In [1]:
```
# @title Importing Required Libraries
#In this task we are performing EDA,these are the required libraries to per
import pandas as pd
import numpy as np
import seaborn as sns
```

In [2]:
```
# @title Reading The Indian Agriculture Dataset.
# Reading the csv file
Agriculture_Indian = pd.read_csv("ICRISAT-District Level Data.csv")
# Here we are displaying the top 5 rows.
Agriculture_Indian.head(5)
```

Out[2]:

	Dist Code	Year	State Code	State Name	Dist Name	RICE AREA (1000 ha)	RICE PRODUCTION (1000 tons)	RICE YIELD (Kg per ha)	WHEAT AREA (1000 ha)	WH PRODUCT (1000 t
0	1	1966	14	Chhattisgarh	Durg	548.0	185.0	337.59	44.0	
1	1	1967	14	Chhattisgarh	Durg	547.0	409.0	747.71	50.0	
2	1	1968	14	Chhattisgarh	Durg	556.3	468.0	841.27	53.7	
3	1	1969	14	Chhattisgarh	Durg	563.4	400.8	711.40	49.4	
4	1	1970	14	Chhattisgarh	Durg	571.6	473.6	828.55	44.2	

5 rows × 80 columns

In [3]:
```
# Here we are displaying the last 5 rows.
Agriculture_Indian.tail(5)
```

Out[3]:

	Dist Code	Year	State Code	State Name	Dist Name	RICE AREA (1000 ha)	RICE PRODUCTION (1000 tons)	RICE YIELD (Kg per ha)	WHEAT AREA (1000 ha)	
16141	917	2013	15	Jharkhand	Singhbhum	267.06	579.70	2170.67	1.53	
16142	917	2014	15	Jharkhand	Singhbhum	256.33	586.63	2288.57	5.36	
16143	917	2015	15	Jharkhand	Singhbhum	263.21	264.71	1005.70	1.99	
16144	917	2016	15	Jharkhand	Singhbhum	224.05	319.01	1423.84	0.38	
16145	917	2017	15	Jharkhand	Singhbhum	386.91	669.97	1731.62	0.00	

5 rows × 80 columns

In [4]:
```
# @title Checking The DataTypes
# Using the dtypes command we are checking the types of data.
Agriculture_Indian.dtypes
```

Out[4]:
```
Dist Code                                int64
Year                                     int64
State Code                               int64
State Name                              object
Dist Name                               object
                                         ...
VEGETABLES AREA (1000 ha)              float64
FRUITS AND VEGETABLES AREA (1000 ha)   float64
POTATOES AREA (1000 ha)                float64
ONION AREA (1000 ha)                   float64
FODDER AREA (1000 ha)                  float64
Length: 80, dtype: object
```

In [5]:
```
# @title Checking The Missing Values
# Using the isnull() command i am checking the missing values in the dataset
Agriculture_Indian.isnull().sum()
```

Out[5]:
```
Dist Code                                0
Year                                     0
State Code                               0
State Name                               0
Dist Name                                0
                                       ..
VEGETABLES AREA (1000 ha)                0
FRUITS AND VEGETABLES AREA (1000 ha)     0
POTATOES AREA (1000 ha)                  0
ONION AREA (1000 ha)                     0
FODDER AREA (1000 ha)                    0
Length: 80, dtype: int64
```

In [6]:
```
# @title Printing the Columns For The Agriculture Data
# Printing the columns of Data Agriculture
print(Agriculture_Indian.columns)
```

```
Index(['Dist Code', 'Year', 'State Code', 'State Name', 'Dist Name',
       'RICE AREA (1000 ha)', 'RICE PRODUCTION (1000 tons)',
       'RICE YIELD (Kg per ha)', 'WHEAT AREA (1000 ha)',
       'WHEAT PRODUCTION (1000 tons)', 'WHEAT YIELD (Kg per ha)',
       'KHARIF SORGHUM AREA (1000 ha)',
       'KHARIF SORGHUM PRODUCTION (1000 tons)',
       'KHARIF SORGHUM YIELD (Kg per ha)', 'RABI SORGHUM AREA (1000 ha)',
       'RABI SORGHUM PRODUCTION (1000 tons)', 'RABI SORGHUM YIELD (Kg per h
a)',
       'SORGHUM AREA (1000 ha)', 'SORGHUM PRODUCTION (1000 tons)',
       'SORGHUM YIELD (Kg per ha)', 'PEARL MILLET AREA (1000 ha)',
       'PEARL MILLET PRODUCTION (1000 tons)', 'PEARL MILLET YIELD (Kg per h
a)',
       'MAIZE AREA (1000 ha)', 'MAIZE PRODUCTION (1000 tons)',
       'MAIZE YIELD (Kg per ha)', 'FINGER MILLET AREA (1000 ha)',
       'FINGER MILLET PRODUCTION (1000 tons)',
       'FINGER MILLET YIELD (Kg per ha)', 'BARLEY AREA (1000 ha)',
       'BARLEY PRODUCTION (1000 tons)', 'BARLEY YIELD (Kg per ha)',
       'CHICKPEA AREA (1000 ha)', 'CHICKPEA PRODUCTION (1000 tons)',
       'CHICKPEA YIELD (Kg per ha)', 'PIGEONPEA AREA (1000 ha)',
       'PIGEONPEA PRODUCTION (1000 tons)', 'PIGEONPEA YIELD (Kg per ha)',
       'MINOR PULSES AREA (1000 ha)', 'MINOR PULSES PRODUCTION (1000 ton
s)',
       'MINOR PULSES YIELD (Kg per ha)', 'GROUNDNUT AREA (1000 ha)',
       'GROUNDNUT PRODUCTION (1000 tons)', 'GROUNDNUT YIELD (Kg per ha)',
       'SESAMUM AREA (1000 ha)', 'SESAMUM PRODUCTION (1000 tons)',
       'SESAMUM YIELD (Kg per ha)', 'RAPESEED AND MUSTARD AREA (1000 ha)',
       'RAPESEED AND MUSTARD PRODUCTION (1000 tons)',
       'RAPESEED AND MUSTARD YIELD (Kg per ha)', 'SAFFLOWER AREA (1000 h
a)',
       'SAFFLOWER PRODUCTION (1000 tons)', 'SAFFLOWER YIELD (Kg per ha)',
       'CASTOR AREA (1000 ha)', 'CASTOR PRODUCTION (1000 tons)',
       'CASTOR YIELD (Kg per ha)', 'LINSEED AREA (1000 ha)',
       'LINSEED PRODUCTION (1000 tons)', 'LINSEED YIELD (Kg per ha)',
       'SUNFLOWER AREA (1000 ha)', 'SUNFLOWER PRODUCTION (1000 tons)',
       'SUNFLOWER YIELD (Kg per ha)', 'SOYABEAN AREA (1000 ha)',
       'SOYABEAN PRODUCTION (1000 tons)', 'SOYABEAN YIELD (Kg per ha)',
       'OILSEEDS AREA (1000 ha)', 'OILSEEDS PRODUCTION (1000 tons)',
       'OILSEEDS YIELD (Kg per ha)', 'SUGARCANE AREA (1000 ha)',
       'SUGARCANE PRODUCTION (1000 tons)', 'SUGARCANE YIELD (Kg per ha)',
       'COTTON AREA (1000 ha)', 'COTTON PRODUCTION (1000 tons)',
       'COTTON YIELD (Kg per ha)', 'FRUITS AREA (1000 ha)',
       'VEGETABLES AREA (1000 ha)', 'FRUITS AND VEGETABLES AREA (1000 ha)',
       'POTATOES AREA (1000 ha)', 'ONION AREA (1000 ha)',
       'FODDER AREA (1000 ha)'],
      dtype='object')
```

In [7]:
```
# @title Dataset Statistics
# Shape of our dataset
Agriculture_Indian.shape

# Info our dataset
Agriculture_Indian.info()

# Describe our dataset
Agriculture_Indian.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16146 entries, 0 to 16145
Data columns (total 80 columns):
 #   Column                                          Non-Null Count  Dtype  
---  ------                                          --------------  -----  
 0   Dist Code                                       16146 non-null  int64  
 1   Year                                            16146 non-null  int64  
 2   State Code                                      16146 non-null  int64  
 3   State Name                                      16146 non-null  object 
 4   Dist Name                                       16146 non-null  object 
 5   RICE AREA (1000 ha)                             16146 non-null  float64
 6   RICE PRODUCTION (1000 tons)                     16146 non-null  float64
 7   RICE YIELD (Kg per ha)                          16146 non-null  float64
 8   WHEAT AREA (1000 ha)                            16146 non-null  float64
 9   WHEAT PRODUCTION (1000 tons)                    16146 non-null  float64
 10  WHEAT YIELD (Kg per ha)                         16146 non-null  float64
 11  KHARIF SORGHUM AREA (1000 ha)                   16146 non-null  float64
 12  KHARIF SORGHUM PRODUCTION (1000 tons)           16146 non-null  float64
 13  KHARIF SORGHUM YIELD (Kg per ha)                16146 non-null  float64
 14  RABI SORGHUM AREA (1000 ha)                     16146 non-null  float64
 15  RABI SORGHUM PRODUCTION (1000 tons)             16146 non-null  float64
 16  RABI SORGHUM YIELD (Kg per ha)                  16146 non-null  float64
 17  SORGHUM AREA (1000 ha)                          16146 non-null  float64
 18  SORGHUM PRODUCTION (1000 tons)                  16146 non-null  float64
 19  SORGHUM YIELD (Kg per ha)                       16146 non-null  float64
 20  PEARL MILLET AREA (1000 ha)                     16146 non-null  float64
 21  PEARL MILLET PRODUCTION (1000 tons)             16146 non-null  float64
 22  PEARL MILLET YIELD (Kg per ha)                  16146 non-null  float64
 23  MAIZE AREA (1000 ha)                            16146 non-null  float64
 24  MAIZE PRODUCTION (1000 tons)                    16146 non-null  float64
 25  MAIZE YIELD (Kg per ha)                         16146 non-null  float64
 26  FINGER MILLET AREA (1000 ha)                    16146 non-null  float64
 27  FINGER MILLET PRODUCTION (1000 tons)            16146 non-null  float64
 28  FINGER MILLET YIELD (Kg per ha)                 16146 non-null  float64
 29  BARLEY AREA (1000 ha)                           16146 non-null  float64
 30  BARLEY PRODUCTION (1000 tons)                   16146 non-null  float64
 31  BARLEY YIELD (Kg per ha)                        16146 non-null  float64
 32  CHICKPEA AREA (1000 ha)                         16146 non-null  float64
 33  CHICKPEA PRODUCTION (1000 tons)                 16146 non-null  float64
 34  CHICKPEA YIELD (Kg per ha)                      16146 non-null  float64
 35  PIGEONPEA AREA (1000 ha)                        16146 non-null  float64
 36  PIGEONPEA PRODUCTION (1000 tons)                16146 non-null  float64
 37  PIGEONPEA YIELD (Kg per ha)                     16146 non-null  float64
 38  MINOR PULSES AREA (1000 ha)                     16146 non-null  float64
 39  MINOR PULSES PRODUCTION (1000 tons)             16146 non-null  float64
 40  MINOR PULSES YIELD (Kg per ha)                  16146 non-null  float64
 41  GROUNDNUT AREA (1000 ha)                        16146 non-null  float64
 42  GROUNDNUT PRODUCTION (1000 tons)                16146 non-null  float64
 43  GROUNDNUT YIELD (Kg per ha)                     16146 non-null  float64
 44  SESAMUM AREA (1000 ha)                          16146 non-null  float64
 45  SESAMUM PRODUCTION (1000 tons)                  16146 non-null  float64
 46  SESAMUM YIELD (Kg per ha)                       16146 non-null  float64
 47  RAPESEED AND MUSTARD AREA (1000 ha)             16146 non-null  float64
 48  RAPESEED AND MUSTARD PRODUCTION (1000 tons)     16146 non-null  float64
 49  RAPESEED AND MUSTARD YIELD (Kg per ha)          16146 non-null  float64
 50  SAFFLOWER AREA (1000 ha)                        16146 non-null  float64
 51  SAFFLOWER PRODUCTION (1000 tons)                16146 non-null  float64
 52  SAFFLOWER YIELD (Kg per ha)                     16146 non-null  float64
 53  CASTOR AREA (1000 ha)                           16146 non-null  float64
 54  CASTOR PRODUCTION (1000 tons)                   16146 non-null  float64
 55  CASTOR YIELD (Kg per ha)                        16146 non-null  float64
 56  LINSEED AREA (1000 ha)                          16146 non-null  float64
 57  LINSEED PRODUCTION (1000 tons)                  16146 non-null  float64
 58  LINSEED YIELD (Kg per ha)                       16146 non-null  float64
```

```
 59  SUNFLOWER AREA (1000 ha)                    16146 non-null  float64
 60  SUNFLOWER PRODUCTION (1000 tons)            16146 non-null  float64
 61  SUNFLOWER YIELD (Kg per ha)                 16146 non-null  float64
 62  SOYABEAN AREA (1000 ha)                     16146 non-null  float64
 63  SOYABEAN PRODUCTION (1000 tons)             16146 non-null  float64
 64  SOYABEAN YIELD (Kg per ha)                  16146 non-null  float64
 65  OILSEEDS AREA (1000 ha)                     16146 non-null  float64
 66  OILSEEDS PRODUCTION (1000 tons)             16146 non-null  float64
 67  OILSEEDS YIELD (Kg per ha)                  16146 non-null  float64
 68  SUGARCANE AREA (1000 ha)                    16146 non-null  float64
 69  SUGARCANE PRODUCTION (1000 tons)            16146 non-null  float64
 70  SUGARCANE YIELD (Kg per ha)                 16146 non-null  float64
 71  COTTON AREA (1000 ha)                       16146 non-null  float64
 72  COTTON PRODUCTION (1000 tons)               16146 non-null  float64
 73  COTTON YIELD (Kg per ha)                    16146 non-null  float64
 74  FRUITS AREA (1000 ha)                       16146 non-null  float64
 75  VEGETABLES AREA (1000 ha)                   16146 non-null  float64
 76  FRUITS AND VEGETABLES AREA (1000 ha)        16146 non-null  float64
 77  POTATOES AREA (1000 ha)                     16146 non-null  float64
 78  ONION AREA (1000 ha)                        16146 non-null  float64
 79  FODDER AREA (1000 ha)                       16146 non-null  float64
dtypes: float64(75), int64(3), object(2)
memory usage: 9.9+ MB
```

