Crop_Production_analysis

August 2, 2024

---

1 Crop Production Analysis in India

1.1 Project Overview

This project is dedicated to analyzing crop production across India, employing advanced data science techniques to uncover key factors influencing crop yields. By leveraging historical data on crop production, the goal is to provide actionable insights and predictions that benefit stakeholders in the agricultural sector. The analysis aims to enhance decision-making processes and support strategic planning for improved crop management.

1.2 Technologies

- **Programming Language**: Python
- **Data Analysis & Visualization**: Pandas, NumPy, Matplotlib, Seaborn, Plotly
- **Documentation**: Jupyter Notebooks, Markdown

1.3 Dataset

The project utilizes a comprehensive dataset detailing crop production in India over multiple years. The dataset includes attributes related to crop yields, geographical regions, and temporal aspects.

1.4 Project Structure

1. **Data Exploration**: Initial exploration of the dataset to understand its structure and contents.
2. **Data Preprocessing**: Cleaning and transforming data to ensure it is ready for analysis.
3. **Exploratory Data Analysis (EDA)**: Statistical analysis and visualization to identify key patterns and insights.
4. **Feature Engineering**: Creation and selection of features that impact crop production.
5. **Insights and Analysis**: Extraction of significant insights and trends from the data.
6. **Visualization and Dashboards**: Development of interactive dashboards and visualizations to effectively communicate findings.
7. **Reporting and Documentation**: Detailed documentation of methodologies, results, and conclusions.

1.4.1 *IMPORT NECESSARY LIBRARY*

[3]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import scipy.stats as stats
import plotly.io as pt
import plotly.express as px
import plotly.graph_objects as go
```

1.4.2 *READING THE DATA*

[5]:
```
df = pd.read_csv("Crop Production data.csv")
```

1.4.3 *EXPLORING THE DATA*

[7]:
```
df
```

[7]:

	State_Name	District_Name	Crop_Year	Season	Crop	Area	Production
0	Andaman and Nicobar Islands	NICOBARS	2000	Kharif	Arecanut	1254.0	2000.0
1	Andaman and Nicobar Islands	NICOBARS	2000	Kharif	Other Kharif pulses	2.0	1.0
2	Andaman and Nicobar Islands	NICOBARS	2000	Kharif	Rice	102.0	321.0
3	Andaman and Nicobar Islands	NICOBARS	2000	Whole Year	Banana	176.0	641.0
4	Andaman and Nicobar Islands	NICOBARS	2000	Whole Year	Cashewnut	720.0	165.0
…	…	…	…	…	…	…	…
246086	West Bengal	PURULIA	2014	Summer	Rice	306.0	801.0
246087	West Bengal	PURULIA	2014	Summer	Sesamum	627.0	463.0
246088	West Bengal	PURULIA	2014	Whole Year	Sugarcane	324.0	16250.0
246089	West Bengal	PURULIA	2014	Winter	Rice	279151.0	597899.0
246090	West Bengal	PURULIA	2014	Winter	Sesamum	175.0	88.0

[246091 rows x 7 columns]

[8]:
```
df.head(10)
```

[8]:

	State_Name	District_Name	Crop_Year	Season	Crop	Area	Production
0	Andaman and Nicobar Islands	NICOBARS	2000	Kharif	Arecanut	1254.0	2000.0
1	Andaman and Nicobar Islands	NICOBARS	2000	Kharif	Other Kharif pulses	2.0	1.0
2	Andaman and Nicobar Islands	NICOBARS	2000	Kharif	Rice	102.0	321.0
3	Andaman and Nicobar Islands	NICOBARS	2000	Whole Year	Banana	176.0	641.0
4	Andaman and Nicobar Islands	NICOBARS	2000	Whole Year	Cashewnut	720.0	165.0
5	Andaman and Nicobar Islands	NICOBARS	2000	Whole Year	Coconut	18168.0	65100000.0
6	Andaman and Nicobar Islands	NICOBARS	2000	Whole Year	Dry ginger	36.0	100.0
7	Andaman and Nicobar Islands	NICOBARS	2000	Whole Year	Sugarcane	1.0	2.0
8	Andaman and Nicobar Islands	NICOBARS	2000	Whole Year	Sweet potato	5.0	15.0
9	Andaman and Nicobar Islands	NICOBARS	2000	Whole Year	Tapioca	40.0	169.0

[9]:
```
df.tail(10)
```

[9]:

	State_Name	District_Name	Crop_Year	Season	Crop	Area	Production
246081	West Bengal	PURULIA	2014	Rabi	Rapeseed &Mustard	1885.0	1508.0
246082	West Bengal	PURULIA	2014	Rabi	Safflower	54.0	37.0
246083	West Bengal	PURULIA	2014	Rabi	Urad	220.0	113.0
246084	West Bengal	PURULIA	2014	Rabi	Wheat	1622.0	3663.0
246085	West Bengal	PURULIA	2014	Summer	Maize	325.0	2039.0
246086	West Bengal	PURULIA	2014	Summer	Rice	306.0	801.0
246087	West Bengal	PURULIA	2014	Summer	Sesamum	627.0	463.0
246088	West Bengal	PURULIA	2014	Whole Year	Sugarcane	324.0	16250.0
246089	West Bengal	PURULIA	2014	Winter	Rice		
246090	West Bengal	PURULIA	2014	Winter	Sesamum		

```
246089  279151.0    597899.0
246090     175.0        88.0
```

[10]:
```
df.sample(10)
```

[10]:
```
               State_Name District_Name  Crop_Year       Season  \
114233     Madhya Pradesh        MANDLA       2001  Kharif
32181               Bihar     GOPALGANJ       2004  Kharif
100009             Kerala      KOTTAYAM       2007  Whole Year
17641               Assam    DIMA HASAO       2013  Rabi
9923    Arunachal Pradesh          ANJAW       2013  Whole Year
167189          Rajasthan      BHILWARA       2008  Rabi
229476      Uttar Pradesh       SITAPUR       2010  Rabi
104345     Madhya Pradesh          BETUL       2003  Whole Year
8618       Andhra Pradesh  VIZIANAGARAM       2000  Rabi
193364         Telangana    MAHBUBNAGAR       2004  Rabi

                   Crop     Area  Production
114233        Soyabean    350.0       268.0
32181            Jowar     83.0        84.0
100009    Sweet potato      1.0         8.0
17641             Urad    587.0       340.0
9923            Potato    120.0      1022.0
167189  other oilseeds   2793.0       940.0
229476          Masoor  25657.0     20782.0
104345          Banana      2.0        38.0
8618        Horse-gram  25968.0      2934.0
193364      Horse-gram    692.0       224.0
```

[11]:
```
df.dtypes
```

[11]:
```
State_Name        object
District_Name     object
Crop_Year          int64
Season            object
Crop              object
Area             float64
Production       float64
dtype: object
```

[12]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 246091 entries, 0 to 246090
Data columns (total 7 columns):
 #   Column         Non-Null Count   Dtype
---  ------         --------------   -----
 0   State_Name     246091 non-null  object
```

```
 1   District_Name  246091 non-null  object
 2   Crop_Year      246091 non-null  int64
 3   Season         246091 non-null  object
 4   Crop           246091 non-null  object
 5   Area           246091 non-null  float64
 6   Production     242361 non-null  float64
dtypes: float64(2), int64(1), object(4)
memory usage: 13.1+ MB
```

[13]:
```
df.describe().T
```

[13]:
```
                count           mean           std      min     25%     50%  \
Crop_Year    246091.0    2005.643018  4.952164e+00  1997.00  2002.0  2006.0
Area         246091.0   12002.820864  5.052340e+04     0.04    80.0   582.0
Production   242361.0  582503.442251  1.706581e+07     0.00    88.0   729.0

                75%           max
Crop_Year    2010.0  2.015000e+03
Area         4392.0  8.580100e+06
Production   7023.0  1.250800e+09
```

[14]:
```
df.isnull().sum()
```

[14]:
```
State_Name          0
District_Name       0
Crop_Year           0
Season              0
Crop                0
Area                0
Production       3730
dtype: int64
```

[15]:
```
df.shape
```

[15]:
```
(246091, 7)
```

[16]:
```
total_missing_value=(3730/246091)*100
print(total_missing_value,"%")
```

```
1.5156994770227274 %
```

- ***The Null Values in the data is 1.52% of the full data it is a small amount of the null values so droped null values***

[18]:
```
df.dropna(inplace=True)
```

[19]:
```
df.isnull().sum()
```

[19]:
```
State_Name       0
District_Name    0
Crop_Year        0
Season           0
Crop             0
Area             0
Production       0
dtype: int64
```

[20]:
```
df.shape
```

[20]:
```
(242361, 7)
```

[21]:
```
df.duplicated().sum()
```

[21]:
```
0
```

[22]:
```
states=df.State_Name.unique()
```

[23]:
```
states =states.size
print(states)
```

```
33
```

- ***This dataset encodes agriculture data for 33 Indian states which also include the Union Terretories As well***

[25]:
```
df.State_Name.unique()
```

[25]:
```
array(['Andaman and Nicobar Islands', 'Andhra Pradesh',
       'Arunachal Pradesh', 'Assam', 'Bihar', 'Chandigarh',
       'Chhattisgarh', 'Dadra and Nagar Haveli', 'Goa', 'Gujarat',
       'Haryana', 'Himachal Pradesh', 'Jammu and Kashmir ', 'Jharkhand',
       'Karnataka', 'Kerala', 'Madhya Pradesh', 'Maharashtra', 'Manipur',
       'Meghalaya', 'Mizoram', 'Nagaland', 'Odisha', 'Puducherry',
       'Punjab', 'Rajasthan', 'Sikkim', 'Tamil Nadu', 'Telangana ',
       'Tripura', 'Uttar Pradesh', 'Uttarakhand', 'West Bengal'],
      dtype=object)
```

[26]:
```
df.District_Name.nunique()
```

[26]:
```
646
```

[27]:
```
df.District_Name.unique()
```

[27]:
```
array(['NICOBARS', 'NORTH AND MIDDLE ANDAMAN', 'SOUTH ANDAMANS',
       'ANANTAPUR', 'CHITTOOR', 'EAST GODAVARI', 'GUNTUR', 'KADAPA',
       'KRISHNA', 'KURNOOL', 'PRAKASAM', 'SPSR NELLORE', 'SRIKAKULAM',
       'VISAKHAPATANAM', 'VIZIANAGARAM', 'WEST GODAVARI', 'ANJAW',
```

