Unemployment Analysis

In [4]:
```
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt
import seaborn as sns

import calendar

import datetime as dt

import plotly.io as pio
import plotly.express as px
import plotly.graph_objects as go
import plotly.figure_factory as ff
from IPython.display import HTML
```

```
C:\ProgramData\anaconda3\Lib\site-packages\paramiko\transport.py:219: CryptographyDeprec
ationWarning: Blowfish has been deprecated
  "class": algorithms.Blowfish,
```

In [14]:
```
df = pd.read_csv('data.csv')
#displaying the dataframe
df.head()
```

Out[14]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Area
0	Andhra Pradesh	31-05-2019	Monthly	3.65	11999139.0	43.24	Rural
1	Andhra Pradesh	30-06-2019	Monthly	3.05	11755881.0	42.05	Rural
2	Andhra Pradesh	31-07-2019	Monthly	3.75	12086707.0	43.50	Rural
3	Andhra Pradesh	31-08-2019	Monthly	3.32	12285693.0	43.97	Rural
4	Andhra Pradesh	30-09-2019	Monthly	5.17	12256762.0	44.68	Rural

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

Out[15]:
```
(768, 7)
```

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 7 columns):
 #   Column                                   Non-Null Count  Dtype
---  ------                                   --------------  -----
 0   Region                                   740 non-null    object
 1    Date                                    740 non-null    object
 2    Frequency                               740 non-null    object
 3    Estimated Unemployment Rate (%)         740 non-null    float64
 4    Estimated Employed                      740 non-null    float64
 5    Estimated Labour Participation Rate (%) 740 non-null    float64
 6   Area                                     740 non-null    object
dtypes: float64(3), object(4)
memory usage: 42.1+ KB
```

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
Region                                   28
 Date                                    28
 Frequency                               28
 Estimated Unemployment Rate (%)         28
 Estimated Employed                      28
 Estimated Labour Participation Rate (%)  28
Area                                     28
dtype: int64
```

In [15]:
```
df=df.dropna()
```

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

Out[14]:
```
Region                                   0
 Date                                    0
 Frequency                               0
 Estimated Unemployment Rate (%)         0
 Estimated Employed                      0
 Estimated Labour Participation Rate (%)  0
Area                                     0
dtype: int64
```

In [16]:
```
df.columns
```

Out[16]:
```
Index(['Region', ' Date', ' Frequency', ' Estimated Unemployment Rate (%)',
       ' Estimated Employed', ' Estimated Labour Participation Rate (%)',
       'Area'],
      dtype='object')
```

In [17]:
```
df.columns =['Region','Date','Frequency','Estimated Unemployment Rate (%)','Estimated Em
```

In [18]:
```
df.head(3)
```

Out[18]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Area
0	Andhra Pradesh	31-05-2019	Monthly	3.65	11999139.0	43.24	Rural
1	Andhra Pradesh	30-06-2019	Monthly	3.05	11755881.0	42.05	Rural
2	Andhra Pradesh	31-07-2019	Monthly	3.75	12086707.0	43.50	Rural

In [19]:
```
df.describe()
```

Out[19]:

	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)
count	740.000000	7.400000e+02	740.000000
mean	11.787946	7.204460e+06	42.630122
std	10.721298	8.087988e+06	8.111094
min	0.000000	4.942000e+04	13.330000
25%	4.657500	1.190404e+06	38.062500
50%	8.350000	4.744178e+06	41.160000
75%	15.887500	1.127549e+07	45.505000
max	76.740000	4.577751e+07	72.570000

In [21]:
```
round(df[['Estimated Unemployment Rate (%)', 'Estimated Employed', 'Estimated Labour Par
```

Out[21]:

	count	mean	std	min	25%	50%	75%	max
Estimated Unemployment Rate (%)	740.0	11.79	10.72	0.00	4.66	8.35	15.89	76.74
Estimated Employed	740.0	7204460.03	8087988.43	49420.00	1190404.50	4744178.50	11275489.50	45777509.00
Estimated Labour Participation Rate (%)	740.0	42.63	8.11	13.33	38.06	41.16	45.50	72.57

In [22]:
```
#grouping by 'Region' and finding mean values for the numerical columns
areaStats = df.groupby(['Area'])[['Estimated Unemployment Rate (%)',
                                  'Estimated Employed',
                                  'Estimated Labour Participation Rate (%)']].mean()

#rounding the values to 2 decimal points
round(areaStats,2)#grouping by 'Region' and finding mean values for the numerical column
```

Out[22]:

	Area	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)
0	Rural	10.32	10192852.57	44.46
1	Urban	13.17	4388625.58	40.90

In [23]:
```
regionStats = df.groupby(['Region'])[['Estimated Unemployment Rate (%)',
                                      'Estimated Employed',
                                      'Estimated Labour Participation Rate (%)']].mean()

#rounding the values to 2 decimal points
round(regionStats,2)#grouping by 'Region' and finding mean values for the numerical colu
```

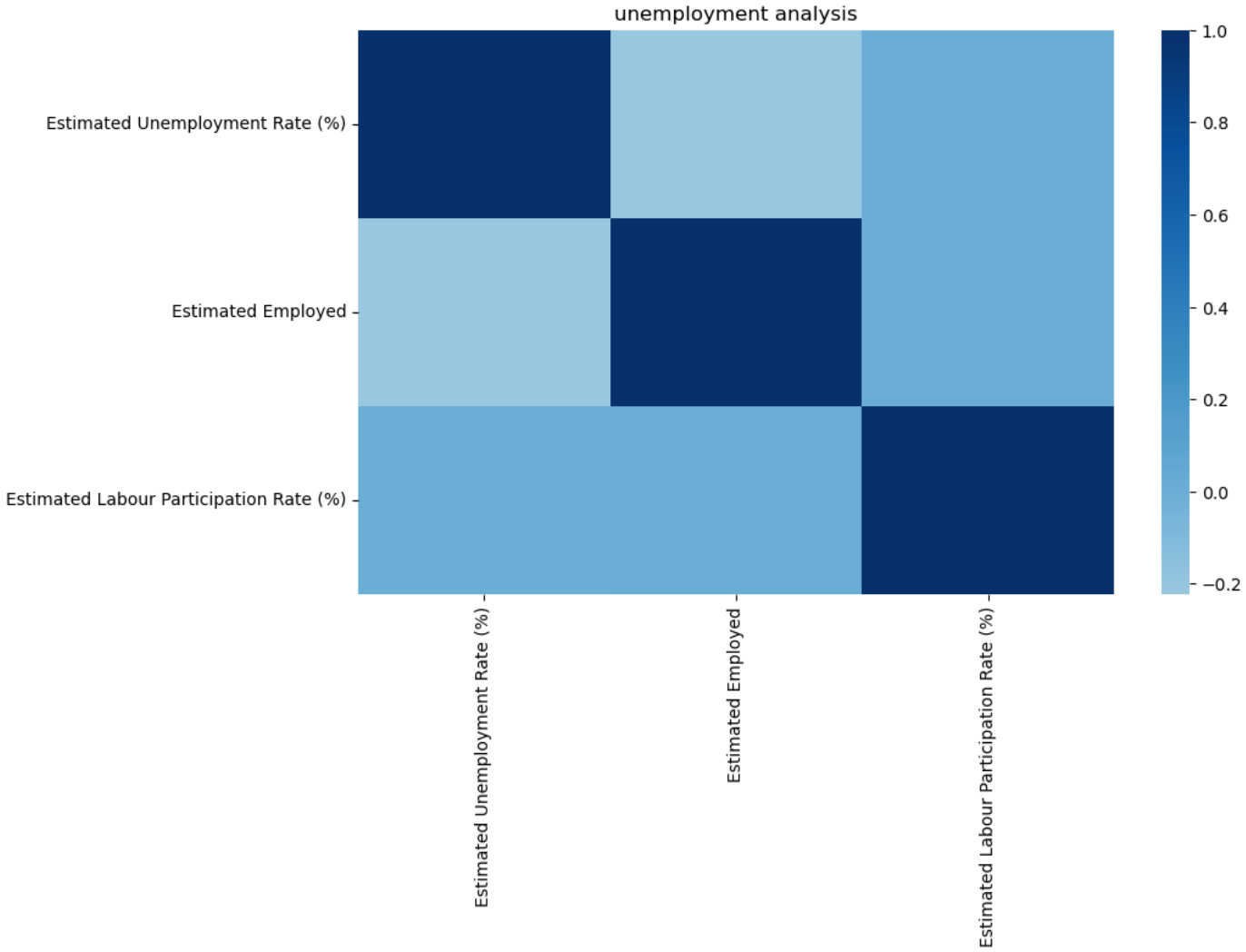
Out[23]:

	Region	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)
0	Andhra Pradesh	7.48	8154093.18	39.38
1	Assam	6.43	5354772.15	44.87
2	Bihar	18.92	12366189.14	38.15
3	Chandigarh	15.99	316831.25	39.34
4	Chhattisgarh	9.24	4303498.57	42.81
5	Delhi	16.50	2627512.86	38.93
6	Goa	9.27	226308.33	39.25
7	Gujarat	6.66	11402012.79	46.10
8	Haryana	26.28	3557072.46	42.74
9	Himachal Pradesh	18.54	1059823.71	44.22
10	Jammu & Kashmir	16.19	1799931.67	41.03
11	Jharkhand	20.58	4469240.43	41.67
12	Karnataka	6.68	10667119.29	41.35
13	Kerala	10.12	4425899.50	34.87
14	Madhya Pradesh	7.41	11115484.32	38.82
15	Maharashtra	7.56	19990195.86	42.30
16	Meghalaya	4.80	689736.81	57.08
17	Odisha	5.66	6545746.96	38.93
18	Puducherry	10.22	212278.08	38.99
19	Punjab	12.03	4539362.00	41.14
20	Rajasthan	14.06	10041064.75	39.97
21	Sikkim	7.25	106880.71	46.07
22	Tamil Nadu	9.28	12269546.75	40.87
23	Telangana	7.74	7939662.75	53.00
24	Tripura	28.35	717002.64	61.82
25	Uttar Pradesh	12.55	28094832.18	39.43
26	Uttarakhand	6.58	1390228.11	33.78
27	West Bengal	8.12	17198538.00	45.42

In [25]:
```
fig, ax = plt.subplots(figsize=(10,6))
sns.heatmap(df.corr(), center=0, cmap='Blues')
ax.set_title('unemployment analysis')
```

```
C:\Users\User\AppData\Local\Temp\ipykernel_3484\3600111740.py:2: FutureWarning: The defa
ult value of numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only to sile
nce this warning.
  sns.heatmap(df.corr(), center=0, cmap='Blues')
```

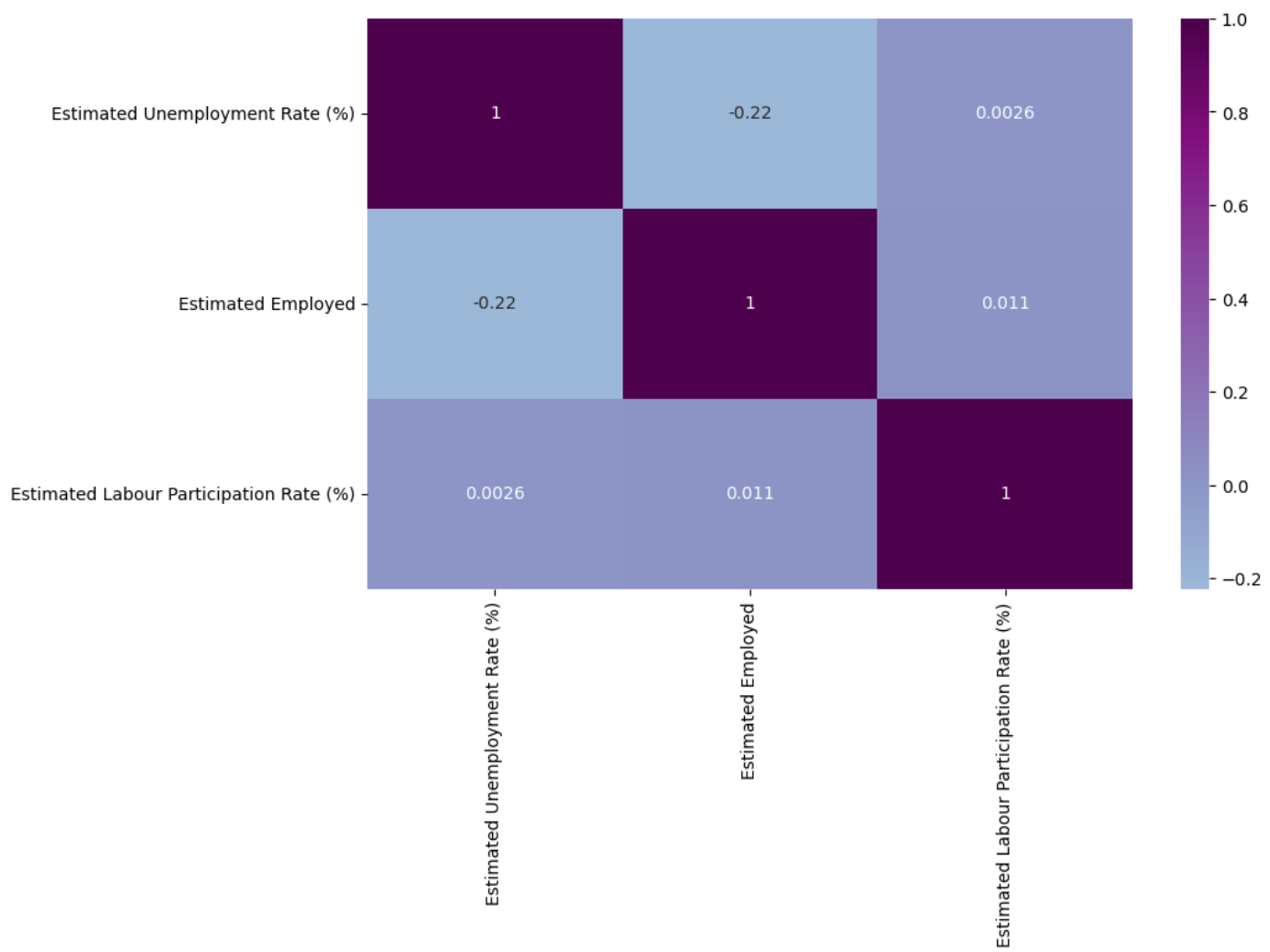
Out[25]:
```
Text(0.5, 1.0, 'unemployment analysis')
```



In [31]:
```
fig, ax = plt.subplots(figsize=(10,6))
sns.heatmap(df.corr(), center=0, cmap='BuPu', annot=True)
```

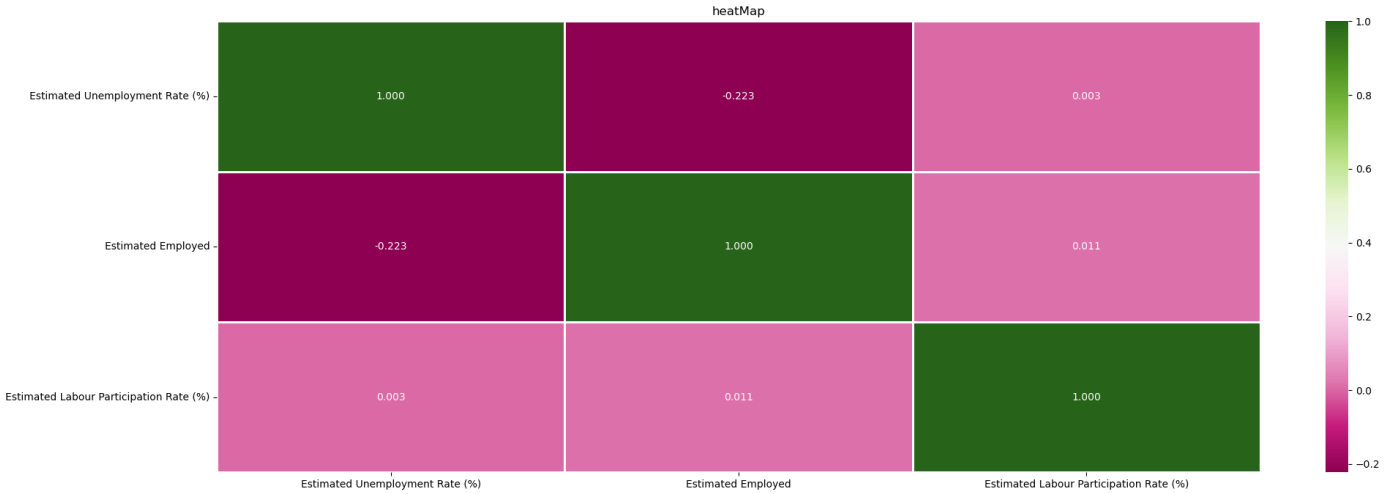
```
C:\Users\User\AppData\Local\Temp\ipykernel_3484\611880948.py:2: FutureWarning: The defau
lt value of numeric_only in DataFrame.corr is deprecated. In a future version, it will d
efault to False. Select only valid columns or specify the value of numeric_only to silen
ce this warning.
  sns.heatmap(df.corr(), center=0, cmap='BuPu', annot=True)
```

Out[31]:
```
<Axes: >
```



In [36]:

```
heatMap = df[['Estimated Unemployment Rate (%)', 'Estimated Employed',
              'Estimated Labour Participation Rate (%)']]

#constructing on heatMap with correlation values
heatMap = heatMap.corr()

#plotting the figure
plt.figure(figsize=(23,8))
sns.heatmap(heatMap, annot=True,cmap='PiYG', fmt='.3f', linewidths=1)
plt.title('heatMap')
plt.show()
```



In [16]:

```
df = pd.read_csv('data.csv')
#displaying the dataframe
df.head()
```

Out[16]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Area
0	Andhra Pradesh	31-05-2019	Monthly	3.65	11999139.0	43.24	Rural
1	Andhra Pradesh	30-06-2019	Monthly	3.05	11755881.0	42.05	Rural
2	Andhra Pradesh	31-07-2019	Monthly	3.75	12086707.0	43.50	Rural
3	Andhra Pradesh	31-08-2019	Monthly	3.32	12285693.0	43.97	Rural
4	Andhra Pradesh	30-09-2019	Monthly	5.17	12256762.0	44.68	Rural

In [25]:
```
u_emp=df[['Area',' Estimated Unemployment Rate (%)']].groupby('Area').sum().sort_values(
u_emp
```

Out[25]:

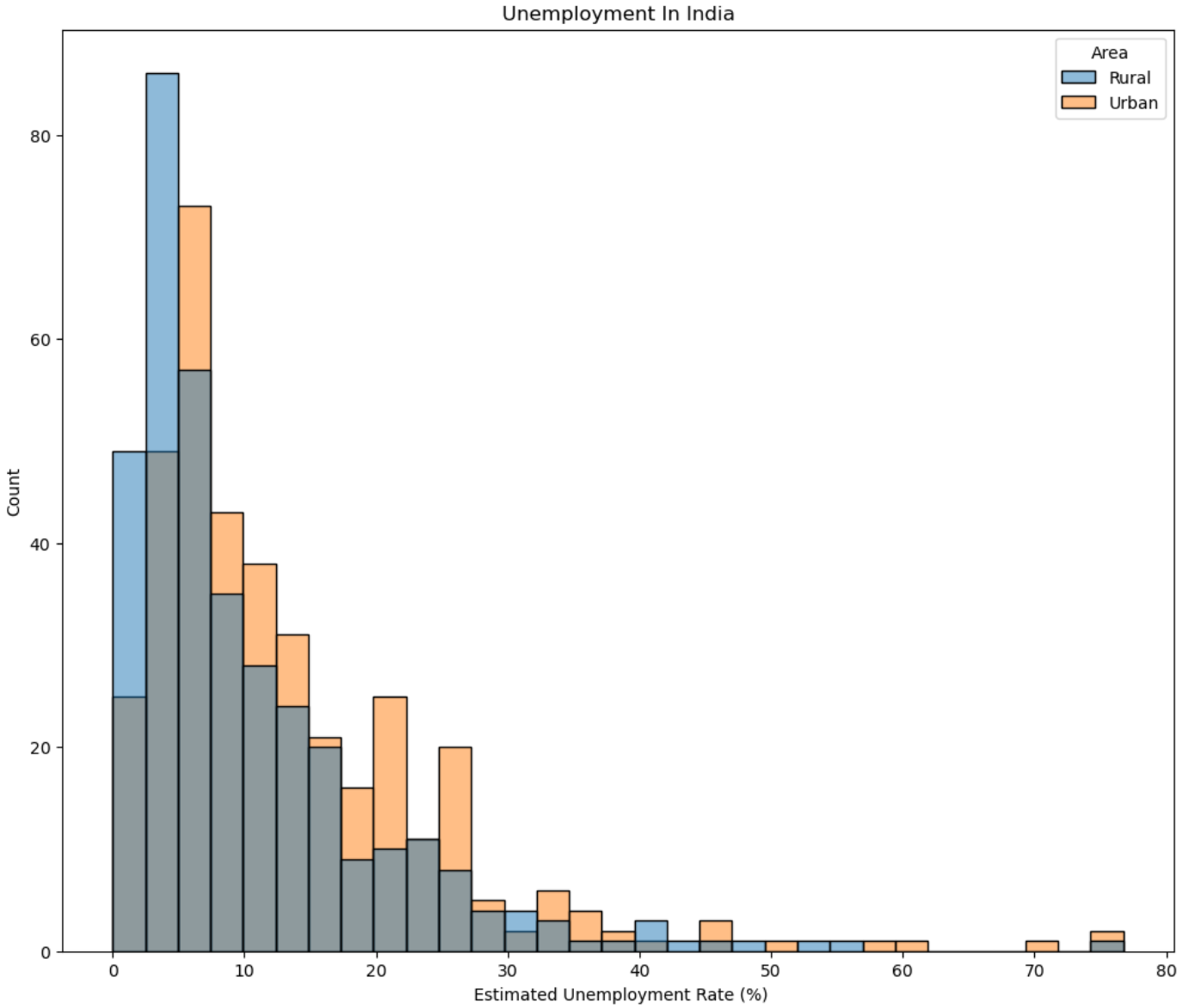
Area	Estimated Unemployment Rate (%)
Urban	5016.48
Rural	3706.60

In [29]:
```
import plotly.express as pl
!pip install kaleido
```

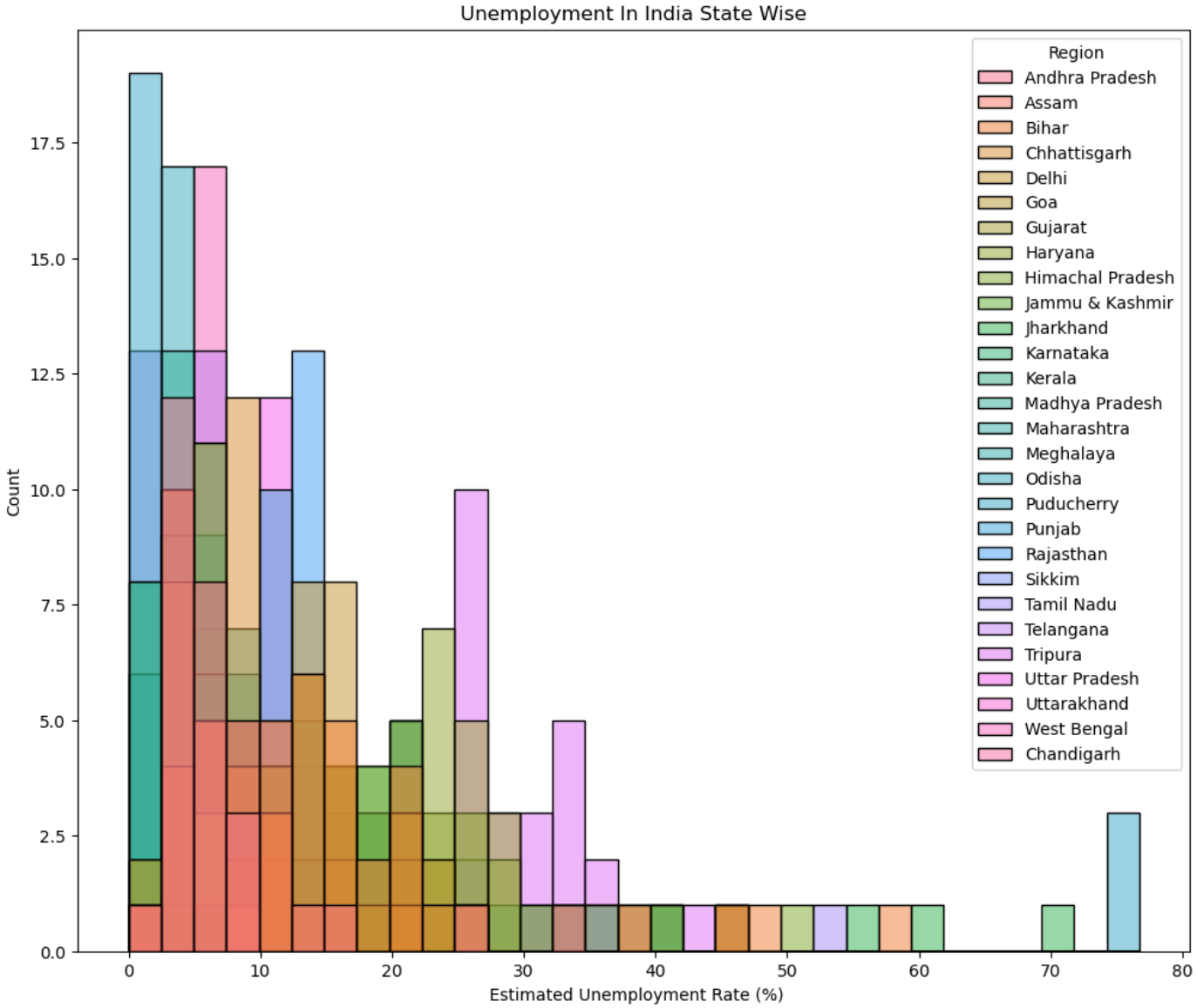
```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: kaleido in c:\users\user\appdata\roaming\python\python311
\site-packages (0.2.1)
```

In [32]:
```
import plotly.express as px
df = pd.read_csv('data.csv')

fig = px.scatter(df, x="Area", y=' Estimated Unemployment Rate (%)', color=' Estimated L
                 title="Scatterplot")


fig.show(renderer='colab')
fig.show(renderer='notebook')
```

In [35]:
```
plt.figure(figsize=(12,10))
plt.title('Unemployment In India')
sns.histplot(x=' Estimated Unemployment Rate (%)', hue="Area", data=df)
plt.show()
```



In [37]:
```
plt.figure(figsize=(12,10))
plt.title('Unemployment In India State Wise')
sns.histplot(x=' Estimated Unemployment Rate (%)', hue="Region", data=df)
plt.show()
```



So this is how you can analyze the unemployment rate by using the Python programming language. Unemployment is measured by the unemployment rate which is the number of people who are unemployed as a percentage of the total labour force.