Out[7]:

	Dist Code	Year	State Code	RICE AREA (1000 ha)	RICE PRODUCTION (1000 tons)	RICE YIEL (Kg per ha
count	16146.000000	16146.000000	16146.000000	16146.000000	16146.000000	16146.00000
mean	269.769231	1991.496841	9.568562	128.593192	224.889565	1486.92478
std	278.309125	15.011185	4.988538	160.078825	326.629828	956.18528
min	1.000000	1966.000000	1.000000	-1.000000	-1.000000	-1.00000
25%	78.000000	1978.000000	6.000000	10.400000	9.460000	800.00000
50%	156.000000	1991.000000	10.000000	66.800000	95.840000	1333.21000
75%	241.000000	2005.000000	12.000000	191.390000	315.715000	2113.51750
max	917.000000	2017.000000	20.000000	1154.230000	3215.010000	5653.83000

8 rows × 78 columns

In [10]:

```
# @title Reading The Global Country Information Dataset.
# Reading the csv file
Country_Global = pd.read_csv("Global_Country.csv")
# Here we are displaying the top 5 rows.
Country_Global.head(5)
```

Out[10]:

	Rank	State	Capital	Population	% of Total Population	Males	Females	Sex Ratio	Li
0	1	Uttar Pradesh	Lucknow	199,812,341	16.50	104,480,510	95,331,831	912	
1	2	Maharashtra	Mumbai	112,374,333	9.28	58,243,056	54,131,277	929	
2	3	Bihar	Patna	104,099,452	8.60	54,278,157	49,821,295	918	
3	4	West Bengal	Kolkata	91,276,115	7.54	46,809,027	44,467,088	950	
4	5	Andhra Pradesh	Hyderabad	84,580,777	6.99	42,442,146	42,138,631	993	

In [11]:
```
# Here we are displaying the last 5 rows.
Country_Global.tail(5)
```

Out[11]:

	Rank	State	Capital	Population	% of Total Population	Males	Females	Sex Ratio	Literacy Rate (%)
30	31	Sikkim	Gangtok	610,577	0.05	323,070	287,507	890	81.42
31	32	Andaman and Nicobar Islands	Port Blair	380,581	0.03	202,871	177,710	876	86.63
32	33	Dadra and Nagar Haveli	Silvassa	343,709	0.03	193,760	149,949	774	76.24
33	34	Daman and Diu	Daman	243,247	0.02	150,301	92,946	618	87.10
34	35	Lakshadweep	Kavaratti	64,473	0.01	33,123	31,350	946	91.85

In [12]:
```
# @title Checking The DataTypes
# Using the dtypes command we are checking the types of data.
Country_Global.dtypes
```

Out[12]:
```
Rank                        int64
State                      object
Capital                    object
Population                 object
% of Total Population     float64
Males                      object
Females                    object
Sex Ratio                  object
Literacy Rate (%)         float64
Rural Population           object
Urban Population           object
Area (km*km)               object
Density (1/km*km)          object
Decadal Growth (%)         object
dtype: object
```

In [13]:
```
# @title Checking The Missing Values
# Using the isnull() command i am checking the missing values in the dataset
Country_Global.isnull().sum()
```

Out[13]:
```
Rank                     0
State                    0
Capital                  1
Population               0
% of Total Population    0
Males                    0
Females                  0
Sex Ratio                0
Literacy Rate (%)        0
Rural Population         0
Urban Population         0
Area (km*km)             0
Density (1/km*km)        0
Decadal Growth (%)       0
dtype: int64
```

In [14]:
```
# @title Printing the Columns For The Agriculture Data
# Printing the columns of Data Agriculture
print(Country_Global.columns)
```

```
Index(['Rank', 'State', 'Capital', 'Population', '% of Total Population',
       'Males', 'Females', 'Sex Ratio', 'Literacy Rate (%)',
       'Rural Population', 'Urban Population', 'Area (km*km)',
       'Density (1/km*km)', 'Decadal Growth (%)'],
      dtype='object')
```

In [15]:
```
# @title Dataset Statistics
# Shape of our dataset
Country_Global.shape

# Info our dataset
Country_Global.info()

# Describe our dataset
Country_Global.describe()
```