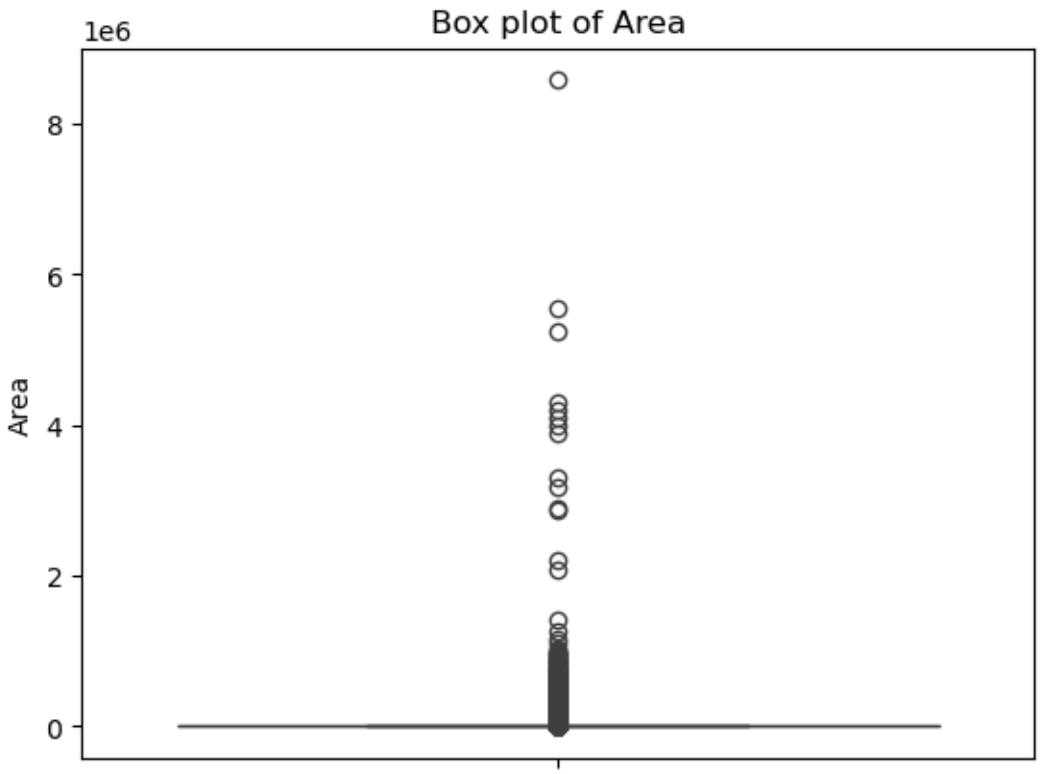
'CHANGLANG', 'DIBANG VALLEY', 'EAST KAMENG', 'EAST SIANG',
'KURUNG KUMEY', 'LOHIT', 'LONGDING', 'LOWER DIBANG VALLEY',
'LOWER SUBANSIRI', 'NAMSAI', 'PAPUM PARE', 'TAWANG', 'TIRAP',
'UPPER SIANG', 'UPPER SUBANSIRI', 'WEST KAMENG', 'WEST SIANG',
'BAKSA', 'BARPETA', 'BONGAIGAON', 'CACHAR', 'CHIRANG', 'DARRANG',
'DHEMAJI', 'DHUBRI', 'DIBRUGARH', 'DIMA HASAO', 'GOALPARA',
'GOLAGHAT', 'HAILAKANDI', 'JORHAT', 'KAMRUP', 'KAMRUP METRO',
'KARBI ANGLONG', 'KARIMGANJ', 'KOKRAJHAR', 'LAKHIMPUR', 'MARIGAON',
'NAGAON', 'NALBARI', 'SIVASAGAR', 'SONITPUR', 'TINSUKIA',
'UDALGURI', 'ARARIA', 'ARWAL', 'AURANGABAD', 'BANKA', 'BEGUSARAI',
'BHAGALPUR', 'BHOJPUR', 'BUXAR', 'DARBHANGA', 'GAYA', 'GOPALGANJ',
'JAMUI', 'JEHANABAD', 'KAIMUR (BHABUA)', 'KATIHAR', 'KHAGARIA',
'KISHANGANJ', 'LAKHISARAI', 'MADHEPURA', 'MADHUBANI', 'MUNGER',
'MUZAFFARPUR', 'NALANDA', 'NAWADA', 'PASHCHIM CHAMPARAN', 'PATNA',
'PURBI CHAMPARAN', 'PURNIA', 'ROHTAS', 'SAHARSA', 'SAMASTIPUR',
'SARAN', 'SHEIKHPURA', 'SHEOHAR', 'SITAMARHI', 'SIWAN', 'SUPAUL',
'VAISHALI', 'CHANDIGARH', 'BALOD', 'BALODA BAZAR', 'BALRAMPUR',
'BASTAR', 'BEMETARA', 'BIJAPUR', 'BILASPUR', 'DANTEWADA',
'DHAMTARI', 'DURG', 'GARIYABAND', 'JANJGIR-CHAMPA', 'JASHPUR',
'KABIRDHAM', 'KANKER', 'KONDAGAON', 'KORBA', 'KOREA', 'MAHASAMUND',
'MUNGELI', 'NARAYANPUR', 'RAIGARH', 'RAIPUR', 'RAJNANDGAON',
'SUKMA', 'SURAJPUR', 'SURGUJA', 'DADRA AND NAGAR HAVELI',
'NORTH GOA', 'SOUTH GOA', 'AHMADABAD', 'AMRELI', 'ANAND',
'BANAS KANTHA', 'BHARUCH', 'BHAVNAGAR', 'DANG', 'DOHAD',
'GANDHINAGAR', 'JAMNAGAR', 'JUNAGADH', 'KACHCHH', 'KHEDA',
'MAHESANA', 'NARMADA', 'NAVSARI', 'PANCH MAHALS', 'PATAN',
'PORBANDAR', 'RAJKOT', 'SABAR KANTHA', 'SURAT', 'SURENDRANAGAR',
'TAPI', 'VADODARA', 'VALSAD', 'AMBALA', 'BHIWANI', 'FARIDABAD',
'FATEHABAD', 'GURGAON', 'HISAR', 'JHAJJAR', 'JIND', 'KAITHAL',
'KARNAL', 'KURUKSHETRA', 'MAHENDRAGARH', 'MEWAT', 'PALWAL',
'PANCHKULA', 'PANIPAT', 'REWARI', 'ROHTAK', 'SIRSA', 'SONIPAT',
'YAMUNANAGAR', 'CHAMBA', 'HAMIRPUR', 'KANGRA', 'KINNAUR', 'KULLU',
'LAHUL AND SPITI', 'MANDI', 'SHIMLA', 'SIRMAUR', 'SOLAN', 'UNA',
'ANANTNAG', 'BADGAM', 'BANDIPORA', 'BARAMULLA', 'DODA',
'GANDERBAL', 'JAMMU', 'KARGIL', 'KATHUA', 'KISHTWAR', 'KULGAM',
'KUPWARA', 'LEH LADAKH', 'POONCH', 'PULWAMA', 'RAJAURI', 'RAMBAN',
'REASI', 'SAMBA', 'SHOPIAN', 'SRINAGAR', 'UDHAMPUR', 'BOKARO',
'CHATRA', 'DEOGHAR', 'DHANBAD', 'DUMKA', 'EAST SINGHBUM', 'GARHWA',
'GIRIDIH', 'GODDA', 'GUMLA', 'HAZARIBAGH', 'JAMTARA', 'KHUNTI',
'KODERMA', 'LATEHAR', 'LOHARDAGA', 'PAKUR', 'PALAMU', 'RAMGARH',
'RANCHI', 'SAHEBGANJ', 'SARAIKELA KHARSAWAN', 'SIMDEGA',
'WEST SINGHBHUM', 'BAGALKOT', 'BANGALORE RURAL', 'BELGAUM',
'BELLARY', 'BENGALURU URBAN', 'BIDAR', 'CHAMARAJANAGAR',
'CHIKBALLAPUR', 'CHIKMAGALUR', 'CHITRADURGA', 'DAKSHIN KANNAD',
'DAVANGERE', 'DHARWAD', 'GADAG', 'GULBARGA', 'HASSAN', 'HAVERI',
'KODAGU', 'KOLAR', 'KOPPAL', 'MANDYA', 'MYSORE', 'RAICHUR',
'RAMANAGARA', 'SHIMOGA', 'TUMKUR', 'UDUPI', 'UTTAR KANNAD',

