Crop Production Analysis

Problem Statement:
As a crucial link in the supply chain, the agriculture industry is predicted to undergo significant change in the next years due to advancements on the Future Internet front. In order to promote the efficient and adaptable collaboration of several stakeholders from related business domains, this paper introduces a novel Business-to-Business collaboration platform from the perspective of the agri-food sector.

Code:

Mounting to Google Drive

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

Importing necessary libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Loading Data set File

```
df=pd.read_csv("/content/drive/MyDrive/Crop Production data.csv")
df
```

	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 × 7 columns

There are 246091 rows and 7 columns present in the data set

Checking the Variable Types

```
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
```

This Data set contains 4 categorical variables (State_Name, District_Name, Season, Crop) and 3 continuous variables (Crop_Year, Area, Production)

Checking for Missing values

```
df.isnull().sum()
```

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

```
100*df.isnull().mean()
```

```
State_Name         0.000000
District_Name      0.000000
Crop_Year          0.000000
Season             0.000000
Crop               0.000000
Area               0.000000
Production         1.515699
dtype: float64
```

There are 3730 missing values in the production variable, or 1.51% of the entire sample size; no other variable has any missing values.

Dropping Production column from the Data set

```
df.drop("Production",axis=1)
```

	State_Name	District_Name	Crop_Year	Season	Crop	Area
0	Andaman and Nicobar Islands	NICOBARS	2000	Kharif	Arecanut	1254.0
1	Andaman and Nicobar Islands	NICOBARS	2000	Kharif	Other Kharif pulses	2.0
2	Andaman and Nicobar Islands	NICOBARS	2000	Kharif	Rice	102.0
3	Andaman and Nicobar Islands	NICOBARS	2000	Whole Year	Banana	176.0
4	Andaman and Nicobar Islands	NICOBARS	2000	Whole Year	Cashewnut	720.0
...	...	...	...	...	...	...
246086	West Bengal	PURULIA	2014	Summer	Rice	306.0
246087	West Bengal	PURULIA	2014	Summer	Sesamum	627.0
246088	West Bengal	PURULIA	2014	Whole Year	Sugarcane	324.0
246089	West Bengal	PURULIA	2014	Winter	Rice	279151.0
246090	West Bengal	PURULIA	2014	Winter	Sesamum	175.0

246091 rows × 6 columns

After dropping the Production column from the Data set, Now we are having 246091 rows and 6 columns for data analysis.

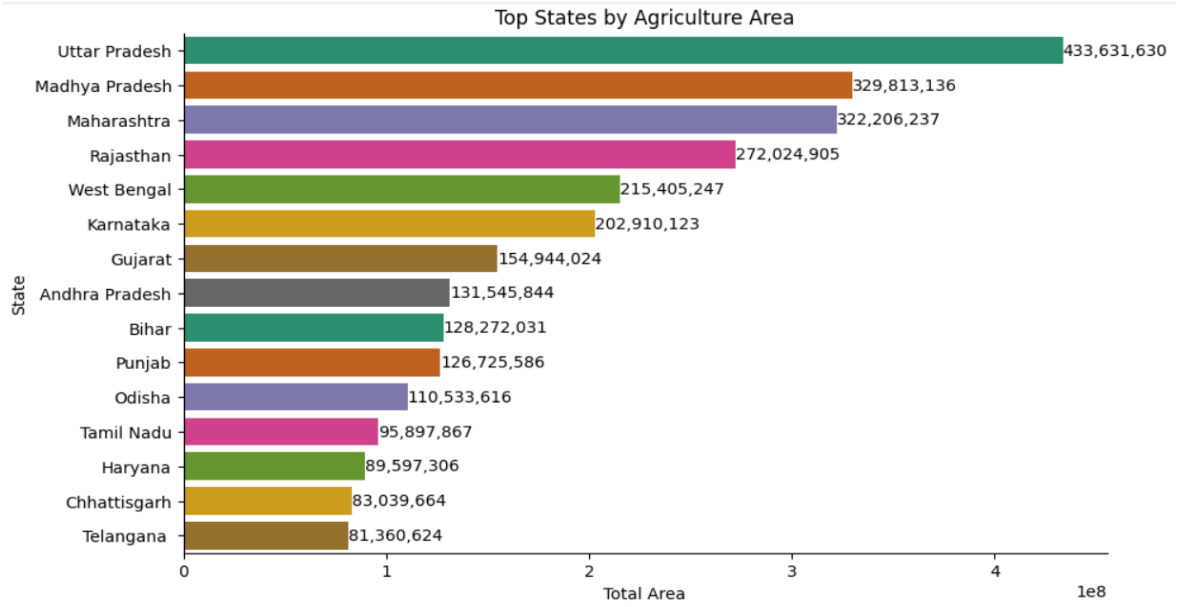
Top 3 Agriculture rich states

```
Top_States=df.groupby('State_Name')['Area'].sum().sort_values(ascending=False).astype(int).head(15)
print(Top_States)
```

```
State_Name
Uttar Pradesh     433631630
Madhya Pradesh    329813136
Maharashtra       322206237
Rajasthan         272024905
West Bengal       215405247
Karnataka         202910123
Gujarat           154944024
Andhra Pradesh    131545844
Bihar             128272031
Punjab            126725586
Odisha            110533616
Tamil Nadu         95897867
Haryana            89597306
Chhattisgarh       83039664
Telangana          81360624
Name: Area, dtype: int64
```

From the above information, we came to know that ‘Uttar Pradesh’, ‘Madhya Pradesh’ and ‘Maharashtra’ are the top 3 agriculture rich states.

```
plt.figure(figsize=(10, 6))
sns.barplot(x=Top_States.values, y=Top_States.index, palette='Dark2')
for index, value in enumerate(Top_States):
    plt.text(value, index, f'{value:,}', va='center', ha='left')
plt.xlabel('Total Area')
plt.ylabel('State')
plt.title('Top States by Agriculture Area')
plt.gca().spines[['top', 'right']].set_visible(False)
plt.show()
```



District wise Analysis

```
df.District_Name.nunique()
```

```
646
```

```
df.District_Name.value_counts()
```

```
District_Name
BIJAPUR      945
TUMKUR       936
BELGAUM      925
HASSAN       895
BELLARY      887
            ...
HYDERABAD      8
KHUNTI         6
RAMGARH        6
NAMSAI         1
MUMBAI         1
Name: count, Length: 646, dtype: int64
```

There are total 646 districts taken into consideration for data analysis, among all of them ‘BIJAPUR’, ‘TUMKUR’ and ‘BELGAUM’ are majorly contributed.

Crop Year Analysis

```
df.Crop_Year.value_counts()
```

```
Crop_Year
2003    17287
2002    16671
2008    14550
2007    14526
2006    14328
2004    14117
2009    14116
2011    14071
2010    14065
2005    13799
2000    13658
2013    13650
2012    13410
2001    13361
1999    12515
1998    11533
2014    10973
1997     8899
2015      562
Name: count, dtype: int64
```

```
df.Crop_Year.nunique()
```

```
19
```

The Sample set contains 19 unique years among all of them the top 3 years are 2003, 2002 and 2008.

Seasonal Analysis

```
df.Season.nunique()
```

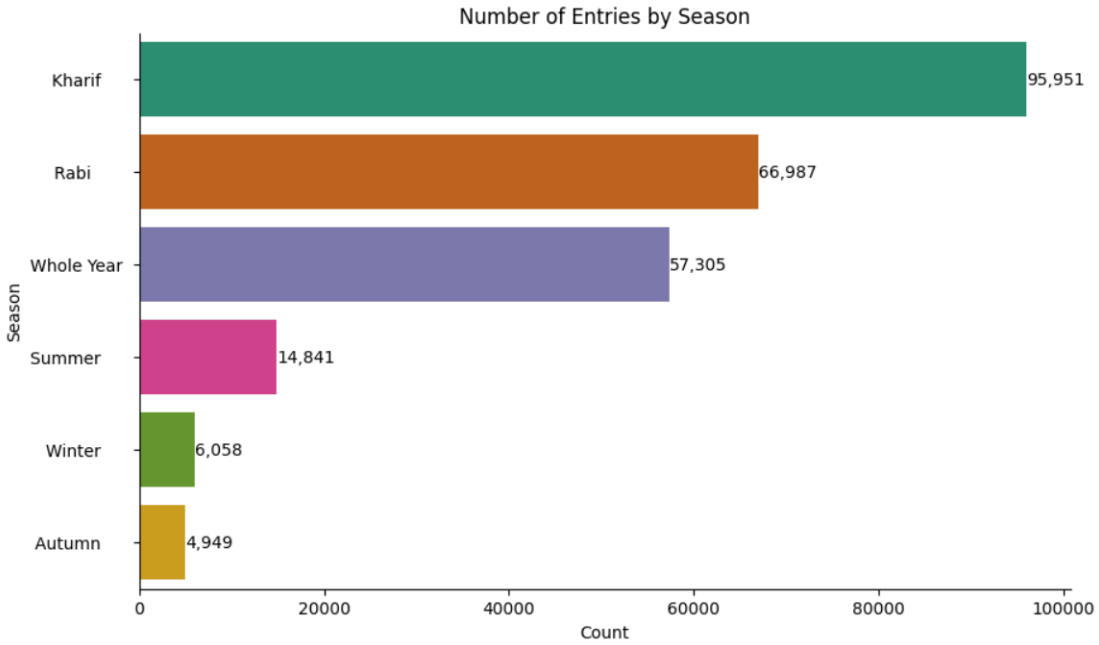
```
6
```

```
season_counts=df.Season.value_counts()
print(season_counts)
```

```
Season
Kharif         95951
Rabi           66987
Whole Year     57305
Summer         14841
Winter          6058
Autumn          4949
Name: count, dtype: int64
```

The data set is having 6 types of seasons and top 3 seasons are Kharif, Rabi and Whole Year.

```
plt.figure(figsize=(10, 6))
sns.barplot(x=season_counts.values, y=season_counts.index, palette='Dark2')
for index, value in enumerate(season_counts):
    plt.text(value, index, f'{value:,}', va='center', ha='left')
plt.xlabel('Count')
plt.ylabel('Season')
plt.title('Number of Entries by Season')
plt.gca().spines[['top', 'right']].set_visible(False)
plt.show()
```



Crop Analysis

```
df.Crop.nunique()
```

```
124
```

```
df.Crop.value_counts()
```

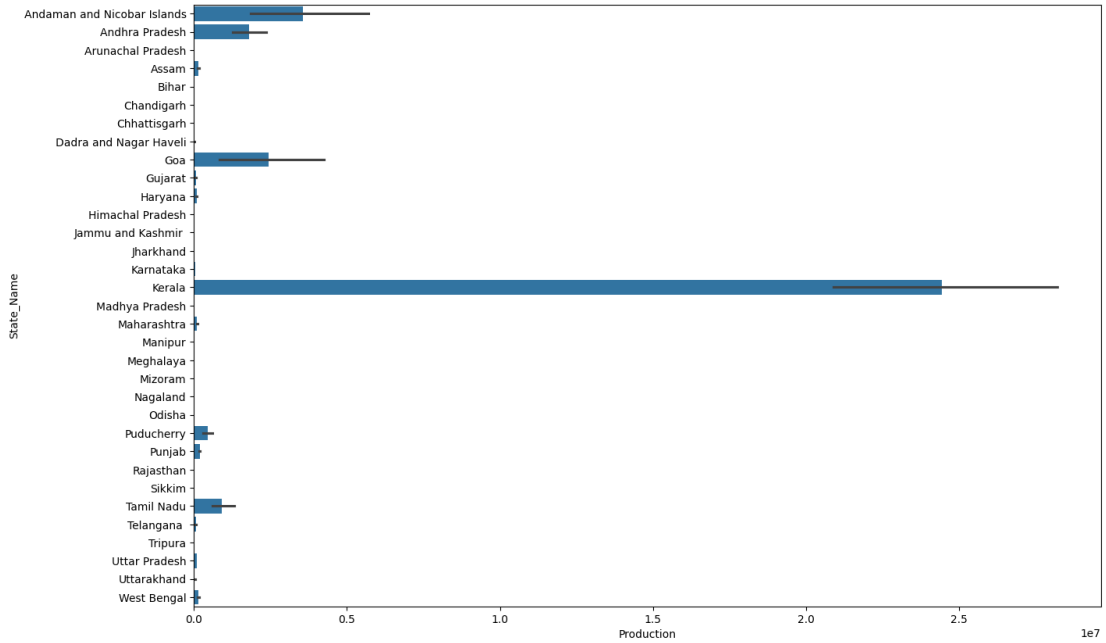
```
Crop
Rice                15104
Maize               13947
Moong(Green Gram)   10318
Urad                 9850
Sesamum              9046
                    ...
Litchi                  6
Coffee                  6
Apple                   4
Peach                   4
Other Dry Fruit         1
Name: count, Length: 124, dtype: int64
```

The data set is having 124 types of crops in which ‘Rice’, ‘Maize’ and ‘Moong(Green Gram)’ are the top 3 crops.

Bivariate Analysis

Analysis of Production of crops in state wise

```
plt.figure(figsize=(15,10))
sns.barplot(x=df["Production"], y=df["State_Name"])
```



From the above plot shows that Kerala is the top production state followed by Andaman and Nicobar Islands and Goa.

Categorizing Data Zone wise

```
North = ['Jammu and Kashmir', 'Punjab', 'Himachal Pradesh', 'Haryana', 'Uttarakhand', 'Uttar Pradesh', 'Chandigarh']
East = ['Bihar', 'Odisha', 'Jharkhand', 'West Bengal']
South = ['Andhra Pradesh', 'Karnataka', 'Kerala', 'Tamil Nadu','Telangana']
West = ['Rajasthan', 'Gujarat', 'Goa', 'Maharashtra']
Central_India = ['Madhya Pradesh', 'Chhattisgarh']
North_East=['Assam', 'Sikkim', 'Nagaland', 'Meghalaya', 'Manipur', 'Mizoram', 'Tripura', 'Arunachal Pradesh']
Union_Territories = ['Andaman and Nicobar Islands', 'Dadra and Nagar Haveli', 'Puducherry']
def zones_names (row):
    if row['State_Name'].strip() in North:
        val= 'North Zone'
    elif row['State_Name'].strip() in South:
        val= 'South Zone'
    elif row['State_Name'].strip() in East:
        val= 'East Zone'
    elif row['State_Name'].strip() in West:
        val= 'West Zone'
    elif row['State_Name'].strip() in Central_India:
        val= 'Central Zone'
    elif row['State_Name'].strip() in North_East:
        val= 'NorthEast Zone'
    elif row['State_Name'].strip() in Union_Territories:
        val= 'Union Territory'
    else:
        val = 'No Value'
    return val
```

```
df['zones'] = df.apply(zones_names, axis=1)
df.zones.value_counts()
```

```
zones
South Zone          54207
North Zone          51468
East Zone           43339
West Zone           33786
Central Zone        33652
NorthEast Zone      28297
Union Territory      1342
Name: count, dtype: int64
```

The above information shows that the South Zone, North Zone and East Zones are the Top Zones.

Categorizing Crops

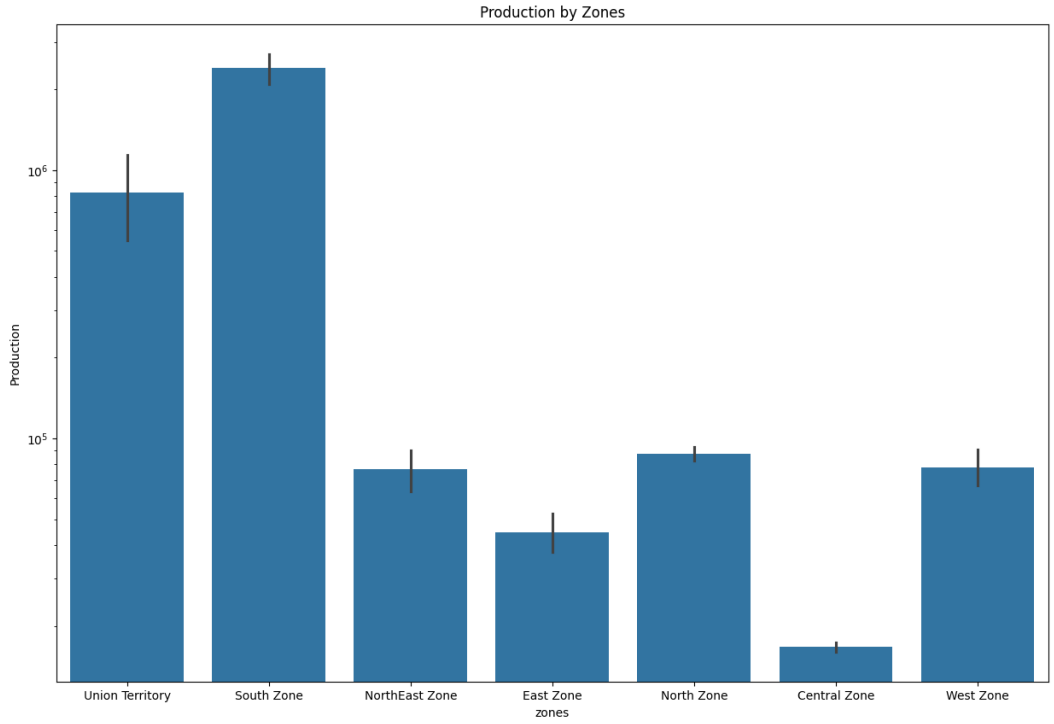
```
crop=df['Crop']
def categ_crop(crop):
    if crop in ['Rice', 'Maize', 'Wheat', 'Barley', 'Varagu', 'Other Cereals & Millets', 'Ragi, Small millets', 'Bajra', 'Jowar', 'Paddy', 'Tot
        return 'Cereal'
    elif crop in ['Moong', 'Urad', 'Arhar/Tur', 'Peas & beans', 'Masoor', 'Other Kharif pulses', 'other misc.pulses',
              'Ricebean', 'Rajmash (nagadal)','Kholar', 'Lentil', 'Samai', 'Blackgram', 'Korra', 'Cowpea (Lobia)',
              'Other Rabi pulses', 'Other Kharif pulses', 'Peas & beans (Pulses)', 'Pulses total', 'Gram']:
        return 'Pulses'
    elif crop in ['Peach', 'Apple', 'Litchi', 'Pear', 'Plums', 'Ber', 'Sapota', 'Lemon', 'PomeGranet',
              'Other Citrus Fruit', 'Water Melon', 'JackFruit', 'Grapes', 'Pineapple', 'Orange',
              'Pome FruitCitrus Fruit', 'Other FreshFruits', 'Mango', 'Papaya', 'Coconut', 'Banana']:
        return 'Fruits'
    elif crop in ['Bean', 'Lab-Lab', 'Moth', 'Guar seed', 'Soyabean', 'Horse- gram']:
        return 'Beans'
    elif crop in ['Turnip', 'Peas', 'Beet Root', 'Carrot', 'Yam', 'Ribed Guard', 'Ash Gourd', 'PumpKin', 'Redish', 'Snak Guard', 'Bottle Gourd'
         'Cauliflower', 'Beans & Mutter (Vegetable)', 'Cabbage', 'Bhindi', 'Tomato', 'Brinjal', 'Khesari', 'Sweet potato', 'Potato', 'Onion',
        return 'Vegetables'
    elif crop in ['Perilla', 'Ginger', 'Cardamom', 'Black pepper', 'Dry ginger, Garlic', 'Coriander', 'Turmeric', 'Dry chillies', 'Cond-spcs ot
        return 'spices'
    elif crop in ['other fibres', 'Kapas', 'Jute & mesta', 'Jute', 'Mesta', 'Cotton (lint)', 'Sannhamp']:
        return 'fibres'
    elif crop in ['Arcanut (Processed)', 'Atcanut (Raw)', 'Cashewnut Processed', 'Cashewnut Raw', 'Cashewnut', 'Arecanut', 'Groundnut']:
        return 'Nuts'
    elif crop in ['other oilseeds', 'Safflower', 'Niger seed', 'Castor seed', 'Linseed', 'Sunflower', 'Rapeseed &Mustard', 'Sesamum', 'Oilseeds
        return 'oilseeds'
    elif crop in ['Tobacco', 'Coffee', 'Tea', 'Sugarcane', 'Rubber']:
        return 'Commercial'
    else:
        return None
```

```
df['categ_crop'] = df['Crop'].apply(categ_crop)
df_filtered = df.dropna(subset=['categ_crop'])
df_filtered["categ_crop"].value_counts()
```

```
categ_crop
Cereal         55003
Pulses         37824
oilseeds       34454
Vegetables     23109
spices         15689
Nuts           11588
Commercial     10716
fibres          5677
Beans           5453
Fruits          5057
Name: count, dtype: int64
```

Cereals, Pulses and Oil seeds are the Top 3 crop categories.

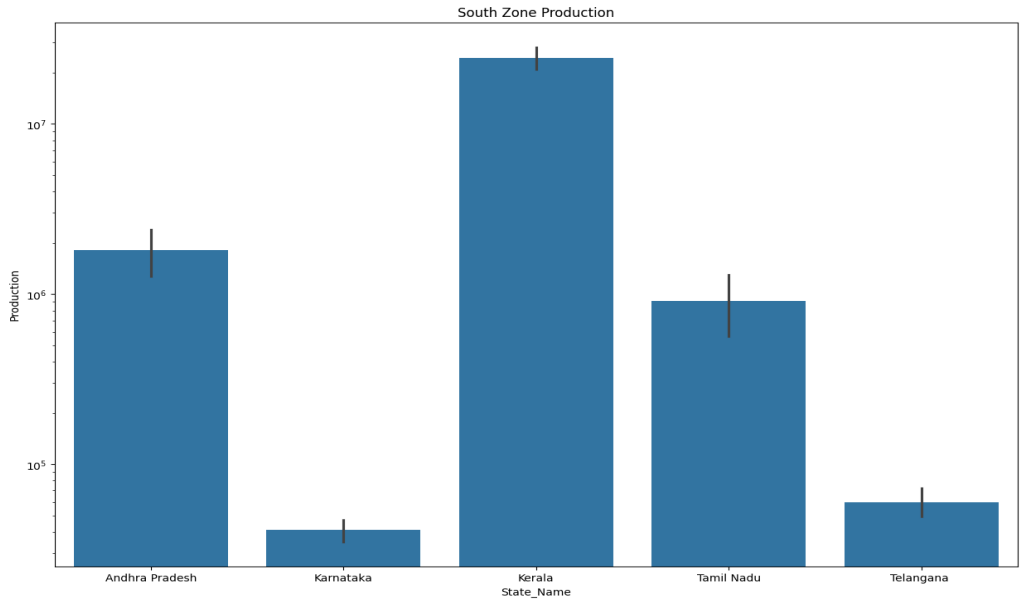
```
plt.figure(figsize=(15,10))
sns.barplot(x=df.zones, y=df.Production)
plt.yscale('log')
plt.title('Production by Zones')
plt.show()
```



From the above barplot, the South Zone is the most production zone.

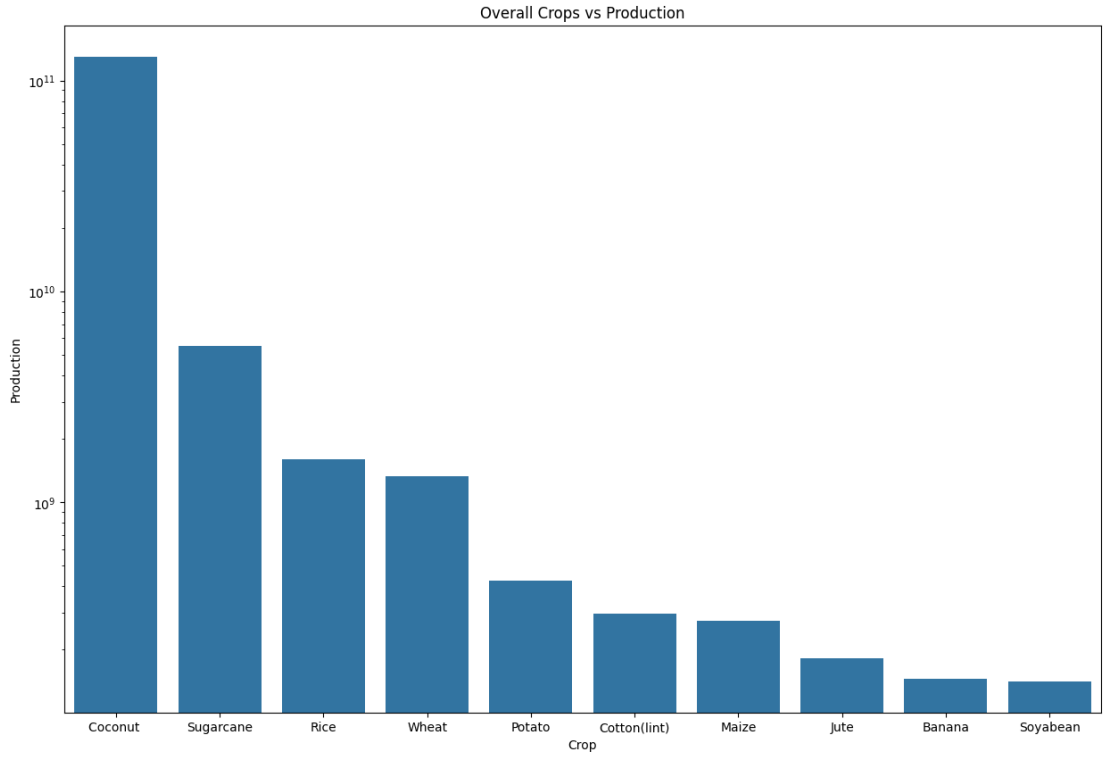
Analyzing South Zone

```
Southern_part=df[(df [ "zones"]=="South Zone")]
plt.figure(figsize=(15,10))
sns.barplot (x=Southern_part.State_Name,y=Southern_part. Production)
plt.yscale('log')
plt.title("South Zone Production")
```



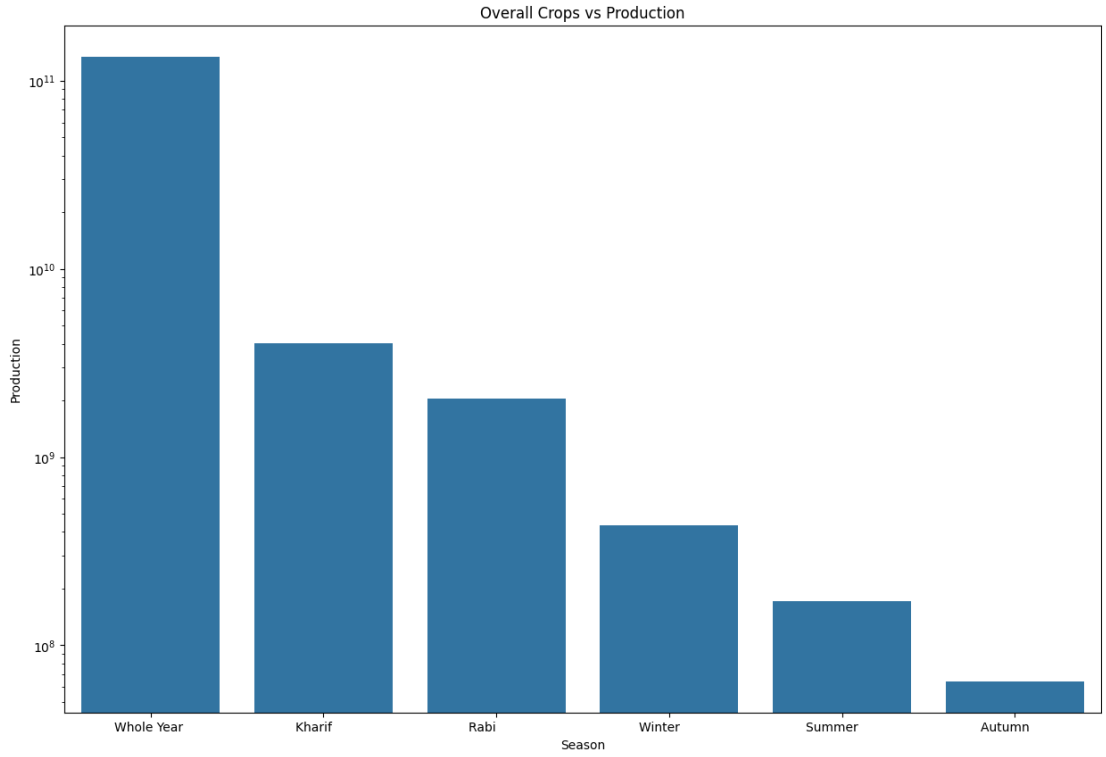
Crop Production Prediction

```
crop= df.groupby(by='Crop')['Production'].sum().reset_index().sort_values (by='Production',ascending=False).head (10)
crop
fig, ax = plt.subplots (figsize=(15,10))
sns.barplot(x=crop.Crop, y=crop. Production)
plt.yscale('log')
plt.title('Overall Crops vs Production')
```



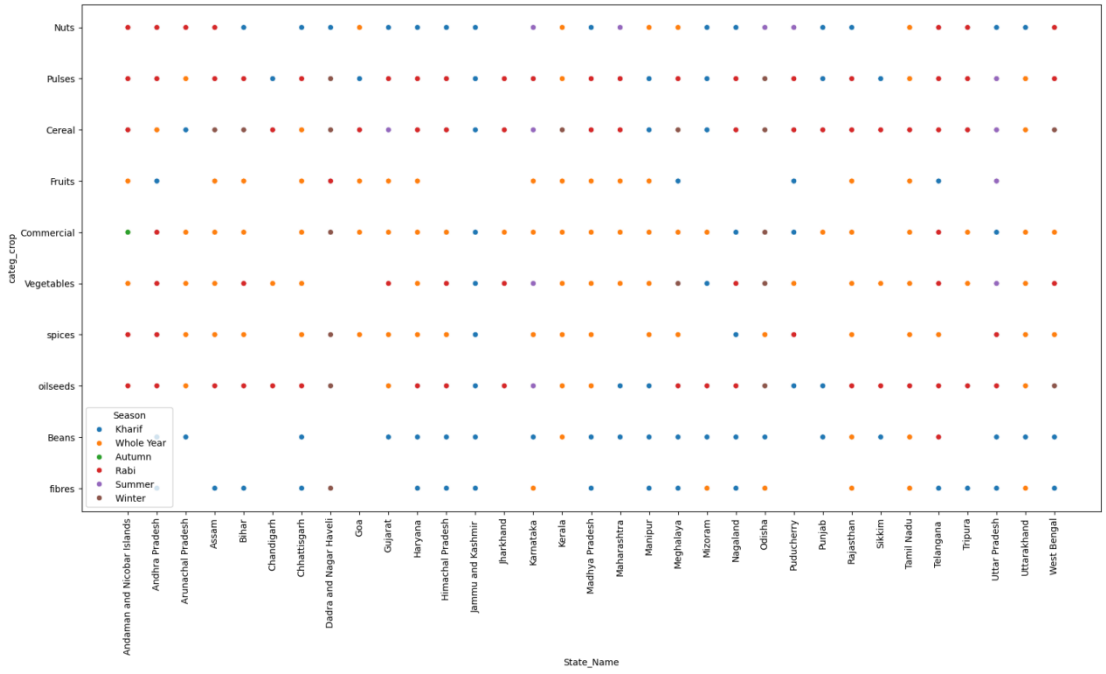
Season wise Production

```
season= df.groupby(by='Season')['Production'].sum().reset_index().sort_values (by='Production',ascending=False).head (10)
season
fig, ax = plt.subplots (figsize=(15,10))
sns.barplot (x=season. Season, y=season. Production)
plt.yscale('log')
plt.title( 'Overall Crops vs Production')
```



Multivariate Analysis

```
plt.figure(figsize=(20,10))
sns.scatterplot(data=df, x=df.State_Name, y="categ_crop", hue="Season")
plt.xticks(rotation=90)
plt.show()
```



Key Findings:

1) The top three agriculturally wealthy states are "Uttar Pradesh", "Madhya Pradesh" and "Maharashtra".

2) "Bijapur", "Tumkur" and "Belgaum" made the largest contributions, followed by other districts.

3) 2003, 2002, and 2008 are the top three years for crop production.

4) Of the 6 different season types in the sample, the top 3 seasons are Kharif, Rabi, and the Whole Year.

5) There are 124 different crop types in the data-set; the three most produced crops are Rice, Maize and Moong.

6) Kerala is the state with the highest production, followed by Andaman and Nicobar Islands and Goa.

7) Southern India, Northern India, and Eastern India are the main three producing zones.

8) The three crop categories with the highest production are cereals, pulses and Oil seeds

9) Kerala, the top-producing state, produces most of its seasonal crops throughout the year.

10) The majority of Kharif, Rabi and Summer crops are produced in the top-producing state of Uttar Pradesh.

----------END----------