Unemployment Analysis

Problem Statement

The price of a car depends on a lot of factors like the goodwill of the brand of the car, features of the car, horsepower and the mileage it gives and many more. Car price prediction is one of the major research areas in machine learning. So if you want to learn how to train a car price prediction model then this project is for you.

Step 1 :- Import Libraries and Dataset

In [1]:
```
# import required library
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
import plotly.express as px
import plotly.graph_objects as go
import seaborn as sns
import datetime as dt
import calendar
%matplotlib inline
```

In [2]:
```
df = pd.read_csv(r"C:\Users\admin\Desktop\Machine Learning\Oasis Infobyte Internship\Task 2\Unemployment_Rate_upto_11_2020.csv")
df.head()
```

Out[2]:

	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

Step 2 :- Data Pre-Processing

1. Data Type and conversion
2. Identifying & Treatment Missing Value
3. Identifying & Treatment Outliers
4. Descriptive Analysis

2.1 Data Type and Conversion

In [3]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 267 entries, 0 to 266
Data columns (total 9 columns):
 #   Column                                   Non-Null Count  Dtype  
---  ------                                   --------------  -----  
 0   Region                                   267 non-null    object 
 1    Date                                    267 non-null    object 
 2    Frequency                               267 non-null    object 
 3    Estimated Unemployment Rate (%)         267 non-null    float64
 4    Estimated Employed                      267 non-null    int64  
 5    Estimated Labour Participation Rate (%) 267 non-null    float64
 6   Region.1                                 267 non-null    object 
 7   longitude                                267 non-null    float64
 8   latitude                                 267 non-null    float64
dtypes: float64(4), int64(1), object(4)
memory usage: 18.9+ KB
```

2.2 Identifying and Treatment of Missing Values

In [4]:
```
# count the number of NaN values in each column
df.isnull().sum()
```

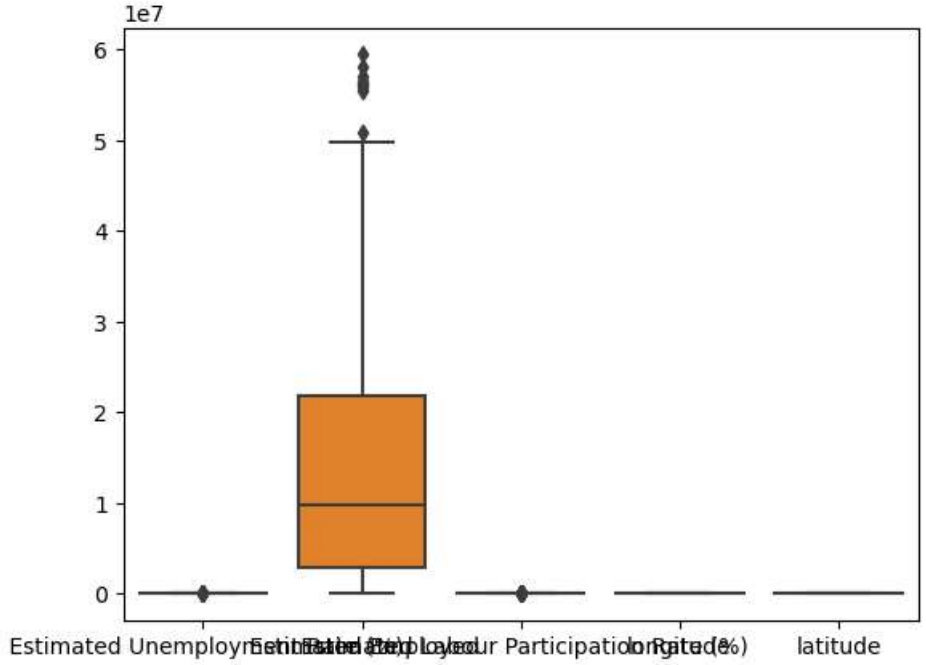
Out[4]:
```
Region                                   0
 Date                                    0
 Frequency                               0
 Estimated Unemployment Rate (%)         0
 Estimated Employed                      0
 Estimated Labour Participation Rate (%)    0
Region.1                                 0
longitude                                0
latitude                                 0
dtype: int64
```

- No Missing values present in dataset

2.3 Identifying And Treatment of Outliers

In [5]:
```
sns.boxplot(data = df)
```

Out[5]: <AxesSubplot:>



2.4 Descriptive Analysis

In [6]:
```
df.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]:
```
df.corr()
```