'YADGIR', 'ALAPPUZHA', 'ERNAKULAM', 'IDUKKI', 'KANNUR',
'KASARAGOD', 'KOLLAM', 'KOTTAYAM', 'KOZHIKODE', 'MALAPPURAM',
'PALAKKAD', 'PATHANAMTHITTA', 'THIRUVANANTHAPURAM', 'THRISSUR',
'WAYANAD', 'AGAR MALWA', 'ALIRAJPUR', 'ANUPPUR', 'ASHOKNAGAR',
'BALAGHAT', 'BARWANI', 'BETUL', 'BHIND', 'BHOPAL', 'BURHANPUR',
'CHHATARPUR', 'CHHINDWARA', 'DAMOH', 'DATIA', 'DEWAS', 'DHAR',
'DINDORI', 'GUNA', 'GWALIOR', 'HARDA', 'HOSHANGABAD', 'INDORE',
'JABALPUR', 'JHABUA', 'KATNI', 'KHANDWA', 'KHARGONE', 'MANDLA',
'MANDSAUR', 'MORENA', 'NARSINGHPUR', 'NEEMUCH', 'PANNA', 'RAISEN',
'RAJGARH', 'RATLAM', 'REWA', 'SAGAR', 'SATNA', 'SEHORE', 'SEONI',
'SHAHDOL', 'SHAJAPUR', 'SHEOPUR', 'SHIVPURI', 'SIDHI', 'SINGRAULI',
'TIKAMGARH', 'UJJAIN', 'UMARIA', 'VIDISHA', 'AHMEDNAGAR', 'AKOLA',
'AMRAVATI', 'BEED', 'BHANDARA', 'BULDHANA', 'CHANDRAPUR', 'DHULE',
'GADCHIROLI', 'GONDIA', 'HINGOLI', 'JALGAON', 'JALNA', 'KOLHAPUR',
'LATUR', 'MUMBAI', 'NAGPUR', 'NANDED', 'NANDURBAR', 'NASHIK',
'OSMANABAD', 'PALGHAR', 'PARBHANI', 'PUNE', 'RAIGAD', 'RATNAGIRI',
'SANGLI', 'SATARA', 'SINDHUDURG', 'SOLAPUR', 'THANE', 'WARDHA',
'WASHIM', 'YAVATMAL', 'BISHNUPUR', 'CHANDEL', 'CHURACHANDPUR',
'IMPHAL EAST', 'IMPHAL WEST', 'SENAPATI', 'TAMENGLONG', 'THOUBAL',
'UKHRUL', 'EAST GARO HILLS', 'EAST JAINTIA HILLS',
'EAST KHASI HILLS', 'NORTH GARO HILLS', 'RI BHOI',
'SOUTH GARO HILLS', 'SOUTH WEST GARO HILLS',
'SOUTH WEST KHASI HILLS', 'WEST GARO HILLS', 'WEST JAINTIA HILLS',
'WEST KHASI HILLS', 'AIZAWL', 'CHAMPHAI', 'KOLASIB', 'LAWNGTLAI',
'LUNGLEI', 'MAMIT', 'SAIHA', 'SERCHHIP', 'DIMAPUR', 'KIPHIRE',
'KOHIMA', 'LONGLENG', 'MOKOKCHUNG', 'MON', 'PEREN', 'PHEK',
'TUENSANG', 'WOKHA', 'ZUNHEBOTO', 'ANUGUL', 'BALANGIR',
'BALESHWAR', 'BARGARH', 'BHADRAK', 'BOUDH', 'CUTTACK', 'DEOGARH',
'DHENKANAL', 'GAJAPATI', 'GANJAM', 'JAGATSINGHAPUR', 'JAJAPUR',
'JHARSUGUDA', 'KALAHANDI', 'KANDHAMAL', 'KENDRAPARA', 'KENDUJHAR',
'KHORDHA', 'KORAPUT', 'MALKANGIRI', 'MAYURBHANJ', 'NABARANGPUR',
'NAYAGARH', 'NUAPADA', 'PURI', 'RAYAGADA', 'SAMBALPUR', 'SONEPUR',
'SUNDARGARH', 'KARAIKAL', 'MAHE', 'PONDICHERRY', 'YANAM',
'AMRITSAR', 'BARNALA', 'BATHINDA', 'FARIDKOT', 'FATEHGARH SAHIB',
'FAZILKA', 'FIROZEPUR', 'GURDASPUR', 'HOSHIARPUR', 'JALANDHAR',
'KAPURTHALA', 'LUDHIANA', 'MANSA', 'MOGA', 'MUKTSAR', 'NAWANSHAHR',
'PATHANKOT', 'PATIALA', 'RUPNAGAR', 'S.A.S NAGAR', 'SANGRUR',
'TARN TARAN', 'AJMER', 'ALWAR', 'BANSWARA', 'BARAN', 'BARMER',
'BHARATPUR', 'BHILWARA', 'BIKANER', 'BUNDI', 'CHITTORGARH',
'CHURU', 'DAUSA', 'DHOLPUR', 'DUNGARPUR', 'GANGANAGAR',
'HANUMANGARH', 'JAIPUR', 'JAISALMER', 'JALORE', 'JHALAWAR',
'JHUNJHUNU', 'JODHPUR', 'KARAULI', 'KOTA', 'NAGAUR', 'PALI',
'PRATAPGARH', 'RAJSAMAND', 'SAWAI MADHOPUR', 'SIKAR', 'SIROHI',
'TONK', 'UDAIPUR', 'EAST DISTRICT', 'NORTH DISTRICT',
'SOUTH DISTRICT', 'WEST DISTRICT', 'ARIYALUR', 'COIMBATORE',
'CUDDALORE', 'DHARMAPURI', 'DINDIGUL', 'ERODE', 'KANCHIPURAM',
'KANNIYAKUMARI', 'KARUR', 'KRISHNAGIRI', 'MADURAI', 'NAGAPATTINAM',

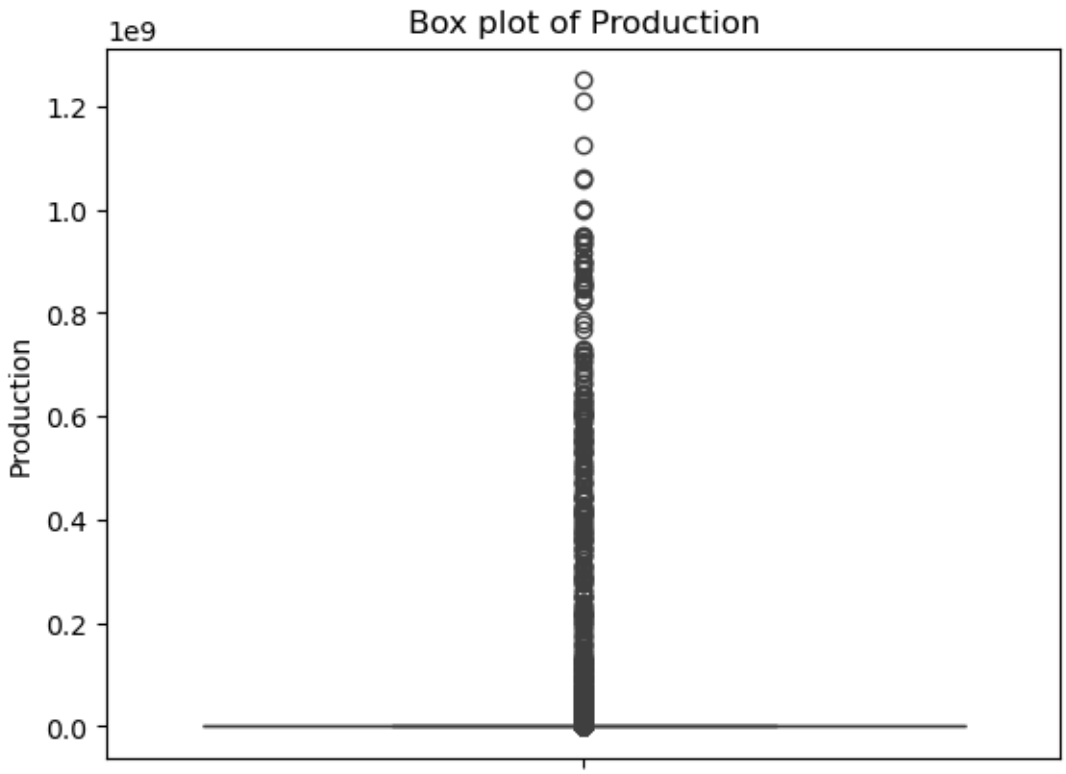
```
'NAMAKKAL', 'PERAMBALUR', 'PUDUKKOTTAI', 'RAMANATHAPURAM', 'SALEM',
'SIVAGANGA', 'THANJAVUR', 'THE NILGIRIS', 'THENI', 'THIRUVALLUR',
'THIRUVARUR', 'TIRUCHIRAPPALLI', 'TIRUNELVELI', 'TIRUPPUR',
'TIRUVANNAMALAI', 'TUTICORIN', 'VELLORE', 'VILLUPURAM',
'VIRUDHUNAGAR', 'ADILABAD', 'HYDERABAD', 'KARIMNAGAR', 'KHAMMAM',
'MAHBUBNAGAR', 'MEDAK', 'NALGONDA', 'NIZAMABAD', 'RANGAREDDI',
'WARANGAL', 'DHALAI', 'GOMATI', 'KHOWAI', 'NORTH TRIPURA',
'SEPAHIJALA', 'SOUTH TRIPURA', 'UNAKOTI', 'WEST TRIPURA', 'AGRA',
'ALIGARH', 'ALLAHABAD', 'AMBEDKAR NAGAR', 'AMETHI', 'AMROHA',
'AURAIYA', 'AZAMGARH', 'BAGHPAT', 'BAHRAICH', 'BALLIA', 'BANDA',
'BARABANKI', 'BAREILLY', 'BASTI', 'BIJNOR', 'BUDAUN',
'BULANDSHAHR', 'CHANDAULI', 'CHITRAKOOT', 'DEORIA', 'ETAH',
'ETAWAH', 'FAIZABAD', 'FARRUKHABAD', 'FATEHPUR', 'FIROZABAD',
'GAUTAM BUDDHA NAGAR', 'GHAZIABAD', 'GHAZIPUR', 'GONDA',
'GORAKHPUR', 'HAPUR', 'HARDOI', 'HATHRAS', 'JALAUN', 'JAUNPUR',
'JHANSI', 'KANNAUJ', 'KANPUR DEHAT', 'KANPUR NAGAR', 'KASGANJ',
'KAUSHAMBI', 'KHERI', 'KUSHI NAGAR', 'LALITPUR', 'LUCKNOW',
'MAHARAJGANJ', 'MAHOBA', 'MAINPURI', 'MATHURA', 'MAU', 'MEERUT',
'MIRZAPUR', 'MORADABAD', 'MUZAFFARNAGAR', 'PILIBHIT', 'RAE BARELI',
'RAMPUR', 'SAHARANPUR', 'SAMBHAL', 'SANT KABEER NAGAR',
'SANT RAVIDAS NAGAR', 'SHAHJAHANPUR', 'SHAMLI', 'SHRAVASTI',
'SIDDHARTH NAGAR', 'SITAPUR', 'SONBHADRA', 'SULTANPUR', 'UNNAO',
'VARANASI', 'ALMORA', 'BAGESHWAR', 'CHAMOLI', 'CHAMPAWAT',
'DEHRADUN', 'HARIDWAR', 'NAINITAL', 'PAURI GARHWAL', 'PITHORAGARH',
'RUDRA PRAYAG', 'TEHRI GARHWAL', 'UDAM SINGH NAGAR', 'UTTAR KASHI',
'24 PARAGANAS NORTH', '24 PARAGANAS SOUTH', 'BANKURA', 'BARDHAMAN',
'BIRBHUM', 'COOCHBEHAR', 'DARJEELING', 'DINAJPUR DAKSHIN',
'DINAJPUR UTTAR', 'HOOGHLY', 'HOWRAH', 'JALPAIGURI', 'MALDAH',
'MEDINIPUR EAST', 'MEDINIPUR WEST', 'MURSHIDABAD', 'NADIA',
'PURULIA'], dtype=object)
```

