Industrial Internship Report on
” Prediction of Agriculture Crop Production in India”

Prepared by
Rishabh Dwivedi, Sweta Yadav, Meenakshi Gupta

Executive Summary

This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT).
This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 6 weeks’ time.
This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship.

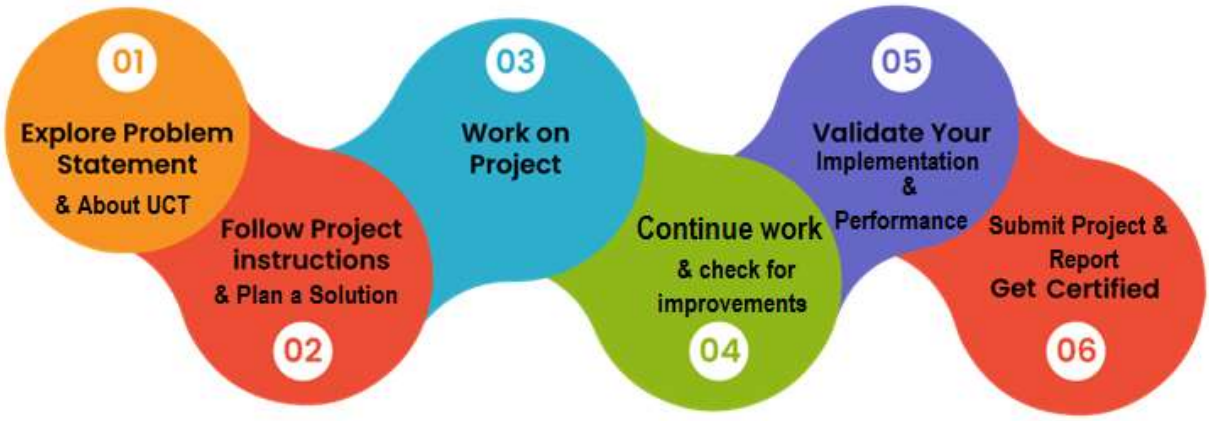
TABLE OF CONTENTS

Preface

India is the second largest country having more than 1.3 billion people and many people here are dependent on agriculture as their primary source of income depends on it. In Agriculture during cultivation and production many problems arises, therefore goal of the project is to analyse these problems and try to solve or reduce these problems.

Crop yield prediction is an important aspect of agriculture that helps farmers make informed decisions about their crops. This involves estimating the number of crops that will be produced in a given area based on various factors such as soil type, weather conditions, and crop management practices. In recent years, data science and machine learning (ML) has emerged as a powerful tool for predicting crop yields.

Data science is the study of data to extract meaningful insights for business. It is a multidisciplinary approach that combines principles and practices from the fields of mathematics, statistics, artificial intelligence, and computer engineering to analyse large amounts of data. This analysis helps data scientists to ask and answer questions like what happened, why it happened, what will happen, and what can be done with the results. Data science is important because it combines tools, methods, and technology to generate meaning from data, whereas machine learning is a branch of artificial intelligence (AI) that allows computers to learn from data without being explicitly programmed. This makes it ideal for crop yield prediction because it can identify patterns and relationships in large amounts of data and make predictions based on these relationships.



Introduction

1.1 About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in Digital Transformation domain and providing Industrial solutions with prime focus on sustainability and RoI.
For developing its products and solutions it is leveraging various **Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end** etc.

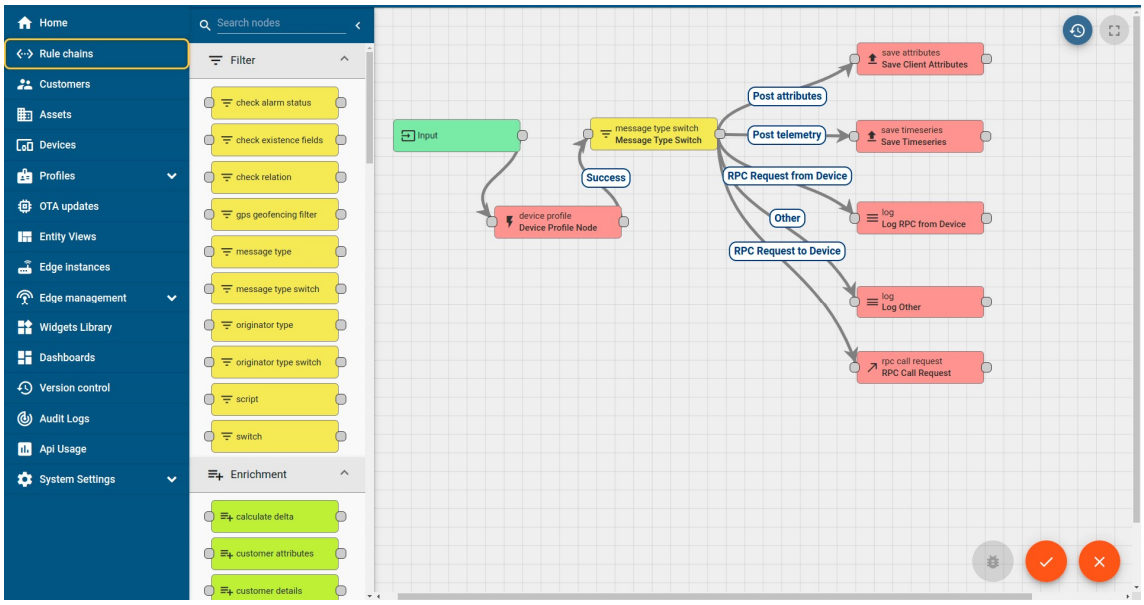
i. UCT IoT Platform (UCT Insight)

UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA
- It supports both cloud and on-premises deployments.

It has features to
• Build Your own dashboard
• Analytics and Reporting
• Alert and Notification

• Integration with third party application(Power BI, SAP, ERP)
• Rule Engine

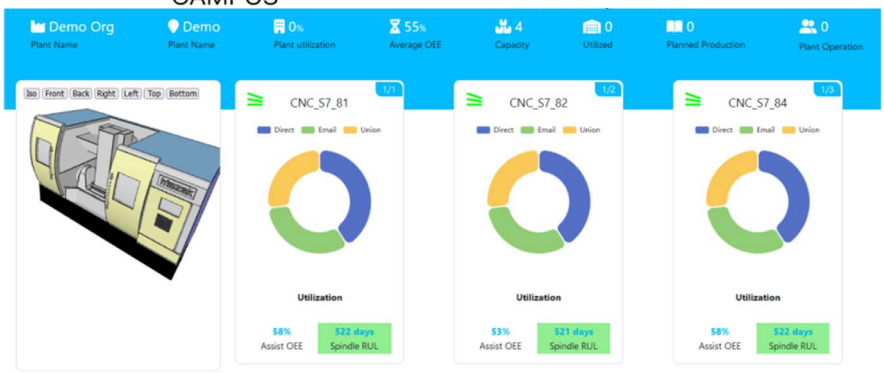


ii. Smart Factory Platform (FACTORY WATCH)

Factory watch is a platform for smart factory needs.
It provides Users/ Factory

- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.
- to unleased the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
- A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users to save time, cost and money.



