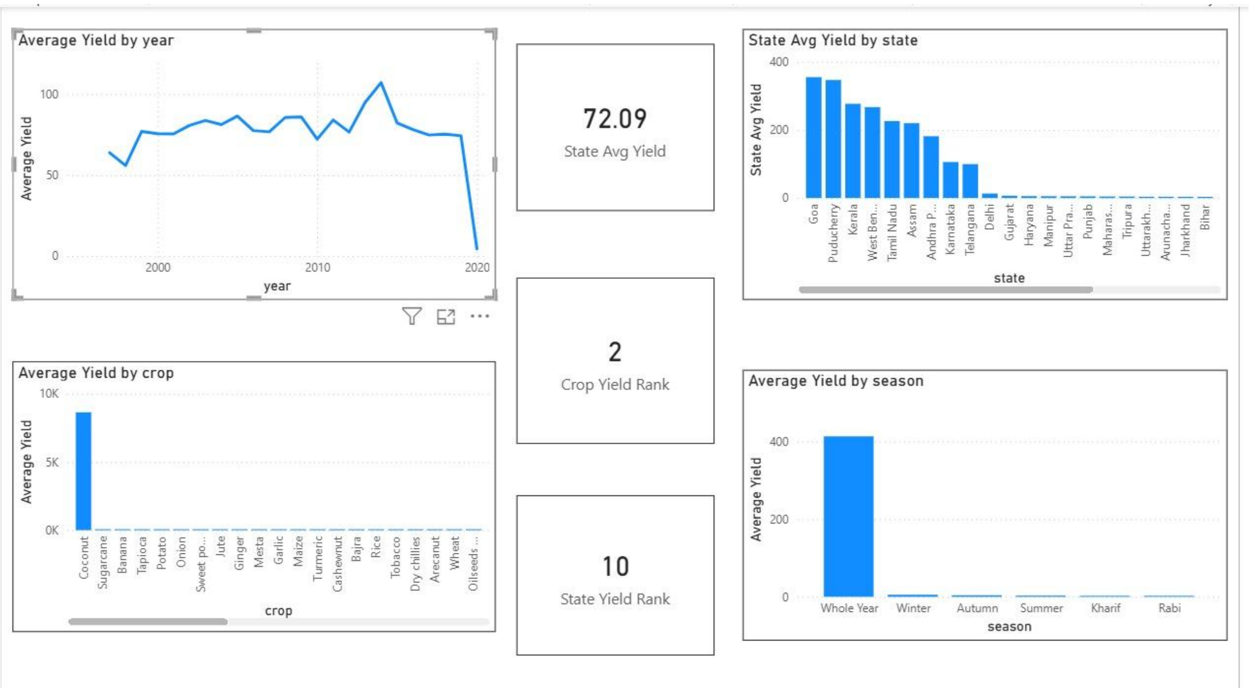
Dashboard 2: Yield Ranking and Crop Performance

Includes:

- State-wise average yield
- Crop yield rank
- State yield rank
- Cropwise average yield

- Season-wise average yield

This dashboard shows which crops and states perform best.