1.4.4 *EXPLORING THE DATA WITH EXPLORATORY DATA ANALYSIS AND ASKED THE QUESTION ON THE DATA*

[29]:
```
sns.boxplot(y='Area', data=df)
plt.title('Box plot of Area')
plt.show()
```

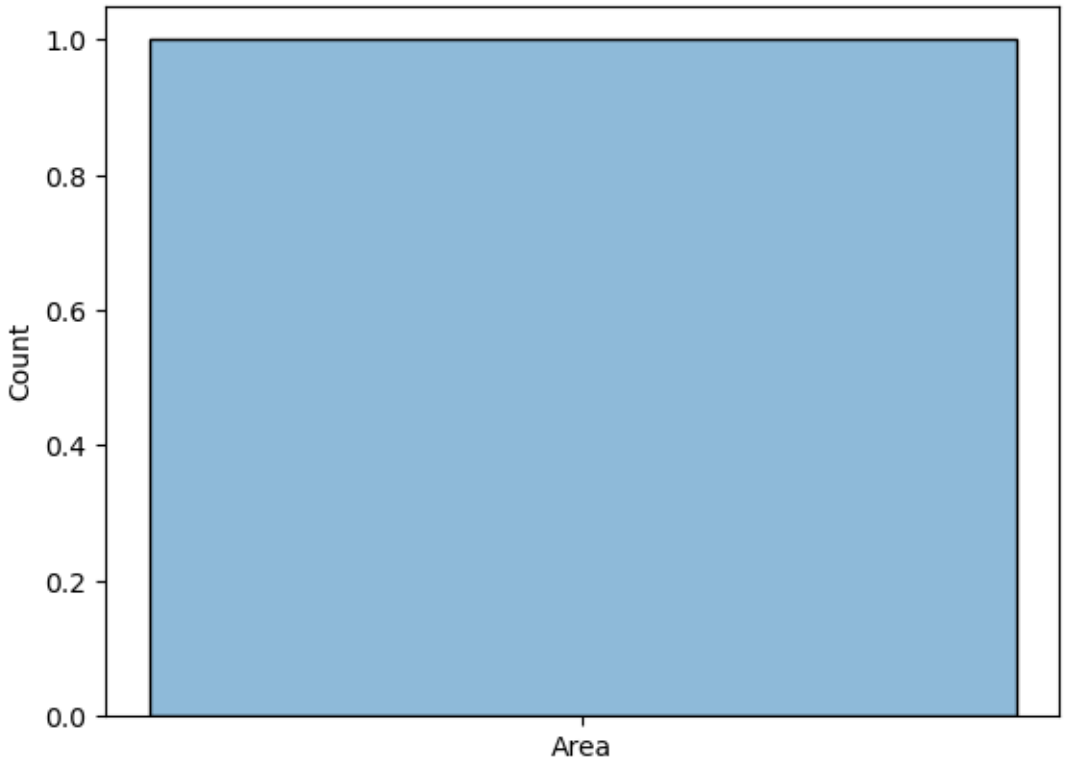


[30]:
```
sns.boxplot(y='Production', data=df)
plt.title('Box plot of Production')
plt.show()
```



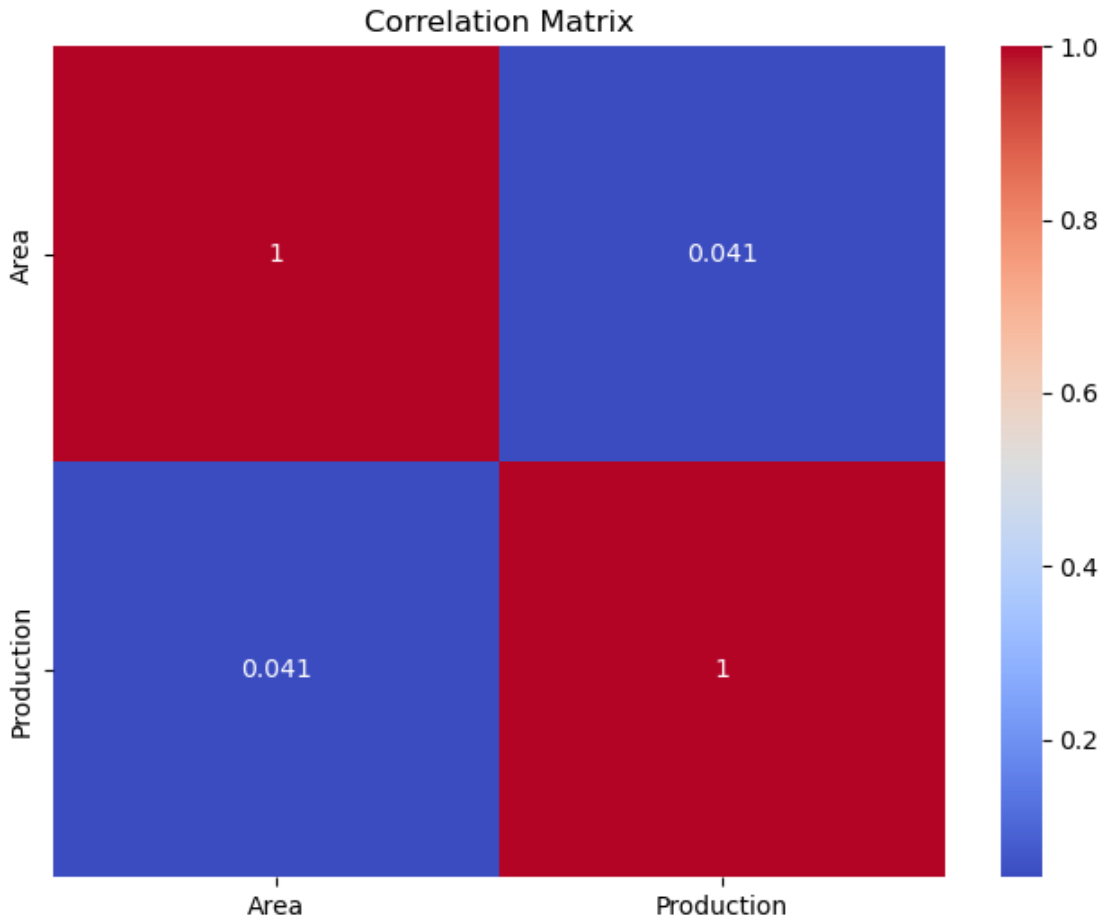
[31]:
```
sns.histplot('Area',kde=True,bins=20)
```

[31]:
```
<Axes: ylabel='Count'>
```



1.4.5 *Co-Relation In Data*

[33]:
```
# Correlation matrix
correlation_matrix = df[['Area', 'Production']].corr()

plt.figure(figsize=(8, 6))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```

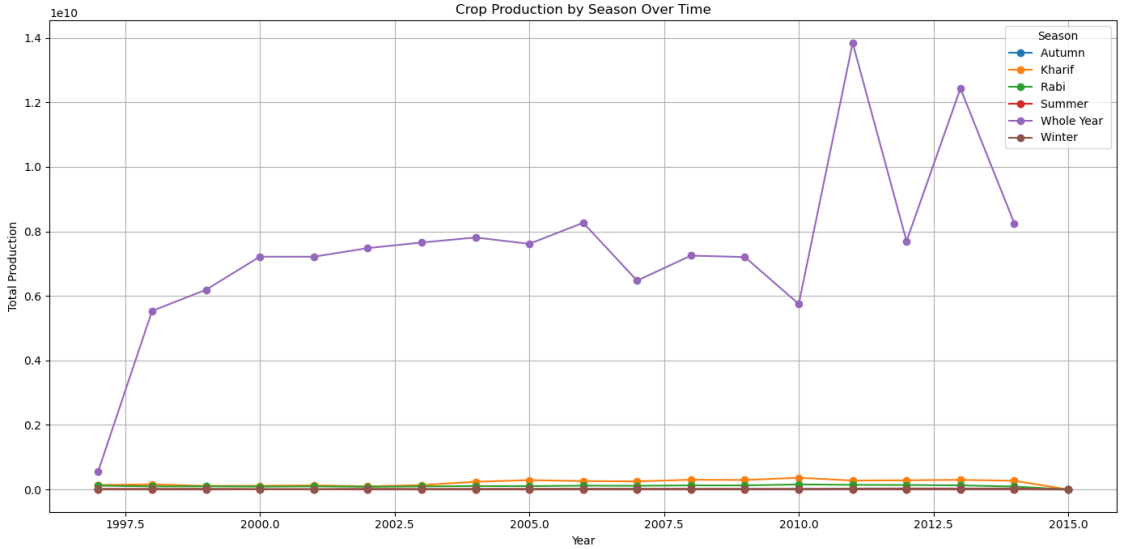


- *There is low co relation in the data*

[35]:
```
#Zone-Wise Production - 1997-2014
north_india = ['Jammu and Kashmir', 'Punjab', 'Himachal Pradesh', 'Haryana',␣
 ↪'Uttarakhand', 'Uttar Pradesh', 'Chandigarh']
east_india = ['Bihar', 'Odisha', 'Jharkhand', 'West Bengal']
south_india = ['Andhra Pradesh', 'Karnataka', 'Kerala' ,'Tamil Nadu',␣
 ↪'Telangana']
west_india = ['Rajasthan' , 'Gujarat', 'Goa','Maharashtra']
central_india = ['Madhya Pradesh', 'Chhattisgarh']
north_east_india = ['Assam', 'Sikkim', 'Nagaland', 'Meghalaya', 'Manipur',␣
 ↪'Mizoram', 'Tripura', 'Arunachal Pradesh']
ut_india = ['Andaman and Nicobar Islands', 'Dadra and Nagar Haveli',␣
 ↪'Puducherry']
```

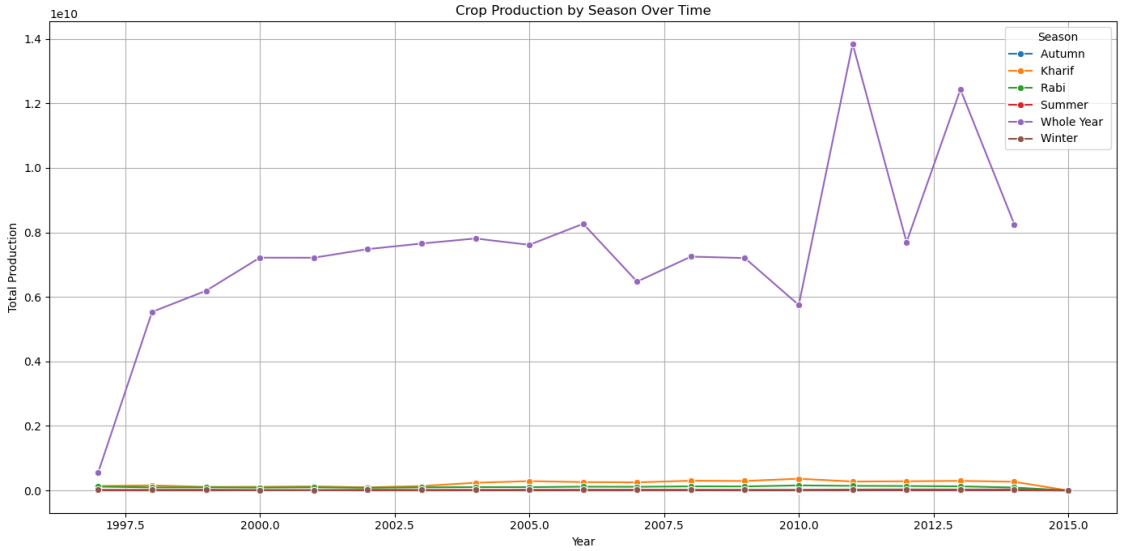
[36]:
```
# Aggregate production by season and year
seasonal_production = df.groupby(['Crop_Year', 'Season'])['Production'].sum().
 ↪reset_index()
```

[37]:
```
# Pivot table for plotting
pivot_table = seasonal_production.pivot(index='Crop_Year', columns='Season',␣
 ↪values='Production')