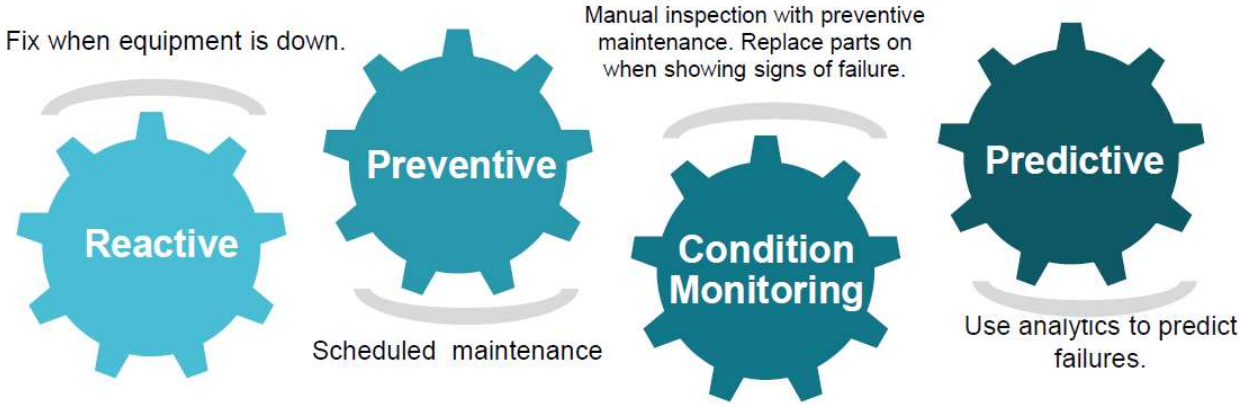
Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i

iii. LoRaWAN based Solution

UCT is one of the early adopters of LoRAWAN teschnology and providing solution in Agritech, Smart cities, Industrial Monitoring, Smart Street Light, Smart Water/ Gas/ Electricity metering solutions etc.

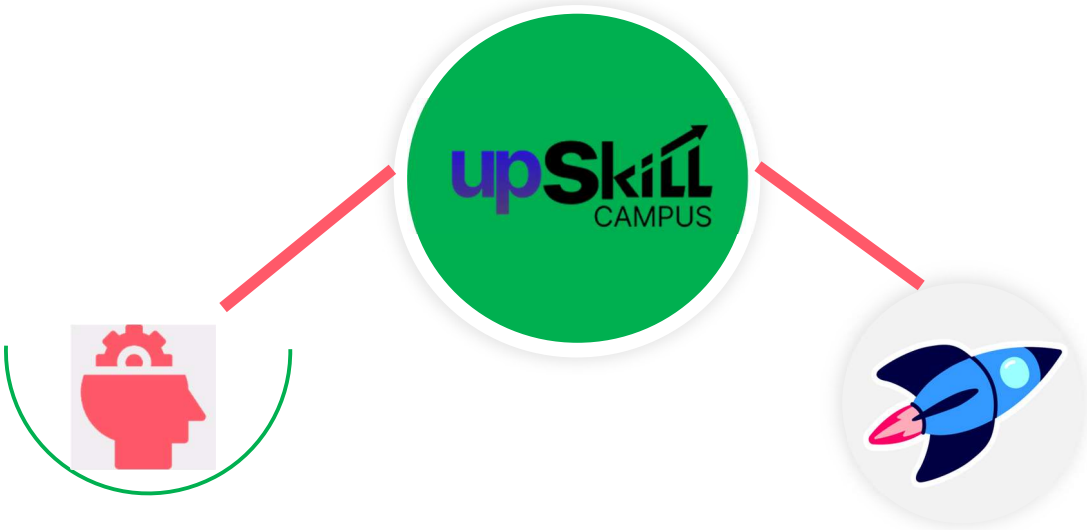
iv. Predictive Maintenance

UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



1.2 About upskill Campus (USC)

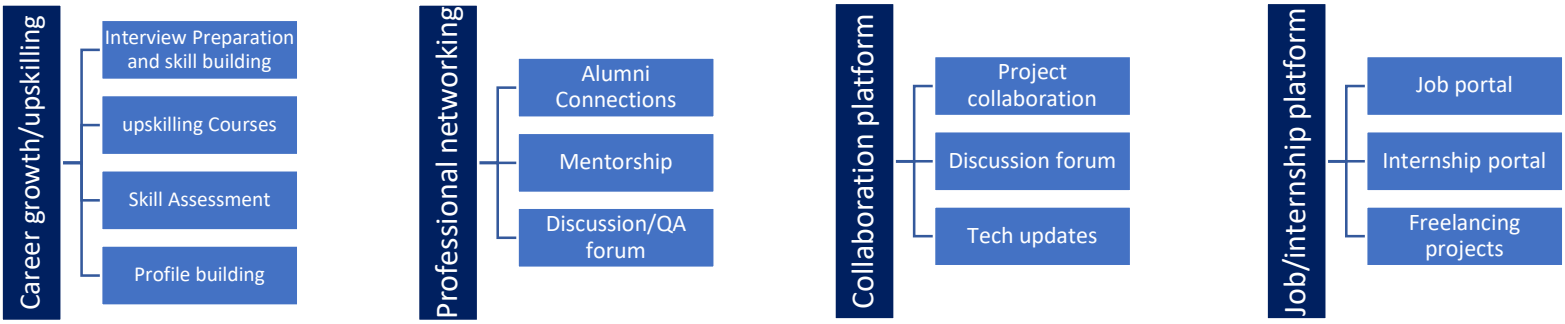
upskill Campus along with The IoT Academy and in association with Uniconverge technologies has facilitated the smooth execution of the complete internship process. USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



Seeing need of upskilling in self paced manner along-with additional support services e.g. Internship, projects, interaction with Industry experts, Career growth Services

upSkill Campus aiming to upskill 1 million learners in next 5 year

https://www.upskillcampus.com/



1.3 The IoT Academy

The IoT academy is EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

1.4 Objectives of this Internship program

The objective for this internship program was to

- ☛ get practical experience of working in the industry.
- ☛ to solve real world problems.
- ☛ to have improved job prospects.
- ☛ to have Improved understanding of our field and its applications.
- ☛ to have Personal growth like better communication and problem solving.

ABSTRACT

The agricultural sector in India plays a crucial role in ensuring food security and driving economic growth for its large population. However, farmers face significant challenges due to unpredictable weather patterns, limited resources, and evolving market demands. To overcome these obstacles and optimize crop production, data science and machine learning (ML) techniques offer promising solutions.

This report provides a comprehensive analysis of the role of data science and ML in agricultural crop production in India. It begins by examining the current landscape of agriculture in India and emphasizes the need for innovative approaches to enhance productivity, sustainability, and profitability. The report explores the potential applications of data science and ML, including predictive modelling, yield estimation, disease detection, and crop recommendation systems.

Furthermore, the report delves into various data sources that can be utilized, such as satellite imagery, weather data, soil information, and historical crop yield records. It discusses the challenges associated with data collection, pre-processing, and integration, underscoring the significance of data quality and reliability in obtaining accurate and actionable insights.

Next, the report investigates different ML algorithms and techniques suitable for agricultural crop production, such as regression models, support vector machines, decision trees, and neural networks. It explores their strengths, limitations, and applicability to specific agricultural contexts.

Moreover, the report examines case studies and real-world data science and ML applications in the Indian agriculture sector. These examples highlight successful implementations in optimizing irrigation schedules, predicting crop diseases, optimizing fertilizer usage, and improving crop yield predictions.

In conclusion, the report discusses the potential impact of data science and ML on India's agriculture sector, including increased productivity, reduced resource wastage, and improved decision-making for farmers. It also identifies key challenges and future research directions, emphasizing the importance of interdisciplinary collaboration and policy support to ensure the widespread adoption of these technologies.

In summary, this report provides a comprehensive overview of integrating data science and ML techniques in agricultural crop production in India. By leveraging data-driven approaches, Indian farmers and policymakers can make informed decisions, optimize resource allocation, and ensure sustainable and efficient agricultural practices amidst evolving challenges.

Keywords- Agriculture, Data Science, Machine Learning, crop.

1. INTRODUCTION

Agricultural crop production plays a crucial role in India's economy, ensuring an ample food supply and supporting the livelihoods of millions of people. Given India's diverse climate and vast agricultural landscape, there are unique challenges to overcome in order to optimize crop yields and improve agricultural practices. One promising avenue for gaining insights into crop production patterns and making informed decisions is the application of data science and machine learning techniques.

This report analyses a comprehensive dataset on agricultural crop production in India. The dataset from Kaggle provides a wide range of information covering different crops, regions, and time periods. Using data science and machine learning, we aim to uncover meaningful patterns, trends, and relationships that deepen our understanding of crop production dynamics in India.

