

Task 2: Unemployment Analysis with Python

2Q) Unemployment is measured by the unemployment rate which is the number of people who are unemployed as a percentage of the total labour force. We have seen a sharp increase in the unemployment rate during Covid-19, so analyzing the unemployment rate can be a good data science project.

```
In [1]: 1 # import all required packages
        2
        3 import numpy as np
        4 import pandas as pd
        5 import matplotlib.pyplot as plt
        6 import seaborn as sns
        7 import datetime as dt
        8 import warnings
        9 warnings.filterwarnings('ignore')
       10 import plotly.express as px
       11 import plotly.graph_objects as go
```

```
In [2]: 1 df1=pd.read_csv('Unemployment in India.csv') # reading the first data
        2 df1
```

Out[2]:

	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

768 rows × 7 columns

In [3]: 1 df1.head()

Out[3]:

	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 [4]: 1 df1.tail()

Out[4]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Area
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

In [5]: 1 df1.shape

Out[5]: (768, 7)

In [6]: 1 df1.isnull().sum() # identifying the null values in the dataset

Out[6]: Region 28
Date 28
Frequency 28
Estimated Unemployment Rate (%) 28
Estimated Employed 28
Estimated Labour Participation Rate (%) 28
Area 28
dtype: int64

In [7]: 1 df1.dropna(axis=0,how='all',inplace=True) # drop the rows with all null

```
In [8]: 1 df1.tail()
```

Out[8]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Area
749	West Bengal	29- 02- 2020	Monthly	7.55	10871168.0	44.09	Urban
750	West Bengal	31- 03- 2020	Monthly	6.67	10806105.0	43.34	Urban
751	West Bengal	30- 04- 2020	Monthly	15.63	9299466.0	41.20	Urban
752	West Bengal	31- 05- 2020	Monthly	15.22	9240903.0	40.67	Urban
753	West Bengal	30- 06- 2020	Monthly	9.86	9088931.0	37.57	Urban

```
In [9]: 1 df1.shape
```

Out[9]: (740, 7)

```
In [10]: 1 df1.columns # here, the column names contain spaces in them
```

Out[10]: Index(['Region', ' Date', ' Frequency', ' Estimated Unemployment Rate (%)',
' Estimated Employed', ' Estimated Labour Participation Rate (%)',
'Area'],
dtype='object')

```
In [11]: 1 df1.rename(columns=lambda x:x.strip(),inplace=True) # remove the spaces
```

In [12]: 1 df1.head()

Out[12]:

	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 [13]: 1 df1.columns

Out[13]: Index(['Region', 'Date', 'Frequency', 'Estimated Unemployment Rate (%)', 'Estimated Employed', 'Estimated Labour Participation Rate (%)', 'Area'], dtype='object')

In [14]: 1 df1.dtypes

Out[14]: Region object
Date object
Frequency object
Estimated Unemployment Rate (%) float64
Estimated Employed float64
Estimated Labour Participation Rate (%) float64
Area object
dtype: object

```
In [15]: 1 df1['Region'].value_counts()
```

```
Out[15]: Andhra Pradesh      28
          Kerala             28
          West Bengal        28
          Uttar Pradesh      28
          Tripura            28
          Telangana          28
          Tamil Nadu         28
          Rajasthan          28
          Punjab             28
          Odisha             28
          Madhya Pradesh     28
          Maharashtra        28
          Karnataka          28
          Jharkhand          28
          Himachal Pradesh   28
          Haryana            28
          Gujarat            28
          Delhi              28
          Chhattisgarh       28
          Bihar              28
          Meghalaya          27
          Uttarakhand        27
          Assam              26
          Puducherry         26
          Goa                24
          Jammu & Kashmir    21
          Sikkim             17
          Chandigarh         12
          Name: Region, dtype: int64
```

```
In [16]: 1 df1['Frequency'].value_counts() # here, values with same meanings have
```

```
Out[16]: Monthly      381
          Monthly      359
          Name: Frequency, dtype: int64
```

```
In [17]: 1 df1['Area'].value_counts()
```

```
Out[17]: Urban      381
          Rural      359
          Name: Area, dtype: int64
```

```
In [18]: 1 # mapping to give same notations to similar values
          2 df1['Frequency']=df1['Frequency'].map({'Monthly':'Month',' Monthly':'Mc
```

```
In [19]: 1 df1['Frequency'].value_counts() # values of the column after changes
```

```
Out[19]: Month      740
          Name: Frequency, dtype: int64
```

```
In [20]: 1 df2=pd.read_csv('Unemployment_Rate_upto_11_2020.csv') # reading the sec
2 df2
```

Out[20]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Region.1	longitude
0	Andhra Pradesh	31-01-2020	M	5.48	16635535	41.02	South	15.9129
1	Andhra Pradesh	29-02-2020	M	5.83	16545652	40.90	South	15.9129
2	Andhra Pradesh	31-03-2020	M	5.79	15881197	39.18	South	15.9129
3	Andhra Pradesh	30-04-2020	M	20.51	11336911	33.10	South	15.9129
4	Andhra Pradesh	31-05-2020	M	17.43	12988845	36.46	South	15.9129
...
262	West Bengal	30-06-2020	M	7.29	30726310	40.39	East	22.9868
263	West Bengal	31-07-2020	M	6.83	35372506	46.17	East	22.9868
264	West Bengal	31-08-2020	M	14.87	33298644	47.48	East	22.9868
265	West Bengal	30-09-2020	M	9.35	35707239	47.73	East	22.9868
266	West Bengal	31-10-2020	M	9.98	33962549	45.63	East	22.9868

267 rows × 9 columns



In [21]: 1 df2.head()

Out[21]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Region.1	longitude	I
0	Andhra Pradesh	31-01-2020	M	5.48	16635535	41.02	South	15.9129	
1	Andhra Pradesh	29-02-2020	M	5.83	16545652	40.90	South	15.9129	
2	Andhra Pradesh	31-03-2020	M	5.79	15881197	39.18	South	15.9129	
3	Andhra Pradesh	30-04-2020	M	20.51	11336911	33.10	South	15.9129	
4	Andhra Pradesh	31-05-2020	M	17.43	12988845	36.46	South	15.9129	



In [22]: 1 df2.tail()

Out[22]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Region.1	longitude	
262	West Bengal	30-06-2020	M	7.29	30726310	40.39	East	22.9868	
263	West Bengal	31-07-2020	M	6.83	35372506	46.17	East	22.9868	
264	West Bengal	31-08-2020	M	14.87	33298644	47.48	East	22.9868	
265	West Bengal	30-09-2020	M	9.35	35707239	47.73	East	22.9868	
266	West Bengal	31-10-2020	M	9.98	33962549	45.63	East	22.9868	



In [23]: 1 df2.shape

Out[23]: (267, 9)

```
In [24]: 1 df2.columns # here, the columns contain spaces in their names
```

```
Out[24]: Index(['Region', ' Date', ' Frequency', ' Estimated Unemployment Rate (%)',  
              ' Estimated Employed', ' Estimated Labour Participation Rate (%)',  
              'Region.1', 'longitude', 'latitude'],  
              dtype='object')
```

```
In [25]: 1 df2.isnull().sum()
```

```
Out[25]: Region          0  
         Date            0  
         Frequency       0  
         Estimated Unemployment Rate (%)  0  
         Estimated Employed  0  
         Estimated Labour Participation Rate (%)  0  
         Region.1        0  
         longitude       0  
         latitude       0  
         dtype: int64
```

```
In [26]: 1 df2.rename(columns=lambda x:x.strip(), inplace=True) # removing spaces
```

```
In [27]: 1 df2.columns
```

```
Out[27]: Index(['Region', 'Date', 'Frequency', 'Estimated Unemployment Rate (%)',  
              'Estimated Employed', 'Estimated Labour Participation Rate (%)',  
              'Region.1', 'longitude', 'latitude'],  
              dtype='object')
```



```
In [28]: 1 df2['Region'].value_counts()
```

```
Out[28]: Andhra Pradesh      10
          Assam              10
          Uttarakhand        10
          Uttar Pradesh      10
          Tripura             10
          Telangana           10
          Tamil Nadu          10
          Rajasthan          10
          Punjab              10
          Puducherry          10
          Odisha              10
          Meghalaya           10
          Maharashtra         10
          Madhya Pradesh      10
          Kerala              10
          Karnataka           10
          Jharkhand           10
          Himachal Pradesh    10
          Haryana             10
          Gujarat             10
          Goa                 10
          Delhi               10
          Chhattisgarh        10
          Bihar               10
          West Bengal          10
          Jammu & Kashmir      9
          Sikkim               8
          Name: Region, dtype: int64
```

```
In [29]: 1 df2['Frequency'].value_counts()
```

```
Out[29]: M      267
          Name: Frequency, dtype: int64
```

```
In [30]: 1 df2['Region.1'].value_counts()
```

```
Out[30]: North      79
          South      60
          West       50
          East       40
          Northeast  38
          Name: Region.1, dtype: int64
```

Analysis of DataFrame 1

In [31]:

```
1 df1
```

Out[31]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Area
0	Andhra Pradesh	31-05-2019	Month	3.65	11999139.0	43.24	Rural
1	Andhra Pradesh	30-06-2019	Month	3.05	11755881.0	42.05	Rural
2	Andhra Pradesh	31-07-2019	Month	3.75	12086707.0	43.50	Rural
3	Andhra Pradesh	31-08-2019	Month	3.32	12285693.0	43.97	Rural
4	Andhra Pradesh	30-09-2019	Month	5.17	12256762.0	44.68	Rural
...
749	West Bengal	29-02-2020	Month	7.55	10871168.0	44.09	Urban
750	West Bengal	31-03-2020	Month	6.67	10806105.0	43.34	Urban
751	West Bengal	30-04-2020	Month	15.63	9299466.0	41.20	Urban
752	West Bengal	31-05-2020	Month	15.22	9240903.0	40.67	Urban
753	West Bengal	30-06-2020	Month	9.86	9088931.0	37.57	Urban

740 rows × 7 columns

In [32]:

```
1 df1.dtypes
```

Out[32]:

Region	object
Date	object
Frequency	object
Estimated Unemployment Rate (%)	float64
Estimated Employed	float64
Estimated Labour Participation Rate (%)	float64
Area	object
dtype:	object

```
In [33]: 1 df1['Date']=pd.to_datetime(df1['Date']) # converting date column from
2 df1['Months']=df1['Date'].dt.month_name() # creating a new column to id
3 df1
```

Out[33]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Area	Months
0	Andhra Pradesh	2019-05-31	Month	3.65	11999139.0	43.24	Rural	May
1	Andhra Pradesh	2019-06-30	Month	3.05	11755881.0	42.05	Rural	June
2	Andhra Pradesh	2019-07-31	Month	3.75	12086707.0	43.50	Rural	July
3	Andhra Pradesh	2019-08-31	Month	3.32	12285693.0	43.97	Rural	August
4	Andhra Pradesh	2019-09-30	Month	5.17	12256762.0	44.68	Rural	September
...
749	West Bengal	2020-02-29	Month	7.55	10871168.0	44.09	Urban	February
750	West Bengal	2020-03-31	Month	6.67	10806105.0	43.34	Urban	March
751	West Bengal	2020-04-30	Month	15.63	9299466.0	41.20	Urban	April
752	West Bengal	2020-05-31	Month	15.22	9240903.0	40.67	Urban	May
753	West Bengal	2020-06-30	Month	9.86	9088931.0	37.57	Urban	June