# Plot the data
plt.figure(figsize=(14, 7))
for season in pivot_table.columns:
    plt.plot(pivot_table.index, pivot_table[season], marker='o', label=season)

plt.title('Crop Production by Season Over Time')
plt.xlabel('Year')
plt.ylabel('Total Production')
plt.legend(title='Season')
plt.grid(True)
plt.tight_layout()
plt.show()
```

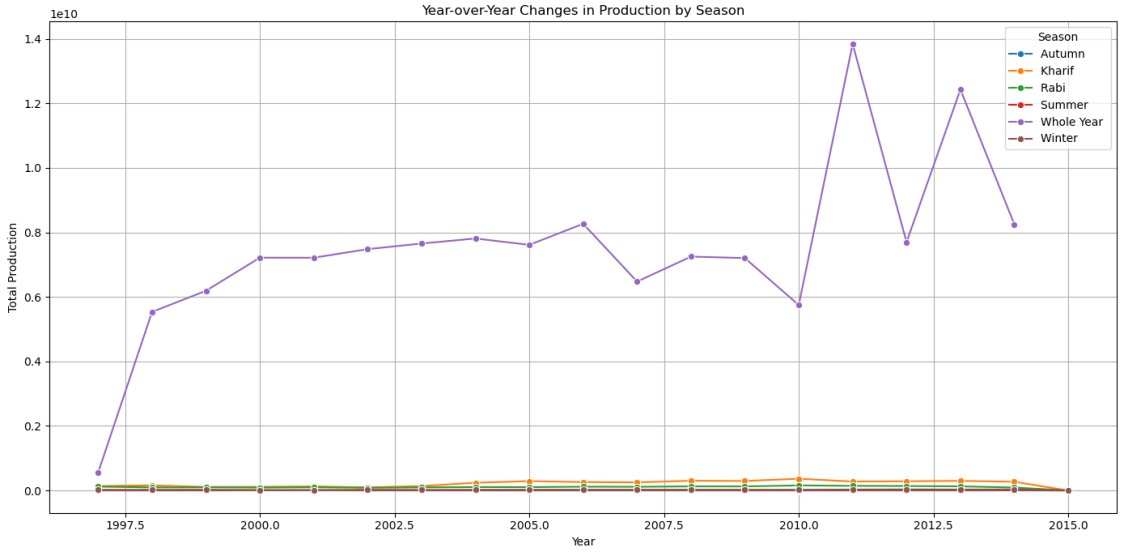


[38]:
```
import seaborn as sns

# Plot using Seaborn
plt.figure(figsize=(14, 7))
sns.lineplot(data=seasonal_production, x='Crop_Year', y='Production',␣
 ↪hue='Season', marker='o')

plt.title('Crop Production by Season Over Time')
plt.xlabel('Year')
plt.ylabel('Total Production')
plt.grid(True)
```

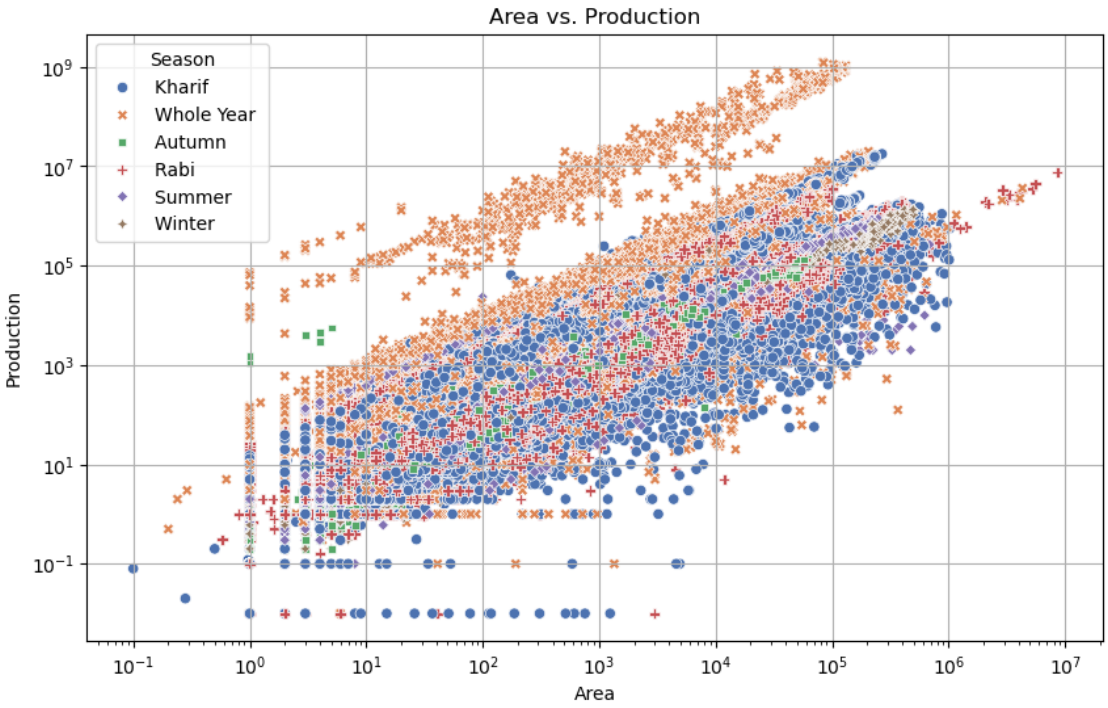
```
plt.tight_layout()
plt.show()
```



[39]:
```
# Plotting year-over-year changes
plt.figure(figsize=(14, 7))
sns.lineplot(data=seasonal_production, x='Crop_Year', y='Production',␣
 ↪hue='Season', marker='o')
plt.title('Year-over-Year Changes in Production by Season')
plt.xlabel('Year')
plt.ylabel('Total Production')
plt.grid(True)
plt.tight_layout()
plt.show()
```



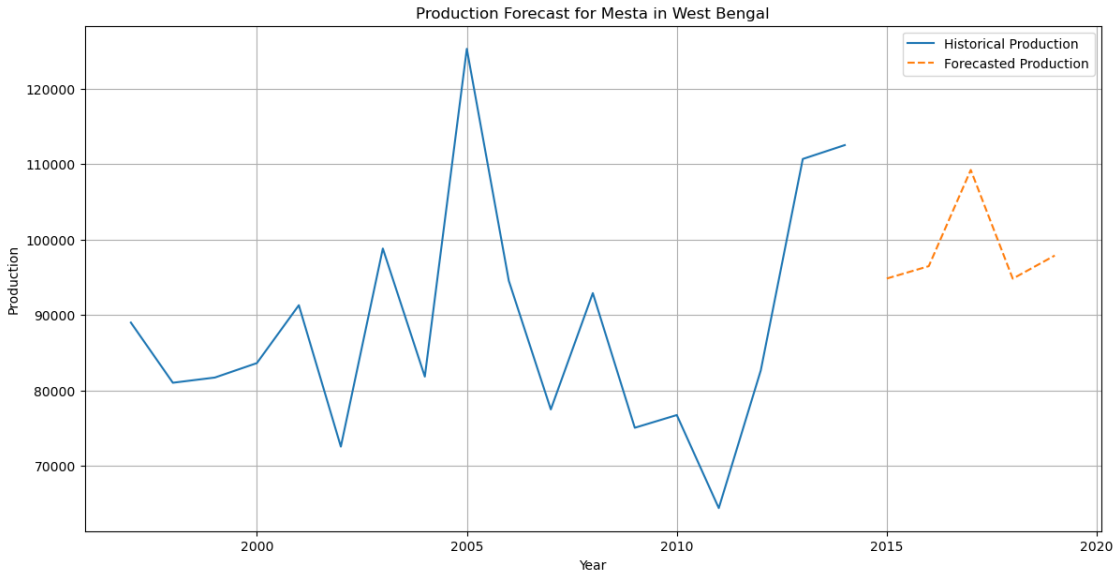
[40]:
```
# Plotting Area vs. Production
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='Area', y='Production', hue='Season',␣
 ↪style='Season', palette='deep')
plt.title('Area vs. Production')
plt.xlabel('Area')
plt.ylabel('Production')
plt.xscale('log')
plt.yscale('log')
plt.grid(True)
plt.show()
```



[41]:
```
from statsmodels.tsa.api import ExponentialSmoothing