The dataset consists of detailed records documenting crop production quantities, seasonal variations, cost of cultivation per hectare, yield, and other relevant parameters. By exploring this dataset and employing various analytical techniques, we can gain valuable insights into the

factors that influence crop production, identify regions with high yields, and identify opportunities to improve agricultural practices.

This project addresses essential questions such as: How does crop production vary across different regions of India? Are there significant fluctuations in crop yields over the years? Can machine learning models effectively predict crop production based on environmental and regional factors? By addressing these questions, we aim to contribute to the existing agricultural research body and support evidence-based decision-making to foster sustainable crop production in India.

The subsequent sections of this report will delve into the dataset, conduct exploratory data analysis, analyze temporal and regional patterns, explore trends specific to different crops, and discuss the potential of machine learning models in predicting crop yields. By synthesizing these findings, we aim to provide valuable insights into agricultural crop production in India and emphasize the importance of data-driven approaches for optimizing agricultural practices.

2. DATASET

The dataset used in this project was obtained from Kaggle and is titled "Agriculture Crops Cultivation/Production in India." It is a licensed dataset sourced from https://data.gov.in/.
The dataset provides a comprehensive overview of India's cultivated and produced crops. It includes multiple columns that capture essential information about various crop cultivation and production aspects. These columns include:

1. **Crop:** This column contains the name of the crop as a text value.

2. **Variety**: This column specifies the specific variety or subtype of the crop and is stored as a text value.

3. **State:** This column indicates the Indian state where the crop is cultivated or produced and is stored as a text value.

4. **Quantity:** This column represents the crop yield, measured in quintals per hectare, and is stored as an integer value.

5. **Production:** This column records the total production of the crop over a specific time and is stored as an integer value.

6. **Season**: This column indicates the duration of the crop season, measured in medium and long durations, using a DateTime data type.

7. **Unit:** This column denotes the unit of measurement for crop production, typically in tons, and is stored as a text value.

8. **Cost:** This column captures the cost of cultivating and producing the crop and is stored as an integer value.

9. **Recommended Zone:** This column provides information on the recommended cultivation zones for specific crops, including details such as the state, Mandal, and village. It is stored as a text value.

The dataset comprises five sub-datasets, each stored as a separate CSV file. Here is a description of each data file:

1. **Crop Variety Dataset:**
    - This dataset contains information about various crop varieties cultivated in India, including cultivation costs and monthly production per hectare.
    - It provides insights into crop yields in quintals per hectare, offering valuable information about productivity and profitability.
    - Analyzing this dataset enables a deeper understanding of crop distribution across different regions and associated cultivation costs, facilitating informed decision-making in the agricultural sector.

2. **Crop Yield and Production Dataset (2006-07 to 2010-11):**
    - This dataset covers five years, from 2006-07 to 2010-11, focusing on crop yields, production, and the required cultivation area.
    - It provides comprehensive information on annual crop yields, production quantities, and land utilization patterns.
    - Analyzing this dataset over time helps identify trends, patterns, and potential factors influencing crop productivity and production levels.

3. **Crop Varieties and Recommended Zones Dataset:**
    - This dataset offers valuable agricultural planning and decision-making information by providing details on different crop varieties and their optimal cultivation zones.
    - It assists farmers in making informed choices regarding crop selection based on factors such as soil conditions, climate, and geographic suitability.

4. **Crop Yield Dataset (2006-2011):**
    - This dataset provides a detailed representation of crop yields from 2006 to 2011.
    - It offers comprehensive data on crop quantities harvested each year, enabling analysis of variations in crop yield and identification of potential correlations with environmental factors and agricultural practices.

5. **Crop Production Dataset (2006-2011):**
    - The crop production dataset presents total crop production quantities in tonnes/months from 2006 to 2011.
    - It offers insights into overall agricultural output and production trends during this period.
    - Analyzing this dataset allows researchers and policymakers to identify patterns, shifts, and potential areas for improvement in crop production.

These five datasets collectively provide a comprehensive and detailed representation of agricultural production in India. Analyzing the various aspects of crop variety, cultivation costs, production rates, yield quantities, recommended zones, and production trends can better understand agricultural practices and support evidence-based decision-making in the sector.
By utilizing this dataset, the report aims to explore and analyze the cultivation and production patterns of crops in India. The insights derived from this dataset will contribute to a deeper understanding of agricultural practices and can inform decision-making processes in the agricultural sector.

3. PRE-PROCESSING OF DATASET

This section outlines the methodology to pre-process the dataset before conducting exploratory data analysis (EDA) and building machine learning (ML) models. Pre-processing is crucial to ensure data quality, compatibility with ML algorithms, and the ability to derive meaningful insights. The following steps were taken during this phase:

1. Data Cleaning:

- Duplicate Removal: Duplicate records in the dataset were identified and removed to maintain data integrity.
- Handling Missing Values: Missing values were carefully examined and addressed accordingly. Records with missing values were removed using appropriate

techniques, such as mean, median, or regression imputation, to retain necessary information.

- Outlier Detection and Treatment: Outliers were identified using statistical methods. Depending on the nature of the data and analysis objectives, outliers were removed or transformed using suitable techniques to minimize their impact on the analysis.

2. Data Integration:

- Relevant Datasets Combination: If multiple datasets were available, we integrated them based on standard fields or keys to create a comprehensive analysis dataset. This integration ensured that all relevant information was included in the final dataset.
- Inconsistency Resolution: Inconsistencies, such as different naming conventions or encoding formats for categorical variables, were addressed to ensure data consistency across the integrated dataset.

3. Data Transformation:

- Feature Scaling: Numerical features were normalized or standardized to ensure they were on a similar scale. This step prevented any particular feature from dominating the analysis and ensured fair comparisons.
- Handling Categorical Variables: Categorical variables were transformed into numerical representations suitable for ML algorithms. Techniques like one-hot encoding or label encoding were applied based on the nature of the categorical variables.
- Feature Engineering: New features were created, and meaningful information was extracted from existing features to enhance the predictive power of the ML models. This step involved domain knowledge and careful feature selection to create informative and relevant features.

4. Feature Selection:

- Relevant Feature Identification: The correlation between features and the target variable was analyzed to identify the most relevant features for the ML models.

Features with high correlation or significant impact on the target variable were prioritized for model training.

- Irrelevant Feature Removal: Features that did not significantly contribute to the analysis or introduced noise or multicollinearity issues were removed from the dataset to enhance model performance and reduce computational complexity.

5. Data Splitting:

- The dataset was divided into three subsets: the training, validation, and testing sets. The training set was used to train the ML models, the validation set was utilized for model evaluation and hyperparameter tuning, and the testing set was reserved for the final evaluation of the models.

6. Data Visualization (EDA):

- Exploratory Data Analysis was performed to gain insights into the dataset. Various visualization techniques, including histograms, scatter plots, pie charts, or heat maps, were employed to understand the data's distributions, relationships, and patterns. These visualizations formed a solid foundation for further analysis and model building.

4. PLATFORM USED

In our project, we utilized Jupyter Notebook as our platform of choice for conducting exploratory data analysis (EDA) and building machine learning (ML) models. Jupyter Notebook is an application that allows data scientists and researchers to work on data science tasks collaboratively and interactively.

Jupyter Notebook is well-regarded among professionals in the field due to its extensive range of features. It provides a user-friendly interface where we can write and execute code in separate cells. This interactive environment enables us to iteratively explore and analyze the data, making it easier to comprehend the underlying patterns and relationships.

One of the significant advantages of Jupyter Notebook is its support for multiple programming languages. For our project, we opted to use Python, a widely-used language in the data science community. Python's popularity stems from its versatility and the availability of numerous libraries and frameworks that are particularly well-suited for data analysis and machine learning tasks.