Out[7]:

	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	longitude	latitude
Estimated Unemployment Rate (%)	1.000000	-0.245176	-0.073540	0.149976	-0.023976
Estimated Employed	-0.245176	1.000000	-0.047948	-0.113664	-0.119321
Estimated Labour Participation Rate (%)	-0.073540	-0.047948	1.000000	0.080372	0.397836
longitude	0.149976	-0.113664	0.080372	1.000000	0.125895
latitude	-0.023976	-0.119321	0.397836	0.125895	1.000000

Step 3 : Exploratory Data Analysis

In [8]:
```
df.columns
```

Out[8]:
```
Index(['Region', ' Date', ' Frequency', ' Estimated Unemployment Rate (%)',
       ' Estimated Employed', ' Estimated Labour Participation Rate (%)',
       'Region.1', 'longitude', 'latitude'],
      dtype='object')
```

In [9]:
```
# create a new column for month

df['date'] = pd.to_datetime(df[' Date'], dayfirst=True)
df['month_int'] = pd.DatetimeIndex(df['date']).month
df['month'] =  df['month_int'].apply(lambda x: calendar.month_abbr[x])
df = df.drop(['month_int'],axis=1)
df.head()
```

Out[9]:

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

In [10]:
```
df.columns
```

Out[10]:
```
Index(['Region', ' Date', ' Frequency', ' Estimated Unemployment Rate (%)',
       ' Estimated Employed', ' Estimated Labour Participation Rate (%)',
       'Region.1', 'longitude', 'latitude', 'date', 'month'],
      dtype='object')
```

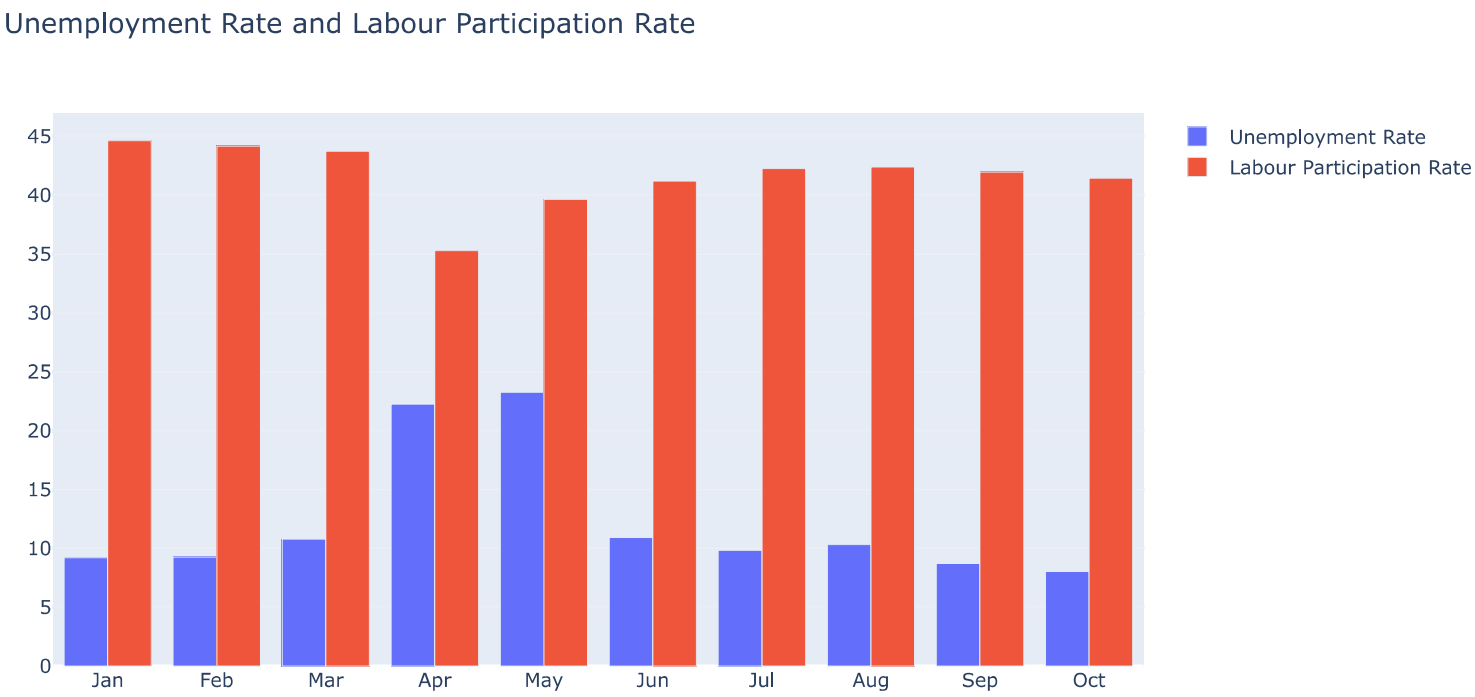
In [11]:
```
# Numeric data grouped by months
IND =  df.groupby(["month"])[[' Estimated Unemployment Rate (%)', ' Estimated Employed',
                              ' Estimated Labour Participation Rate (%)']].mean()
IND = pd.DataFrame(IND).reset_index()
IND.head()
```