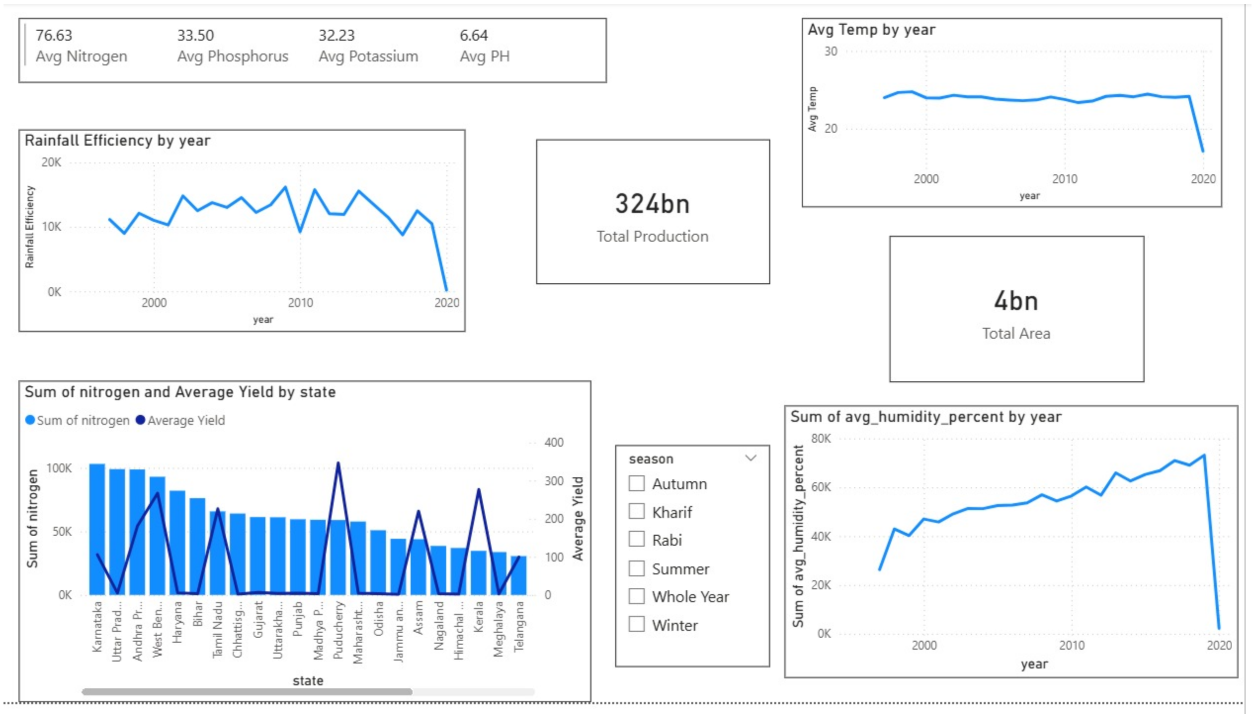


Dashboard 3: Soil & Weather Impact Analysis

Contains:

- Avg Nitrogen, Phosphorus, Potassium, pH
- Rainfall efficiency
- Avg temperature trends
- Avg humidity trends
- Nitrogen vs Yield comparison by state
- Total production & area KPIs

Useful for understanding environmental impacts.



76.63 Avg Nitrogen

33.50 Avg Phosphorus

32.23 Avg Potassium

6.64 Avg PH

324bn Total Production

4bn Total Area

season

☐ Autumn

☐ Kharif

☐ Rabi

☐ Summer

☐ Whole Year

☐ Winter

Crop Yield Analysis Dashboard

- **State-wise Production:** Overall State-wise Yields (for Kerala, Tamil Nadu, and Karnataka) show the highest cumulative Crops produced.

- **Year-wise Production:** Aggregate Yields (all years combined) show a continuing upward trend until about 2015; a sharp drop in total Yields occurred in 2020-2022.

- **Crop-wise Analysis:** Specific Crop Type Yields show that Coconut has the highest cumulative Yields of the different types of crops.

- **Yield Trends**: Coconut; Sugarcane & Cassava yield on average stable daily rates.

- **Seasonal Analysis:** The average Yield by the season (Spring, Fall, Summer) is greatest in Spring and least in Fall.

- **Soil & Fertilizer Impact:** Higher concentrations of Nitrogen, Phosphorus, & Potassium (found in Soils) proportionally correlate with higher Crop yields.

- **Weather Impact:** Environmental influences (weather) affect Crop Yields due to the average Weather conditions affecting Yield.

- **Efficiency Metrics**: The average daily efficiencies of Rain & Humidity affect Crop Performance significantly. In addition, the extreme differences in Water & Humidity levels for Yields were evident in 2020 (when producing relatively difficult conditions).

- **Insights for Planning:** The Crop Performance Visualization Dashboard provides a unique perspective & example for developing the best performing states, crops, seasons, and method for intervention regarding Crop planning, allocation of resources, etc.

5.Machine Learning Model Implementation

1. Feature Engineering & Preprocessing

The following operations were performed:

- Outliers above 100 in the yield column were eliminated. (yield ≤ 100 retained).
- Categorical variables (crop, season, state) were encoded with StringIndexer and OneHotEncoder.
- Logarithmic transformation was applied to yield in order to decrease skewness.
- Creation of engineered features:
  - yield_per_area = production / area
  - humidity_temp_ratio = avg_humidity_percent / avg_temp_c
  - rainfall_ph_interaction = total_rainfall_mm * ph
- Feature assembly using *VectorAssembler*.
- Feature scaling using *StandardScaler* to generate scaled_features.

All preprocessing steps were combined using a PySpark Pipeline.

2. Models Used

The following four regression models were trained:

1. Linear Regression
2. Random Forest Regressor
3. Gradient Boosted Tree (GBT) Regressor
4. Decision Tree Regressor

Each model was trained on 80% of the dataset and tested on the remaining 20%.

Performance was evaluated using:

- RMSE (Root Mean Squared Error)
- R^2 (Coefficient of Determination)

Metrics were calculated on both:

- Log scale (using yield_log)
- Original scale

3.Results Obtained

Model Used	log scale		original scale	
	RMSE	R^2	RMSE	R^2
Linear Regression	0.2824	0.8861	10.2053	-0.0558
Random Forest	0.1849	0.9512	3.0573	0.9052
Gradient Boosted Tree	0.0905	0.9883	1.4205	0.9795
Decision Tree	0.1114	0.9823	2.5207	0.9356

This analysis showed that the best-performing model was the **GBT Regressor**, which performed well in identifying complex non-linear relationships between soil properties, weather and yield.

5. Conclusion from PySpark Models

- Ensemble learning (Random Forests and Gradient Boosted Trees) provided greater accuracy than linear regression methods to forecast agriculture yield.
- The Gradient Boosted Tree Regressor (GBT) predicted the most accurately with almost 98% of its variance explained when tested.
- Major predictors of crop yield include weather variables (precipitation, humidity and temperature), and soil properties (N, P and K and pH).

- Machine Learning using PySpark MLlib was scalable to large datasets generated from Hadoop and Hive.

Future Work

As a continuity of integrated analysis for crop yield derived from soil, weather and cultivation data, additional improving methods for prediction accuracy and more detailed analytical results are recommended for future use:

1. Inclusion of Satellite-Based Remote Sensing Data

Environmental Modelling, and Detection of Drought/Crop Stress By adding additional satellite data (NDVI, Soil Moisture, and Land Surface Temperature), we can now monitor the following in real time:

- Track real-time vegetation health
- Improve environmental modelling
- Detect drought and crop stress more accurately

2. Seasonal Forecasting Using Time-Series Models

Using historical time series analysis (ARIMA, LSTM networks), we can generate future forecasts regarding the following:

- Crop production trends
- Rainfall and temperature variations
- Long-term yield fluctuations

3. Expansion to District-Level or Farm-Level Data

Using more granular datasets can enhance model precision and help in:

- Micro-level yield estimation
- Identifying region-specific challenges
- Delivering localized recommendations

4. Development of a Farmer-Friendly Dashboard / Web Portal

A web interface or mobile app can be built to:

- Show farmers yield predictions
- Share weather alerts and soil recommendations
- Provide guidelines for fertilizer and pesticide use based on analytics

5. Automation of the End-to-End Data Pipeline

Using tools like Apache Spark, Airflow, and Kafka to build a fully automated pipeline:

- Continuous Ingestion of Data
- Real-time Updates of Models
- Automated Dashboards in Power BI

6. Incorporation of Economic and Market Data

Future work will include the following:

- Market Price Forecasting
- Profitability Modelling

- Supply Chain Optimisation

References

1. **Power BI Documentation – Microsoft Learn** – Basics of dashboards, DAX, and data modeling.
2. **Kaggle – Agricultural Crop Yield Datasets** (for understanding data formats and analysis approaches).