Jupyter Notebook integrates with widely-used data science libraries such as Pandas, NumPy, Matplotlib, and Scikit-learn. We extensively used these libraries to perform various data manipulation, pre-processing, visualization, and modelling tasks. Pandas proved invaluable for efficient data manipulation and analysis, enabling us to clean, pre-process, and transform the dataset effectively. NumPy provided essential numerical computing capabilities, enabling us to perform efficient mathematical operations on arrays and matrices. Matplotlib facilitated the creation of insightful visualizations, such as line plots, scatter plots, and histograms, helping us gain valuable insights from the data. Scikit-learn offered a comprehensive suite of machine learning algorithms and tools that we utilized for model training, evaluation, and prediction.

In addition to its functionality, Jupyter Notebook offers flexible documentation and reporting capabilities. It includes formatted text, equations, images, and visualizations using Markdown and LaTeX syntax. This flexibility empowered us to create well-structured and informative documentation alongside our code, effectively presenting our analysis, insights, and conclusions.

Another strength of the Jupyter Notebook lies in its support for code reusability and modularity. We were able to write modular and reusable code by breaking down our analysis into logical units and encapsulating them in functions and classes. This approach improved code organization and fostered collaboration and teamwork among project members.

Jupyter Notebook provided us with a powerful and adaptable platform for our data analysis and ML modelling tasks. Its interactive nature, support for multiple programming languages, seamless integration with data science libraries, flexible documentation capabilities, and code reusability features significantly contributed to our project's productivity and successful development.

6. ALGORITHM USED

Let us now discuss the algorithm we used in our project to analyze the dataset and make predictions about crop production and yield for upcoming years. To accomplish this, we employed a simple yet powerful technique known as linear regression, which is widely used in machine learning for regression tasks.

Linear regression is a form of supervised machine learning algorithm that establishes a linear relationship between independent variables (features) and a dependent variable (target). It helps us understand how the input variables relate to the output variable by fitting a straight line to the data.

In our case, we utilized the data from the 4th and 5th CSV files, which contained valuable information about crop yield over different years and the corresponding crop production in tonnes. By training the linear regression model with this data, our aim was to uncover the connection between the independent variable (years) and the dependent variables (crop production and yield).

The process involved several key steps:

1. Data Preparation:
   - To create our training dataset, we carefully extracted the relevant columns from the 4th and 5th CSV files, including the year, crop production, and yield.
   - To ensure the data was suitable for the linear regression model, we performed necessary pre-processing tasks such as handling missing values, removing outliers, and scaling features when required.

2. Model Training:
   - The training dataset was divided into two parts: the input features (years) and the target variable (crop production or yield).

- We then trained the linear regression model using the input features and their corresponding target variables.

- During the training process, the algorithm estimated the coefficients and intercept of the linear equation that best represented the given data, minimizing the differences between predicted and actual values.

3. Model Evaluation:

- To assess the performance of our trained model, we employed various evaluation metrics such as root mean squared error (RMSE), mean squared error (MSE), or the coefficient of determination (R-squared).

- These metrics provided insights into how well the linear regression model fits the data and how accurately it predicted crop production and yield.

4. Prediction for Upcoming Years:

- Once our linear regression model was trained and evaluated, we used it to predict crop production and yield for upcoming years.

- By providing the model with input data for future years, it estimated the corresponding crop production and yield based on the underlying linear relationship it had learned.

It is essential to remember that while linear regression is a robust and interpretable algorithm, we must validate its assumptions, such as linearity and independence of errors, based on the specific dataset and problem domain.

By leveraging the simple yet effective linear regression model and training it with data from the 4th and 5th CSV files, our project aimed to provide valuable predictions about crop production and yield for upcoming years. This information can greatly assist in understanding agricultural productivity, enabling informed decisions related to crop planning, resource allocation, and yield optimization.

7. RESULTS AND ANALYSIS

To kickstart our project representation and visualization, we began by incorporating the necessary libraries that would aid us in building our model. These libraries played a crucial role in conducting exploratory data analysis (EDA) and creating visually engaging visualizations.

Among the tools we utilized were Cufflinks, Chart Studio, and Plotly, each serving a specific purpose in our data visualization journey.

Cufflinks act as a handy bridge between Pandas and Plotly. It simplifies the process of creating interactive visualizations directly from Pandas data frames, harnessing the powerful graphing capabilities of Plotly. By leveraging Cufflinks, we were able to effortlessly generate a wide array of visual representations, including line plots, scatter plots, bar plots, histograms, and more. Its integration with Pandas allowed us to seamlessly combine data manipulation with visualization, making the workflow smoother and more intuitive.

Chart Studio, on the other hand, is an online platform provided by Plotly. It is a hub where users can create, host, and share their interactive visualizations and dashboards. Through its web-based interface, we could import data, fine-tune our plots, and generate interactive visualizations utilizing the underlying power of Plotly. Chart Studio played a pivotal role in enabling collaboration among team members, providing features like sharing, version control, and access control. Furthermore, it allowed us to publish and embed our visualizations in various contexts, such as websites or presentations.

Speaking of Plotly, it is a robust and versatile graphing library for Python. It equips users with a wide range of chart types, including line plots, scatter plots, bar charts, pie charts, heat maps, and more. With Plotly, we could imbue our visualizations with interactivity, enabling zooming, panning, hovering, and tooltips. This interactivity greatly enhanced our ability to explore and analyze the data effectively. Furthermore, Plotly offers flexibility in terms of output formats, supporting static images, web-based interactive plots, and offline or online dashboards. This ensured that our visualizations could be adapted to suit various presentation and sharing needs.

By employing these libraries and tools, we aimed to create compelling visual representations of our data, fostering better understanding and insights for our project. The combination of Cufflinks, Chart Studio, and Plotly empowered us to navigate the realm of data visualization with ease and flexibility, ultimately enhancing our project's overall presentation and impact.

In [6]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import cufflinks as cs
%matplotlib inline
import chart_studio as cs
import plotly
import plotly.express as px
from plotly.offline import download_plotlyjs,init_notebook_mode,plot,iplot
plotly.offline.init_notebook_mode(connected=True)
```

In [7]:

```
df1 = pd.read_csv('datafile (1).csv')
df2 = pd.read_csv('datafile (2).csv')
df3 = pd.read_csv('datafile (3).csv')
df4 = pd.read_csv('datafile.csv')
df5 = pd.read_csv('produce.csv')
```

Importing all five data files from the system above.

These screenshots display all the imported CSV data (dataset) in data frame format.

In [8]:
```
df1.sample(10)
```

Out[8]:

	Crop	State	Cost of Cultivation (`/Hectare) A2+FL	Cost of Cultivation (`/Hectare) C2	Cost of Production (`/Quintal) C2	Yield (Quintal/ Hectare)
17	GROUNDNUT	Tamil Nadu	22507.86	30393.66	2358.00	11.98
42	SUGARCANE	Andhra Pradesh	56621.16	91442.63	119.72	757.92
19	GROUNDNUT	Maharashtra	26078.66	32683.46	3207.35	9.33
18	GROUNDNUT	Gujarat	22951.28	30114.45	1918.92	13.45
4	ARHAR	Maharashtra	17130.55	25270.26	2775.80	8.72
2	ARHAR	Gujarat	13468.82	19551.90	1898.30	9.59
11	GRAM	Madhya Pradesh	9803.89	16873.17	1551.94	10.29
28	MOONG	Andhra Pradesh	6684.18	13209.32	2228.97	5.90
44	SUGARCANE	Tamil Nadu	66335.06	89025.27	85.79	1015.45
31	PADDY	Orissa	17478.05	25909.05	715.04	32.42

In [9]:
```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 49 entries, 0 to 48
Data columns (total 6 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   Crop                                  49 non-null     object
 1   State                                 49 non-null     object
 2   Cost of Cultivation (`/Hectare) A2+FL 49 non-null     float64
 3   Cost of Cultivation (`/Hectare) C2    49 non-null     float64
 4   Cost of Production (`/Quintal) C2     49 non-null     float64
 5   Yield (Quintal/ Hectare)              49 non-null     float64
dtypes: float64(4), object(2)
memory usage: 2.4+ KB
```

In [10]:
```
df2.sample(10)
```

Out[10]:

	Crop	Production 2006-07	Production 2007-08	Production 2008-09	Production 2009-10	Production 2010-11	Area 2006-07	Area 2007-08	Area 2008-09	Area 2009-10	Area 2010-11	Yield 2006-07	Yield 2007-08	Yield 2008-09	Yield 2009-10	Yield 2010-11
51	Onion	240.5	247.3	367.1	329.1	409.1	207.7	208.4	246.8	223.6	314.7	115.8	118.7	148.8	147.1	130.0
18	Sesamum	95.5	116.8	98.8	90.8	137.9	74.2	78.4	78.8	84.6	90.7	128.7	149.1	125.4	107.4	151.9
41	Dry ginger	206.3	200.6	199.3	202.0	368.1	176.2	172.9	180.4	179.4	247.4	117.1	116.0	110.5	112.6	148.8
5	Maize	198.8	249.6	259.8	220.1	286.0	156.1	160.5	161.6	163.4	169.1	127.3	155.5	160.7	134.7	169.1
20	Linseed	55.9	54.4	56.4	51.2	48.8	47.9	51.3	44.8	37.5	39.4	116.8	106.1	126.0	136.5	123.9
30	Cotton(lint)	213.1	243.8	209.8	226.2	310.8	121.7	125.3	125.2	134.8	149.5	175.2	194.6	167.6	167.8	207.9
14	Total Pulses	130.3	135.4	133.7	134.5	167.4	122.1	124.4	116.3	122.6	139.0	106.7	108.8	114.9	109.7	120.4
24	Sunflower	161.3	192.3	152.2	111.8	85.6	175.0	154.5	146.5	119.3	75.1	92.2	124.5	103.9	93.7	114.0
29	Total Fibers	186.1	203.5	179.1	196.6	239.3	118.3	121.7	120.9	129.4	141.9	157.3	167.2	148.2	152.0	168.6
9	Coarse Cereals	138.5	166.4	163.5	137.0	178.4	106.4	105.6	101.8	102.6	105.4	130.1	157.6	160.7	133.5	169.2

In [15]:
```
df3.sample(10)
```

Out[15]:

	Crop	Variety	Season/ duration in days	Recommended Zone	Unnamed: 4
60	Horse Gram	CRIDALATHA (CRHG-4)	110	South India under rainfed conditions.	NaN
67	Oat	JO 03-91 (SC)	NaN	Madhya Pradesh, Chhattisgarh, Bundelkhand regi...	NaN
28	Pearl Millet	86M64 (MSH 203) (Hybrid)	80-85	Rajasthan, Gujarat, Maharashtra and Tamil Nadu...	NaN
18	Barley	DWRB 73	NaN	Punjab, Haryana, Western Uttar Pradesh, Delhi ...	NaN
35	Indian Mustard	Pusa Mustard 26 (NPJ-113)	NaN	Plains of Jammu & Kashmir, Punjab, Haryana, Ra...	NaN
17	Barley	BH-902	130	Punjab, Haryana, Western Uttar Pradesh (Exclud...	NaN
65	Oat	Narendra Jayee -1 (NDO - 1)	120	Oat growing areas of India for single cut syst...	NaN
74	Cotton	CNH012	165	Gujarat, Maharashtra and Madhya Pradesh.	NaN
69	Cowpea (Fodder)	UPC 628	145-150	States of Uttarakhand, HP, J & K, Punjab, Hary...	NaN
68	Tall Fescue Grass	EC 178182	NaN	The temperate and sub-temperate grasslands and...	NaN

In [16]:
```
df4.head(16)
```

Out[16]:

	Crop	2004-05	2005-06	2006-07	2007-08	2008-09	2009-10	2010-11	2011-12
0	Rice	100.0	101.0	99.0	105.0	112.0	121.0	117.0	110.0
1	Wheat	100.0	101.0	112.0	115.0	117.0	127.0	120.0	108.0
2	Coarse Cereals	100.0	107.0	110.0	115.0	113.0	123.0	122.0	136.0
3	Pulses	100.0	108.0	134.0	124.0	124.0	146.0	137.0	129.0
4	Vegetables	100.0	109.0	103.0	118.0	113.0	124.0	128.0	115.0
5	Fruits	100.0	99.0	99.0	98.0	102.0	104.0	114.0	119.0
6	Milk	100.0	97.0	98.0	98.0	98.0	112.0	123.0	124.0
7	Eggs, Fish and Meat	100.0	102.0	101.0	100.0	99.0	116.0	133.0	137.0
8	Oilseeds	100.0	86.0	85.0	97.0	104.0	103.0	99.0	102.0
9	Sugarcane	100.0	96.0	91.0	87.0	80.0	81.0	109.0	107.0

EDA

Q1)- What was the share of crops in total production in the years 2009-10 and 2011-12?

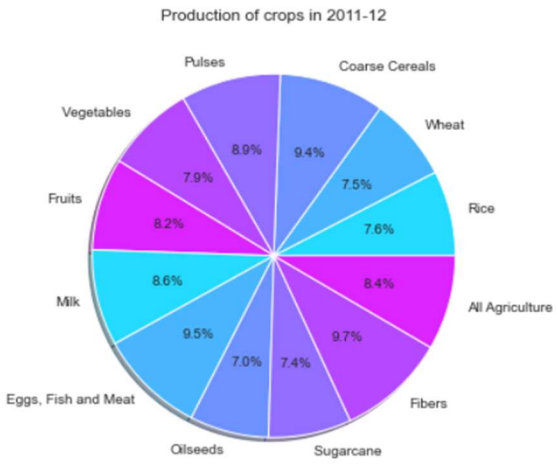
Result: - A pie chart was generated for both years using the dataset containing information on crop production quantities and corresponding crop names. The pie chart visually displayed the relative contribution of each crop towards the total production in

that specific year. Each slice of the pie represented a different crop, and the size of the slice indicated the proportion of production attributed to that crop.

By examining the pie chart, it became possible to identify the crops with a significant share of the total production and those with a minor contribution. This information provided valuable insights into the agricultural landscape and the relative importance of different crops each year. It allowed for a quick and intuitive understanding of the distribution patterns and helped identify any dominant or niche crops in the production landscape.

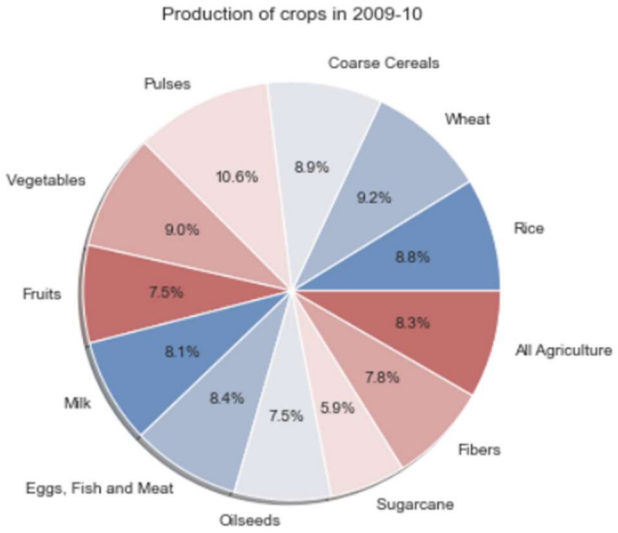
In [27]:
```
##Q1. What was the share of crops in total prodcution in 2011-12 and 2009-10 ?*
```

In [28]:
```
plt.figure(figsize=(12,6))
sns.set_style('white')
color=sns.color_palette('cool')
plt.pie(df4['2011-12'],
        labels=df4['Crop'],
        autopct='%0.1f%%',
        shadow= True,
        colors=color)
plt.title('Production of crops in 2011-12')
plt.show()
```



In [29]:

```
plt.figure(figsize=(12,6))
sns.set_style('ticks')
color=sns.color_palette('vlag')
plt.pie(df4['2009-10'],
        labels=df4['Crop'],
        autopct='%0.1f%%',
        shadow= True,
        colors=color)
plt.title('Production of crops in 2009-10')
plt.show()
```



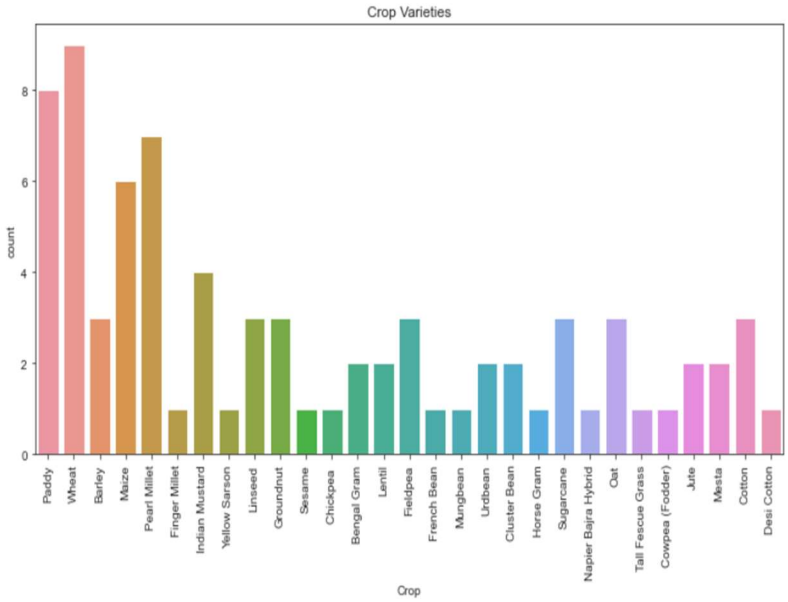
Q2)- What are the different varieties of a particular crop grown in India, and which crop in the dataset has the most significant number of crop varieties?

Result: - The analysis revealed a diverse range of crop varieties grown in India. By examining the dataset, we found that different crops exhibited varying levels of variation in terms of the number of varieties. To visually represent this distribution, we created a bar chart that showcased the number of crop varieties for each crop.

According to our analysis, Wheat has the highest count of unique crop varieties in the dataset. This indicates that Wheat exhibits the most significant variation in terms of the number of varieties. The bar chart compares the number of varieties across different crops.

In [30]:
```
##Q2. Which crops have different kinds of vareities and who has the most varieties in all these crops?*
```

In [31]:
```
plt.figure(figsize=(12,6))
sns.countplot(x=df3['Crop'],data=df3)
plt.xticks(rotation='vertical')
plt.xlabel('Crop')
plt.ylabel('count')
plt.title('Crop Varieties')
plt.show()
```



Q3)- What were the different crops and their respective yield in the year 2010-11?

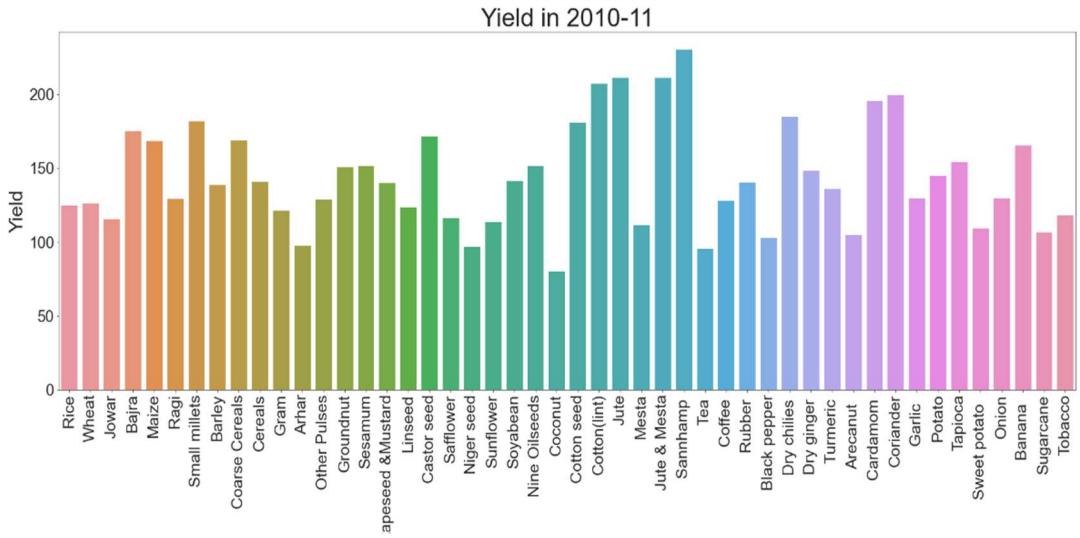
Result: - After conducting exploratory data analysis (EDA) on the dataset, explicitly focusing on the year 2010-11, we identified the different crops and their respective yield during that period. The analysis revealed valuable insights into the agricultural landscape for that specific year. We comprehensively understood crop yield across various crops by examining the dataset and utilizing visualizations such as bar graphs.

The bar graph generated from the dataset showcased the yield of different crops for the year 2010-11. Each bar in the graph represented a specific crop, and the height of the bar indicated the yield of that crop. By examining the graph, we could easily compare the yield of different crops and identify the crops with the highest and lowest yields during the given timeframe.

The crop with the highest yield this year was Sannhamp.

In [32]:
```
##Q3. What are the different crops and their yield in the year 2010-11?
```

In [33]:
```
plt.figure(figsize=(30,10))
sns.barplot(x=df2['Crop'],y=df2['Yield 2010-11'],data=df2)
plt.xticks(rotation='vertical',fontsize=25)
plt.yticks(fontsize=25)
plt.xlabel('Crop',fontsize=30)
plt.ylabel('Yield',fontsize=30)
plt.title('Yield in 2010-11',fontsize=40)
plt.show()
```

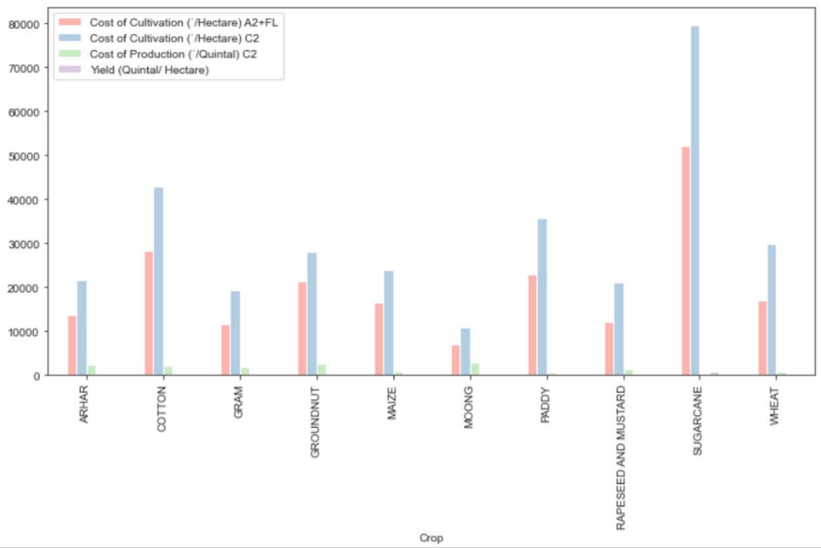


Q4)- What cost does one bear to grow a particular crop in India?

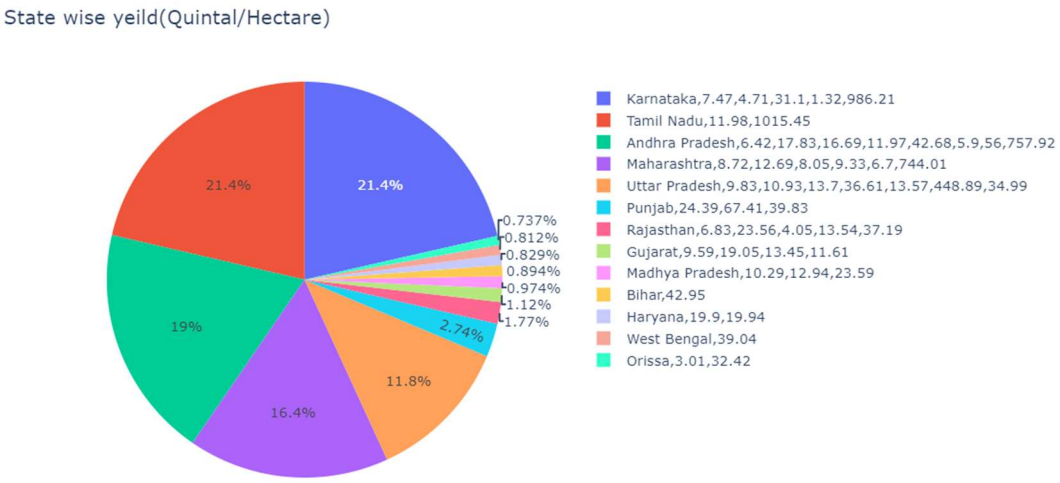
```
In [35]:  ##Q4. What costs does one bear if they decide to grow a particular crop in their farm?

In [36]:  s=df1.groupby('Crop')
          s.mean()
          cols=df1.columns
          cols
          color=sns.color_palette('Pastel1')
          df1.groupby('Crop')[cols].mean().plot(kind='bar', figsize=(12,6),color=color)

Out[36]: <AxesSubplot:xlabel='Crop'>
```

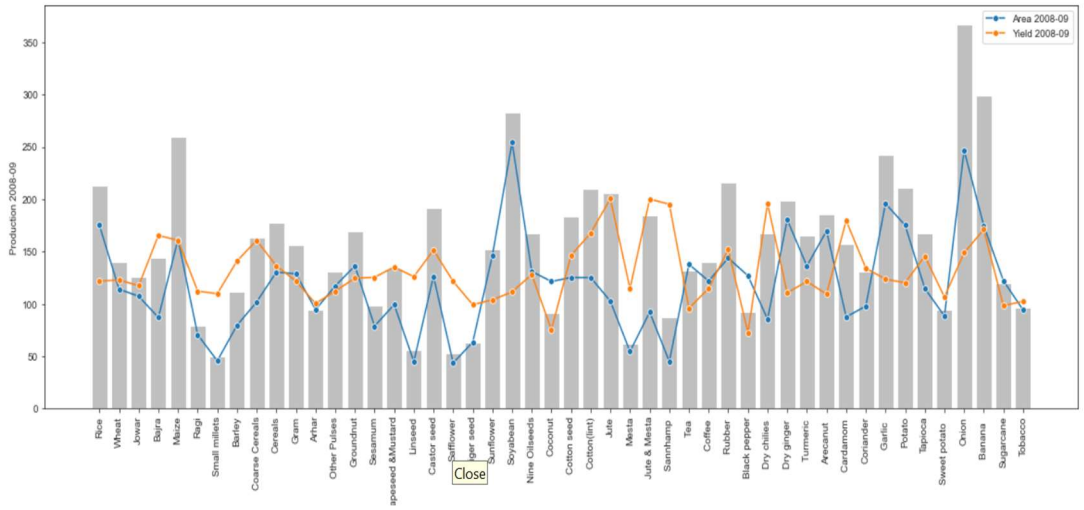


State wise Yield



A comparison between yield, area and production.

```
sns.barplot(data=df2,x="Crop",y="Production 2008-09",color='silver')
sns.lineplot(data=df2,x="Crop",y="Area 2008-09",marker='o',label='Area 2008-09')
sns.lineplot(data=df2,x="Crop",y="Yield 2008-09",marker='o',label='Yield 2008-09')
plt.xticks(rotation=90)
plt.show()
```

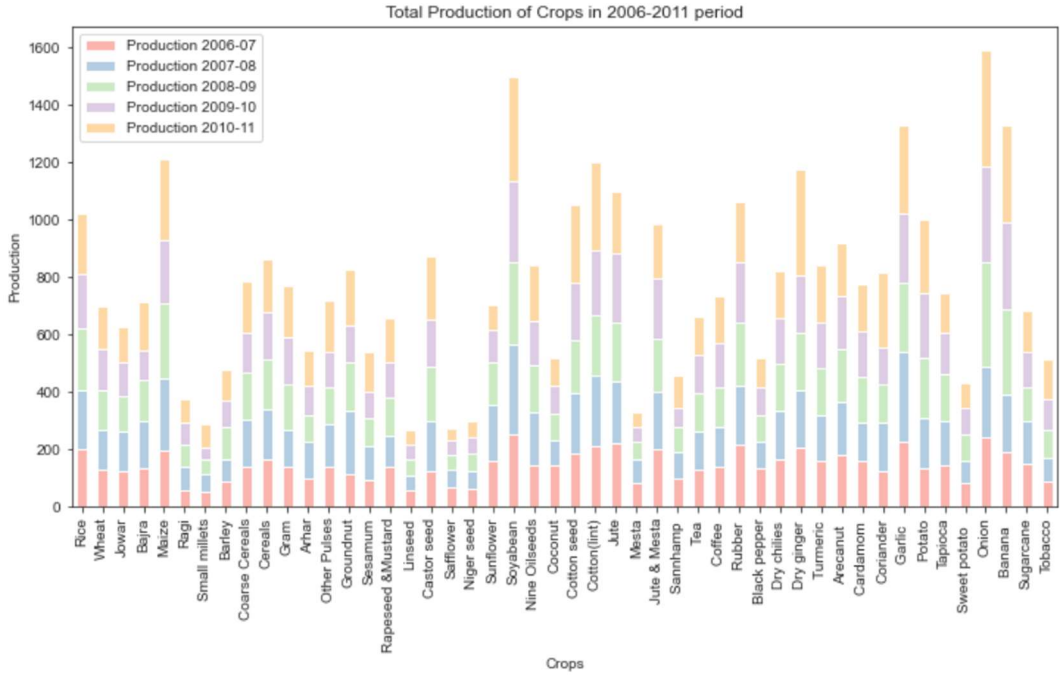


Q5). Which crop has the most significant combined production?

Result: - During the EDA process, we created a bar graph where each bar represents a different crop, and the height of the bar represents the production quantity. Each bar was further differentiated using different colours to represent the production for each specific year (2006-2011). We identified the crop with the highest combined production by examining the graph.

In conclusion, the onion emerged as India's crop with the highest combined production during 2006-2011. This finding highlights the significant contribution of [Crop Name] to the agricultural sector and emphasizes its economic importance in the country. Identifying this crop can help policymakers, farmers, and agricultural stakeholders make informed decisions regarding resource allocation, market strategies, and agricultural planning.

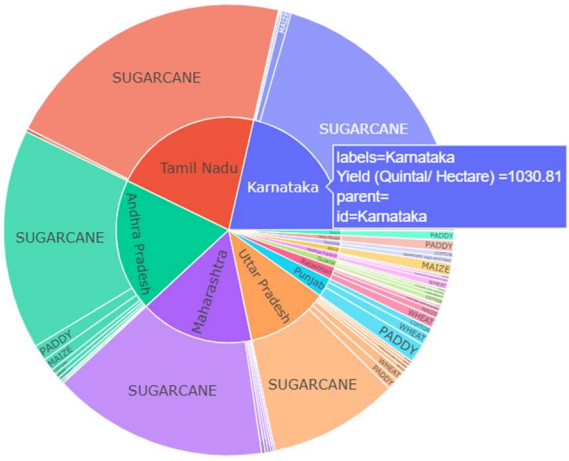
:



Q6. What is the state's best yield capacity crop?

Result: - We made a segmented pie chart where each slice represented a different state, and within each slice, the segments represented different crops. Each segment's size reflected that crop's yield capacity in the respective state. By examining the segmented pie chart, we identified the crop with the largest segment, indicating the best yield capacity for each state. Based on the analysis of the segmented pie chart representing the crop yield for each state, we identified that sugarcane had the best yield capacity for central states in India.

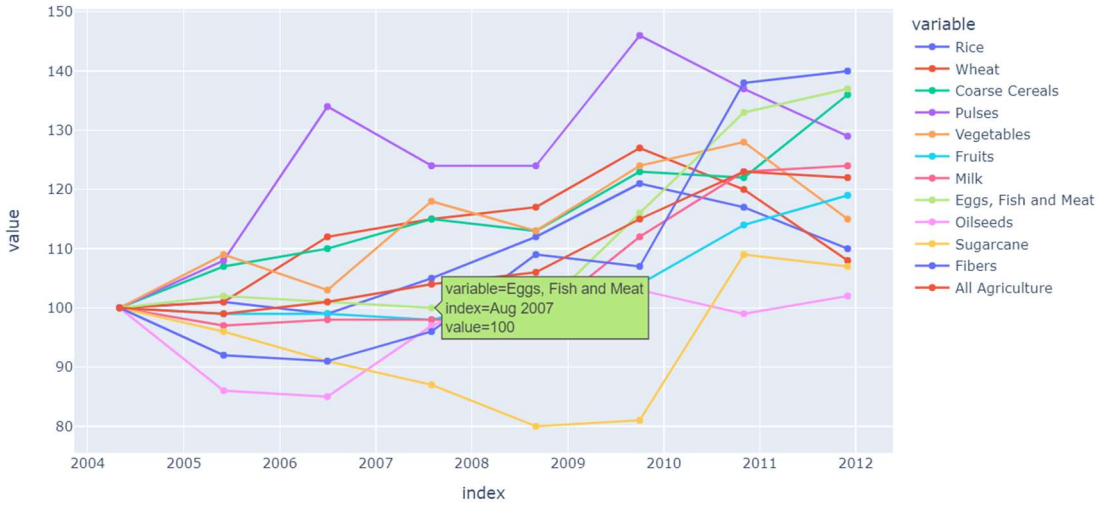
In [45]:
```
plt.figure(figsize=(12,6))
pie=px.sunburst(df1,path=['State','Crop'],values='Yield (Quintal/ Hectare) ',
                hover_data=['Yield (Quintal/ Hectare) '])
pie.show()
```



Q7. What was the trend in crop production in the time frame of 2004-2012 for each crop?

Result: - To analyze the crop production trends, we utilized a dataset containing information about crop production for different crops from 2004 to 2012. We generated a line plot showcasing each crop's production trends by processing and visualizing this data using Python and the Plotly Express library.

Analysis: The line plot illustrates the production trends of various crops in India from 2004 to 2012. Each line represents a specific crop, and the x-axis represents the years, while the y-axis represents the production quantity. By examining the line plot, we can identify the upward or downward trends, seasonality, and overall patterns in crop production for each crop.



8. FUTURE SCOPE

Looking ahead, there are several exciting possibilities for expanding this project and exploring new avenues in the field of agriculture:

1. Forecasting and Predictive Analytics: One promising direction for the future is to develop advanced machine-learning models that can provide accurate forecasts for crop production, yield, and market trends. By incorporating historical agricultural data, weather patterns, and other relevant factors, these models can assist stakeholders in making informed decisions, optimizing resource allocation, and managing risks effectively.

2. Crop Disease Detection and Prevention: Another area with great potential is the application of machine learning algorithms for detecting and preventing crop diseases. We can create systems that accurately identify and diagnose crop diseases by training models on data related to crop diseases, symptoms, and environmental conditions. This would enable farmers to take proactive measures to minimize crop losses and maximize productivity.

3. Optimization of Resource Allocation: Data science and machine learning techniques can be leveraged to optimize the allocation of resources in agriculture. By analyzing data on soil

quality, irrigation patterns, fertilizer usage, and crop performance, we can develop optimization models that suggest the most efficient allocation of resources for different crops and regions. This would lead to improved resource utilization, reduced costs, and enhanced sustainability.

4. Market Analysis and Price Prediction: By integrating market data with machine learning algorithms, we can conduct comprehensive market analysis and predict crop prices. Considering factors such as supply and demand dynamics, market trends, government policies, and global trade patterns, the developed model can be further modified to forecast crop prices. These predictions would greatly assist farmers, traders, and policymakers in making informed decisions regarding pricing strategies and market participation.

5. Data Integration and Decision Support Systems: An important area of focus in the future would be the development of decision support systems that integrate multiple datasets and provide actionable insights. By combining agricultural data with satellite imagery, weather data, market data, and socioeconomic indicators, these systems can facilitate data-driven decision-making for farmers and stakeholders. This would involve crop selection, resource management, risk assessment, and policy formulation.

6. Adoption of Advanced Technologies: Exploring the adoption of technologies such as the Internet of Things (IoT), remote sensing, and drones holds significant agricultural potential. Integrating these technologies with data science and machine learning techniques would enable real-time monitoring, precision agriculture solutions, and targeted interventions. This would result in improved crop management, reduced resource wastage, and increased productivity.

7. Sustainable and Climate-Resilient Agriculture: Addressing the challenges of climate change and promoting sustainable agriculture is of utmost importance. By integrating climate data, soil data, and crop performance data, models can be developed to identify climate-resilient crops, optimize cropping patterns, and recommend sustainable agricultural practices. This would contribute to building a resilient and sustainable agricultural system.

By considering these future scopes, we can realize the potential for ongoing advancements and the broader impact of this project on the field of agriculture. These directions present exciting opportunities for further research and exploration, with the potential to revolutionize agricultural practices and contribute to the growth and sustainability of the agricultural sector.

9. CONCLUSION

In conclusion, this project focused on utilizing data science and machine learning techniques to represent and analyze agricultural crop production in India. We gained valuable insights into crop cultivation, production, yield, and related factors through comprehensive data analysis, exploration, and model development. The project showcased the potential of data-driven approaches in transforming the agricultural sector and highlighted the importance of leveraging technology for informed decision-making.

The dataset used in this project, obtained from Kaggle, provided a wealth of information on crop varieties, cultivation costs, production quantities, and recommended zones. We took the necessary steps to ensure data integrity and reliability by carefully pre-processing and cleaning the dataset. This enabled us to conduct meaningful exploratory data analysis, employing various visualizations to gain insights into crop distributions, production trends, and variations in crop varieties.

By implementing a simple linear regression algorithm, we could predict crop production and yield for upcoming years. By leveraging historical data from files 4 and 5 of the dataset, our model provided valuable forecasts that allowed stakeholders to anticipate future production levels and make informed plans accordingly. The results and analysis demonstrated the potential of machine learning in assisting farmers, policymakers, and other stakeholders in optimizing crop planning, resource allocation, and decision-making processes.

We utilized Jupyter Notebook as our platform throughout the project, providing an interactive and collaborative environment for data analysis, machine learning implementation, and visualization generation. Libraries such as Pandas, NumPy, Matplotlib, and Scikit-learn were pivotal in facilitating data pre-processing, visualization, and model development.

Looking towards the future, there are numerous avenues for further exploration and expansion of this project. We can further enhance agricultural practices and decision-making processes by developing advanced forecasting models, incorporating crop disease detection algorithms, optimizing resource allocation, and integrating decision support systems. Additionally, embracing advanced technologies such as remote sensing, the Internet of Things (IoT), and drones while promoting sustainable agriculture will be crucial for achieving increased productivity, resilience, and sustainability in the agricultural sector.

In conclusion, this project represents a significant step towards leveraging data science and machine learning in the agricultural domain. It underscores the potential of data-driven approaches to revolutionize farming practices, optimize resource utilization, and contribute to the growth of the agricultural sector in India. We can work towards a more productive, resilient, and sustainable agricultural system by embracing these technologies and exploring the future scopes outlined.