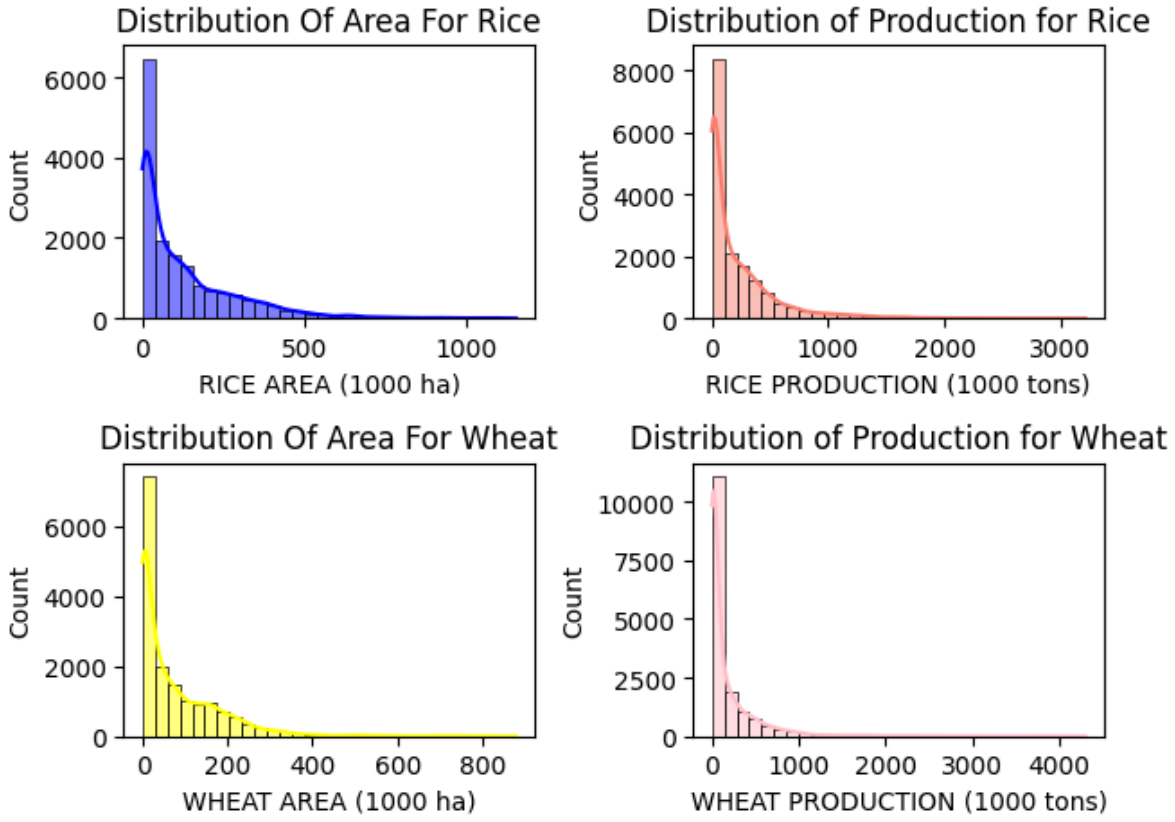
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 35 entries, 0 to 34
Data columns (total 14 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Rank                   35 non-null     int64
 1   State                  35 non-null     object
 2   Capital                34 non-null     object
 3   Population             35 non-null     object
 4   % of Total Population  35 non-null     float64
 5   Males                  35 non-null     object
 6   Females                35 non-null     object
 7   Sex Ratio              35 non-null     object
 8   Literacy Rate (%)      35 non-null     float64
 9   Rural Population       35 non-null     object
 10  Urban Population       35 non-null     object
 11  Area (km*km)           35 non-null     object
 12  Density (1/km*km)      35 non-null     object
 13  Decadal Growth (%)     35 non-null     object
dtypes: float64(2), int64(1), object(11)
memory usage: 4.0+ KB
```

Out[15]:

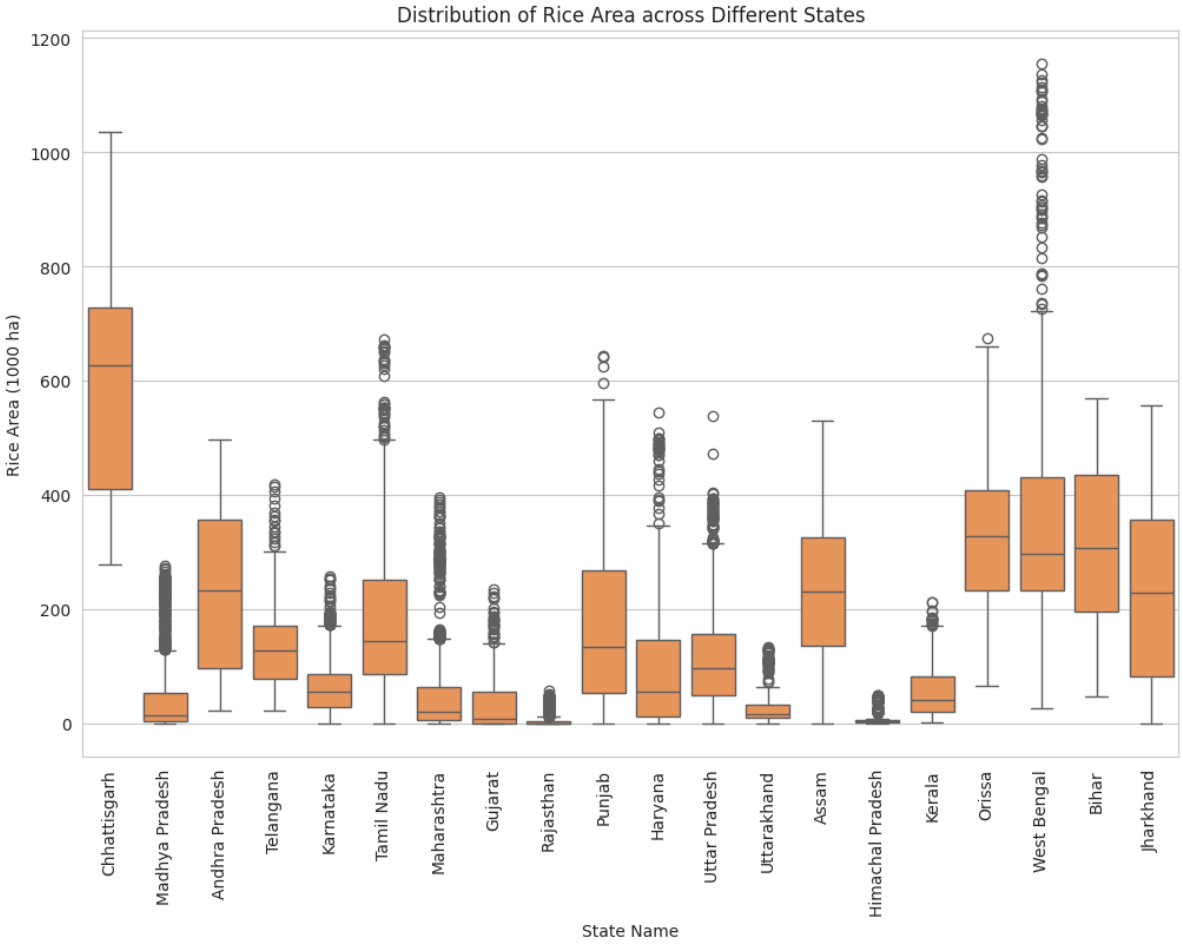
	Rank	% of Total Population	Literacy Rate (%)
count	35.000000	35.000000	35.000000
mean	18.000000	2.856857	77.940286
std	10.246951	3.671943	8.598837
min	1.000000	0.010000	61.800000
25%	9.500000	0.115000	71.235000
50%	18.000000	1.390000	78.030000
75%	26.500000	5.020000	85.950000
max	35.000000	16.500000	94.000000

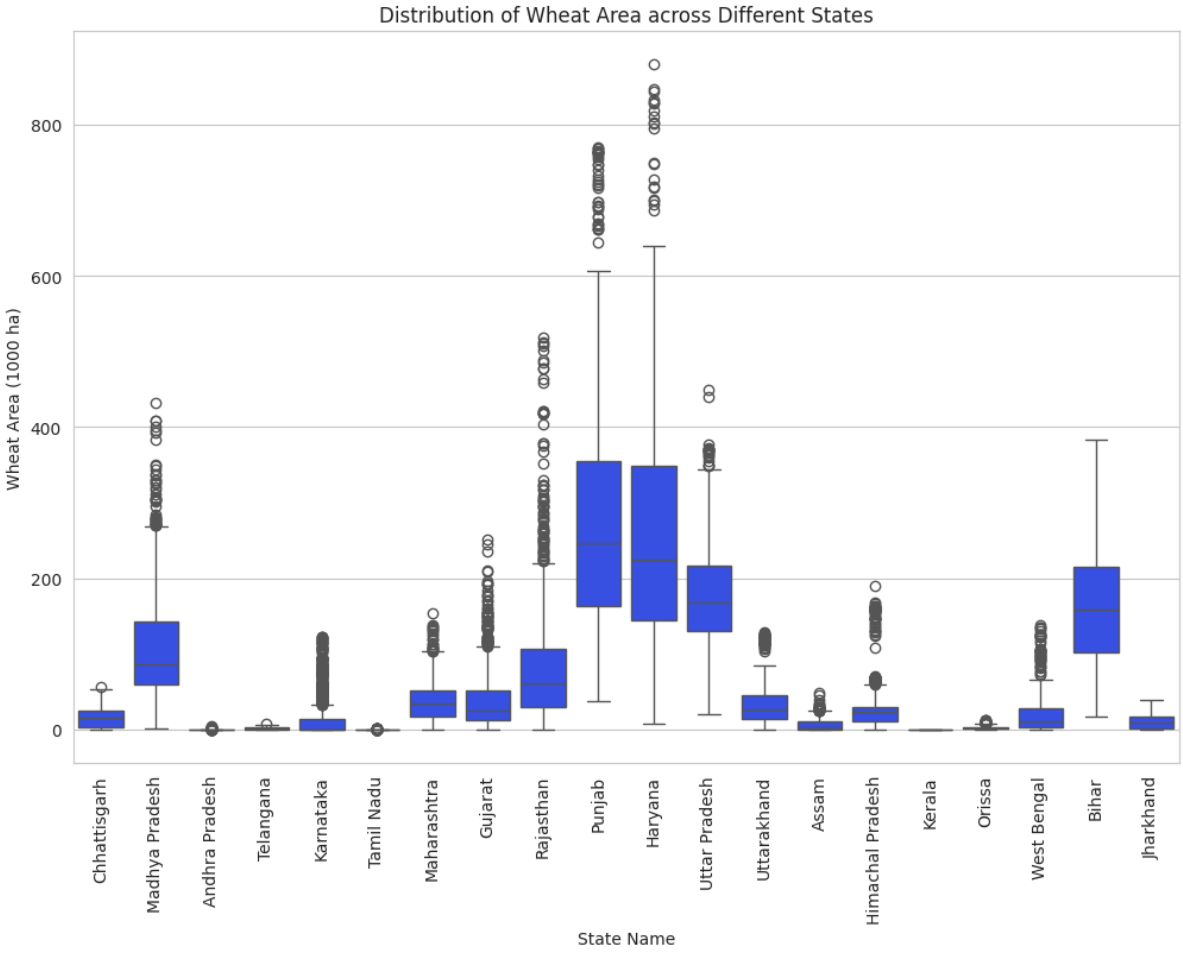
In [16]:
```
# @title Plotting The Histogram For the Rice Area,Production And Wheat Area,
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

