

Unemployment Analysis With Python

Importing required libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
## Supress warnings
import warnings
warnings.filterwarnings("ignore")

data = pd.read_csv("/content/Unemployment in India.csv")
print("data has been successfully loaded")
```

data has been successfully loaded

Checking and cleaning the dataset

data

	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
...	...	...	...	...	...	...	...
763	NaN	NaN	NaN	NaN	NaN	NaN	NaN
764	NaN	NaN	NaN	NaN	NaN	NaN	NaN
765	NaN	NaN	NaN	NaN	NaN	NaN	NaN
766	NaN	NaN	NaN	NaN	NaN	NaN	NaN
767	NaN	NaN	NaN	NaN	NaN	NaN	NaN

data.head

<bound method NDFrame.head of				Region	Date	Frequency	Estimated Unemployment Rate (%)	\
0	Andhra Pradesh	31-05-2019	Monthly				3.65	
1	Andhra Pradesh	30-06-2019	Monthly				3.05	
2	Andhra Pradesh	31-07-2019	Monthly				3.75	
3	Andhra Pradesh	31-08-2019	Monthly				3.32	
4	Andhra Pradesh	30-09-2019	Monthly				5.17	
..	...	...	...				...	
763	NaN	NaN	NaN				NaN	
764	NaN	NaN	NaN				NaN	
765	NaN	NaN	NaN				NaN	
766	NaN	NaN	NaN				NaN	
767	NaN	NaN	NaN				NaN	
				Estimated Employed	Estimated Labour Participation Rate (%)		Area	

```
0      11999139.0      43.24 Rural
1      11755881.0      42.05 Rural
2      12086707.0      43.50 Rural
3      12285693.0      43.97 Rural
4      12256762.0      44.68 Rural
..      ...
763      NaN      NaN      NaN
764      NaN      NaN      NaN
765      NaN      NaN      NaN
766      NaN      NaN      NaN
767      NaN      NaN      NaN
```

[768 rows x 7 columns]>

data.tail

```
<bound method NDFrame.tail of
0  Andhra Pradesh 31-05-2019 Monthly      3.65
1  Andhra Pradesh 30-06-2019 Monthly      3.05
2  Andhra Pradesh 31-07-2019 Monthly      3.75
3  Andhra Pradesh 31-08-2019 Monthly      3.32
4  Andhra Pradesh 30-09-2019 Monthly      5.17
..      ...
763      NaN      NaN      NaN      NaN
764      NaN      NaN      NaN      NaN
765      NaN      NaN      NaN      NaN
766      NaN      NaN      NaN      NaN
767      NaN      NaN      NaN      NaN
```

```
      Estimated Employed      Estimated Labour Participation Rate (%)      Area
0      11999139.0      43.24 Rural
1      11755881.0      42.05 Rural
2      12086707.0      43.50 Rural
3      12285693.0      43.97 Rural
4      12256762.0      44.68 Rural
..      ...
763      NaN      NaN      NaN
764      NaN      NaN      NaN
765      NaN      NaN      NaN
766      NaN      NaN      NaN
767      NaN      NaN      NaN
```

[768 rows x 7 columns]>

data.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
```

data.shape

(768, 7)

data.describe()

	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



✓ Let's see if this dataset contains missing values if not:

```
print(data.isnull().sum())
```

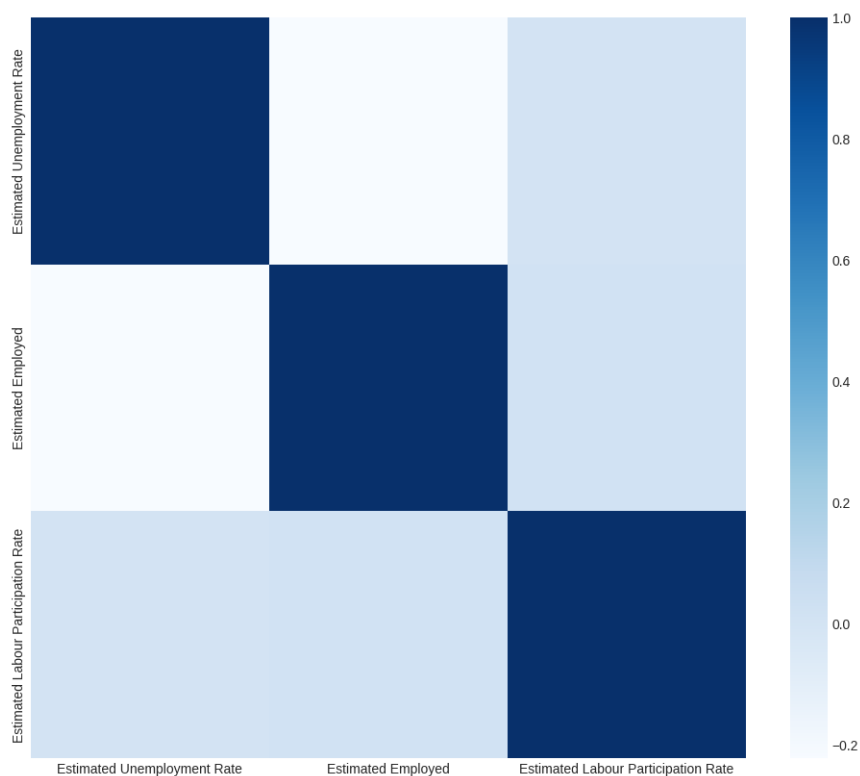
```
Region                28
Date                  28
Frequency              28
Estimated Unemployment Rate (%)  28
Estimated Employed      28
Estimated Labour Participation Rate (%)  28
Area                  28
dtype: int64
```

✓ dataframe

```
data.columns= ["States", "Date", "Frequency",
               "Estimated Unemployment Rate",
               "Estimated Employed",
               "Estimated Labour Participation Rate",
               "Region",]
```

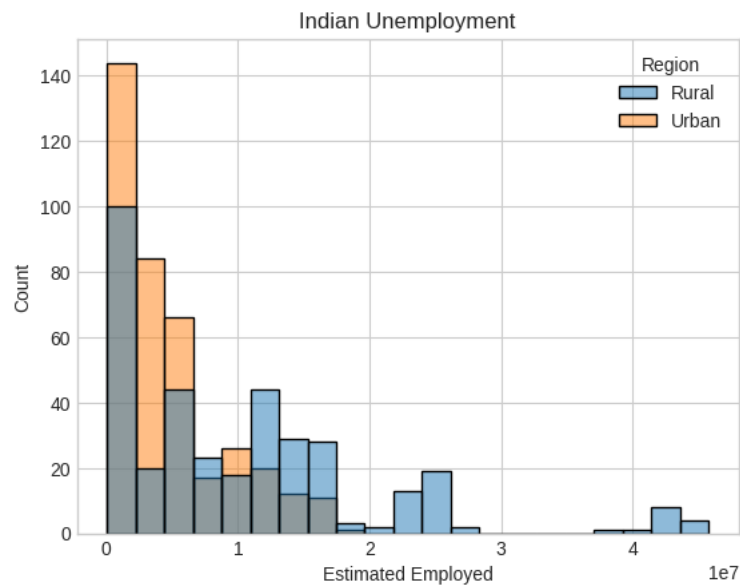
✓ heat map

```
plt.style.use('seaborn-whitegrid')
plt.figure(figsize=(12, 10))
sns.heatmap(data.corr(), cmap="Blues")
plt.show()
```



## ✓ Unemployment Rate Analysis: Data Visualization

```
data.columns= ["States","Date","Frequency",  
              "Estimated Unemployment Rate","Estimated Employed",  
              "Estimated Labour Participation Rate","Region"]  
plt.title("Indian Unemployment")  
sns.histplot(x="Estimated Employed", hue="Region", data=data)  
plt.show()
```



```
plt.figure(figsize=(12, 10))  
plt.title("Indian Unemployment")  
sns.histplot(x="Estimated Unemployment Rate", hue="Region", data=data)  
plt.show()
```



```
data.Region.nunique()
```

```
2
```

```
make_total = data.pivot_table("Estimated Unemployment Rate",index=['Region'],aggfunc='mean')
topstate=make_total.sort_values(by='Estimated Unemployment Rate',ascending=False)[:47]
print(topstate)
```

Estimated Unemployment Rate	
Region	
Urban	13.166614
Rural	10.324791

```
maketotal_1 = data.pivot_table(values='Estimated Unemployment Rate',index=['Region'],aggfunc=np.std)
df1 = maketotal_1.reset_index().dropna(subset=['Estimated Unemployment Rate'])
df2 = df1.loc[df1.groupby('Region')['Estimated Unemployment Rate'].idxmax()]
for index,row in df2.iterrows():
    print(row['Region'], "Region which", row['Region'], "has the highest yearly fluncation.")
```

```
Rural Region which Rural has the highest yearly fluncation.
Urban Region which Urban has the highest yearly fluncation.
```