740 rows × 8 columns



```
In [34]: 1 df1.dtypes
```

```
Out[34]: Region                                object
Date                                datetime64[ns]
Frequency                            object
Estimated Unemployment Rate (%)      float64
Estimated Employed                    float64
Estimated Labour Participation Rate (%) float64
Area                                object
Months                              object
dtype: object
```

```
In [35]: 1 df1.select_dtypes(include='float64') # creating a dataframe consisting
```

Out[35]:

	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)
0	3.65	11999139.0	43.24
1	3.05	11755881.0	42.05
2	3.75	12086707.0	43.50
3	3.32	12285693.0	43.97
4	5.17	12256762.0	44.68
...
749	7.55	10871168.0	44.09
750	6.67	10806105.0	43.34
751	15.63	9299466.0	41.20
752	15.22	9240903.0	40.67
753	9.86	9088931.0	37.57

740 rows × 3 columns

```
In [36]: 1 df1.select_dtypes(include='float64').columns
```

Out[36]: Index(['Estimated Unemployment Rate (%)', 'Estimated Employed',
'Estimated Labour Participation Rate (%)'],
dtype='object')

```
In [37]: 1 # subsetting num_col from all the other cols  
2  
3 num_col=[]  
4 for col in df1.select_dtypes(include='float64').columns:  
5     num_col.append(col)  
6 num_col
```

Out[37]: ['Estimated Unemployment Rate (%)',
'Estimated Employed',
'Estimated Labour Participation Rate (%)']

```
In [38]: 1 df1.select_dtypes(include='object') # subsetting a dataframe consisting
```

Out[38]:

	Region	Frequency	Area	Months
0	Andhra Pradesh	Month	Rural	May
1	Andhra Pradesh	Month	Rural	June
2	Andhra Pradesh	Month	Rural	July
3	Andhra Pradesh	Month	Rural	August
4	Andhra Pradesh	Month	Rural	September
...
749	West Bengal	Month	Urban	February
750	West Bengal	Month	Urban	March
751	West Bengal	Month	Urban	April
752	West Bengal	Month	Urban	May
753	West Bengal	Month	Urban	June

740 rows × 4 columns

```
In [39]: 1 df1.select_dtypes(include='object').columns
```

Out[39]: Index(['Region', 'Frequency', 'Area', 'Months'], dtype='object')

```
In [40]: 1 # separating obj_col from all the other columns
2
3 obj_col=[]
4 for col in df1.select_dtypes(include='object').columns:
5     obj_col.append(col)
6 obj_col
```

Out[40]: ['Region', 'Frequency', 'Area', 'Months']

```
In [41]: 1 # group data by 'region' and take average of columns in float type colu
2 # reset the index column back to numbers
3
4 States=df1.groupby('Region')['Estimated Unemployment Rate (%)', 'Estima
5 States=pd.DataFrame(States).reset_index()
6 States
```

Out[41]:

	Region	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)
0	Andhra Pradesh	7.477143	8.154093e+06	39.375714
1	Assam	6.428077	5.354772e+06	44.868462
2	Bihar	18.918214	1.236619e+07	38.153929
3	Chandigarh	15.991667	3.168312e+05	39.336667
4	Chhattisgarh	9.240357	4.303499e+06	42.810714
5	Delhi	16.495357	2.627513e+06	38.929643
6	Goa	9.274167	2.263083e+05	39.249583
7	Gujarat	6.663929	1.140201e+07	46.101071
8	Haryana	26.283214	3.557072e+06	42.737143
9	Himachal Pradesh	18.540357	1.059824e+06	44.222143
10	Jammu & Kashmir	16.188571	1.799932e+06	41.030952
11	Jharkhand	20.585000	4.469240e+06	41.670714
12	Karnataka	6.676071	1.066712e+07	41.345357
13	Kerala	10.123929	4.425900e+06	34.867857
14	Madhya Pradesh	7.406429	1.111548e+07	38.821429
15	Maharashtra	7.557500	1.999020e+07	42.303214
16	Meghalaya	4.798889	6.897368e+05	57.080741
17	Odisha	5.657857	6.545747e+06	38.926429
18	Puducherry	10.215000	2.122781e+05	38.992692
19	Punjab	12.031071	4.539362e+06	41.138214
20	Rajasthan	14.058214	1.004106e+07	39.973214
21	Sikkim	7.249412	1.068807e+05	46.070000
22	Tamil Nadu	9.284286	1.226955e+07	40.872143
23	Telangana	7.737857	7.939663e+06	53.002500
24	Tripura	28.350357	7.170026e+05	61.823929
25	Uttar Pradesh	12.551429	2.809483e+07	39.432500
26	Uttarakhand	6.582963	1.390228e+06	33.775556
27	West Bengal	8.124643	1.719854e+07	45.417500

```
In [42]: 1 for i,j in enumerate(num_col):  
2         print(i,j)
```

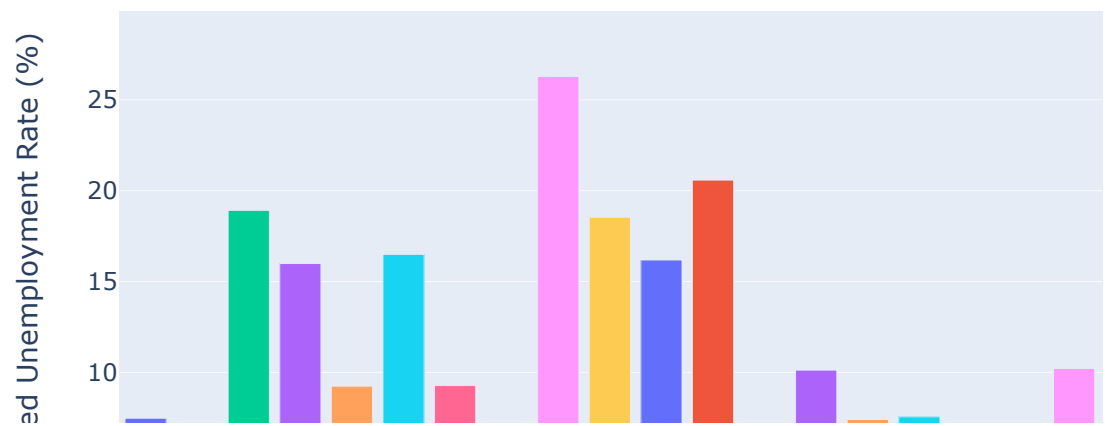
```
0 Estimated Unemployment Rate (%)  
1 Estimated Employed  
2 Estimated Labour Participation Rate (%)
```

```
In [43]: 1 for i,j in enumerate(obj_col):  
2         print(i,j)
```

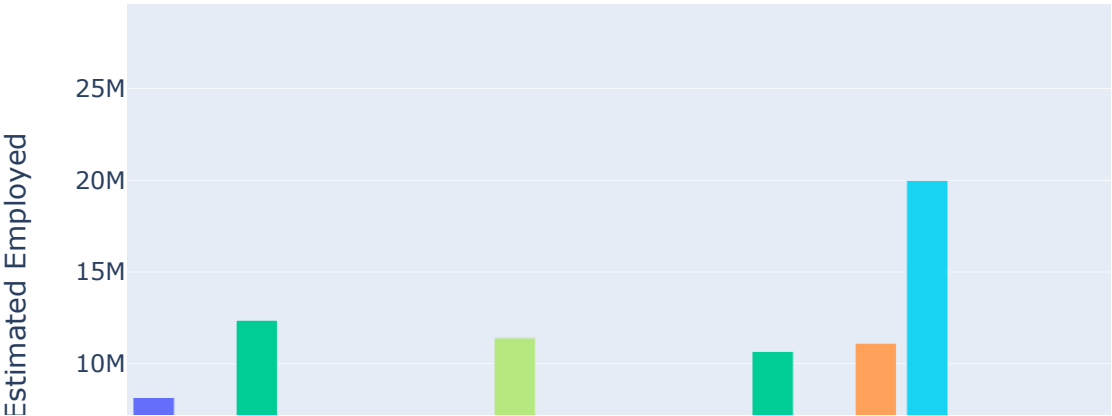
```
0 Region  
1 Frequency  
2 Area  
3 Months
```

```
In [44]: 1 # creating an interactive bar plot of all the float type columns
2
3 for i,j in enumerate(num_col):
4     fig=px.bar(States,x='Region', y=j, color='Region', title='Average '
5     fig.show()
```

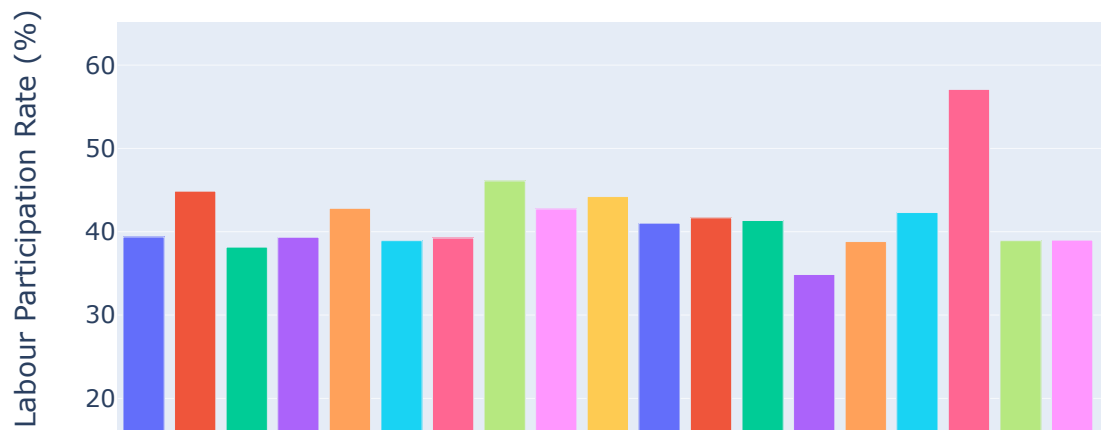
Average Estimated Unemployment Rate (%)



Average Estimated Employed



Average Estimated Labour Participation Rate (%)

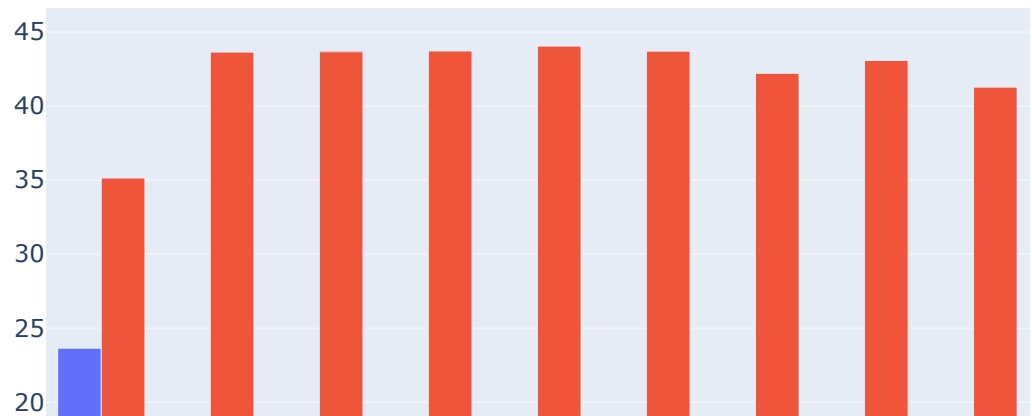


```
In [45]: 1 # group data by 'months' and take average of columns in float type columns
2 # reset the index column back to numbers
3
4 months=df1.groupby(['Months'])['Estimated Unemployment Rate (%)', 'Estimated Labour Participation Rate (%)'].mean()
5 months=pd.DataFrame(months).reset_index()
6 months.head()
```

Out[45]:

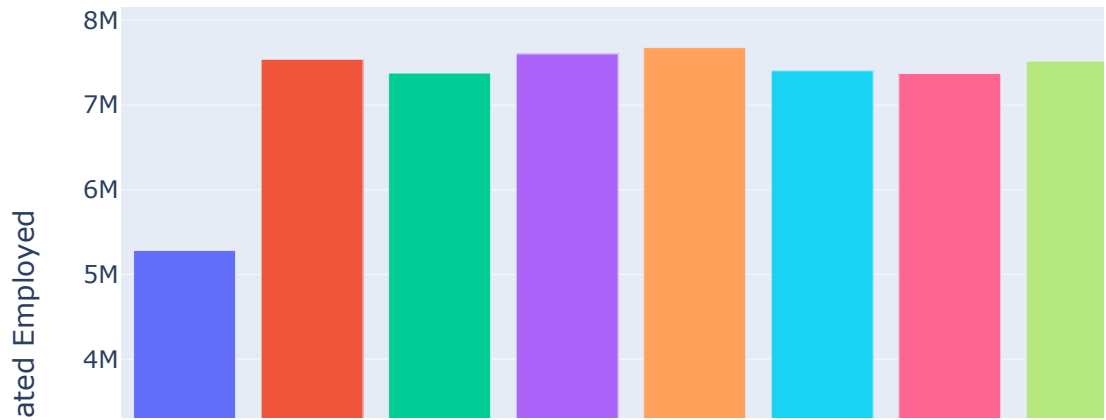
	Months	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)
0	April	23.641569	5.283320e+06	35.141176
1	August	9.637925	7.539815e+06	43.646792
2	December	9.497358	7.377388e+06	43.667358
3	February	9.964717	7.603996e+06	43.723019
4	January	9.950755	7.677344e+06	44.051321

```
In [46]: 1 # create a barplot for 'unemployment rate' and 'labour participation r
2
3 fig=go.Figure()
4 fig.add_trace(go.Bar(x=months['Months'],y=months['Estimated Unemploymer
5 fig.add_trace(go.Bar(x=months['Months'],y=months['Estimated Labour Part
6 fig.show()
```



```
In [47]: 1 fig=px.bar(months,x='Months',y='Estimated Employed', color='Months', ti
2 fig.show()
```

Monthly Average Estimation of Employment



In [48]:

1 df1

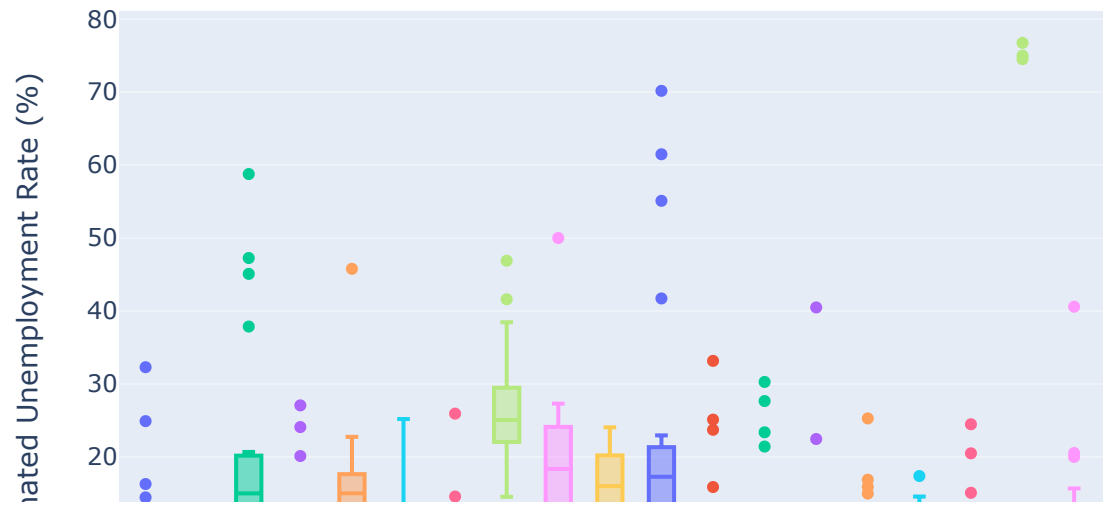
Out[48]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Area	Months
0	Andhra Pradesh	2019-05-31	Month	3.65	11999139.0	43.24	Rural	May
1	Andhra Pradesh	2019-06-30	Month	3.05	11755881.0	42.05	Rural	June
2	Andhra Pradesh	2019-07-31	Month	3.75	12086707.0	43.50	Rural	July
3	Andhra Pradesh	2019-08-31	Month	3.32	12285693.0	43.97	Rural	August
4	Andhra Pradesh	2019-09-30	Month	5.17	12256762.0	44.68	Rural	September
...
749	West Bengal	2020-02-29	Month	7.55	10871168.0	44.09	Urban	February
750	West Bengal	2020-03-31	Month	6.67	10806105.0	43.34	Urban	March
751	West Bengal	2020-04-30	Month	15.63	9299466.0	41.20	Urban	April
752	West Bengal	2020-05-31	Month	15.22	9240903.0	40.67	Urban	May
753	West Bengal	2020-06-30	Month	9.86	9088931.0	37.57	Urban	June

740 rows × 8 columns



```
In [49]: 1 fig=px.box(df1,x='Region',y='Estimated Unemployment Rate (%)',color='Re
2 fig.show()
```



In [50]:

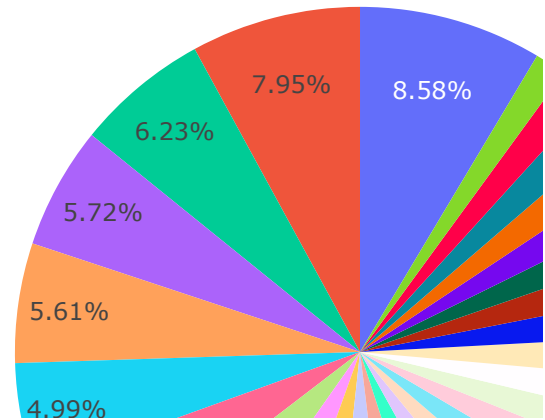
1	States
---	--------

Out[50]:

	Region	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)
0	Andhra Pradesh	7.477143	8.154093e+06	39.375714
1	Assam	6.428077	5.354772e+06	44.868462
2	Bihar	18.918214	1.236619e+07	38.153929
3	Chandigarh	15.991667	3.168312e+05	39.336667
4	Chhattisgarh	9.240357	4.303499e+06	42.810714
5	Delhi	16.495357	2.627513e+06	38.929643
6	Goa	9.274167	2.263083e+05	39.249583
7	Gujarat	6.663929	1.140201e+07	46.101071
8	Haryana	26.283214	3.557072e+06	42.737143
9	Himachal Pradesh	18.540357	1.059824e+06	44.222143
10	Jammu & Kashmir	16.188571	1.799932e+06	41.030952
11	Jharkhand	20.585000	4.469240e+06	41.670714
12	Karnataka	6.676071	1.066712e+07	41.345357
13	Kerala	10.123929	4.425900e+06	34.867857
14	Madhya Pradesh	7.406429	1.111548e+07	38.821429
15	Maharashtra	7.557500	1.999020e+07	42.303214
16	Meghalaya	4.798889	6.897368e+05	57.080741
17	Odisha	5.657857	6.545747e+06	38.926429
18	Puducherry	10.215000	2.122781e+05	38.992692
19	Punjab	12.031071	4.539362e+06	41.138214
20	Rajasthan	14.058214	1.004106e+07	39.973214
21	Sikkim	7.249412	1.068807e+05	46.070000
22	Tamil Nadu	9.284286	1.226955e+07	40.872143
23	Telangana	7.737857	7.939663e+06	53.002500
24	Tripura	28.350357	7.170026e+05	61.823929
25	Uttar Pradesh	12.551429	2.809483e+07	39.432500
26	Uttarakhand	6.582963	1.390228e+06	33.775556
27	West Bengal	8.124643	1.719854e+07	45.417500

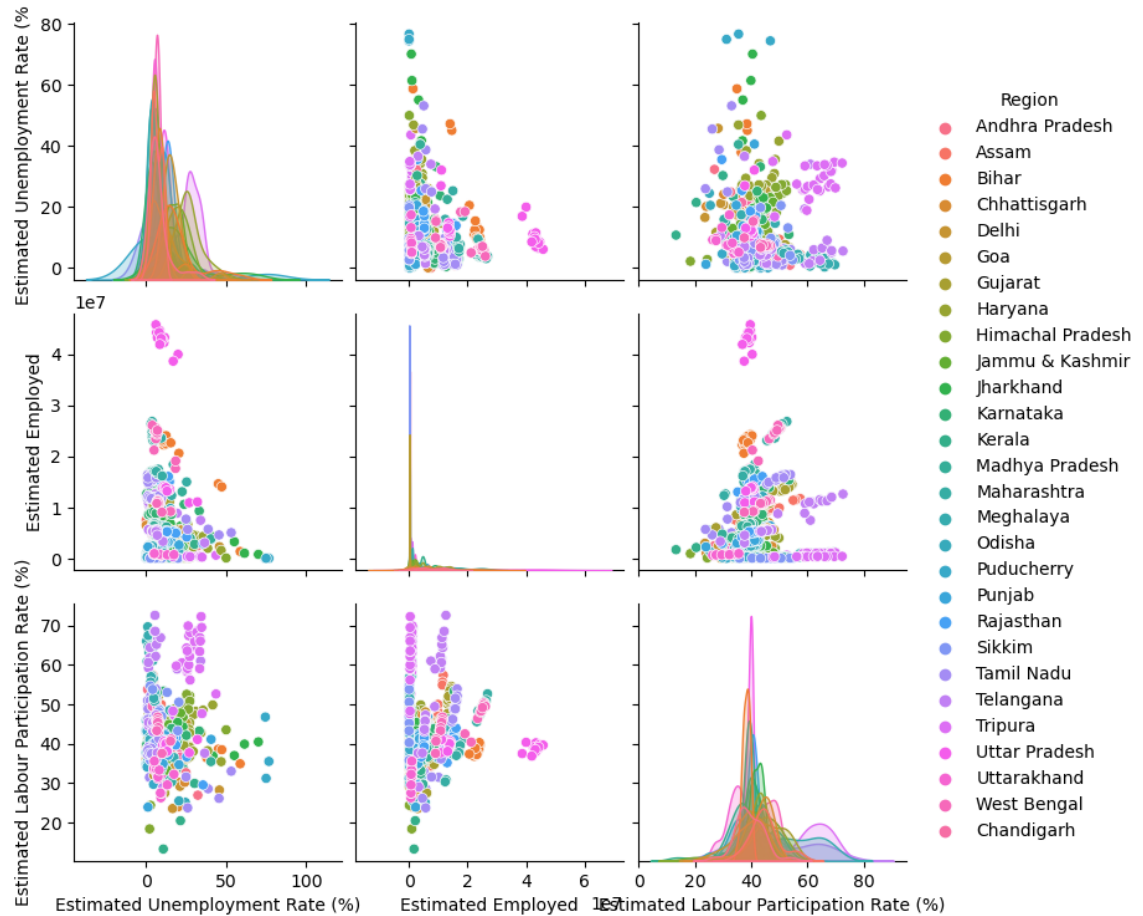
```
In [51]: 1 # pie plot showing the distribution of unemployment by region
2
3 fig=px.pie(States,values='Estimated Unemployment Rate (%)', names='Regi
4 fig.show()
```

Distribution of Estimated Unemployment Rate by Region



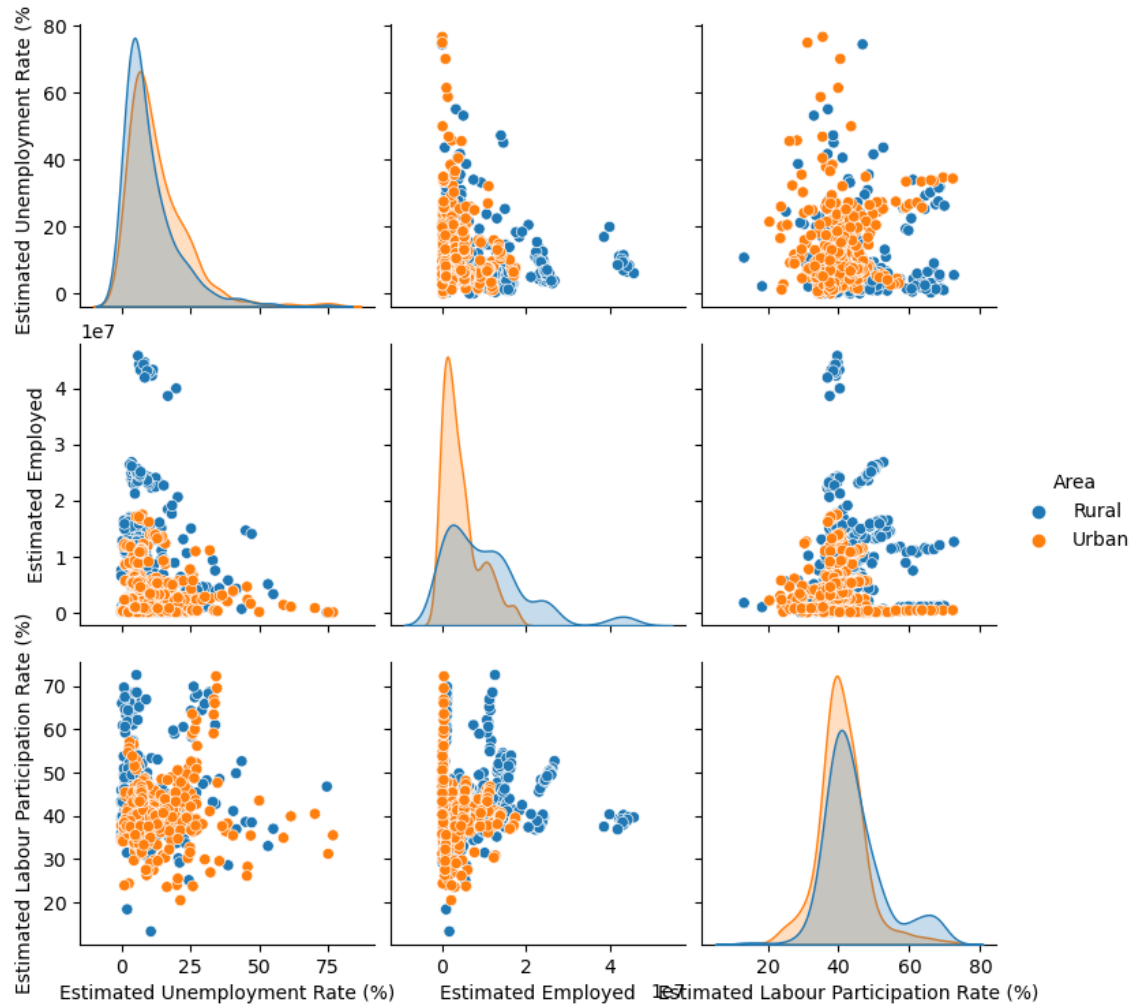

```
In [52]: 1 sns.pairplot(df1,hue='Region')
```

```
Out[52]: <seaborn.axisgrid.PairGrid at 0x24d6d1166d0>
```



```
In [53]: 1 sns.pairplot(df1,hue='Area')
```

```
Out[53]: <seaborn.axisgrid.PairGrid at 0x24d6da96e10>
```



Analysis of DataFrame 2

In [54]:

```
1 df2
```

Out[54]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Region.1	longitude
0	Andhra Pradesh	31-01-2020	M	5.48	16635535	41.02	South	15.9129
1	Andhra Pradesh	29-02-2020	M	5.83	16545652	40.90	South	15.9129
2	Andhra Pradesh	31-03-2020	M	5.79	15881197	39.18	South	15.9129
3	Andhra Pradesh	30-04-2020	M	20.51	11336911	33.10	South	15.9129
4	Andhra Pradesh	31-05-2020	M	17.43	12988845	36.46	South	15.9129
...
262	West Bengal	30-06-2020	M	7.29	30726310	40.39	East	22.9868
263	West Bengal	31-07-2020	M	6.83	35372506	46.17	East	22.9868
264	West Bengal	31-08-2020	M	14.87	33298644	47.48	East	22.9868
265	West Bengal	30-09-2020	M	9.35	35707239	47.73	East	22.9868
266	West Bengal	31-10-2020	M	9.98	33962549	45.63	East	22.9868

267 rows × 9 columns



In [55]:

```
1 df2.dtypes
```

Out[55]:

```
Region      object
Date        object
Frequency    object
Estimated Unemployment Rate (%)  float64
Estimated Employed                int64
Estimated Labour Participation Rate (%) float64
Region.1    object
longitude    float64
latitude     float64
dtype: object
```

```
In [56]: 1 df2['Date']=pd.to_datetime(df2['Date']) # converting date column with
2 df2['Months']=df2['Date'].dt.month_name() # creating a months column to
3 df2
```

Out[56]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Region.1	longitude
0	Andhra Pradesh	2020-01-31	M	5.48	16635535	41.02	South	15.9129
1	Andhra Pradesh	2020-02-29	M	5.83	16545652	40.90	South	15.9129
2	Andhra Pradesh	2020-03-31	M	5.79	15881197	39.18	South	15.9129
3	Andhra Pradesh	2020-04-30	M	20.51	11336911	33.10	South	15.9129
4	Andhra Pradesh	2020-05-31	M	17.43	12988845	36.46	South	15.9129
...
262	West Bengal	2020-06-30	M	7.29	30726310	40.39	East	22.9868
263	West Bengal	2020-07-31	M	6.83	35372506	46.17	East	22.9868
264	West Bengal	2020-08-31	M	14.87	33298644	47.48	East	22.9868
265	West Bengal	2020-09-30	M	9.35	35707239	47.73	East	22.9868
266	West Bengal	2020-10-31	M	9.98	33962549	45.63	East	22.9868

267 rows × 10 columns



```
In [57]: 1 df2.dtypes
```

```
Out[57]: Region                object
Date                datetime64[ns]
Frequency            object
Estimated Unemployment Rate (%)  float64
Estimated Employed    int64
Estimated Labour Participation Rate (%)  float64
Region.1             object
longitude            float64
latitude            float64
Months              object
dtype: object
```

```
In [58]: 1 num_col2=[]
2 for col in df2.select_dtypes(include='float64').columns:
3     num_col2.append(col)
4 num_col2
```

```
Out[58]: ['Estimated Unemployment Rate (%)',
'Estimated Labour Participation Rate (%)',
'longitude',
'latitude']
```

```
In [59]: 1 obj_col2=[]
2 for col in df2.select_dtypes(include='object').columns:
3     obj_col2.append(col)
4 obj_col2
```

```
Out[59]: ['Region', 'Frequency', 'Region.1', 'Months']
```

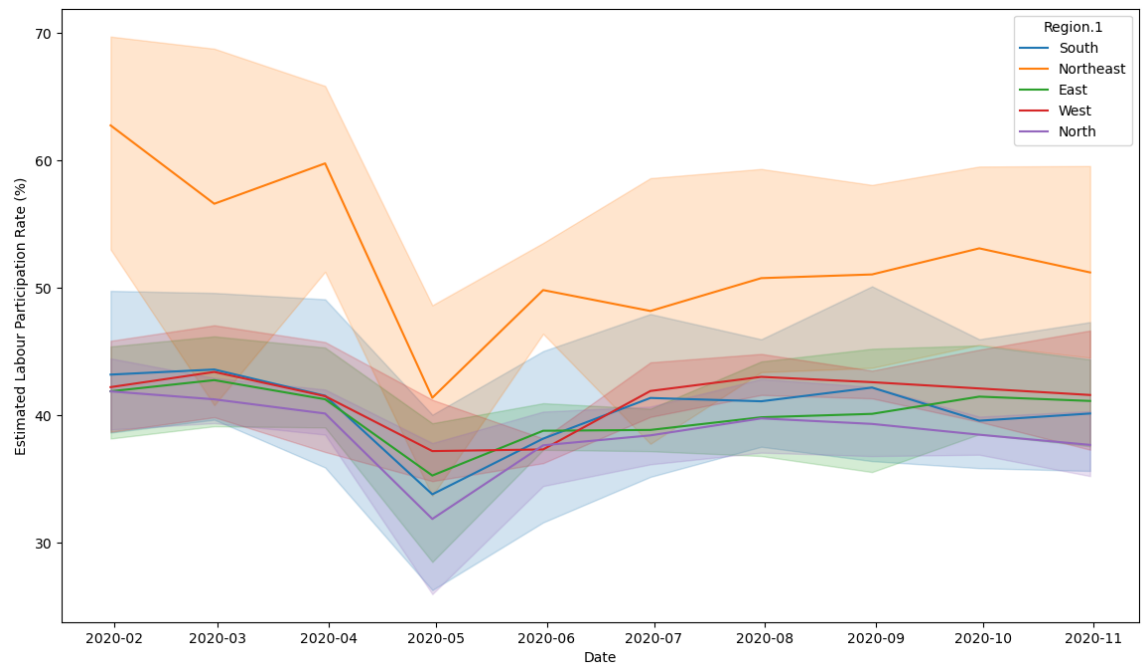
```
In [60]: 1 Regions=df2.groupby('Region.1')['Estimated Unemployment Rate (%)', 'Est
2 Regions=pd.DataFrame(Regions).reset_index()
3 Regions
```

```
Out[60]:
```

	Region.1	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)
0	East	13.916000	1.960237e+07	40.108750
1	North	15.889620	1.307249e+07	38.702658
2	Northeast	10.950263	3.617106e+06	52.055263
3	South	10.454667	1.404059e+07	40.436167
4	West	8.239000	1.862351e+07	41.257000

```
In [61]: 1 # plotting a line plot by region
2
3 plt.figure(figsize=(14,8))
4 sns.lineplot(x='Date',y='Estimated Labour Participation Rate (%)', hue=
```

Out[61]: <Axes: xlabel='Date', ylabel='Estimated Labour Participation Rate (%)'>



```
In [ ]: 1
```