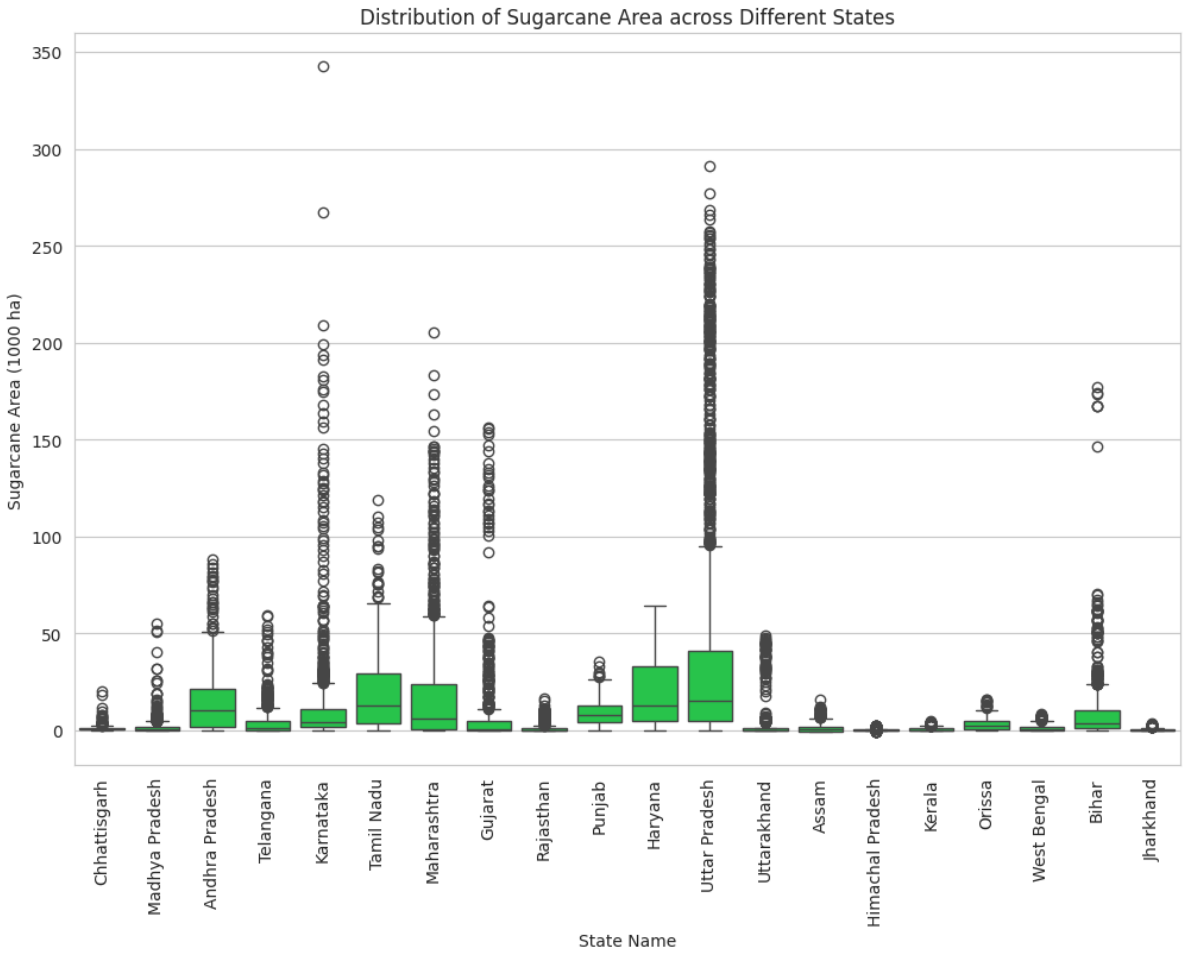
In [17]:
```
# Here we are plotting rice area distribution utlizing histogram
plt.subplot(2, 2, 1)
sns.histplot(Agriculture_Indian["RICE AREA (1000 ha)"], bins=30, kde=True, c
plt.title("Distribution Of Area For Rice")

# Here we are plotting rice production distribution utlizing histogram
plt.subplot(2, 2, 2)
sns.histplot(Agriculture_Indian["RICE PRODUCTION (1000 tons)"], bins=30, kde
plt.title("Distribution of Production for Rice")

# Here we are plotting wheat area distribution utlizing histogram
plt.subplot(2, 2, 3)
sns.histplot(Agriculture_Indian["WHEAT AREA (1000 ha)"], bins=30, kde=True,
plt.title("Distribution Of Area For Wheat")

# Here we are plotting wheat production distribution utlizing histogram
plt.subplot(2, 2, 4)
sns.histplot(Agriculture_Indian["WHEAT PRODUCTION (1000 tons)"], bins=30, kd
plt.title("Distribution of Production for Wheat")

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



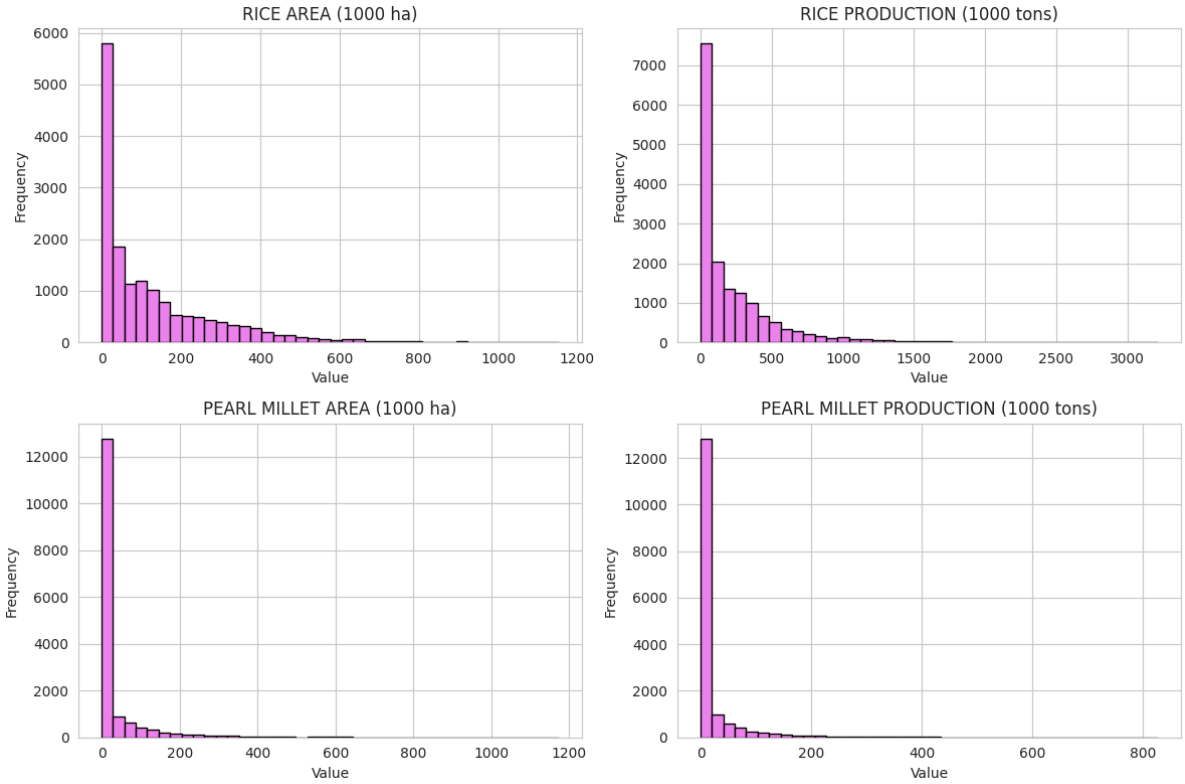
In [18]:
```
# @title Plotting The BoxPlot For the Rice,Sugarcane,Wheat Area across diffe
# Here we are setting the colors
colors = ["#FF9247", "#1E40FF", "#12DD42"]
# Here we are setting the seaborn as style
sns.set_style("whitegrid")


# Plot the box plots with for the rice area and different states
# Here we are setting the figure size
plt.figure(figsize=(12, 8))
sns.boxplot(x='State Name', y='RICE AREA (1000 ha)', data=Agriculture_Indiar
plt.title('Distribution of Rice Area across Different States')
plt.xticks(rotation=90)
plt.ylabel('Rice Area (1000 ha)')
plt.xlabel('State Name')
plt.show()
# Plot the box plots with for the rice area and different states
# Here we are setting the figure size
plt.figure(figsize=(12, 8))
sns.boxplot(x='State Name', y='WHEAT AREA (1000 ha)', data=Agriculture_Indi
plt.title('Distribution of Wheat Area across Different States')
plt.xticks(rotation=90)
plt.ylabel('Wheat Area (1000 ha)')
plt.xlabel('State Name')
plt.show()
# Plot the box plots with for the sugarcane area and different states
# Here we are setting the figure size
plt.figure(figsize=(12, 8))
sns.boxplot(x='State Name', y='SUGARCANE AREA (1000 ha)', data=Agriculture_
plt.title('Distribution of Sugarcane Area across Different States')
plt.xticks(rotation=90)
plt.ylabel('Sugarcane Area (1000 ha)')
plt.xlabel('State Name')
plt.show()
```


Distribution of Rice Area across Different States
Rice Area (1000 ha)
0
200
400
600
800
1000
1200
Chhattisgarh
Madhya Pradesh
Andhra Pradesh
Telangana
Karnataka
Tamil Nadu
Maharashtra
Gujarat
Rajasthan
Punjab
Haryana
Uttar Pradesh
Uttarakhand
Assam
Himachal Pradesh
Kerala
Orissa
West Bengal
Bihar
Jharkhand
State Name

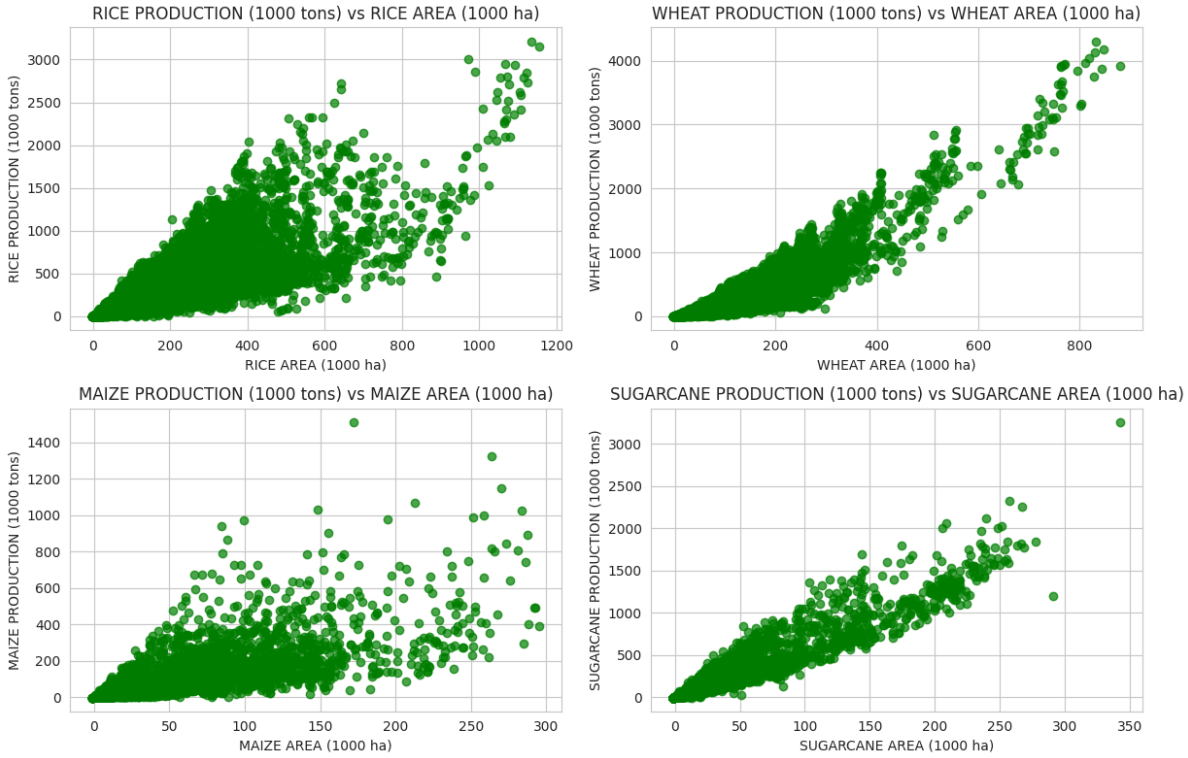

Distribution of Wheat Area across Different States
Wheat Area (1000 ha)
0
200
400
600
800
Chhattisgarh
Madhya Pradesh
Andhra Pradesh
Telangana
Karnataka
Tamil Nadu
Maharashtra
Gujarat
Rajasthan
Punjab
Haryana
Uttar Pradesh
Uttarakhand
Assam
Himachal Pradesh
Kerala
Orissa
West Bengal
Bihar
Jharkhand
State Name



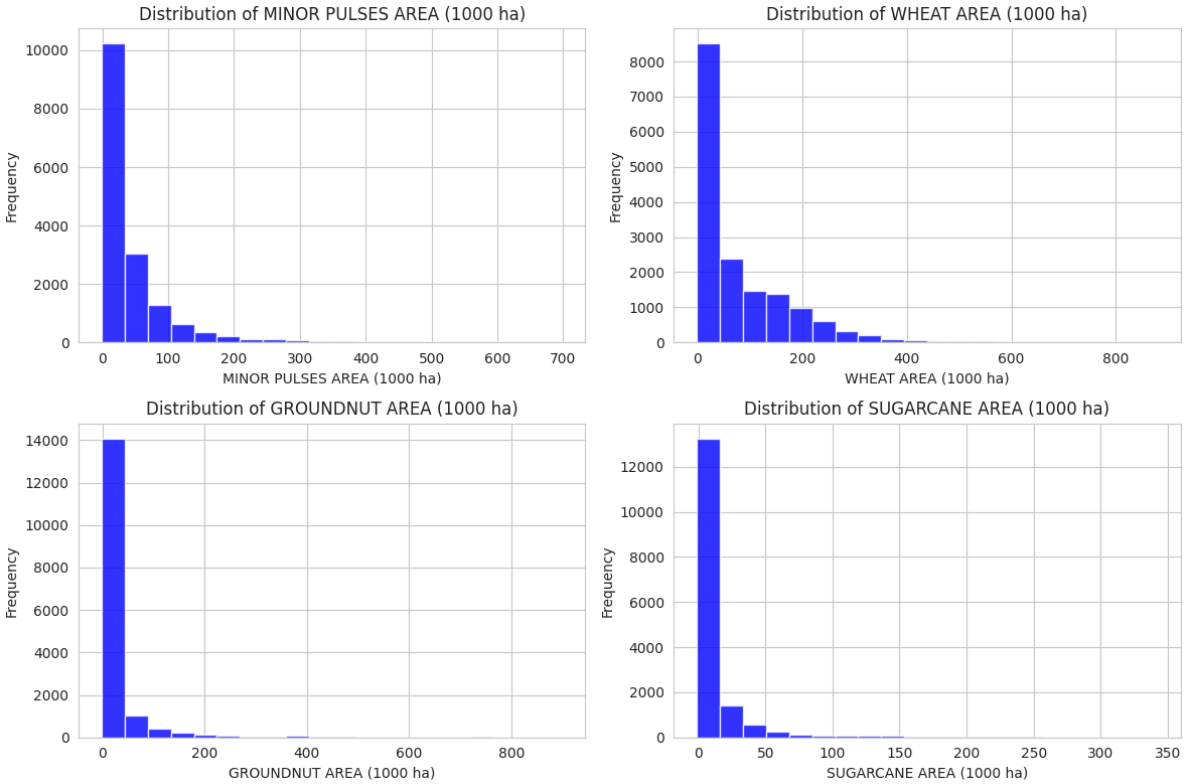
In [19]:
```
# @title Plotting The HistPlot For The RICE,PEARL MILLET Area,Production fo
# Here We are defining the parameters to plot
Agriculture_Indian_parameters = ['RICE AREA (1000 ha)', 'RICE PRODUCTION (1
                                 'PEARL MILLET AREA (1000 ha)', 'PEARL MILLE

# Here we are creating subplots for every parameter
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(12, 8))

# Here we are flatten the axes array to perform easy iteration
axes = axes.flatten()

# Here we are plotting the histograms for each parameter
for i, Parameters_Agriculture in enumerate(Agriculture_Indian_parameters):
    ax = axes[i]
    ax.hist(Agriculture_Indian[Parameters_Agriculture], bins=40, color='vio
    ax.set_title(Parameters_Agriculture)
    ax.set_xlabel('Value')
    ax.set_ylabel('Frequency')

# Adjust layout
plt.tight_layout()
plt.show()
```



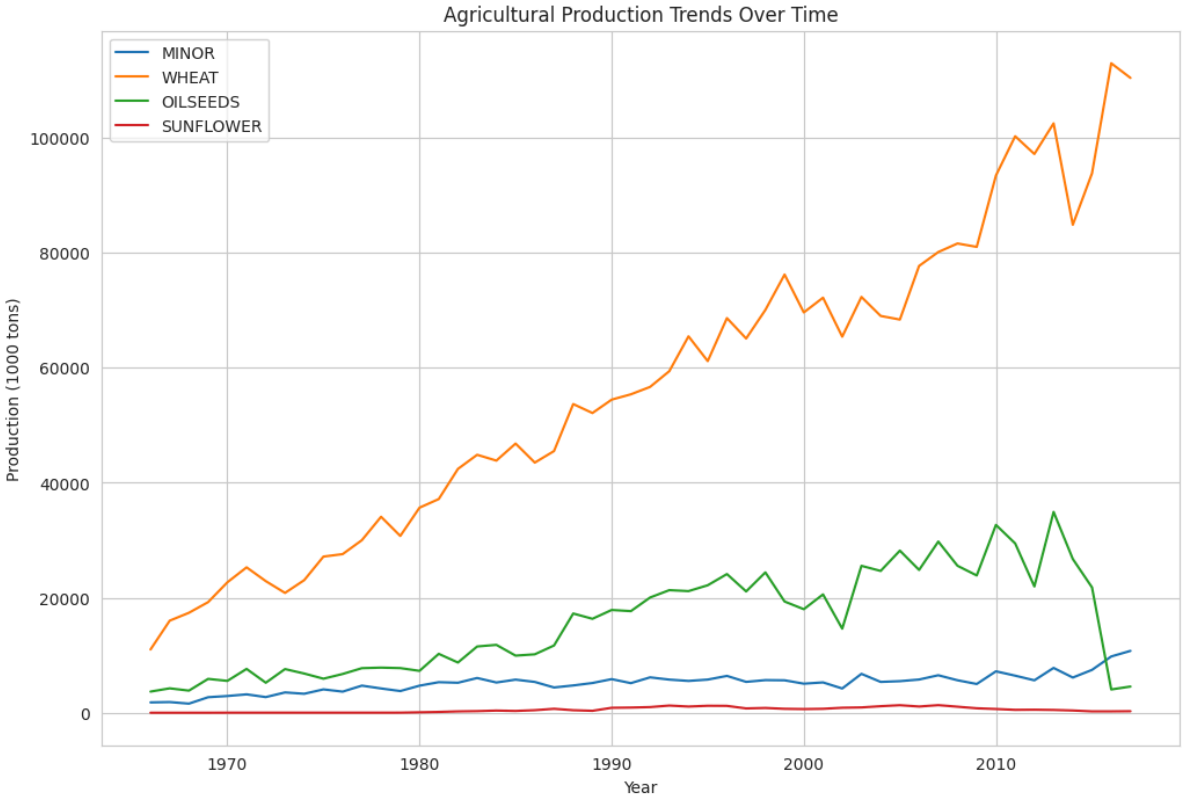
In [ ]:
```
# @title Plotting The Scatterplot For The Rice,Wheat,Maize,Sugarcane Area,Pr
# Here We are defining the parameters to plot
Agriculture_Indian_parameters = [('RICE AREA (1000 ha)', 'RICE PRODUCTION (1
                   ('WHEAT AREA (1000 ha)', 'WHEAT PRODUCTION (1000 tons)'),
                   ('MAIZE AREA (1000 ha)', 'MAIZE PRODUCTION (1000 tons)'),
                   ('SUGARCANE AREA (1000 ha)', 'SUGARCANE PRODUCTION (1000

# Here we are creating subplots for every parameter
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(12, 8))

# Here we are flatten the axes array to perform easy iteration
axes = axes.flatten()

# Here we are plotting the scatterplot for each parameter
for i, pair in enumerate(Agriculture_Indian_parameters):
    ax = axes[i]
    ax.scatter(Agriculture_Indian[pair[0]], Agriculture_Indian[pair[1]], co
    ax.set_title(f"{pair[1]} vs {pair[0]}")
    ax.set_xlabel(pair[0])
    ax.set_ylabel(pair[1])

# Adjust layout
plt.tight_layout()
plt.show()
```



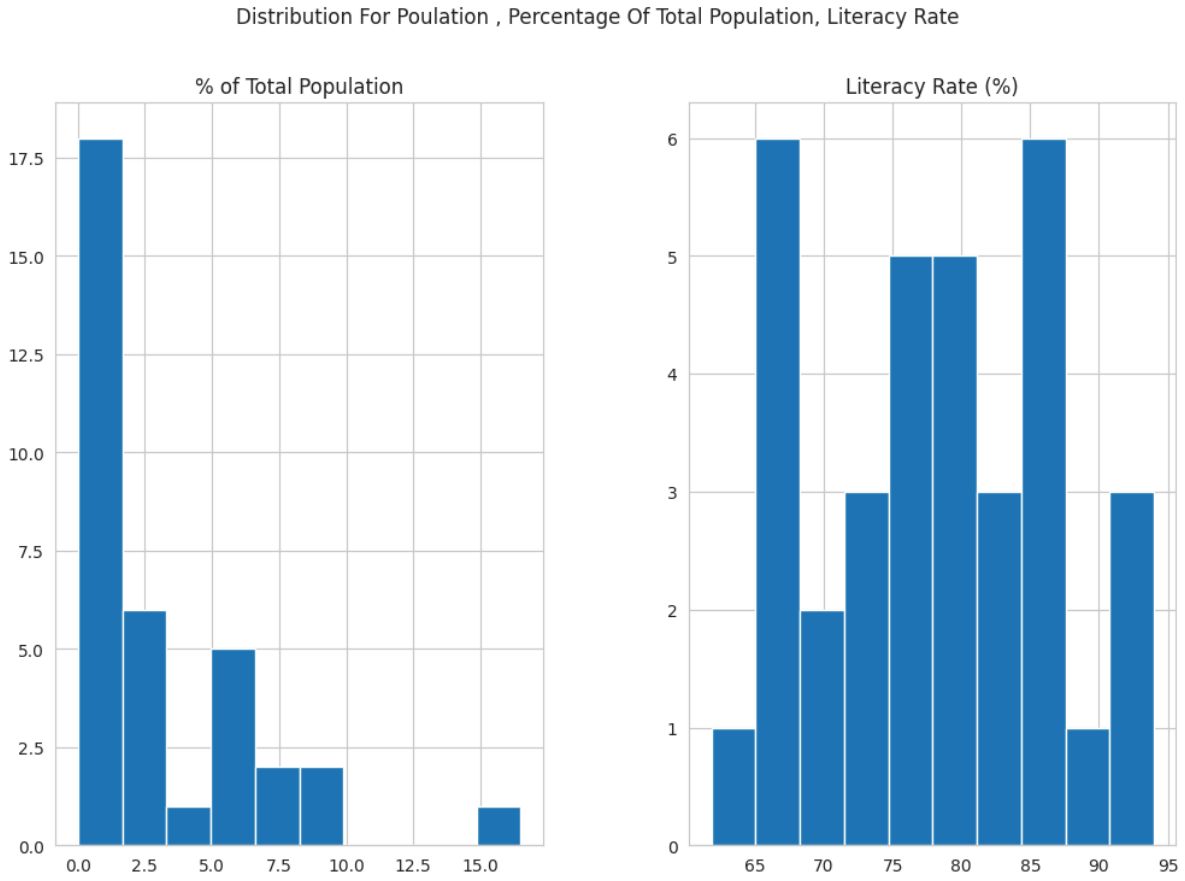
In [ ]:
```
# @title Plotting The Histogram For The Minor Pulses,Wheat,GroundNut,Sugarca
#  Here We are defining the parameters to plot the histogram
Agriculture_Indian_parameters = ['MINOR PULSES AREA (1000 ha)', 'WHEAT AREA

# Here we are creating subplots for every parameter
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(12, 8))

# Here we are flatten the axes array to perform easy iteration
axes = axes.flatten()

# Here we are plotting the scatterplot for each parameter
for i, Parameters_Agriculture in enumerate(Agriculture_Indian_parameters):
    ax = axes[i]
    ax.hist(Agriculture_Indian[Parameters_Agriculture], bins=20, color='blue
    ax.set_title(f"Distribution of {Parameters_Agriculture}")
    ax.set_xlabel(Parameters_Agriculture)
    ax.set_ylabel("Frequency")

# Adjust layout
plt.tight_layout()
plt.show()
```



In [ ]:
```
# @title Plotting The Trends For The Minor Pulses,Wheat,GroundNut,Sunflower
# # Here We are defining the parameters to plot the histogram
#  Here We are defining the parameters to plot the treds for the production
Agriculture_Indian_parameters = ['MINOR PULSES PRODUCTION (1000 tons)', 'WHE

# Group the data by year and calculate the total production for each year
Agriculture_Indian_Production_Over_Time = Agriculture_Indian.groupby('Year')

