

Unemployment analysis with python

Objective of Analysis: The primary aim of this analysis is to assess the repercussions of the COVID-19 pandemic on the labor market in India. We will utilize a comprehensive dataset encompassing the unemployment rates across all states in India. This dataset comprises essential variables, including States, Date, Frequency, Estimated Unemployment Rate (%), Estimated Employed, and Estimated Labour Participation Rate (%).

This dataset contains the unemployment rate of all the states in India

States = states in India
 Date = date which the unemployment rate observed
 Frequency = measuring frequency (Monthly)
 Estimated Unemployment Rate (%) = percentage of people unemployed in each States of India
 Estimated Employed = Number of people employed
 Estimated Labour Participation Rate (%) = The labour force participation rate is the portion of the working population in the 16-64 years' age group in the economy currently in employment or seeking employment.

Importing necessary libraries

```
In [1]: pip install chart-studio
```

```
Requirement already satisfied: chart-studio in c:\users\dell\anaconda3\lib\site-packages (1.1.0)
Requirement already satisfied: plotly in c:\users\dell\anaconda3\lib\site-packages (from chart-studio) (5.14.1)
Requirement already satisfied: requests in c:\users\dell\anaconda3\lib\site-packages (from chart-studio) (2.29.0)
Requirement already satisfied: retrying>=1.3.3 in c:\users\dell\anaconda3\lib\site-packages (from chart-studio) (1.3.3)
Requirement already satisfied: six in c:\users\dell\anaconda3\lib\site-packages (from chart-studio) (1.16.0)
Requirement already satisfied: tenacity>=6.2.0 in c:\users\dell\anaconda3\lib\site-packages (from plotly->chart-studio) (8.2.2)
Requirement already satisfied: packaging in c:\users\dell\anaconda3\lib\site-packages (from plotly->chart-studio) (23.0)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\dell\anaconda3\lib\site-packages (from requests->chart-studio) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\dell\anaconda3\lib\site-packages (from requests->chart-studio) (3.4)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\dell\anaconda3\lib\site-packages (from requests->chart-studio) (1.26.15)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\dell\anaconda3\lib\site-packages (from requests->chart-studio) (2023.5.7)
Note: you may need to restart the kernel to use updated packages.
```

```
In [2]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import datetime as dt
import calendar
```

```
import warnings
warnings.filterwarnings("ignore")
%matplotlib inline
```

In [3]: `data=pd.read_csv(r"D:/Unemployment_Rate_upto_11_2020.csv")
data.head()`

Out[3]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Region.1	longitude	latitude
0	Andhra Pradesh	31-01-2020	M	5.48	16635535	41.02	South	15.9129	79.74
1	Andhra Pradesh	29-02-2020	M	5.83	16545652	40.90	South	15.9129	79.74
2	Andhra Pradesh	31-03-2020	M	5.79	15881197	39.18	South	15.9129	79.74
3	Andhra Pradesh	30-04-2020	M	20.51	11336911	33.10	South	15.9129	79.74
4	Andhra Pradesh	31-05-2020	M	17.43	12988845	36.46	South	15.9129	79.74

◀ ▶

In [4]: `data.shape`

Out[4]: (267, 9)

In [5]: `data.columns`

Out[5]: `Index(['Region', 'Date', 'Frequency', 'Estimated Unemployment Rate (%)', 'Estimated Employed', 'Estimated Labour Participation Rate (%)', 'Region.1', 'longitude', 'latitude'], dtype='object')`

In [6]: `data.describe()`

Out[6]:

	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	longitude	latitude
count	267.000000	2.670000e+02	267.000000	267.000000	267.000000
mean	12.236929	1.396211e+07	41.681573	22.826048	80.532425
std	10.803283	1.336632e+07	7.845419	6.270731	5.831738
min	0.500000	1.175420e+05	16.770000	10.850500	71.192400
25%	4.845000	2.838930e+06	37.265000	18.112400	76.085600
50%	9.650000	9.732417e+06	40.390000	23.610200	79.019300
75%	16.755000	2.187869e+07	44.055000	27.278400	85.279900
max	75.850000	5.943376e+07	69.690000	33.778200	92.937600

```
In [7]: data.isnull().sum()
```

```
Out[7]: 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
```

Basic information about the dataset

```
In [8]: data.shape
```

```
Out[8]: (267, 9)
```

```
In [9]: data.duplicated().any()
```

```
Out[9]: False
```

```
In [10]: data.Region.value_counts()
```

```
Out[10]: 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
```

Rename the attributes

```
In [11]: #updating column names
```

```
data.columns=['state','date','frequency','estimated unemployment rate','estimated emp
```

In [12]: `data.tail()`

Out[12]:

	state	date	frequency	estimated unemployment rate	estimated employed	estimated labour participation rate	region	longitude	latitude
262	West Bengal	30-06-2020	M	7.29	30726310	40.39	East	22.9868	87.855
263	West Bengal	31-07-2020	M	6.83	35372506	46.17	East	22.9868	87.855
264	West Bengal	31-08-2020	M	14.87	33298644	47.48	East	22.9868	87.855
265	West Bengal	30-09-2020	M	9.35	35707239	47.73	East	22.9868	87.855
266	West Bengal	31-10-2020	M	9.98	33962549	45.63	East	22.9868	87.855



Changing the data type of date attribute

In [13]: `data['date']=pd.to_datetime(data['date'],dayfirst=True)`

Extracting month from the date attribute

In [14]: `data['month_int']=data['date'].dt.month
data.head()`

Out[14]:

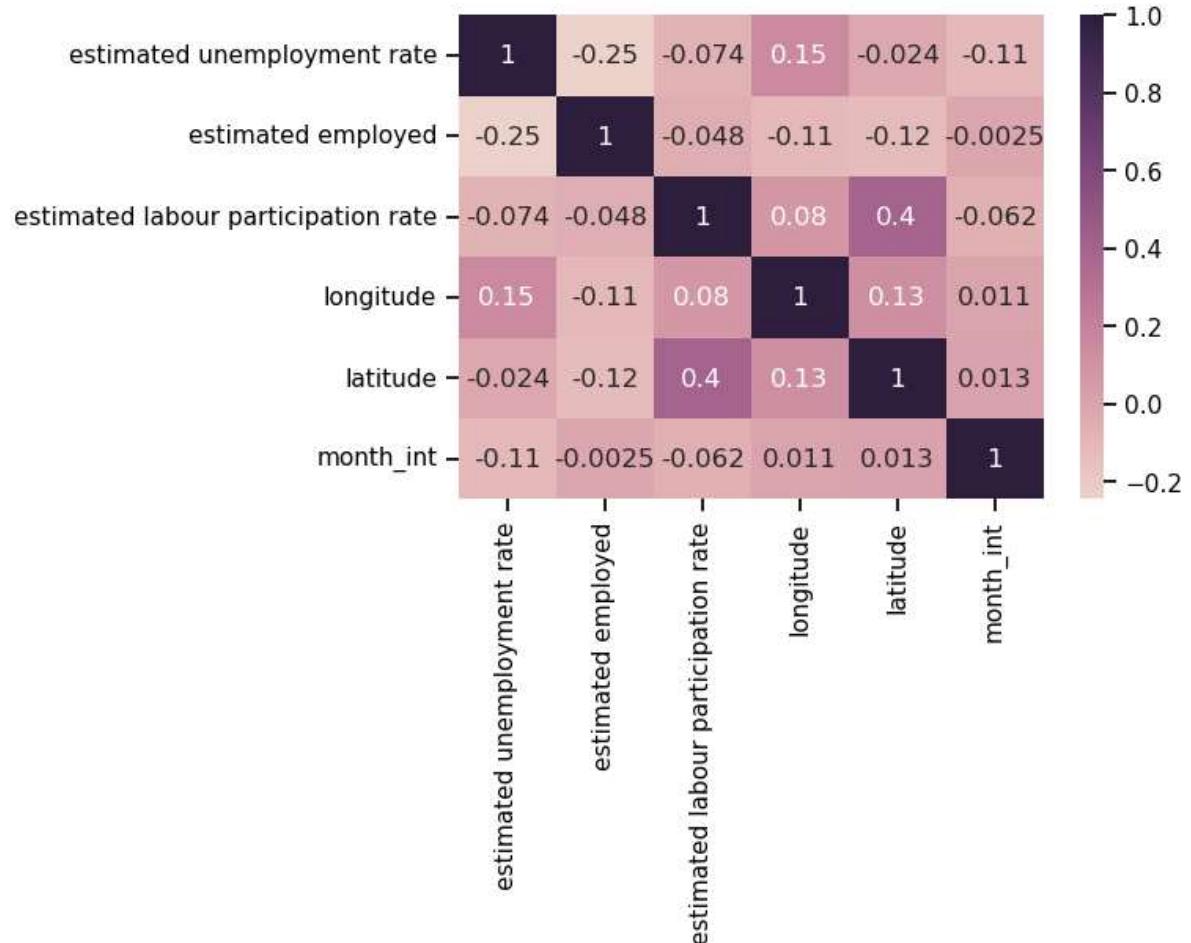
	state	date	frequency	estimated unemployment rate	estimated employed	estimated labour participation rate	region	longitude	latitude
0	Andhra Pradesh	2020-01-31	M	5.48	16635535	41.02	South	15.9129	79.74
1	Andhra Pradesh	2020-02-29	M	5.83	16545652	40.90	South	15.9129	79.74
2	Andhra Pradesh	2020-03-31	M	5.79	15881197	39.18	South	15.9129	79.74
3	Andhra Pradesh	2020-04-30	M	20.51	11336911	33.10	South	15.9129	79.74
4	Andhra Pradesh	2020-05-31	M	17.43	12988845	36.46	South	15.9129	79.74



In [15]: `heatmap= data[['estimated unemployment rate', 'estimated employed', 'estimated labour participation rate']]
heatmap = heatmap.corr()`

```
plt.figure(figsize=(6,4))
sns.set_context('notebook', font_scale=1)
sns.heatmap(data=heatmap, annot=True, cmap=sns.cubehelix_palette(as_cmap=True))
```

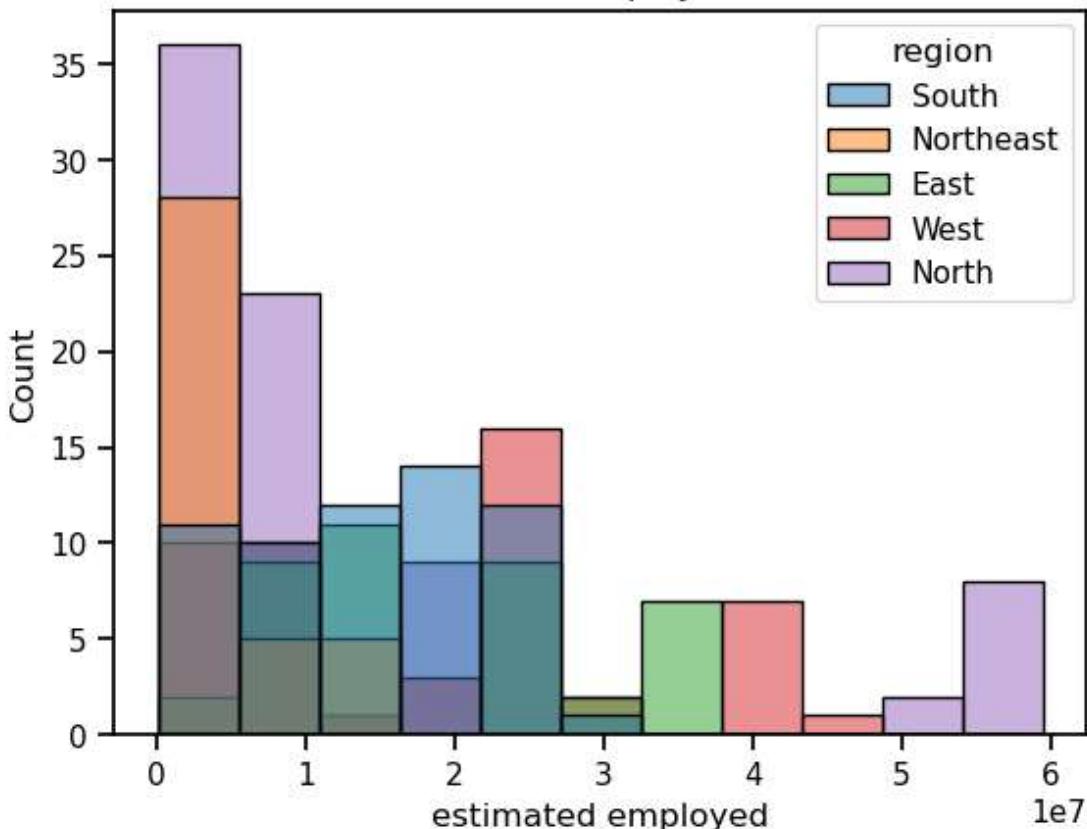
Out[15]: <Axes: >



In [16]: #The estimated number of employees according to different regions of India:

```
data.columns= ["state","date","frequency",
              "estimated unemployment rate","estimated employed",
              "estimated labour participation rate","region",
              "longitude","latitude","month_int"]
plt.title("Indian Unemployment")
sns.histplot(x="estimated employed", hue="region", data=data)
plt.show()
```

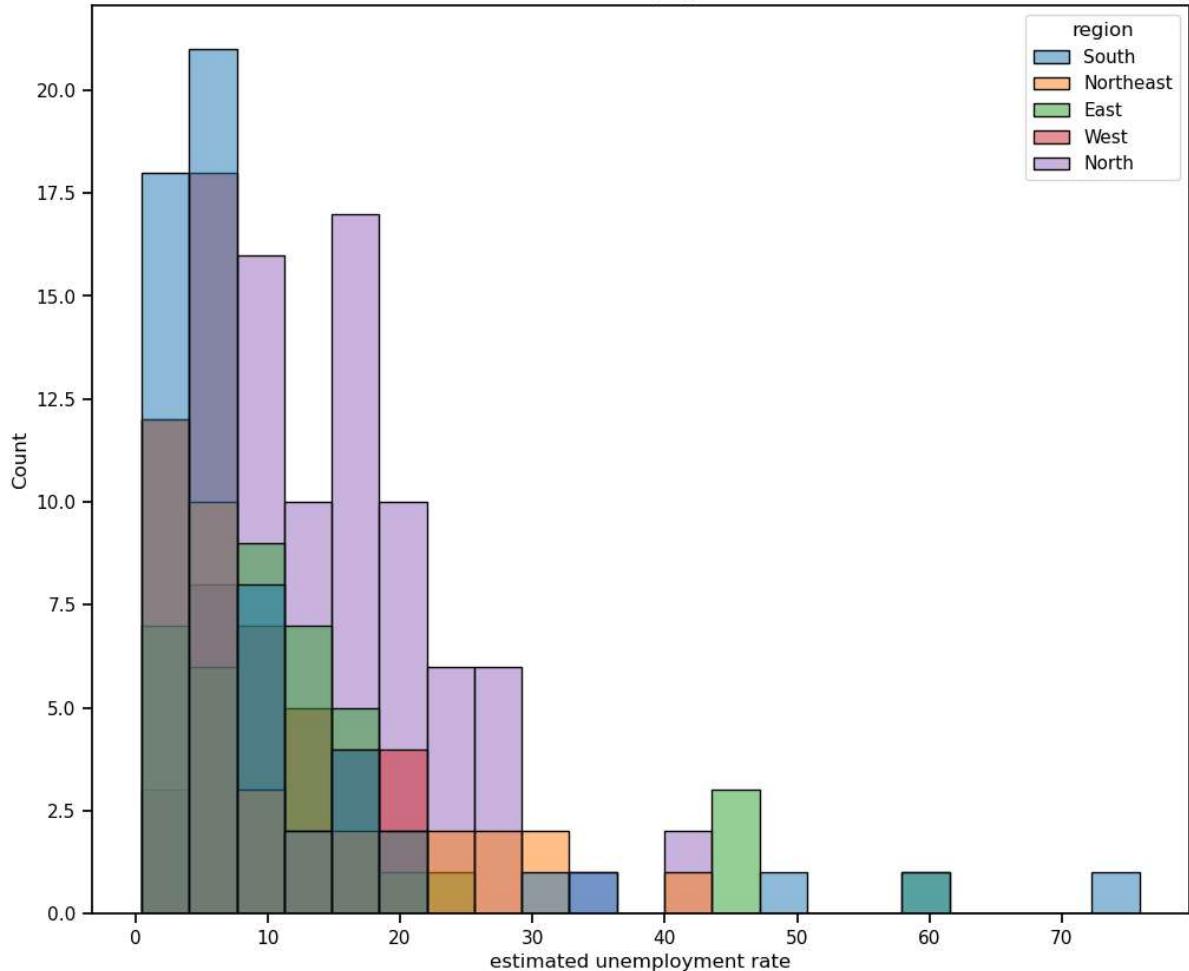
Indian Unemployment



Unemployment Rate Analysis: Data Visualization

```
In [17]: #The unemployment rate according to different regions of India:  
plt.figure(figsize=(12, 10))  
plt.title("Indian Unemployment")  
sns.histplot(x="estimated unemployment rate", hue="region", data=data)  
plt.show()
```

Indian Unemployment



```
In [18]: import plotly.express as px
from plotly.offline import init_notebook_mode, iplot
init_notebook_mode(connected=True)
```

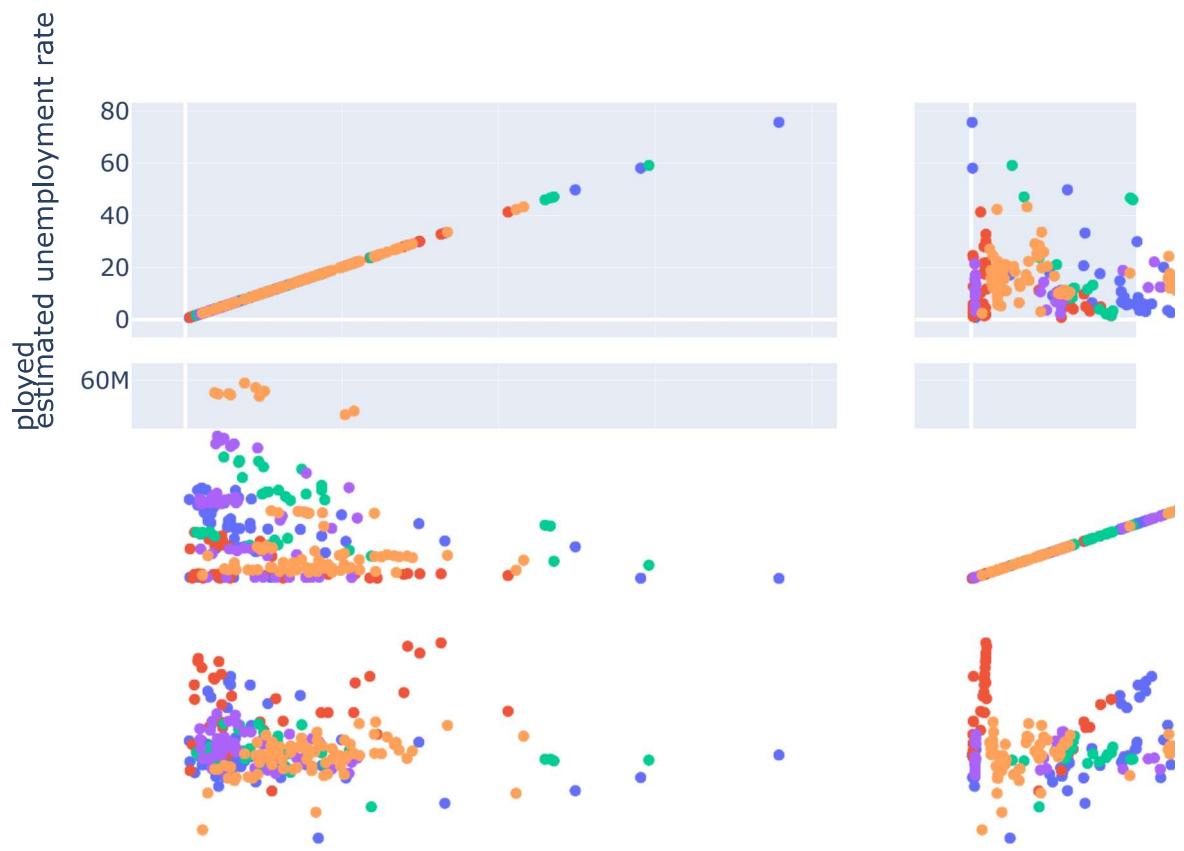
```
In [19]: #Boxplot of Unemployment rate per States
fig = px.box(data, x='state', y='estimated unemployment rate', color='state', title='

# Updating the x-axis category order to be in descending total
fig.update_layout(xaxis={'categoryorder': 'total descending'})
fig.show()
```

Unemployment Analysis

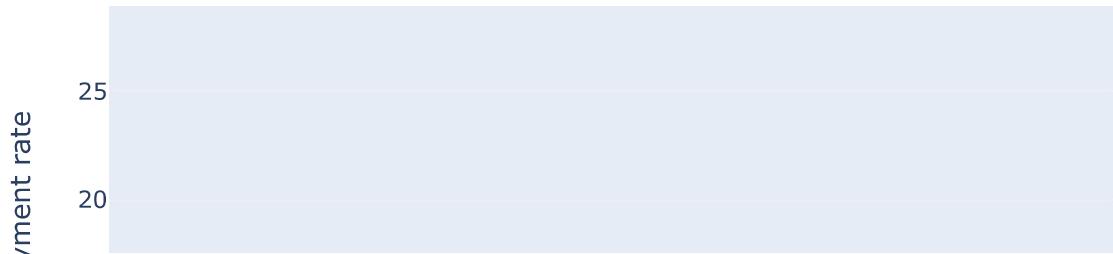


```
In [20]: #Scatter matrix showing relationships between unemployment rate, employment, and Labo
fig = px.scatter_matrix(data, template='plotly',
                         dimensions=['estimated unemployment rate', 'estimated employe
color='region')
fig.show()
```



```
In [21]: #Bar chart displaying the average unemployment rate in each state, color-coded by state
plot_ump = data[['estimated unemployment rate', 'state']]
data_unemp = plot_ump.groupby('state').mean().reset_index()
data_unemp = data_unemp.sort_values('estimated unemployment rate')
fig = px.bar(data_unemp, x='state', y='estimated unemployment rate', color='state',
              title='Average Unemployment Rate in each state', template='plotly')
fig.show()
```

Average Unemployment Rate in each state



```
In [22]: #Let's create a dashboard to analyze the unemployment rate of each Indian state by region
unemployment = data[["state", "region", "estimated unemployment rate"]]
figure = px.sunburst(unemployment, path=["region", "state"],
                     values="estimated unemployment rate",
                     width=700, height=700, color_continuous_scale="RdY1Gn",
                     title="Unemployment Rate in India")
figure.show()
```

Unemployment Rate in India



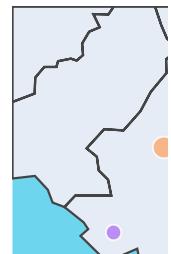
```
In [23]: fig = px.scatter_geo(data,'longitude', 'latitude', color="region",
                           hover_name="state", size="estimated unemployment rate",
                           animation_frame="month_int", scope='asia', template='plotly', title='')

fig.layout.updatemenus[0].buttons[0].args[1]["frame"]["duration"] = 2000

fig.update_geos(lataxis_range=[5,35], lonaxis_range=[65, 100],oceancolor="#6dd5ed",
                showocean=True)

fig.show()
```

Impact of lockdown on employment across regions



```
In [24]: #data representation before and after the Lockdown
after_lockdown = data[(data['month_int'] >= 4) & (data['month_int'] <=6)]
before_lockdown = data[(data['month_int'] >= 1) & (data['month_int'] <=4)]
```



```
In [25]: af_lockdown=after_lockdown.groupby('state')['estimated unemployment rate'].mean().reset_index()
lockdown= before_lockdown.groupby('state')['estimated unemployment rate'].mean().reset_index()
lockdown['unemployment rate before lockdown'] = af_lockdown['estimated unemployment rate']
lockdown.columns=['state','unemployment rate before lockdown','unemployment rate after lockdown']
lockdown.head()
```



```
Out[25]:
```

	state	unemployment rate before lockdown	unemployment rate after lockdown
0	Andhra Pradesh	9.4025	13.750000
1	Assam	6.2250	7.070000
2	Bihar	20.7425	36.806667
3	Chhattisgarh	7.2450	9.380000
4	Delhi	17.6975	25.713333

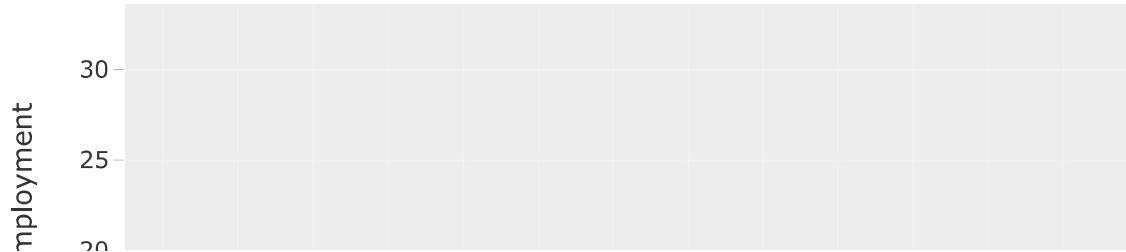
```
In [26]: #Unemployment rate change after Lockdown
lockdown['rate change in unemployment']=round(lockdown['unemployment rate before lockdown']/lockdown['unemployment rate after lockdown'])
```

```
In [27]: import plotly.express as px

fig = px.bar(lockdown, x='state', y='rate change in unemployment', color='rate change in unemployment',
             title='Percentage change in unemployment rate in each state after lockdown')
```

```
fig.update_layout(xaxis={'categoryorder':'total ascending'})  
fig.show()
```

Percentage change in unemployment



Most impacted States/Union Territories

- Puducherry
- Jharkhand
- Bihar
- Haryana
- Tripura

In []: