I have selected **Task 2**

P.S: **UNEMPLOYMENT ANALYSIS WITH PYTHON**

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.

**Step 1:**

Downloaded Dataset from: <https://www.kaggle.com/datasets/gokulrajkmv/unemployment-in-india>

I have selected file: 'Unemployment\_Rate\_upto\_11\_2020.csv for the analysis/

Aim of the Learning

* How Covid-19 affects the employment
* how far the unemployment rate will go

**Programing using Python**

**Importing Necessary Library**

import pandas as pd

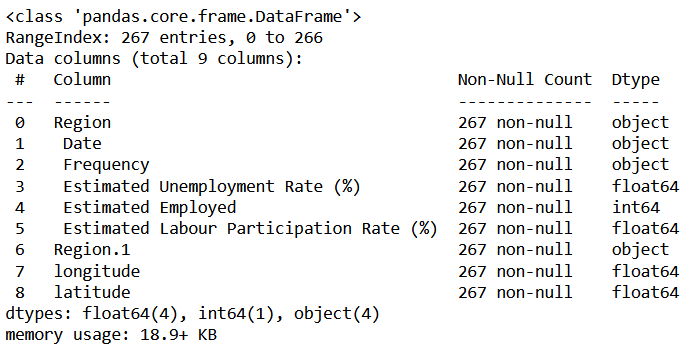
import numpy as np

import calender

**Loading CSV file**

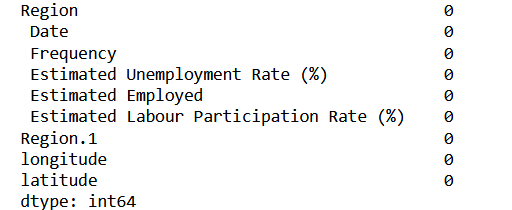
unemployment\_data=pd.read\_csv('Unemployment\_Rate\_upto\_11\_2020.csv')

unemployment\_data.info()



#**check for any null data**

print(unemployment\_data.isnull().sum())



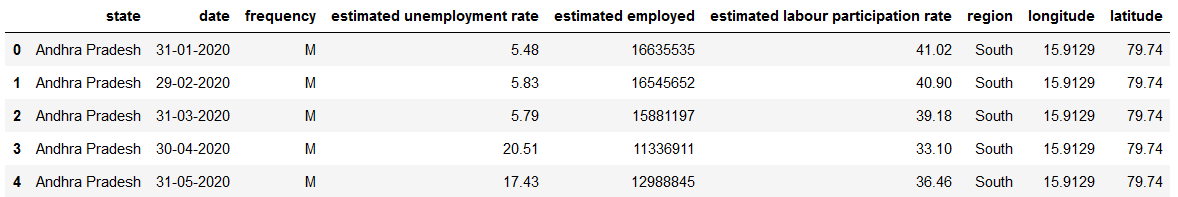
unemployment\_data.count

#**updating column names**

unemployment\_data.columns=['state', 'date', 'frequency', 'estimated unemployment rate', 'estimated employed', 'estimated labour participation rate', 'region', 'longitude', 'latitude']

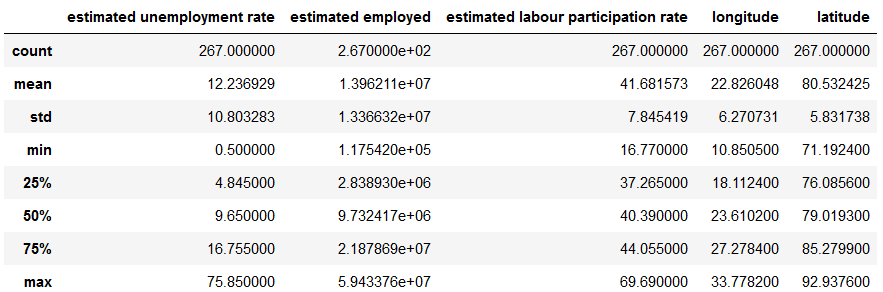
#**print with updated columns name**

unemployment\_data.head()



#**returns descriptive statistics about the data**

unemployment\_data.describe()



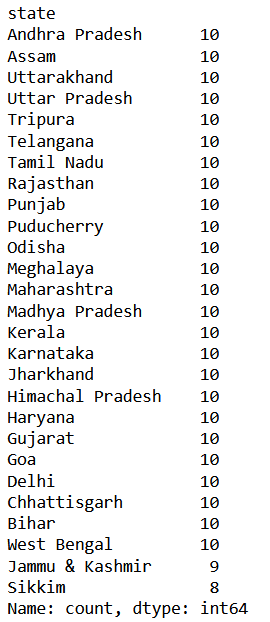
#**check whether any duplicate entry**

unemployment\_data.duplicated().any()

False

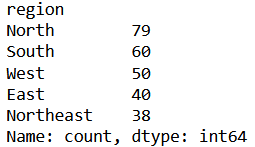
#**list out statewise data count**

unemployment\_data.state.value\_counts()



#**list out region wise data count**

unemployment\_data.region.value\_counts()

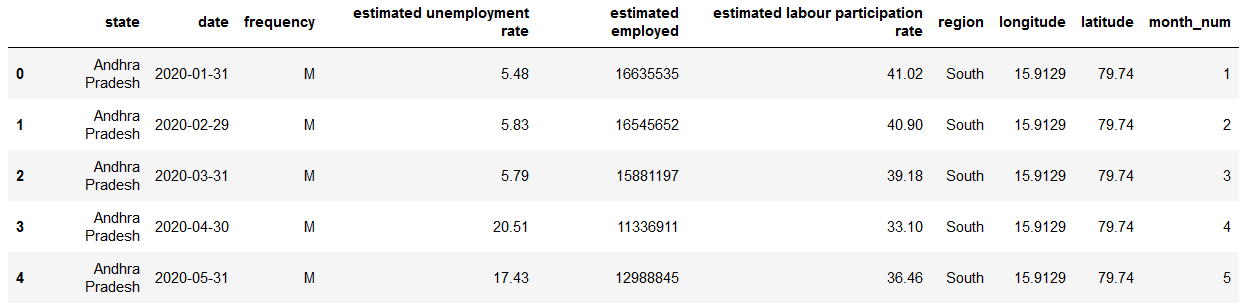


#**Changing the datatype of 'date' from object to datetime**

unemployment\_data['date']=pd.to\_datetime(unemployment\_data['date'],dayfirst=True)

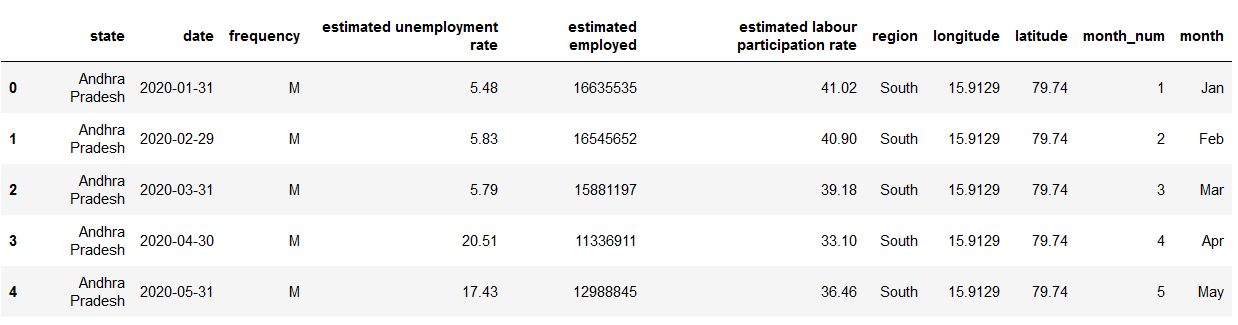
unemployment\_data['month\_num']=unemployment\_data['date'].dt.month

unemployment\_data.head()



unemployment\_data['month'] = unemployment\_data['month\_num'].apply(lambda x: calendar.month\_abbr[x])

unemployment\_data.head()



data = unemployment\_data.groupby(['month'])[['estimated unemployment rate','estimated employed','estimated labour participation rate']].mean()

data=pd.DataFrame(data).reset\_index()

**#plot Unemployment Rate, Labour Participation month wise**

import plotly.graph\_objects as go

month = data.month

unemployment\_rate = data['estimated unemployment rate']

labour\_participation\_rate = data['estimated labour participation rate']

#estimated\_employed\_data=data['estimated employed']

fig = go.Figure()

fig.add\_trace(go.Bar(x = month,y = unemployment\_rate,name = 'Unemployment Rate'))

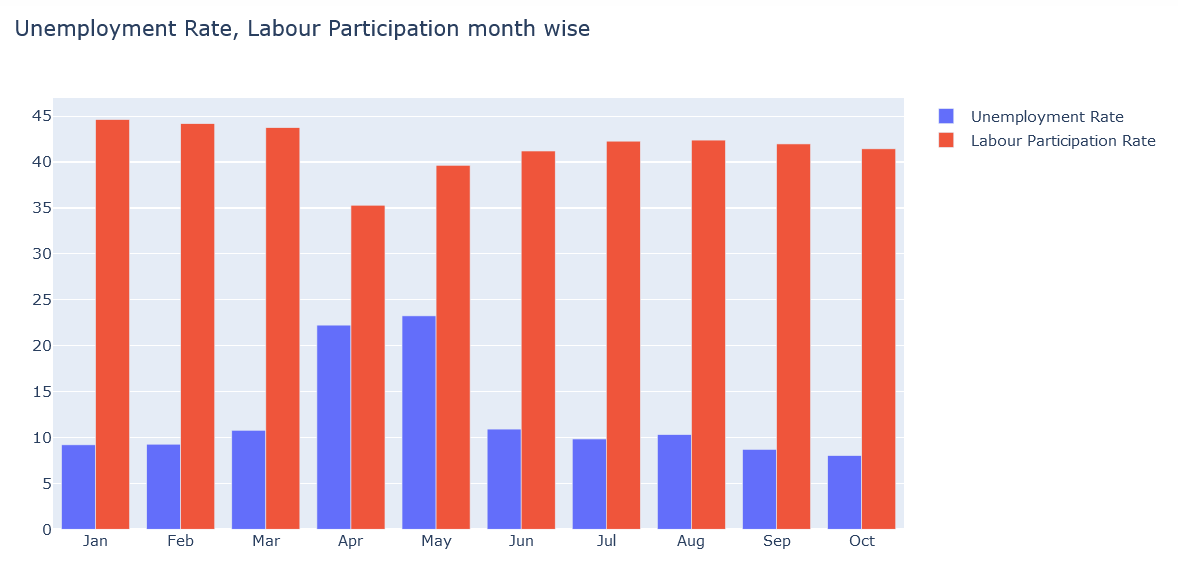
fig.add\_trace(go.Bar(x = month,y = labour\_participation\_rate,name = 'Labour Participation Rate'))

#fig.add\_trace(go.Bar(x = month,y = estimated\_employed\_data,name = 'Estimated Employed'))

fig.update\_layout(title = 'Unemployment Rate, Labour Participation month wise',

xaxis = {'categoryorder':'array','categoryarray':['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct']} )

fig.show()



import plotly.express as px

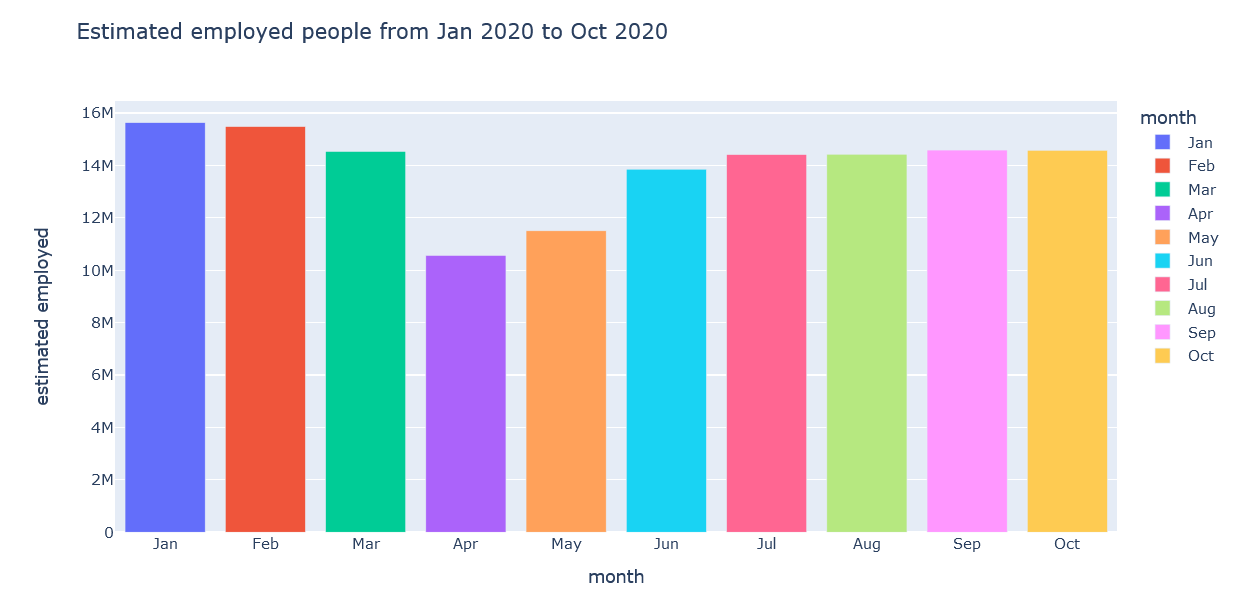
#**plot estimated employed month wise**

fig = px.bar(data, x='month', y='estimated employed', color='month',

category\_orders ={'month':['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct']},

title='Estimated employed people from Jan 2020 to Oct 2020')

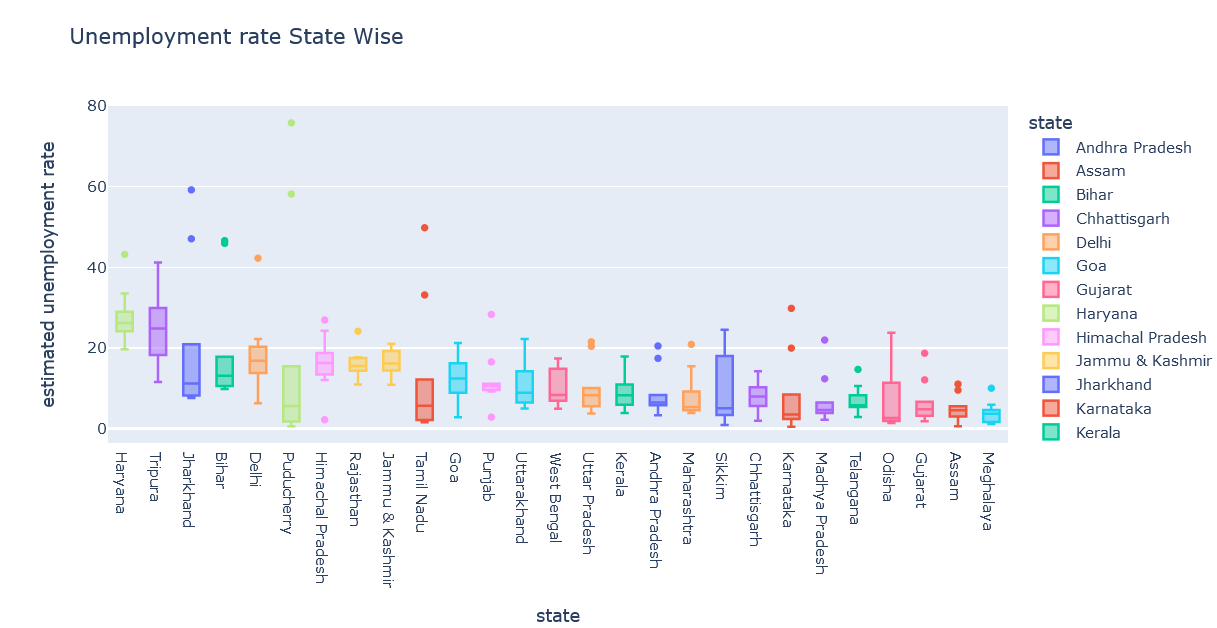
fig.show()



**#State wise Analysis**

state = unemployment\_data.groupby(['state'])[['estimated unemployment rate','estimated employed','estimated labour participation rate']].mean()

state = pd.DataFrame(state).reset\_index()

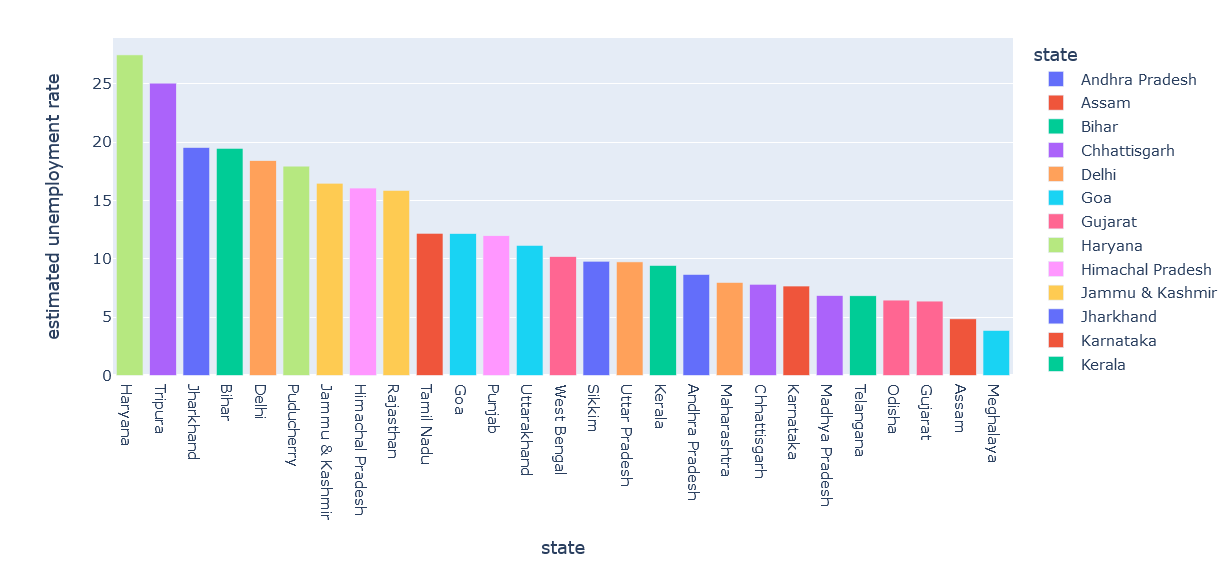


# Box plot

fig = px.box(data\_frame=unemployment\_data,x='state',y='estimated unemployment rate',color='state',title='Unemployment rate State Wise')

fig.update\_layout(xaxis={'categoryorder':'total descending'})

fig.show()



# **average unemployment rate using bar plot**

fig = px.bar(state,x='state',y='estimated unemployment rate',color='state',title='Average unemployment rate (State)')

fig.update\_layout(xaxis={'categoryorder':'total descending'})

fig.show()

**#Hariyana and Tripura were having the highest average amount of Unemployment rate**

**#Meghalaya was having the lowest average amount of Unemployment rate**

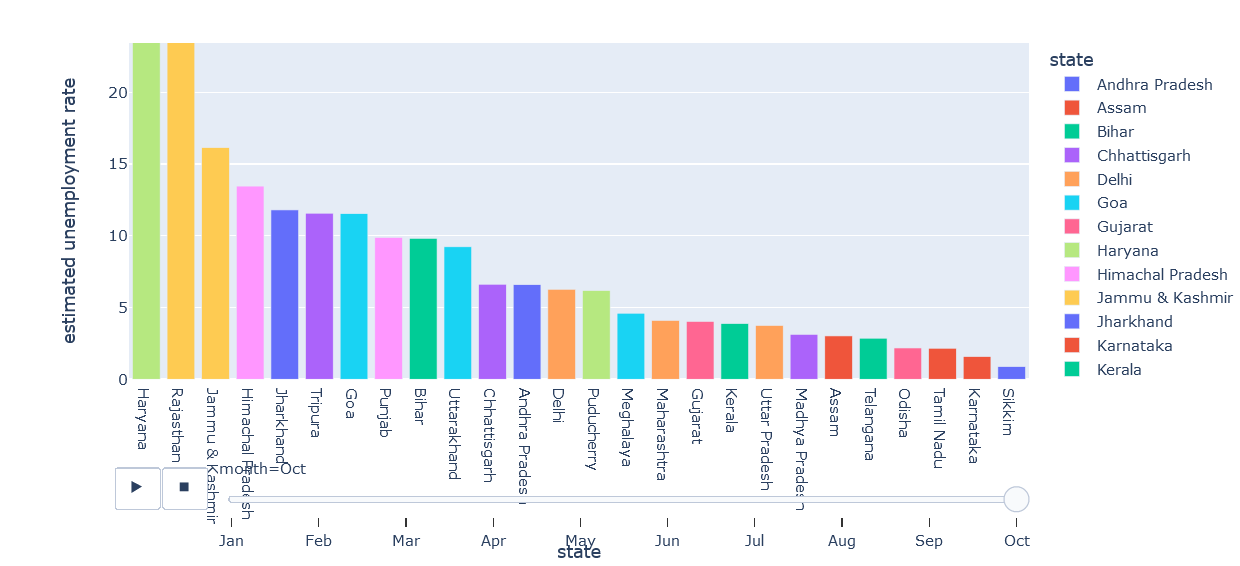
# Bar plot Unemployment Rate (monthly)

fig = px.bar(unemployment\_data,x='state',y='estimated unemployment rate',animation\_frame='month',color='state',

title='Unemployment rate from Jan 2020 to Oct 2020(State)')

fig.update\_layout(xaxis={'categoryorder':'total descending'})

fig.show()



**#We can see monthwise unemployement rate at various state from Jan to Oct (Monthly Unemployment Rate)**

**#Monthly unemployment rate, By clicking play button we can see statewise unemployment rate**

fig=px.scatter\_geo(unemployment\_data,'longitude','latitude',color='state',

hover\_name='state',size='estimated unemployment rate',

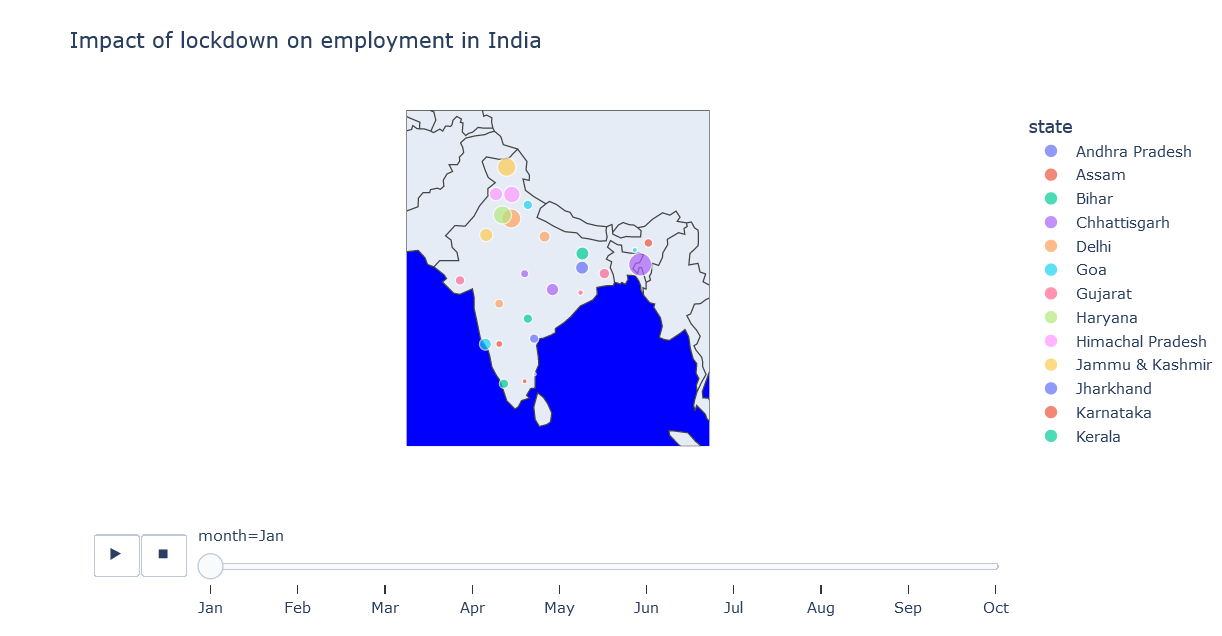
animation\_frame='month',scope='asia',title='Impact of lockdown on employment in India')

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

fig.update\_geos(lataxis\_range=[5,40],lonaxis\_range=[65,100],oceancolor='blue',

showocean=True)

fig.show()



**#Regional wise Analysis**

unemployment\_data.region.unique()

# data grouped by region

region = unemployment\_data.groupby(['region'])[['estimated unemployment rate','estimated employed','estimated labour participation rate']].mean()

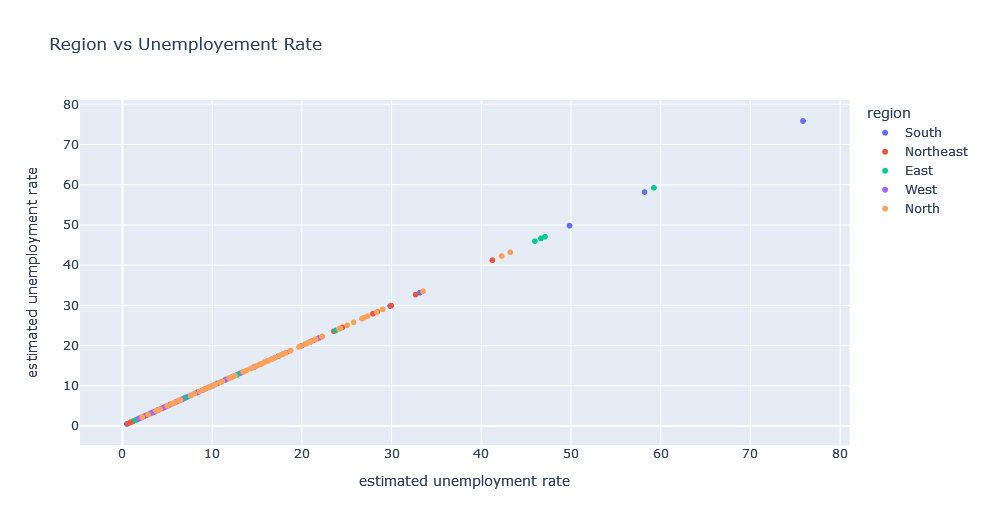
region = pd.DataFrame(region).reset\_index()

#Scatter plot

pd.DataFrame.iteritems = pd.DataFrame.items

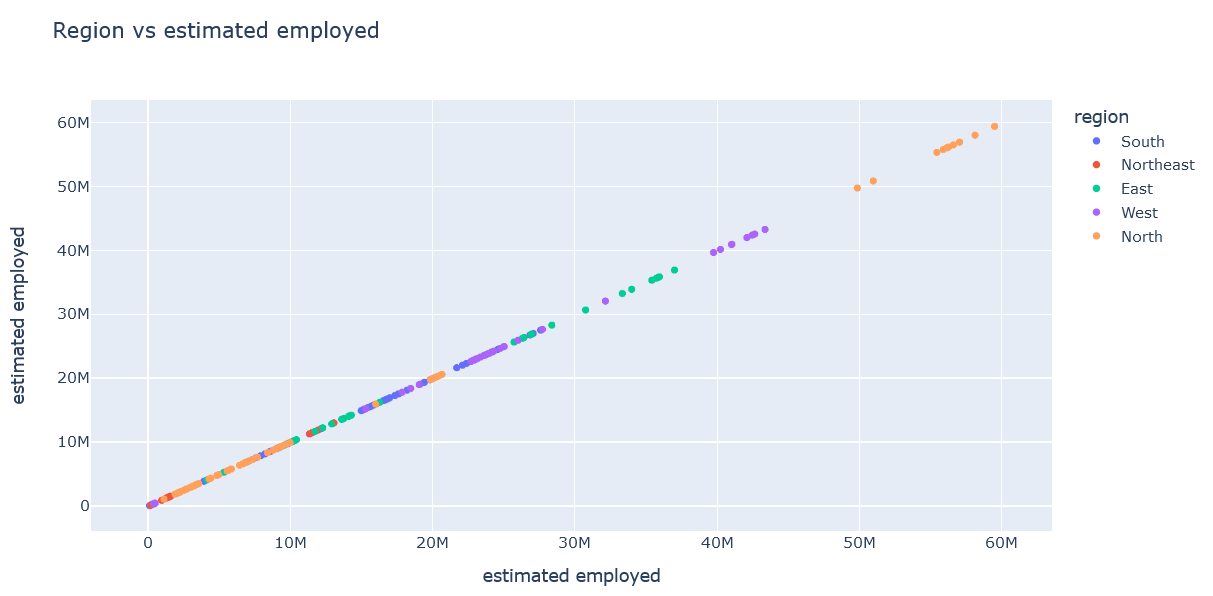
fig= px.scatter\_matrix(unemployment\_data,dimensions=['estimated unemployment rate'],color='region',title="Region vs Unemployement Rate")

fig.show()



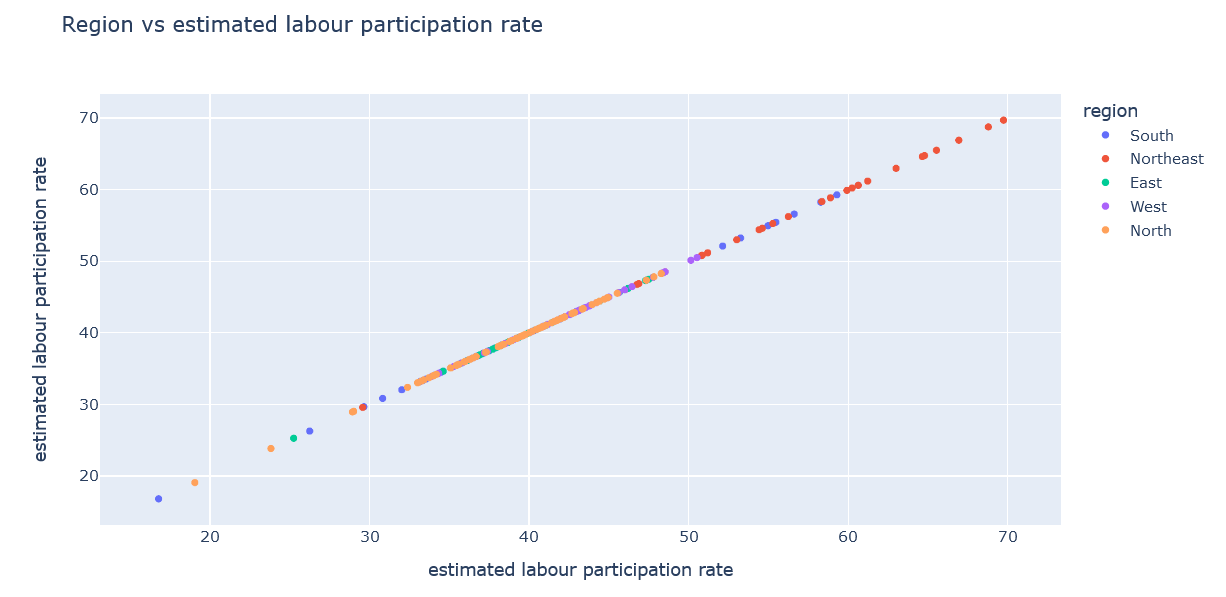
fig= px.scatter\_matrix(unemployment\_data,dimensions=['estimated employed'],color='region',title="Region vs estimated employed")

fig.show()



fig= px.scatter\_matrix(unemployment\_data,dimensions=['estimated labour participation rate'],color='region',title="Region vs estimated labour participation rate")

fig.show()



**# Average Unemployment Rate**

fig = px.bar(region,x='region',y='estimated unemployment rate',color='region',title='Average unemployment rate(region wise)')

fig.update\_layout(xaxis={'categoryorder':'total descending'})

fig.show()

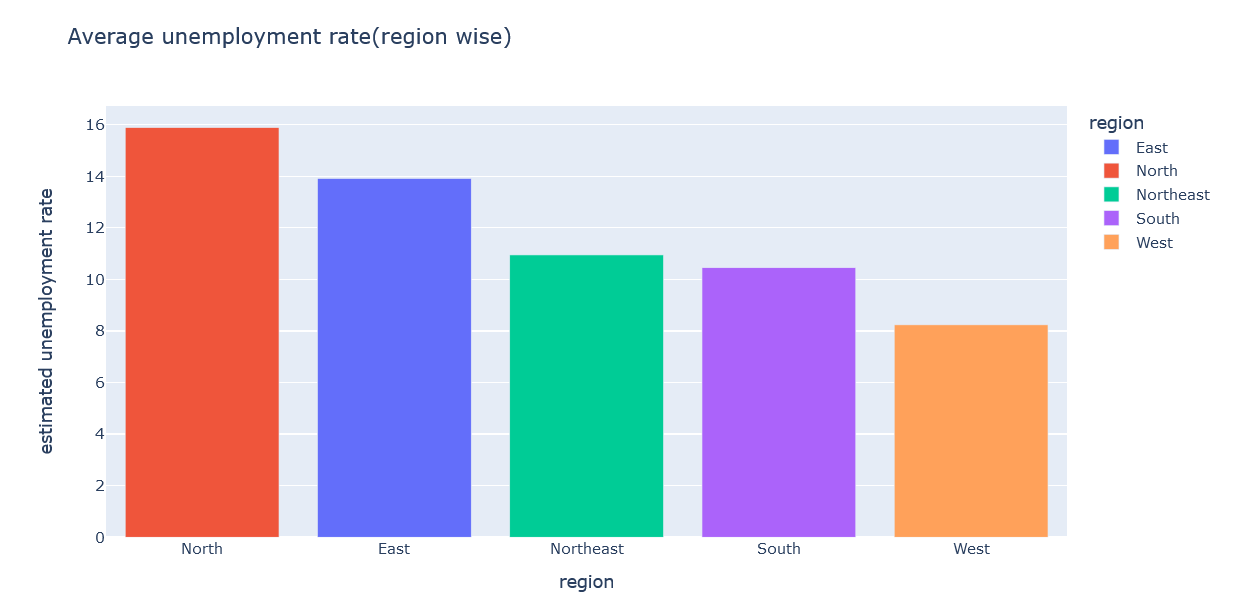


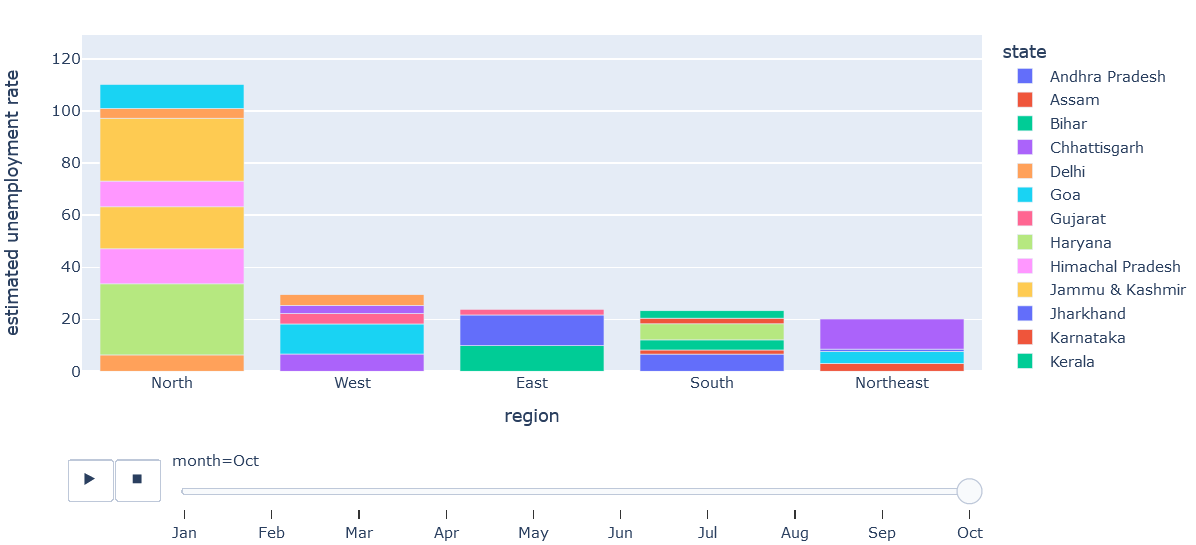
fig = px.bar(unemployment\_data,x='region',y='estimated unemployment rate',animation\_frame='month',color='state',

title='Unemployment rate from Jan 2020 to Oct 2020')

fig.update\_layout(xaxis={'categoryorder':'total descending'})

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

fig.show()



**#Region and State wise estimated unemployment Rate**

unemployment =unemployment\_data.groupby(['region','state'])['estimated unemployment rate'].mean().reset\_index()

unemployment.head(300)

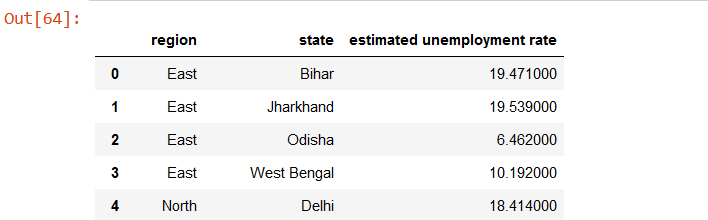
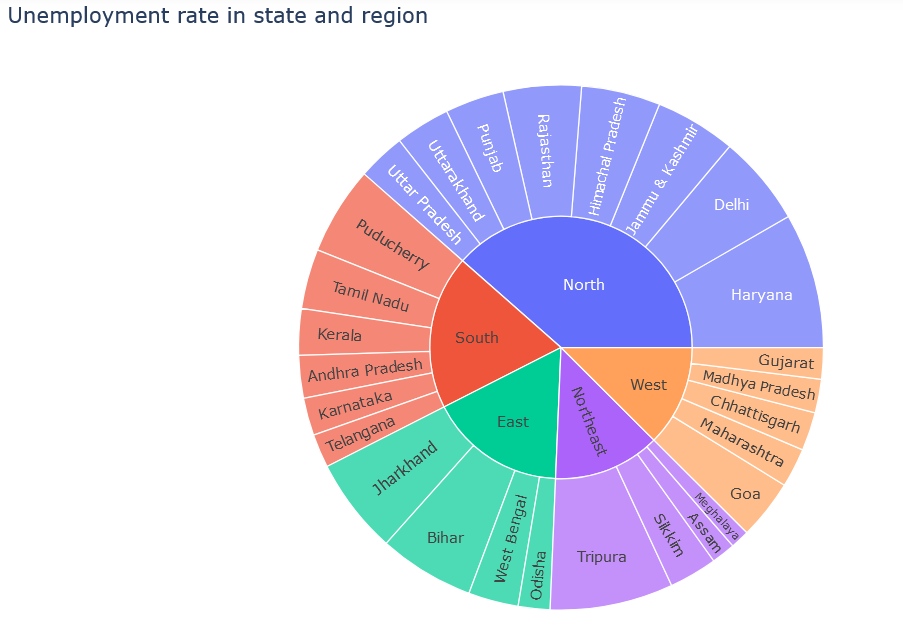


fig = px.sunburst(unemployment,path=['region','state'],values='estimated unemployment rate',

title ='Unemployment rate in state and region',height=600)

fig.show()



**# unemployment Rate before and after lockdown**

before\_lockdown = unemployment\_data[(unemployment\_data['month\_num']>=1) &(unemployment\_data['month\_num'] <4)]

after\_lockdown = unemployment\_data[(unemployment\_data['month\_num'] >=4) & (unemployment\_data['month\_num'] <=6)]

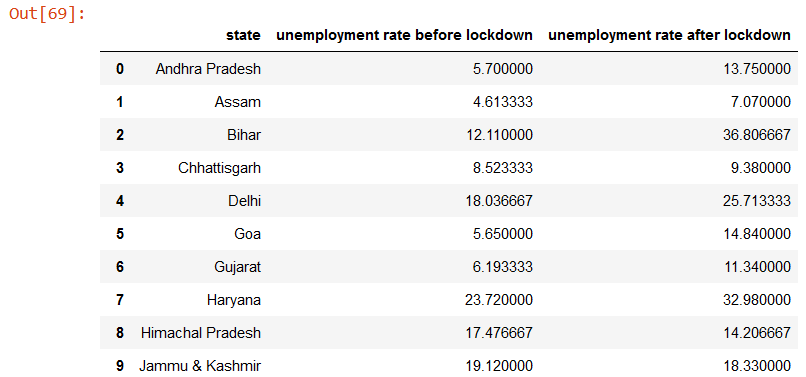
af\_lockdown = after\_lockdown.groupby('state')['estimated unemployment rate'].mean().reset\_index()

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(26)



**# unenployment rate change after lockdown**

lockdown['rate change in unemployment'] =round(lockdown['unemployment rate before lockdown']-lockdown['unemployment rate before lockdown']

/lockdown['unemployment rate after lockdown'],2)

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',template='ggplot2')

fig.update\_layout(xaxis={'categoryorder':'total ascending'})

fig.show()