# Here we are plotting the trends for year
plt.figure(figsize=(12, 8))
for Parameters_Agriculture in Agriculture_Indian_parameters:
    plt.plot(Agriculture_Indian_Production_Over_Time.index, Agriculture_Indi

plt.title("Agricultural Production Trends Over Time")
plt.xlabel("Year")
plt.ylabel("Production (1000 tons)")
plt.legend()
plt.grid(True)
plt.show()
```

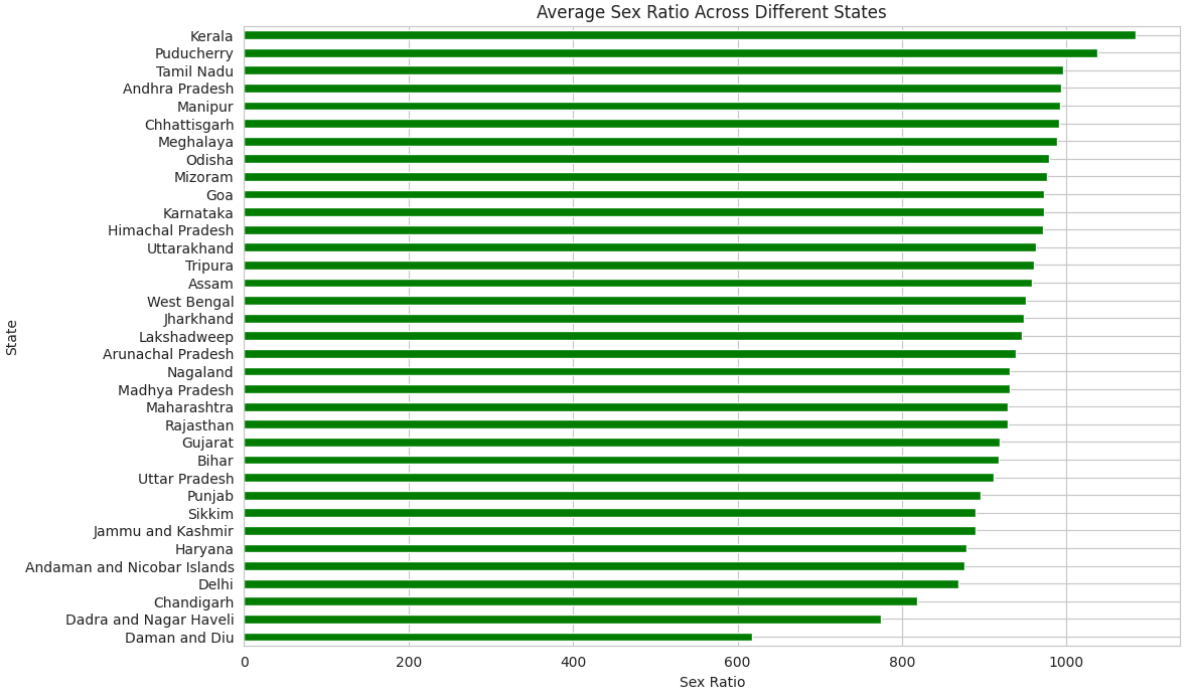


In [ ]:
```
# @title Plotting The HistPlot for The Population, % of Total Population, Li
# Visualize the distribution of numerical variables using histograms
Country_Global_Columns = ['Population', '% of Total Population', 'Literacy R
Country_Global[Country_Global_Columns].hist(figsize=(12, 8))
plt.suptitle("Distribution For Poulation , Percentage Of Total Population, L
plt.show()
```

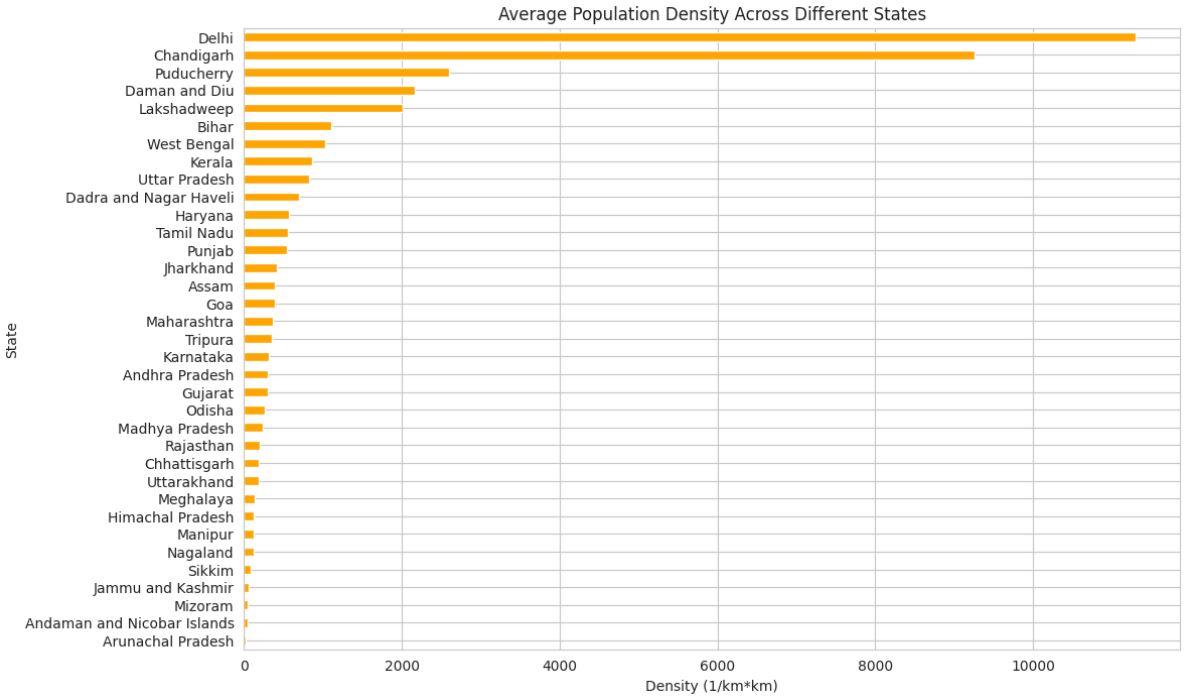


In [ ]:
```
# @title Plotting The BarChart for Male,Female ratio across different states
# Here We are analyzing the sex ratio across different states
```

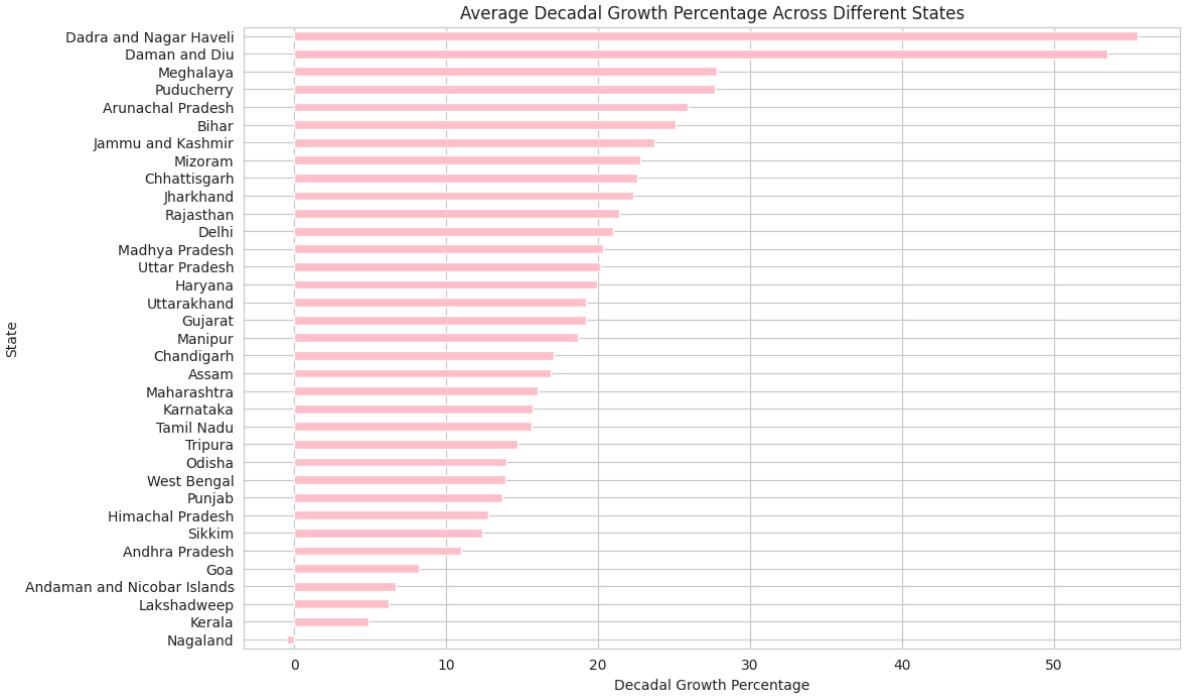
```
plt.figure(figsize=(12, 8))
# In the column of sex ratio there are commas so i am removing he commas and
Country_Global['Sex Ratio'] = Country_Global['Sex Ratio'].str.replace(',',

# Utilizing The Bar Chart We are Plotting the sex ratio average across diffe
Country_Global.groupby('State')['Sex Ratio'].mean().sort_values().plot(kind=
plt.title("Average Sex Ratio Across Different States")
plt.xlabel("Sex Ratio")
plt.ylabel("State")
plt.show()
```



In [ ]:
```
# @title Plotting The BarChart for density of population across states whicl
# Explore the population density across different states
plt.figure(figsize=(12, 8))
# In the column of Population Density there are commas so i am removing he c
Country_Global['Density (1/km*km)'] = Country_Global['Density (1/km*km)'].st
# Utilizing The Bar Chart We are Plotting the population density across dif
Country_Global.groupby('State')['Density (1/km*km)'].mean().sort_values().p
plt.title("Average Population Density Across Different States")
plt.xlabel("Density (1/km*km)")
plt.ylabel("State")
plt.show()
```

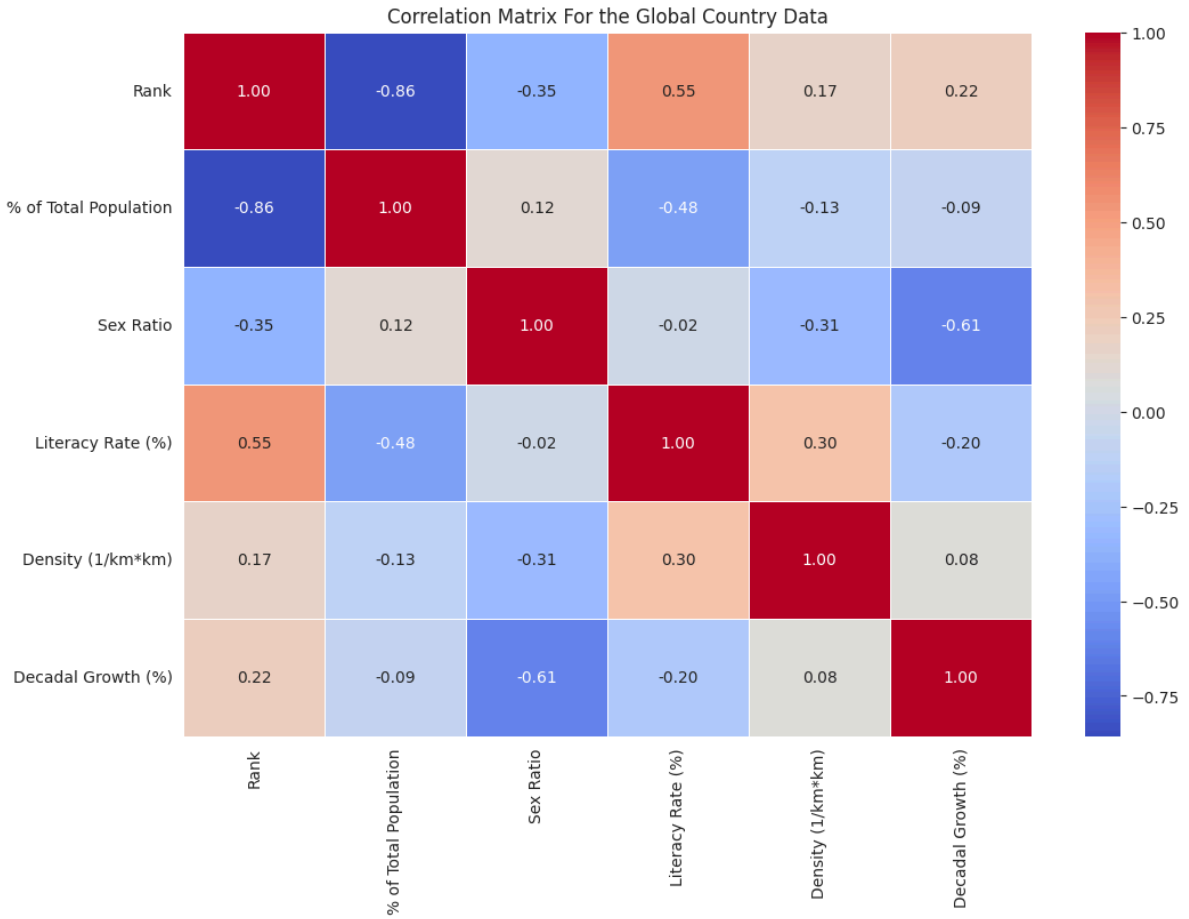


In [ ]:
```
# @title Plotting The BarChart for growth decade percentage across states wh
# Here we are exploring the decadal growth percentage across different state
plt.figure(figsize=(12, 8))
# In the column of Decadal Growth we are removing special characters and cch
Country_Global['Decadal Growth (%)'] = Country_Global['Decadal Growth (%)']

# Utilizing The Bar Chart We are Plotting the average decadal growth across
Country_Global.groupby('State')['Decadal Growth (%)'].mean().sort_values().p
plt.title("Average Decadal Growth Percentage Across Different States")
plt.xlabel("Decadal Growth Percentage")
plt.ylabel("State")
plt.show()
```



In [ ]:
```
# @title Plotting The Corelaion Matrix for the data which is global country
# In this case we are exclude non-numeric columns and calculating the correl
Country_Global_Numeric_Columns = Country_Global.select_dtypes(include=['floa
correlation_matrix = Country_Global[Country_Global_Numeric_Columns].corr()
```

```
# Utilizing the Heat Map we are plotting the correlation matrix
plt.figure(figsize=(12, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f", line
plt.title("Correlation Matrix For the Global Country Data")
plt.show()
```



In [ ]:

```
# @title Merging The Both Agriculuture and Global Country Data
# Here performing inner join and merging the both the datasets on the columr
Agriculture_Global_Merged = pd.merge(Agriculture_Indian, Country_Global, le

# Here We are Displaying the first rows of the dataset which is merged
print(Agriculture_Global_Merged.head())
```

```
   Dist Code  Year  State Code    State Name Dist Name  RICE AREA (1000 ha)
\
0          1  1966          14  Chhattisgarh      Durg                548.0
1          1  1967          14  Chhattisgarh      Durg                547.0
2          1  1968          14  Chhattisgarh      Durg                556.3
3          1  1969          14  Chhattisgarh      Durg                563.4
4          1  1970          14  Chhattisgarh      Durg                571.6

   RICE PRODUCTION (1000 tons)  RICE YIELD (Kg per ha)  WHEAT AREA (1000 h
a)  \
0                        185.0                  337.59                  44.
0
1                        409.0                  747.71                  50.
0
2                        468.0                  841.27                  53.
7
3                        400.8                  711.40                  49.
4
4                        473.6                  828.55                  44.
2

   WHEAT PRODUCTION (1000 tons)  ...  % of Total Population       Males  \
0                          20.0  ...                  2.11  12,832,895
1                          26.0  ...                  2.11  12,832,895
2                          30.0  ...                  2.11  12,832,895
3                          26.5  ...                  2.11  12,832,895
4                          29.0  ...                  2.11  12,832,895

      Females  Sex Ratio  Literacy Rate (%)  Rural Population  \
0  12,712,303        991              70.28        19,603,658
1  12,712,303        991              70.28        19,603,658
2  12,712,303        991              70.28        19,603,658
3  12,712,303        991              70.28        19,603,658
4  12,712,303        991              70.28        19,603,658

   Urban Population  Area (km*km)  Density (1/km*km)  Decadal Growth (%)
0         5,936,538       135,191                189                22.6
1         5,936,538       135,191                189                22.6
2         5,936,538       135,191                189                22.6
3         5,936,538       135,191                189                22.6
4         5,936,538       135,191                189                22.6

[5 rows x 94 columns]
```

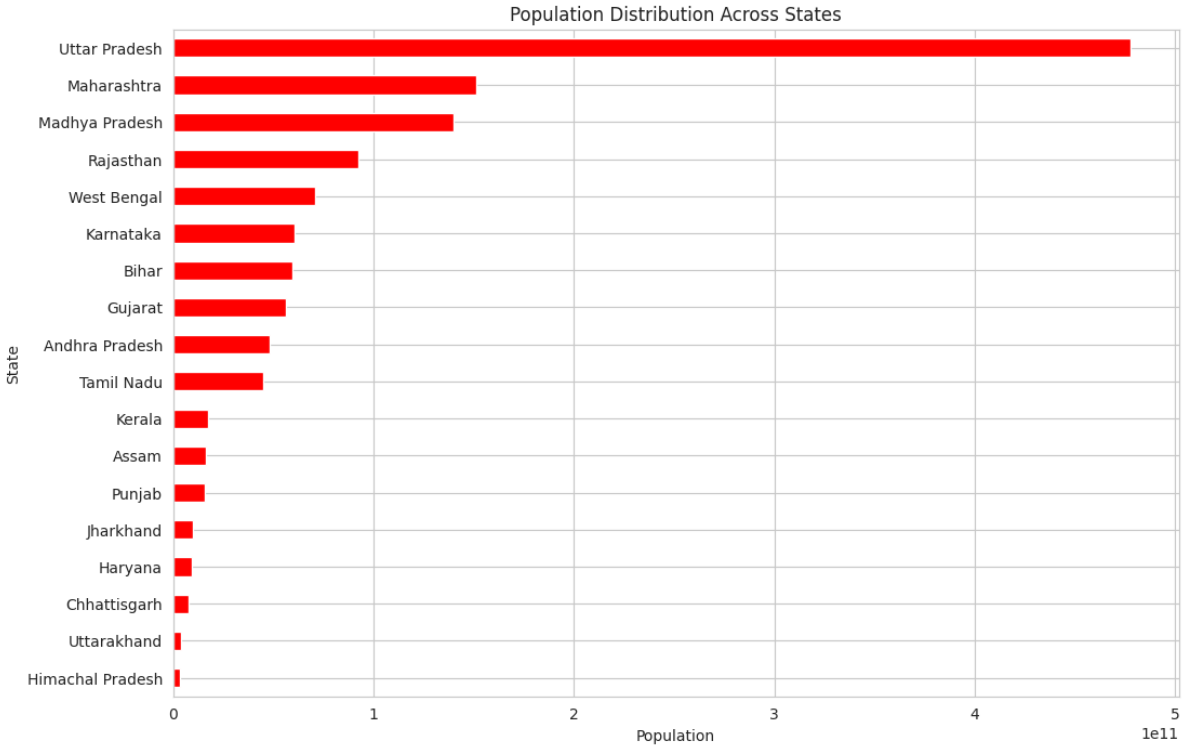
In [ ]:
```
# @title Printing the Columns For The Merged Data
# Printing the columns For both Agriculture and Global Country
Agriculture_Global_Merged.columns
```

Out[ ]:
```
Index(['Dist Code', 'Year', 'State Code', 'State Name', 'Dist Name',
       'RICE AREA (1000 ha)', 'RICE PRODUCTION (1000 tons)',
       'RICE YIELD (Kg per ha)', 'WHEAT AREA (1000 ha)',
       'WHEAT PRODUCTION (1000 tons)', 'WHEAT YIELD (Kg per ha)',
       'KHARIF SORGHUM AREA (1000 ha)',
       'KHARIF SORGHUM PRODUCTION (1000 tons)',
       'KHARIF SORGHUM YIELD (Kg per ha)', 'RABI SORGHUM AREA (1000 ha)',
       'RABI SORGHUM PRODUCTION (1000 tons)', 'RABI SORGHUM YIELD (Kg per h
a)',
       'SORGHUM AREA (1000 ha)', 'SORGHUM PRODUCTION (1000 tons)',
       'SORGHUM YIELD (Kg per ha)', 'PEARL MILLET AREA (1000 ha)',
       'PEARL MILLET PRODUCTION (1000 tons)', 'PEARL MILLET YIELD (Kg per h
a)',
       'MAIZE AREA (1000 ha)', 'MAIZE PRODUCTION (1000 tons)',
       'MAIZE YIELD (Kg per ha)', 'FINGER MILLET AREA (1000 ha)',
       'FINGER MILLET PRODUCTION (1000 tons)',
       'FINGER MILLET YIELD (Kg per ha)', 'BARLEY AREA (1000 ha)',
       'BARLEY PRODUCTION (1000 tons)', 'BARLEY YIELD (Kg per ha)',
       'CHICKPEA AREA (1000 ha)', 'CHICKPEA PRODUCTION (1000 tons)',
       'CHICKPEA YIELD (Kg per ha)', 'PIGEONPEA AREA (1000 ha)',
       'PIGEONPEA PRODUCTION (1000 tons)', 'PIGEONPEA YIELD (Kg per ha)',
       'MINOR PULSES AREA (1000 ha)', 'MINOR PULSES PRODUCTION (1000 ton
s)',
       'MINOR PULSES YIELD (Kg per ha)', 'GROUNDNUT AREA (1000 ha)',
       'GROUNDNUT PRODUCTION (1000 tons)', 'GROUNDNUT YIELD (Kg per ha)',
       'SESAMUM AREA (1000 ha)', 'SESAMUM PRODUCTION (1000 tons)',
       'SESAMUM YIELD (Kg per ha)', 'RAPESEED AND MUSTARD AREA (1000 ha)',
       'RAPESEED AND MUSTARD PRODUCTION (1000 tons)',
       'RAPESEED AND MUSTARD YIELD (Kg per ha)', 'SAFFLOWER AREA (1000 h
a)',
       'SAFFLOWER PRODUCTION (1000 tons)', 'SAFFLOWER YIELD (Kg per ha)',
       'CASTOR AREA (1000 ha)', 'CASTOR PRODUCTION (1000 tons)',
       'CASTOR YIELD (Kg per ha)', 'LINSEED AREA (1000 ha)',
       'LINSEED PRODUCTION (1000 tons)', 'LINSEED YIELD (Kg per ha)',
       'SUNFLOWER AREA (1000 ha)', 'SUNFLOWER PRODUCTION (1000 tons)',
       'SUNFLOWER YIELD (Kg per ha)', 'SOYABEAN AREA (1000 ha)',
       'SOYABEAN PRODUCTION (1000 tons)', 'SOYABEAN YIELD (Kg per ha)',
       'OILSEEDS AREA (1000 ha)', 'OILSEEDS PRODUCTION (1000 tons)',
       'OILSEEDS YIELD (Kg per ha)', 'SUGARCANE AREA (1000 ha)',
       'SUGARCANE PRODUCTION (1000 tons)', 'SUGARCANE YIELD (Kg per ha)',
       'COTTON AREA (1000 ha)', 'COTTON PRODUCTION (1000 tons)',
       'COTTON YIELD (Kg per ha)', 'FRUITS AREA (1000 ha)',
       'VEGETABLES AREA (1000 ha)', 'FRUITS AND VEGETABLES AREA (1000 ha)',
       'POTATOES AREA (1000 ha)', 'ONION AREA (1000 ha)',
       'FODDER AREA (1000 ha)', 'Rank', 'State', 'Capital', 'Population',
       '% of Total Population', 'Males', 'Females', 'Sex Ratio',
       'Literacy Rate (%)', 'Rural Population', 'Urban Population',
       'Area (km*km)', 'Density (1/km*km)', 'Decadal Growth (%)'],
      dtype='object')
```

In [ ]:
```
# @title Checking The DataTypes
# Using the dtypes command we are checking the types of merged data.
Agriculture_Global_Merged.dtypes
```

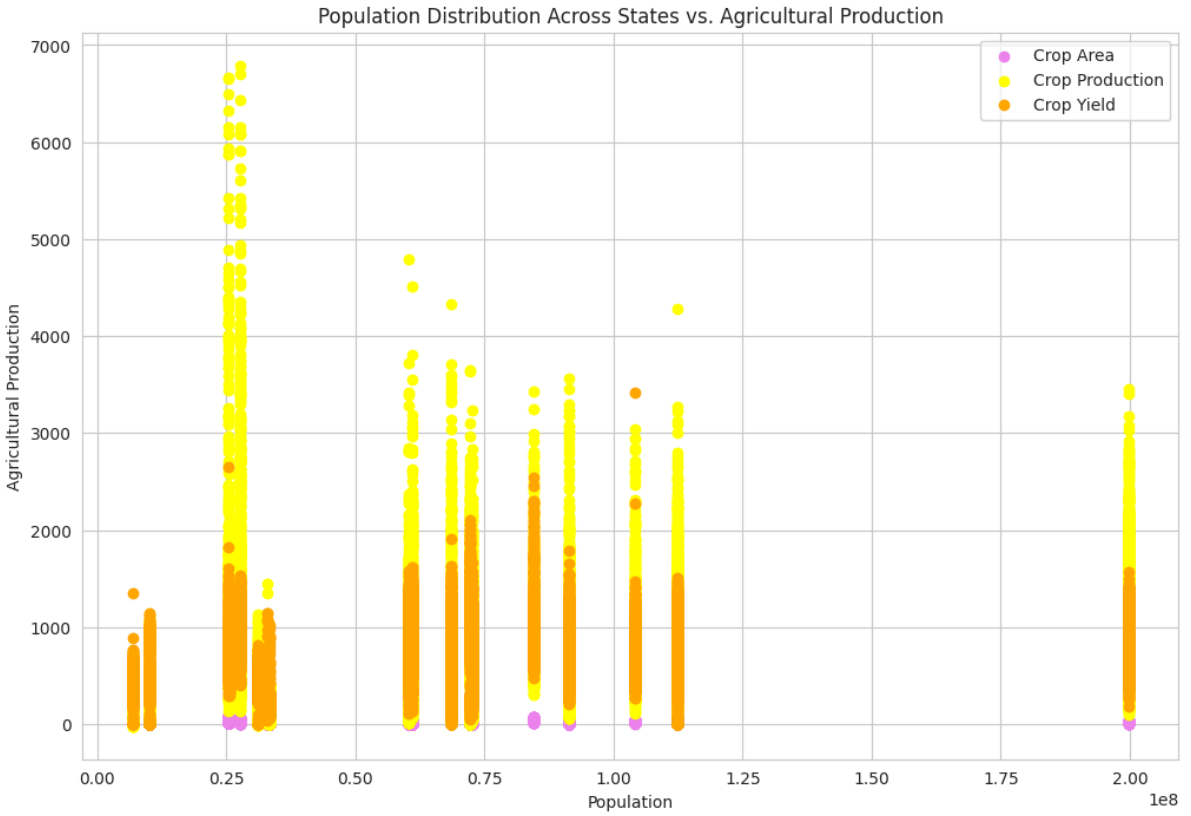
Out[ ]:
```
Dist Code                int64
Year                     int64
State Code               int64
State Name              object
Dist Name               object
                        ...
Rural Population        object
Urban Population        object
Area (km*km)            object
Density (1/km*km)        int64
Decadal Growth (%)     float64
Length: 94, dtype: object
```

In [ ]:
```
# @title Plotting The BarChart for disturbution of population across states
# Here We are Converting the  population related columns to types which are
Agriculture_Global_Merged['Population'] = Agriculture_Global_Merged['Populat
Agriculture_Global_Merged['Males'] = Agriculture_Global_Merged['Males'].str
Agriculture_Global_Merged['Females'] = Agriculture_Global_Merged['Females']
Agriculture_Global_Merged['Rural Population'] = Agriculture_Global_Merged['R
Agriculture_Global_Merged['Urban Population'] = Agriculture_Global_Merged['U
# Utilizing the Barchart we are plotting the across state population disturb
plt.figure(figsize=(12, 8))
Agriculture_Global_Merged.groupby('State Name')['Population'].sum().sort_va
plt.title("Population Distribution Across States")
plt.xlabel("Population")
plt.ylabel("State")
plt.show()
```

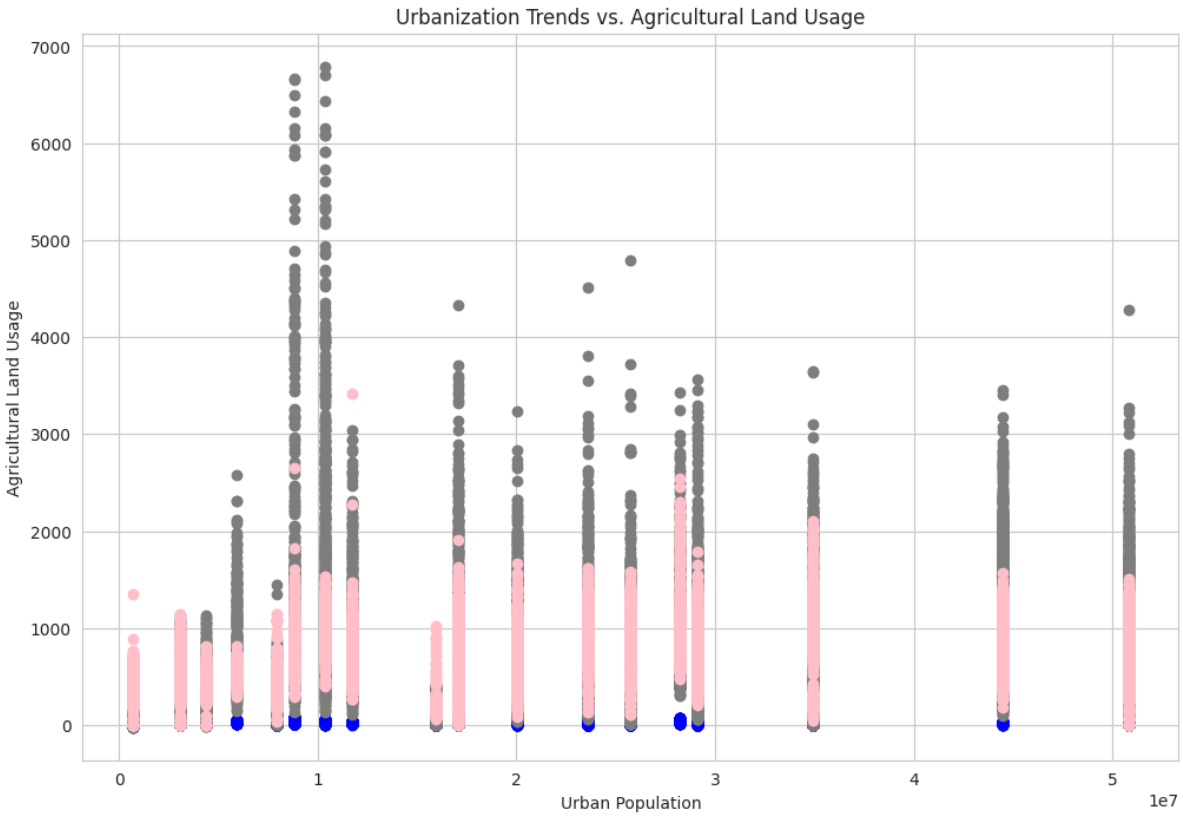


In [ ]:
```
# @title Plotting The ScatterPlot for disturbution of population across stat
# Here we are calculating total agricultural production
Agriculture_Global_Merged['Total Crop Area (1000 ha)'] = Agriculture_Global_
Agriculture_Global_Merged['Total Crop Production (1000 tons)'] = Agriculture
Agriculture_Global_Merged['Average Crop Yield (Kg per ha)'] = Agriculture_G
# Utilizing the scatterplot we are plotting the across state population dist
plt.figure(figsize=(12, 8))
plt.scatter(Agriculture_Global_Merged['Population'], Agriculture_Global_Merg
plt.scatter(Agriculture_Global_Merged['Population'], Agriculture_Global_Merg
plt.scatter(Agriculture_Global_Merged['Population'], Agriculture_Global_Merg
plt.xlabel('Population')
plt.ylabel('Agricultural Production')
```

```
plt.title('Population Distribution Across States vs. Agricultural Productio
plt.legend()
plt.show()
```



In [ ]:
```
# @title Plotting The ScatterPlot for land usage for agriculture and trends
# Here we are setting the Figure Size
plt.figure(figsize=(12, 8))
plt.scatter(Agriculture_Global_Merged['Urban Population'], Agriculture_Globa
plt.scatter(Agriculture_Global_Merged['Urban Population'], Agriculture_Globa
plt.scatter(Agriculture_Global_Merged['Urban Population'], Agriculture_Globa
# Here we are setting xlabel
plt.xlabel('Urban Population')
# Here we are setting ylabel
plt.ylabel('Agricultural Land Usage')
#Here we are setting the title utilizing the title command
plt.title('Urbanization Trends vs. Agricultural Land Usage')
#Here we are setting the legend
plt.legend()
#Utilizing the show command we are displaying the plot
plt.show()
```



In [ ]:
```
# @title Plotting The ScatterPlot for productivity of agriculture and densi
# Here we are setting the Figure Size
plt.figure(figsize=(12, 8))
plt.scatter(Agriculture_Global_Merged['Density (1/km*km)'], Agriculture_Glob
plt.scatter(Agriculture_Global_Merged['Density (1/km*km)'], Agriculture_Glob
plt.xlabel('Population Density (1/km*km)')
plt.ylabel('Agricultural Productivity Metrics')
plt.title('Population Density vs. Agricultural Productivity')
plt.legend()
plt.show()
```