# Example: Forecasting production for one crop in one state
state_crop_data = df[(df['State_Name'] == 'West Bengal') & (df['Crop'] ==␣
 ↪'Mesta')]
state_crop_data = state_crop_data.groupby('Crop_Year')['Production'].sum().
 ↪reset_index()

# Fit the model
model = ExponentialSmoothing(state_crop_data['Production'], trend='add',␣
 ↪seasonal='add', seasonal_periods=4)
fit = model.fit()

# Forecast future production
forecast = fit.forecast(steps=5)

# Plot historical and forecasted production
plt.figure(figsize=(14, 7))
plt.plot(state_crop_data['Crop_Year'], state_crop_data['Production'],␣
 ↪label='Historical Production')
plt.plot(range(state_crop_data['Crop_Year'].max() + 1,␣
 ↪state_crop_data['Crop_Year'].max() + 6), forecast, label='Forecasted␣
 ↪Production', linestyle='--')
```

```
plt.title('Production Forecast for Mesta in West Bengal')
plt.xlabel('Year')
plt.ylabel('Production')
plt.legend()
plt.grid(True)
plt.show()
```

```
C:\Users\HELLO\anaconda\Lib\site-
packages\statsmodels\tsa\holtwinters\model.py:918: ConvergenceWarning:
Optimization failed to converge. Check mle_retvals.
  warnings.warn(
```



[42]:
```
#Zone-Wise Production - 1997-2014
north_india = ['Jammu and Kashmir', 'Punjab', 'Himachal Pradesh', 'Haryana',␣
 ↪'Uttarakhand', 'Uttar Pradesh', 'Chandigarh']
east_india = ['Bihar', 'Odisha', 'Jharkhand', 'West Bengal']
south_india = ['Andhra Pradesh', 'Karnataka', 'Kerala' ,'Tamil Nadu',␣
 ↪'Telangana']
west_india = ['Rajasthan' , 'Gujarat', 'Goa','Maharashtra']
central_india = ['Madhya Pradesh', 'Chhattisgarh']
north_east_india = ['Assam', 'Sikkim', 'Nagaland', 'Meghalaya', 'Manipur',␣
 ↪'Mizoram', 'Tripura', 'Arunachal Pradesh']
ut_india = ['Andaman and Nicobar Islands', 'Dadra and Nagar Haveli',␣
 ↪'Puducherry']
```

[43]:
```
def get_zonal_names(row):
    if row['State_Name'].strip() in north_india:
        val = 'North Zone'
```

```
    elif row['State_Name'].strip()  in south_india:
        val = 'South Zone'
    elif row['State_Name'].strip()  in east_india:
        val = 'East Zone'
    elif row['State_Name'].strip()  in west_india:
        val = 'West Zone'
    elif row['State_Name'].strip()  in central_india:
        val = 'Central Zone'
    elif row['State_Name'].strip()  in north_east_india:
        val = 'NE Zone'
    elif row['State_Name'].strip()  in ut_india:
        val = 'Union Terr'
    else:
        val = 'No Value'
    return val

df['Zones'] = df.apply(get_zonal_names, axis=1)
df['Zones'].unique()
```

[43]:
```
array(['Union Terr', 'South Zone', 'NE Zone', 'East Zone', 'North Zone',
       'Central Zone', 'West Zone'], dtype=object)
```

[44]:
```
crop=df['Crop']
def cat_crop(crop):
    for i in ['Rice','Maize','Wheat','Barley','Varagu','Other Cereals &␣
 ↪Millets','Ragi','Small millets','Bajra','Jowar', 'Paddy','Total␣
 ↪foodgrain','Jobster']:
        if crop==i:
            return 'Cereal'
    for i in ['Moong','Urad','Arhar/Tur','Peas & beans','Masoor',
              'Other Kharif pulses','other misc. pulses','Ricebean (nagadal)',
              'Rajmash␣
 ↪Kholar','Lentil','Samai','Blackgram','Korra','Cowpea(Lobia)',
              'Other  Rabi pulses','Other Kharif pulses','Peas & beans␣
 ↪(Pulses)','Pulses total','Gram']:
        if crop==i:
            return 'Pulses'
    for i in␣
 ↪['Peach','Apple','Litchi','Pear','Plums','Ber','Sapota','Lemon','Pome␣
 ↪Granet',
               'Other Citrus Fruit','Water Melon','Jack␣
 ↪Fruit','Grapes','Pineapple','Orange',
              'Pome Fruit','Citrus Fruit','Other Fresh␣
 ↪Fruits','Mango','Papaya','Coconut','Banana']:
        if crop==i:
            return 'Fruits'
    for i in ['Bean','Lab-Lab','Moth','Guar seed','Soyabean','Horse-gram']:
```

```
        if crop==i:
            return 'Beans'
    for i in ['Turnip','Peas','Beet Root','Carrot','Yam','Ribed Guard','Ash␣
↪Gourd ','Pump Kin','Redish','Snak Guard','Bottle Gourd',
              'Bitter Gourd','Cucumber','Drum Stick','Cauliflower','Beans &␣
↪Mutter(Vegetable)','Cabbage',
              'Bhindi','Tomato','Brinjal','Khesari','Sweet␣
↪potato','Potato','Onion','Tapioca','Colocosia']:
              if crop==i:
                return 'Vegetables'
    for i in ['Perilla','Ginger','Cardamom','Black pepper','Dry␣
↪ginger','Garlic','Coriander','Turmeric','Dry chillies','Cond-spcs other']:
        if crop==i:
            return 'spices'
    for i in ['other fibres','Kapas','Jute &␣
↪mesta','Jute','Mesta','Cotton(lint)','Sannhamp']:
        if crop==i:
            return 'fibres'
    for i in ['Arcanut (Processed)','Atcanut (Raw)','Cashewnut␣
↪Processed','Cashewnut Raw','Cashewnut','Arecanut','Groundnut']:
        if crop==i:
            return 'Nuts'
    for i in ['other oilseeds','Safflower','Niger seed','Castor␣
↪seed','Linseed','Sunflower','Rapeseed &Mustard','Sesamum','Oilseeds total']:
        if crop==i:
            return 'oilseeds'
    for i in ['Tobacco','Coffee','Tea','Sugarcane','Rubber']:
        if crop==i:
            return 'Commercial'