Out[11]:

	month	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)
0	Apr	22.236154	1.057020e+07	35.297308
1	Aug	10.313333	1.442904e+07	42.390741
2	Feb	9.266154	1.548827e+07	44.180769
3	Jan	9.196538	1.563720e+07	44.626538
4	Jul	9.834444	1.441802e+07	42.274815

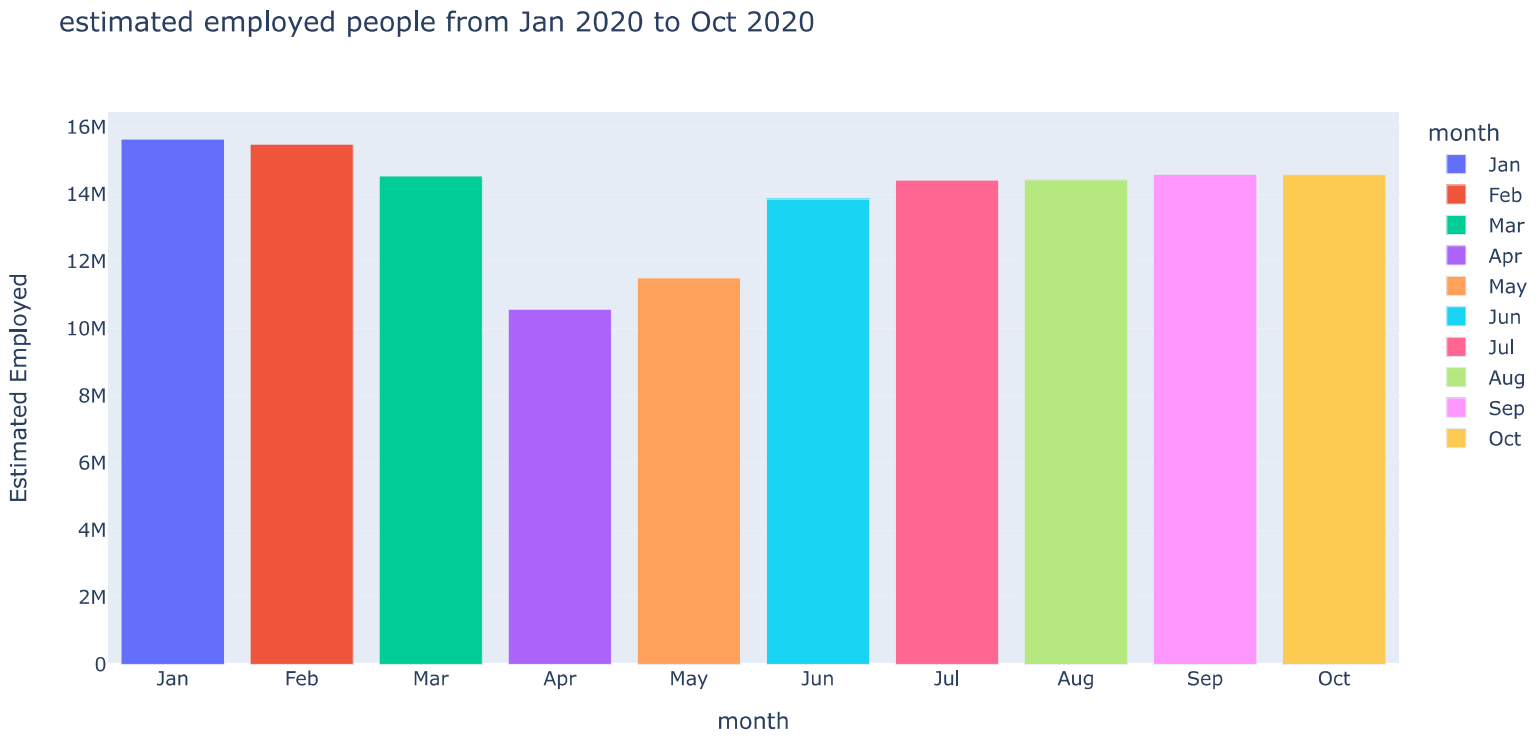
In [12]:
```
month = IND.month
unemployment_rate = IND[" Estimated Unemployment Rate (%)"]
labour_participation_rate = IND[' Estimated Labour Participation Rate (%)']
fig = go.Figure()
fig.add_trace(go.Bar(x = month, y = unemployment_rate, name= "Unemployment Rate"))
fig.add_trace(go.Bar(x = month, y = labour_participation_rate, name= "Labour Participation Rate"))

fig.update_layout(title="Unemployment Rate and Labour Participation Rate",
                  xaxis={"categoryorder":"array", "categoryarray":["Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct"]})
fig.show()
```



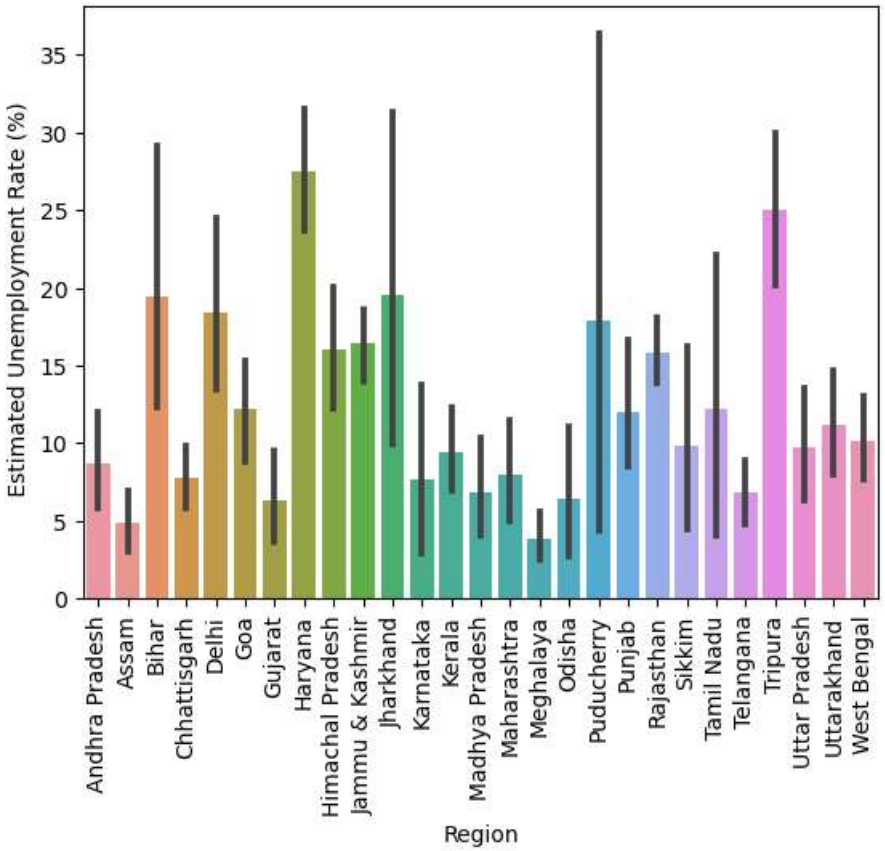
In [13]:
```
fig = px.bar(IND, x='month',y=' Estimated Employed', color='month',
             category_orders = {"month": ["Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct"]},
             title='estimated employed people from Jan 2020 to Oct 2020')

fig.show()
```



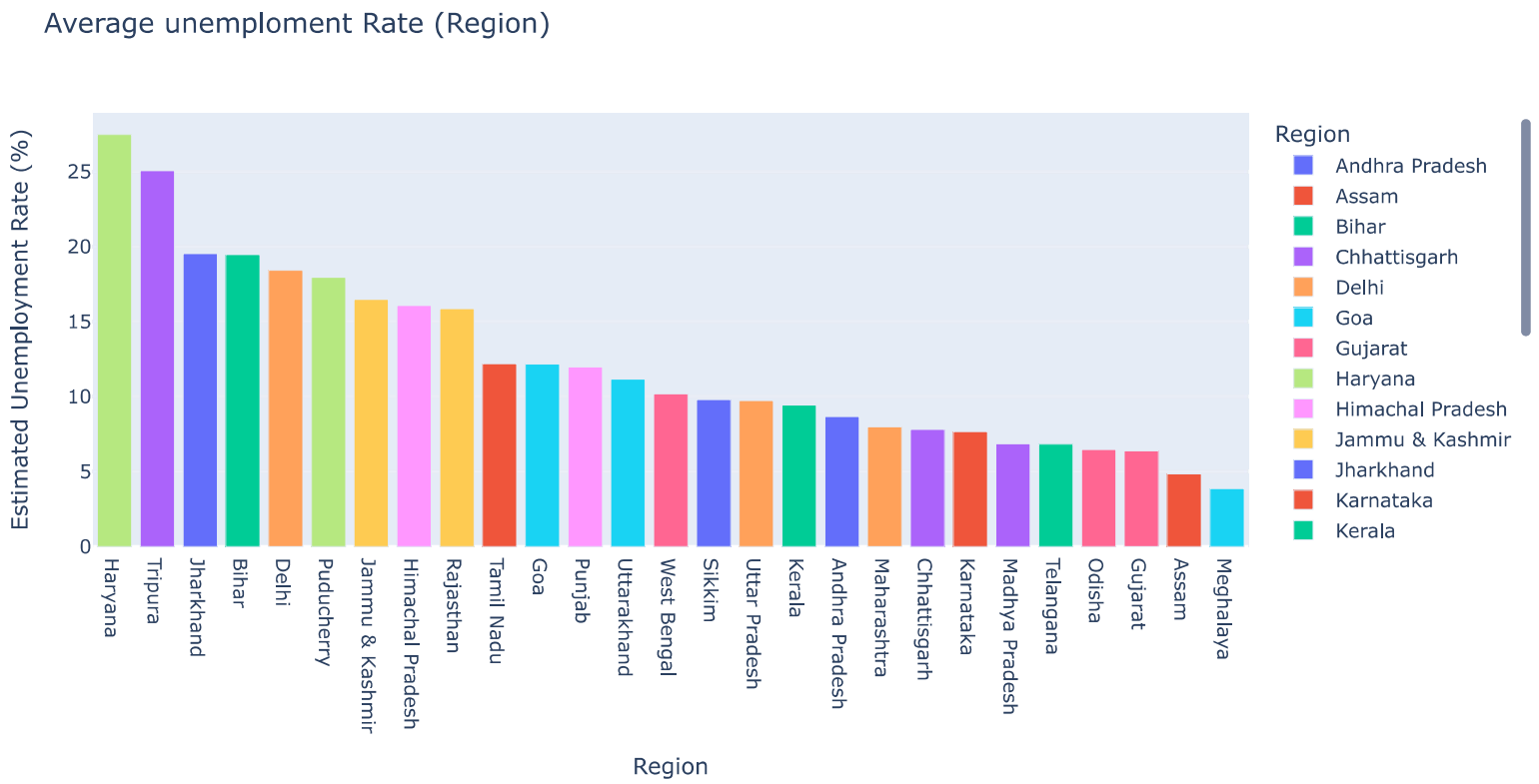
In [14]:
```
state = df.groupby(["Region"])[[" Estimated Unemployment Rate (%)", " Estimated Employed",
                                " Estimated Labour Participation Rate (%)"]].mean()
state = pd.DataFrame(state).reset_index()
```

In [15]:
```
ax = sns.barplot(x='Region',y=' Estimated Unemployment Rate (%)',data=df)
ax.tick_params(axis='x', rotation=90)
```



In [16]:
```
# average unemployment rate bar plot
fig = px.bar(state, x='Region', y=" Estimated Unemployment Rate (%)", color="Region",
             title="Average unemploment Rate (Region)")
fig.update_layout(xaxis={'categoryorder':'total descending'})

fig.show()
```



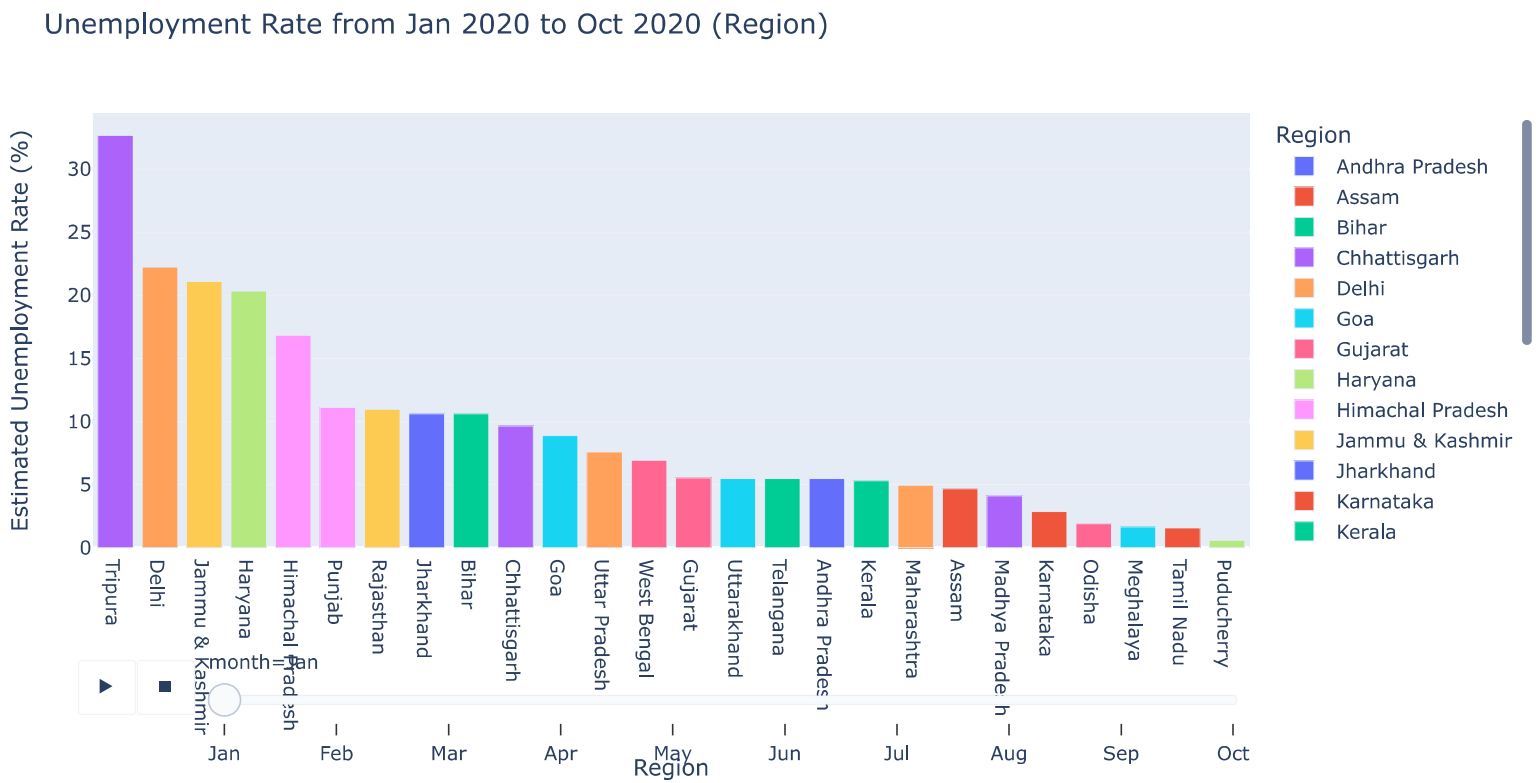
In [17]:
```
# bar plot unemployment rate (monthly)

fig = px.bar(df, x='Region',y=' Estimated Unemployment Rate (%)', animation_frame = 'month', color='Region',
             title='Unemployment Rate from Jan 2020 to Oct 2020 (Region)')

fig.update_layout(xaxis={'categoryorder':'total descending'})

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

fig.show()
```



In [18]:
```
df['Region.1'].unique()
```

Out[18]:
```
array(['South', 'Northeast', 'East', 'West', 'North'], dtype=object)
```

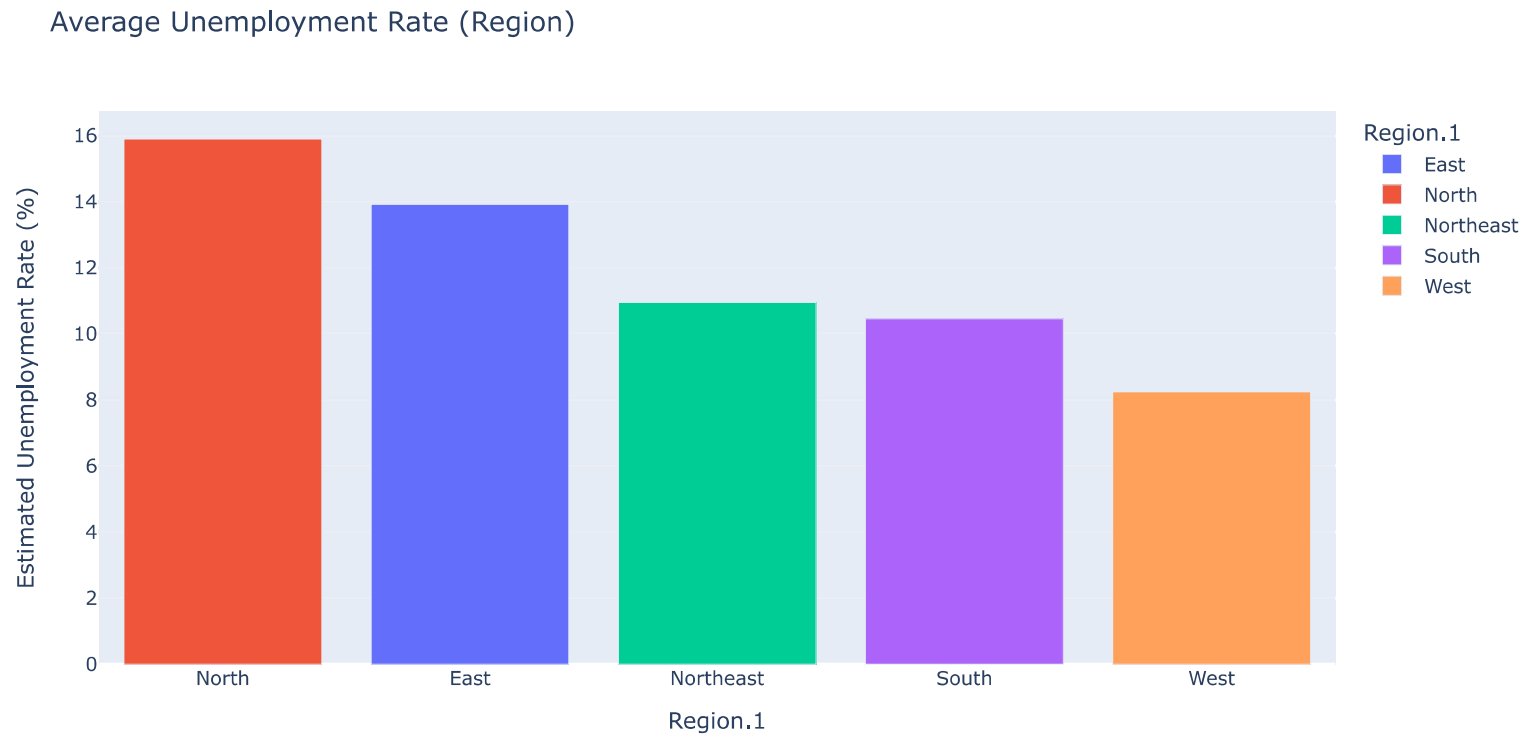
In [19]:
```
# numeric data grouped by region

region = df.groupby(['Region.1'])[[' Estimated Unemployment Rate (%)', " Estimated Employed",
                                   " Estimated Labour Participation Rate (%)"]].mean()
region = pd.DataFrame(region).reset_index()
```

In [20]:
```
# scatter plot

fig = px.scatter_matrix(df, dimensions=[' Estimated Unemployment Rate (%)', " Estimated Employed",
                                        " Estimated Labour Participation Rate (%)"], color='Region.1')
fig.show()
```

In [21]:
```
# Average Unemployment Rate
fig = px.bar(region, x="Region.1", y=" Estimated Unemployment Rate (%)", color="Region.1",
             title="Average Unemployment Rate (Region)")
fig.update_layout(xaxis={'categoryorder':'total descending'})
fig.show()
```

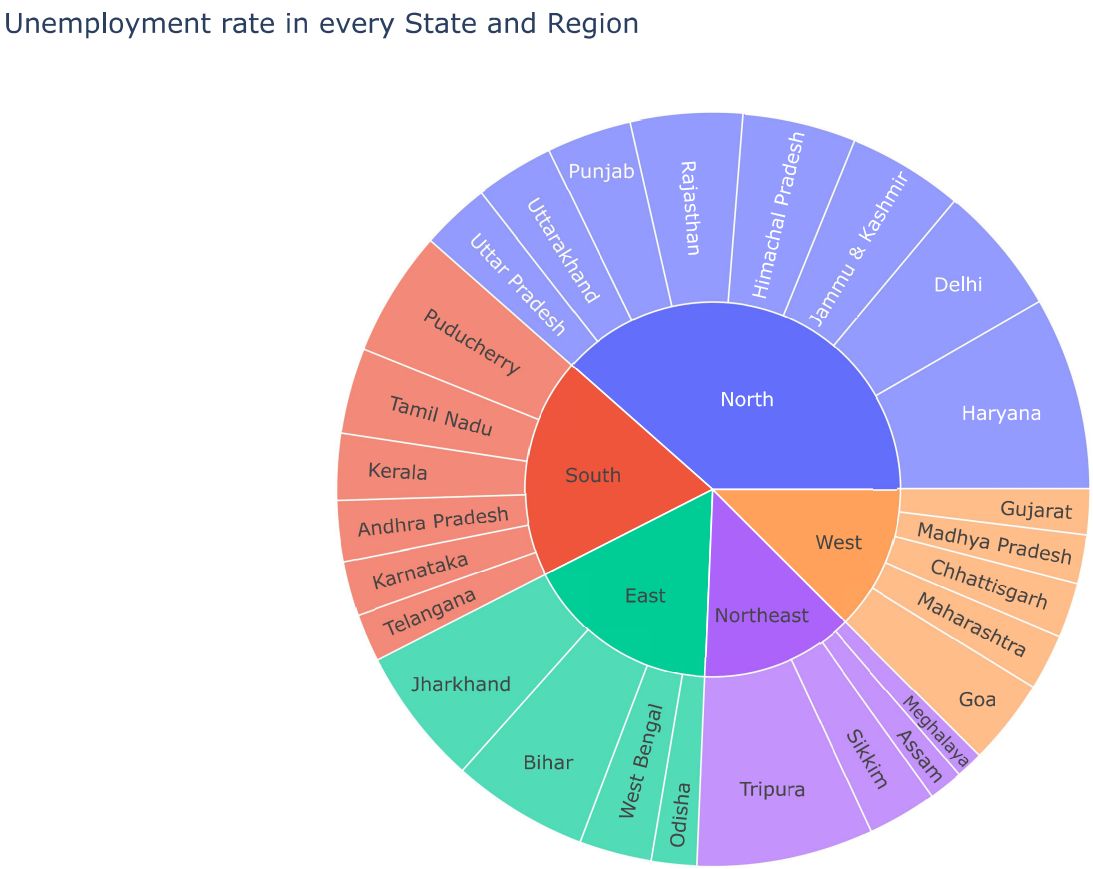


In [22]:
```
unemployment = df.groupby(['Region.1','Region'])[' Estimated Unemployment Rate (%)'].mean().reset_index()
unemployment.head()
```

Out[22]:

	Region.1	Region	Estimated Unemployment Rate (%)
0	East	Bihar	19.471
1	East	Jharkhand	19.539
2	East	Odisha	6.462
3	East	West Bengal	10.192
4	North	Delhi	18.414

In [23]:
```
fig = px.sunburst(unemployment, path=['Region.1','Region'], values=' Estimated Unemployment Rate (%)',
                  title= 'Unemployment rate in every State and Region', height=650)
fig.show()
```



CONCLUSION

- Unployment Rate Outburst was occured in April and May.
- Estimated Employeed was decreased in April and May (After the outburst of COVID-19)
- Hariyana, Tripura, Jharkhand, Bihar and Tamil Nadu was the states in which Unemployment Rate was too high.
- Estimated Unemployment Rate of Hariyana, Tripura, Jharkhand, Bihar, Delhi, puducherry, J&K, Himachal and Rajsthan has more than 15 %.
- Impact of Unemployment Rate was more on North and East Regions.