df['cat_crop']=df['Crop'].apply(cat_crop)
```

[45]:
```
df["cat_crop"].value_counts()
```

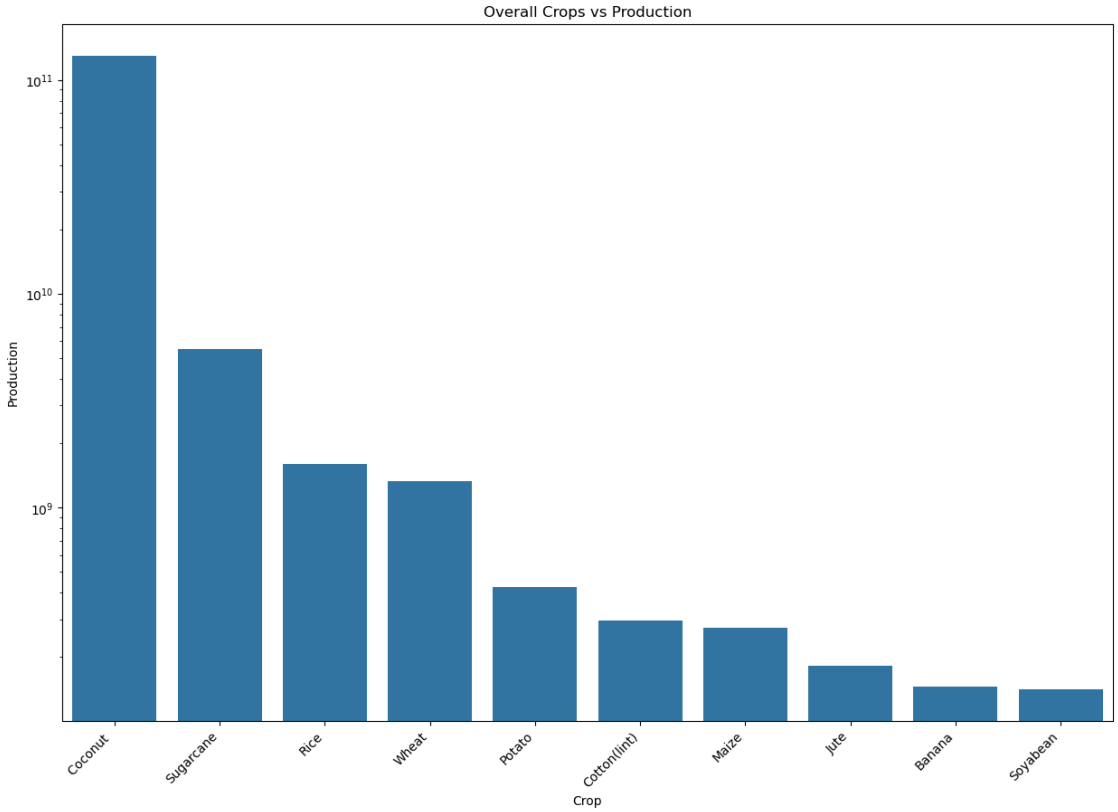
[45]:
```
cat_crop
Cereal        63283
Pulses        40898
oilseeds      33801
Vegetables    23154
spices        21638
Nuts          11472
Commercial    10561
fibres         9785
Beans          9115
Fruits         6153
Name: count, dtype: int64
```

[46]:
```
data_explore = df.copy()
```

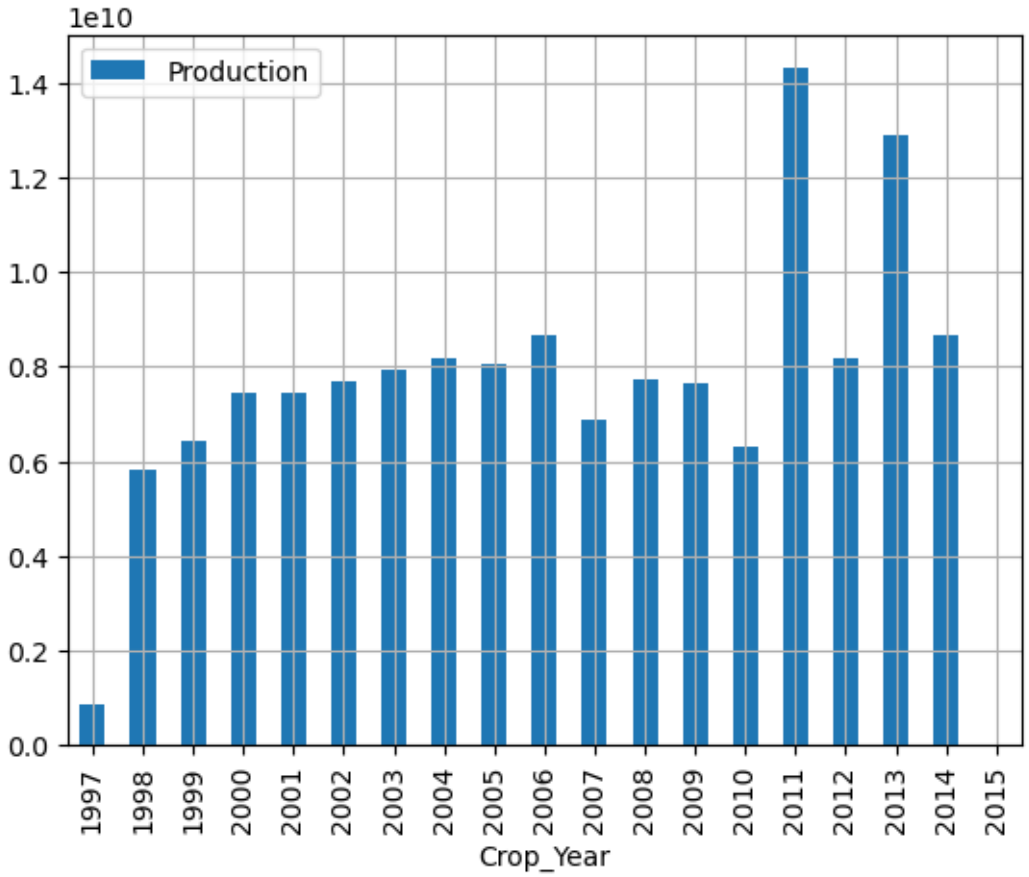
[47]:
```
df.Zones.value_counts()
```

[47]:
```
Zones
South Zone      53500
North Zone      49874
East Zone       43261
West Zone       33134
Central Zone    32972
NE Zone         28284
Union Terr       1336
Name: count, dtype: int64
```

[48]:
```
crop = data_explore.groupby(by='Crop')['Production'].sum().reset_index().
 ↪sort_values(by='Production', ascending=False).head(10)

# Create a bar plot
fig, ax = plt.subplots(figsize=(15,10))
sns.barplot(x=crop.Crop, y=crop.Production, ax=ax)

# Set y-axis to logarithmic scale
plt.yscale('log')

# Add title and labels
plt.title('Overall Crops vs Production')
plt.xlabel('Crop')
plt.ylabel('Production')

# Rotate x-axis labels for better readability
plt.xticks(rotation=45, ha='right')

# Display the plot
plt.show()
```

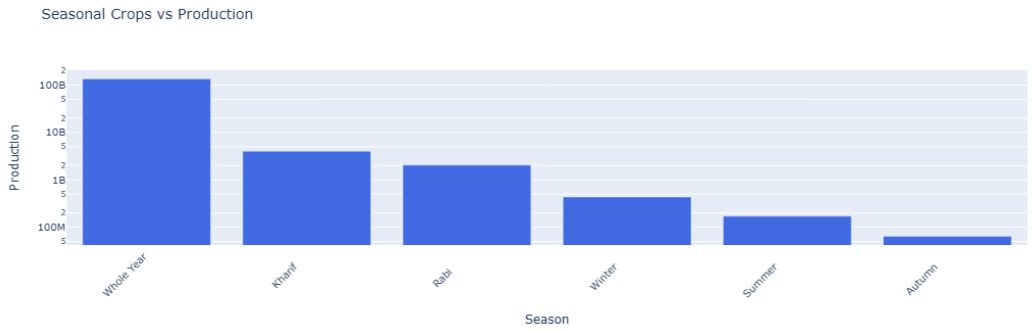


[49]:
```
plt.tick_params(labelsize=10)
data_explore.groupby("Crop_Year")["Production"].agg("sum").plot.bar()
plt.grid()
plt.legend()
plt.show()
```

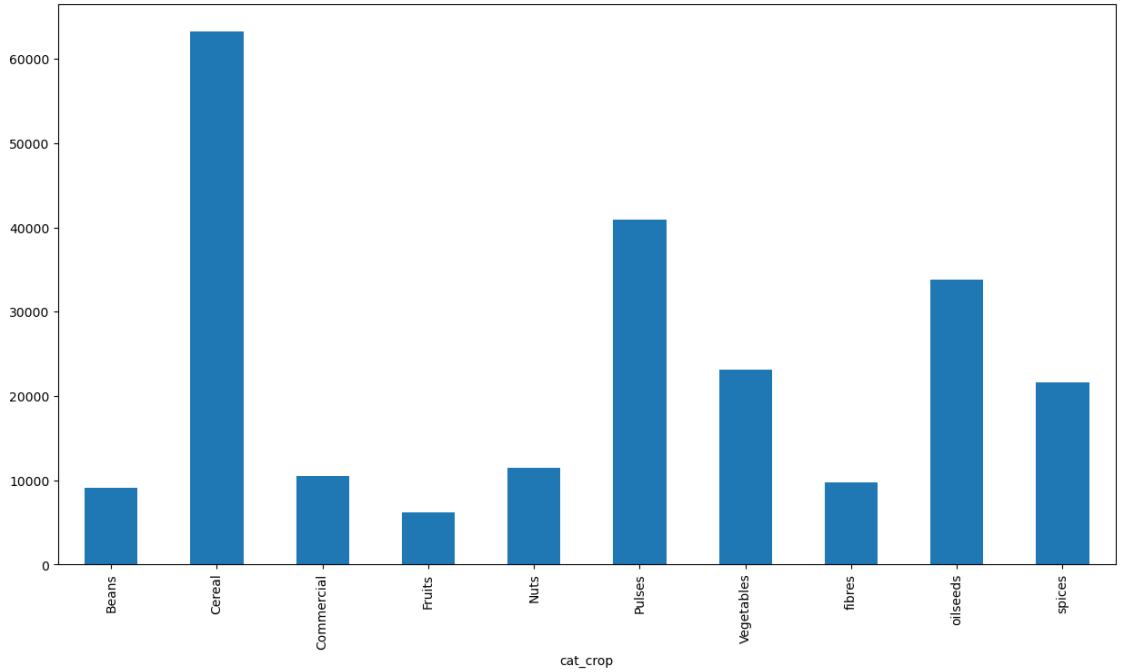


[50]:
```
df_season = data_explore.copy()
season = df_season.groupby(by='Season')['Production'].sum().reset_index().
 ↪sort_values(by='Production', ascending=False).head(10)

# Create a bar plot with Plotly
fig = go.Figure()

# Add bar trace
fig.add_trace(go.Bar(
    x=season['Season'],
    y=season['Production'],
    marker=dict(color='royalblue')  # Customize color as needed
))

# Update layout
fig.update_layout(
    title='Seasonal Crops vs Production',
    xaxis_title='Season',
    yaxis_title='Production',
```

```
    yaxis=dict(type='log'),  # Use logarithmic scale if needed
    xaxis=dict(tickangle=-45),  # Rotate x-axis labels if needed
    font=dict(size=10)  # Set font size for axis labels
)

# Show plot
fig.show()
```

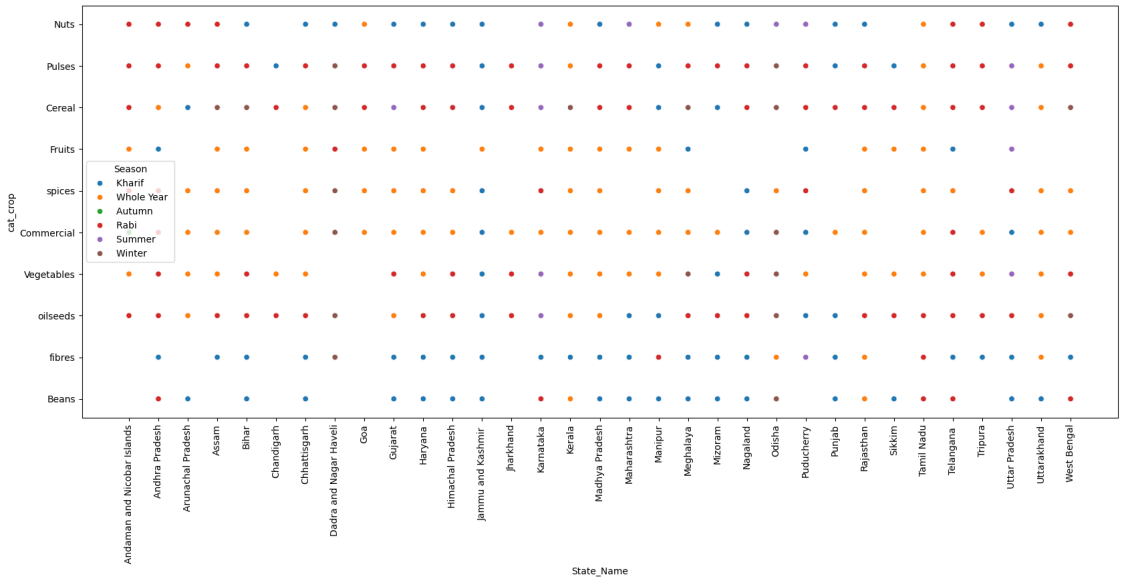


[51]:
```
plt.figure(figsize=(15,8))
plt.tick_params(labelsize=10)
data_explore.groupby("cat_crop")["Production"].agg("count").plot.bar()
plt.show()
```

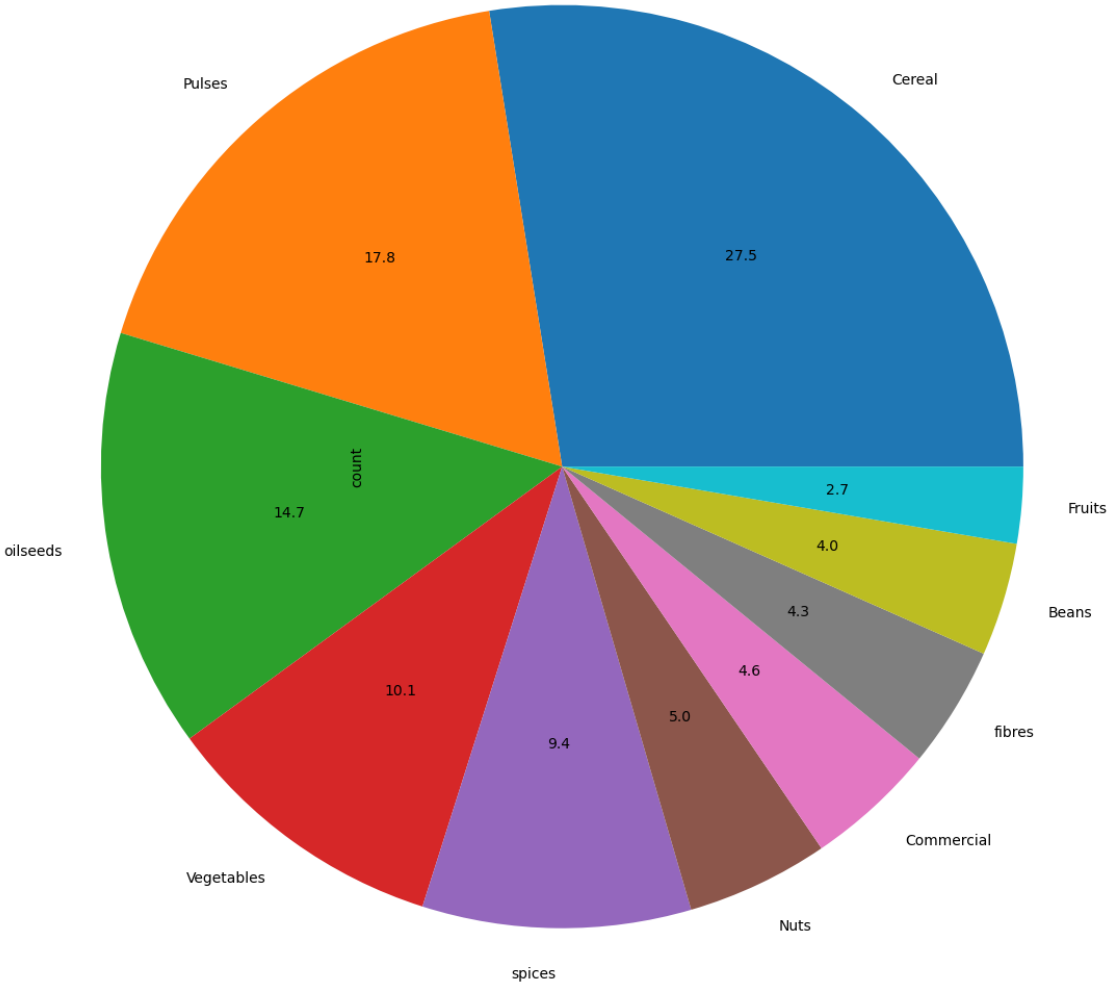


[52]:
```
plt.figure(figsize=(20,8))
sns.scatterplot(data=data_explore,x="State_Name",y="cat_crop",hue="Season")
plt.xticks(rotation=90)
plt.show()
```

```
C:\Users\HELLO\AppData\Roaming\Python\Python312\site-
packages\IPython\core\pylabtools.py:152: UserWarning:

Creating legend with loc="best" can be slow with large amounts of data.
```



[53]:
```
df1=data_explore["cat_crop"].value_counts()
df1.plot(radius=3,kind="pie",autopct="%1.1f",pctdistance=0.6)
plt.tick_params(labelsize=10)
```



- Kerala is top state when we look at the quantum of Production for last 19.years.
- Top production years are 2011, 2013 and 2014.
- Top crop categories which shows high production values are Whole Year(Annual growing plants),Kharif and Rabi crops. It clearly shows these crops heavily dependent on seasonal monssons.
- Top crop categories are Cereal, Pulses and Oilseeds.

Interesting facts: * South zone: i. Top producing state Kerela shows a abundance of whole year seasonal crops * North Zone: ii. Top producing state Uttar Pradesh shows abundance of Kharif, Rabi and Summar crops

1.5 Insights and Conclusions

1.5.1 Dataset Overview

- **Initial Dataset**: Consisted of 246,091 records with 7 attributes.
- **Handling Missing Data**: The `Production` variable had 3,730 missing entries (1.5% of the total). These were excluded, leaving 242,361 records.

- **Multicollinearity Assessment**: Conducted using a correlation heatmap.

1.5.2 Univariate Analysis Insights

- **State_Name**: Represents 33 states and union territories. Major contributors are Uttar Pradesh, Madhya Pradesh, and Karnataka.
- **District_Name**: Covers 646 districts. Leading contributors include Tumkur, Belgaum, and Bijapur from Karnataka.
- **Crop_Year**: Spans from 1997 to 2015, with the highest data concentrations in 2002, 2003, and 2007.
- **Season**: Includes six seasons, with the majority of data from Kharif, Rabi, and Whole Year.
- **Crop**: Data covers 124 crop types, with the most frequent being Rice, Maize, and Moong (Green Gram).
- **Area**: Ranges from 1 to 8,580,100 units, with a highly right-skewed distribution due to numerous outliers.
- **Production**: Ranges from 0 to 1.25e+09, also right-skewed due to outliers.

1.5.3 Bivariate Analysis Observations

- **State_Name vs Production**: Kerala, Andhra Pradesh, and Tamil Nadu are the leading states in terms of production.

1.5.4 Newly Introduced Variables

- **Zones**: States categorized into North, South, East, West, Central, NE, and Union Territories. The dataset shows significant data from South, North, and East zones.
- **Crop Categories**: 124 crops divided into Cereal, Pulses, Oilseeds, Vegetables, Spices, Nuts, Commercial, Fibers, Beans, and Fruits. The most common categories are Cereal, Pulses, and Oilseeds.

1.5.5 Visualization Highlights

- **Zonal Crop Distribution**: The South zone, particularly Kerala, leads in crop production.
- **Crop Production Overview**: Coconut, Sugarcane, and Rice are the top crops by production volume.
- **Production Trends Over Years**: Peak production observed in 2011 and 2013.
- **Seasonal Production Trends**: Whole Year (annual crops), Kharif, and Rabi crops show the highest production, reflecting their dependence on seasonal rains.
- **Crop Category Production Trends**: Cereal, Pulses, and Oilseeds are the dominant categories.
- **State vs Crop Category vs Season Analysis**:
  - Kerala excels in Whole Year crops.
  - Uttar Pradesh is notable for Kharif, Rabi, and Summer crops.
- **Crop Category Proportions**: Cereal (27.5%), Pulses (17.8%), and Oilseeds (14.7%) contribute to 60% of total crop production.

1.5.6 Key Questions Addressed

Q1: Which states lead in crop production across various categories?

- **Dominant State**: Uttar Pradesh excels in numerous crop categories:
  - Beans: 1,112
  - Cereal: 9,719
  - Commercial: 1,741
  - Fruits: 269
  - Nuts: 958
  - Pulses: 6,549
  - Vegetables: 3,734
  - Fibers: 724
  - Oilseeds: 4,028
  - Spices: 2,529

Q2: What is the most prevalent crop, and where is it cultivated?

- **Most Prevalent Crop**: Rice
  - **Growing Conditions**: Requires Winter for maturation.
  - **Top State**: Punjab
  - **Top Districts**: Bardhaman (2.13%), Medinipur West (1.8%), and West Godavari (1.73%).
  - **Peak Production Year**: 2014
  - **Area and Production Correlation**: Higher production correlates with larger cultivation areas.

Q3: Which states are the largest in terms of cultivation area?

- **Top Cultivation States**:
  - Uttar Pradesh: 4.33e+08
  - Madhya Pradesh: 3.29e+08
  - Maharashtra: 3.22e+08
  - **Yearly Trends**:
    - **Uttar Pradesh**: Peak production in 2005; gradual decline afterward.
    - **Madhya Pradesh**: High production in 1998; subsequent decline and recovery with peaks in 2012.
    - **Maharashtra**: Significant drop in 2006, followed by recovery and peak post-2007.
    - **Rajasthan**: Low production in 2002, with recovery by 2010.
    - **West Bengal**: Peak in 2006, with a decline post-2007.

Q4: What are the top crops in Northern India?

- **Leading States in North Zone**:
  - Punjab: 5.86e+08
  - Uttar Pradesh: 3.23e+09
  - Haryana: 3.81e+08
- **Top Crops**: Sugarcane, Wheat, and Rice.

Q5: Status of Coconut Production in South India?

- **Coconut Cultivation**: Continues year-round, unaffected by seasons.
  - **Leading States**: Kerala, Andhra Pradesh, and Tamil Nadu.

- **Top Districts**: Kozhikode (11.75%), Malappuram (11.16%), and Thiruvananthapuram (7.7%).
- **Yearly Trends**: Strong and increasing cultivation, with high correlation to cultivation area.

[ ]: