task2 OASIS

May 16, 2023

[33]:
```
import numpy as np  #numpy library
import pandas as pd  #pandas library
import matplotlib.pyplot as plt #pyplot
import seaborn as sns #seaborn
import plotly.express as px #for visualization
```

[34]:
```
df = pd.read_csv("Unemployment_Rate_upto_11_2020.csv")
df
```

[34]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	\
0	Andhra Pradesh	31-01-2020	M	5.48	
1	Andhra Pradesh	29-02-2020	M	5.83	
2	Andhra Pradesh	31-03-2020	M	5.79	
3	Andhra Pradesh	30-04-2020	M	20.51	
4	Andhra Pradesh	31-05-2020	M	17.43	
..	…	…	…	…	
262	West Bengal	30-06-2020	M	7.29	
263	West Bengal	31-07-2020	M	6.83	
264	West Bengal	31-08-2020	M	14.87	
265	West Bengal	30-09-2020	M	9.35	
266	West Bengal	31-10-2020	M	9.98	

	Estimated Employed	Estimated Labour Participation Rate (%)	Region.1	\
0	16635535	41.02	South	
1	16545652	40.90	South	
2	15881197	39.18	South	
3	11336911	33.10	South	
4	12988845	36.46	South	
..	…	…	…	
262	30726310	40.39	East	
263	35372506	46.17	East	
264	33298644	47.48	East	
265	35707239	47.73	East	
266	33962549	45.63	East	

	longitude	latitude
0	15.9129	79.740

```
1      15.9129    79.740
2      15.9129    79.740
3      15.9129    79.740
4      15.9129    79.740
..         …        …
262    22.9868    87.855
263    22.9868    87.855
264    22.9868    87.855
265    22.9868    87.855
266    22.9868    87.855

[267 rows x 9 columns]
```

[35]:
```
df.isnull().sum()
```

[35]:
```
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
```

[36]:
```
df.shape # before removing null values
```

[36]:
```
(267, 9)
```

[37]:
```
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
```

[38]:
```
df.head(5)   # before preprocessing
```

[38]:
```
           Region         Date  Frequency   Estimated Unemployment Rate (%)  \
0  Andhra Pradesh   31-01-2020          M                              5.48
1  Andhra Pradesh   29-02-2020          M                              5.83
2  Andhra Pradesh   31-03-2020          M                              5.79
3  Andhra Pradesh   30-04-2020          M                             20.51
4  Andhra Pradesh   31-05-2020          M                             17.43

    Estimated Employed   Estimated Labour Participation Rate (%) Region.1  \
0             16635535                                    41.02    South
1             16545652                                    40.90    South
2             15881197                                    39.18    South
3             11336911                                    33.10    South
4             12988845                                    36.46    South

   longitude  latitude
0    15.9129     79.74
1    15.9129     79.74
2    15.9129     79.74
3    15.9129     79.74
4    15.9129     79.74
```

[39]:
```
df.dropna(how='any',inplace=True)
print(df)
```

```
             Region         Date  Frequency   Estimated Unemployment Rate (%)  \
0    Andhra Pradesh   31-01-2020          M                              5.48
1    Andhra Pradesh   29-02-2020          M                              5.83
2    Andhra Pradesh   31-03-2020          M                              5.79
3    Andhra Pradesh   30-04-2020          M                             20.51
4    Andhra Pradesh   31-05-2020          M                             17.43
..              ...          ...        ...                               ...
262     West Bengal   30-06-2020          M                              7.29
263     West Bengal   31-07-2020          M                              6.83
264     West Bengal   31-08-2020          M                             14.87
265     West Bengal   30-09-2020          M                              9.35
266     West Bengal   31-10-2020          M                              9.98

      Estimated Employed   Estimated Labour Participation Rate (%) Region.1  \
0               16635535                                    41.02    South
1               16545652                                    40.90    South
2               15881197                                    39.18    South
3               11336911                                    33.10    South
4               12988845                                    36.46    South
..                   ...                                      ...      ...
```

```
262             30726310                                  40.39     East
263             35372506                                  46.17     East
264             33298644                                  47.48     East
265             35707239                                  47.73     East
266             33962549                                  45.63     East

     longitude  latitude
0      15.9129    79.740
1      15.9129    79.740
2      15.9129    79.740
3      15.9129    79.740
4      15.9129    79.740
..         …        …
262    22.9868    87.855
263    22.9868    87.855
264    22.9868    87.855
265    22.9868    87.855
266    22.9868    87.855

[267 rows x 9 columns]
```

[40]:
```
df.shape # after removing null values
```

[40]:
```
(267, 9)
```

[41]:
```
df.head(5)
```

[41]:
```
           Region         Date  Frequency   Estimated Unemployment Rate (%)  \
0  Andhra Pradesh   31-01-2020          M                              5.48
1  Andhra Pradesh   29-02-2020          M                              5.83
2  Andhra Pradesh   31-03-2020          M                              5.79
3  Andhra Pradesh   30-04-2020          M                             20.51
4  Andhra Pradesh   31-05-2020          M                             17.43

    Estimated Employed   Estimated Labour Participation Rate (%) Region.1  \
0             16635535                                    41.02    South
1             16545652                                    40.90    South
2             15881197                                    39.18    South
3             11336911                                    33.10    South
4             12988845                                    36.46    South

   longitude  latitude
0    15.9129     79.74
1    15.9129     79.74
2    15.9129     79.74
3    15.9129     79.74
4    15.9129     79.74
```

[42]:
```
df.tail()
```

[42]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Region.1	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

[43]:
```
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
```

[44]:
```
df.describe()
```

[44]:

	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	42.240337	22.986800	87.855000

```
std                                  10.803283         1.336632e+07
min                                   0.500000         1.175420e+05
25%                                   4.845000         2.838930e+06
50%                                   9.650000         9.732417e+06
75%                                  16.755000         2.187869e+07
max                                  75.850000         5.943376e+07

       Estimated Labour Participation Rate (%)   longitude    latitude
count                                267.000000  267.000000  267.000000
mean                                  41.681573   22.826048   80.532425
std                                    7.845419    6.270731    5.831738
min                                   16.770000   10.850500   71.192400
25%                                   37.265000   18.112400   76.085600
50%                                   40.390000   23.610200   79.019300
75%                                   44.055000   27.278400   85.279900
max                                   69.690000   33.778200   92.937600
```

[45]:
```
df.dtypes
```

[45]:
```
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
```

[46]:
```
df.index
```

[46]:
```
RangeIndex(start=0, stop=267, step=1)
```

[47]:
```
df.nunique()
```

[47]:
```
Region                                    27
 Date                                     10
 Frequency                                 1
 Estimated Unemployment Rate (%)         252
 Estimated Employed                      267
 Estimated Labour Participation Rate (%) 248
Region.1                                   5
longitude                                 27
latitude                                  24
dtype: int64
```

[48]:
```
df["Region"].unique()
```

[48]:
```
array(['Andhra Pradesh', 'Assam', 'Bihar', 'Chhattisgarh', 'Delhi', 'Goa',
       'Gujarat', 'Haryana', 'Himachal Pradesh', 'Jammu & Kashmir',
       'Jharkhand', 'Karnataka', 'Kerala', 'Madhya Pradesh',
       'Maharashtra', 'Meghalaya', 'Odisha', 'Puducherry', 'Punjab',
       'Rajasthan', 'Sikkim', 'Tamil Nadu', 'Telangana', 'Tripura',
       'Uttar Pradesh', 'Uttarakhand', 'West Bengal'], dtype=object)
```

[49]:
```
df["Region.1"].unique()
```

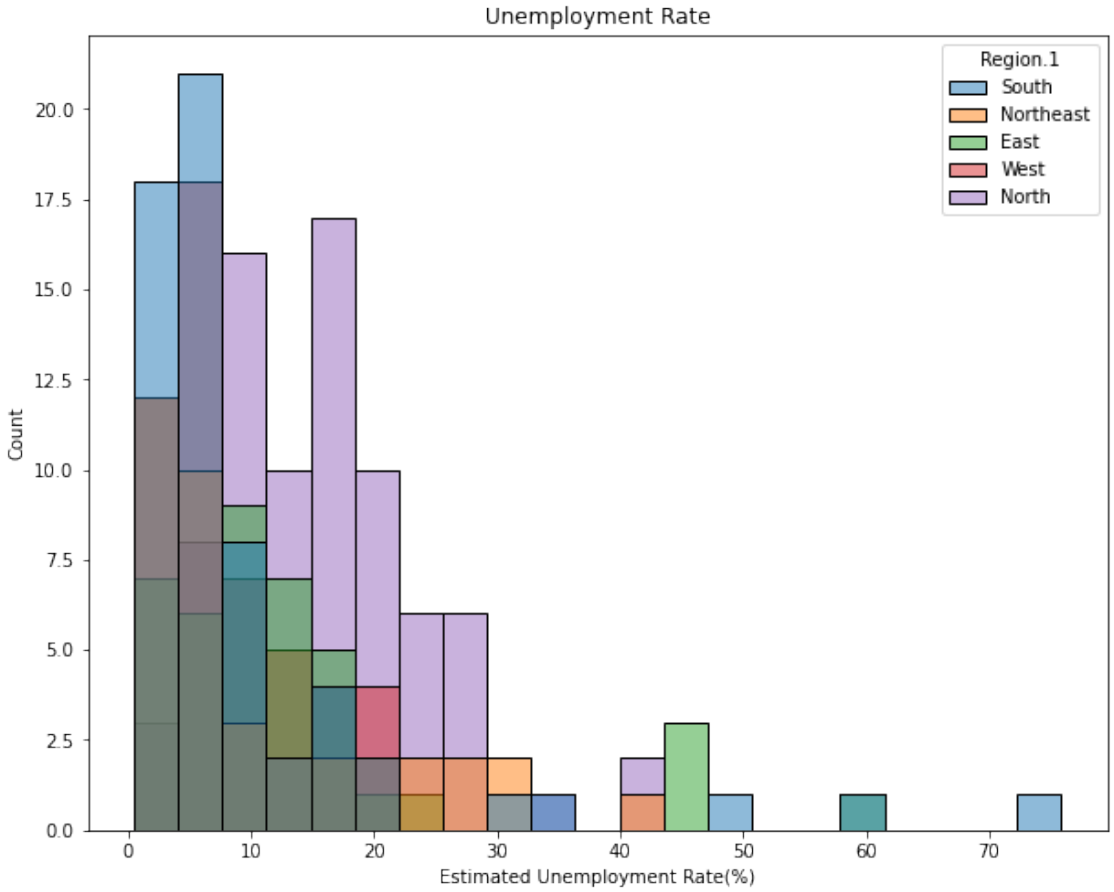
[49]:
```
array(['South', 'Northeast', 'East', 'West', 'North'], dtype=object)
```

[50]:
```
df.groupby("Region.1").size()
```

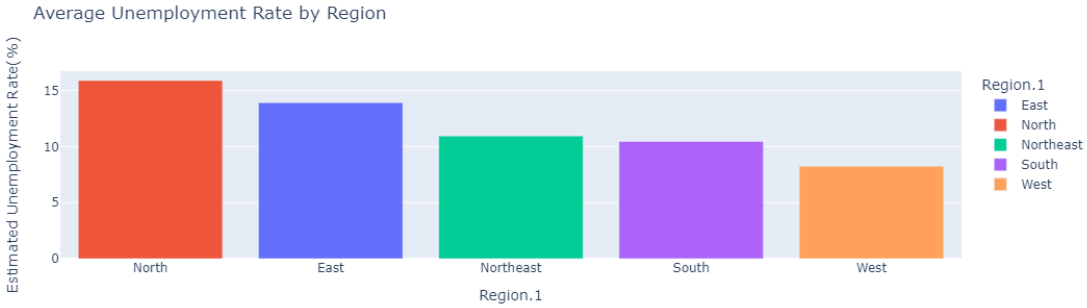
[50]:
```
Region.1
East         40
North        79
Northeast    38
South        60
West         50
dtype: int64
```

[51]:
```
import seaborn
correlation = df.corr ()
fig=plt.figure(figsize=(14,8))
seaborn.heatmap(correlation,annot=True)
plt.show()

                                                    #HEATMAPS
```

[52]:
```
#unemployment rate according to different regions of India
df.columns= ["Region","Date","Frequency","Estimated Unemployment␣
 ↪Rate(%)","Estimated Employed","Estimated Labour Participation␣
 ↪Rate(%)","Region.1","longitude","latitude"]

plt.figure(figsize=(10, 8))
plt.title("Unemployment Rate")
sns.histplot(x="Estimated Unemployment Rate(%)", hue="Region.1", data=df)
plt.show()
```

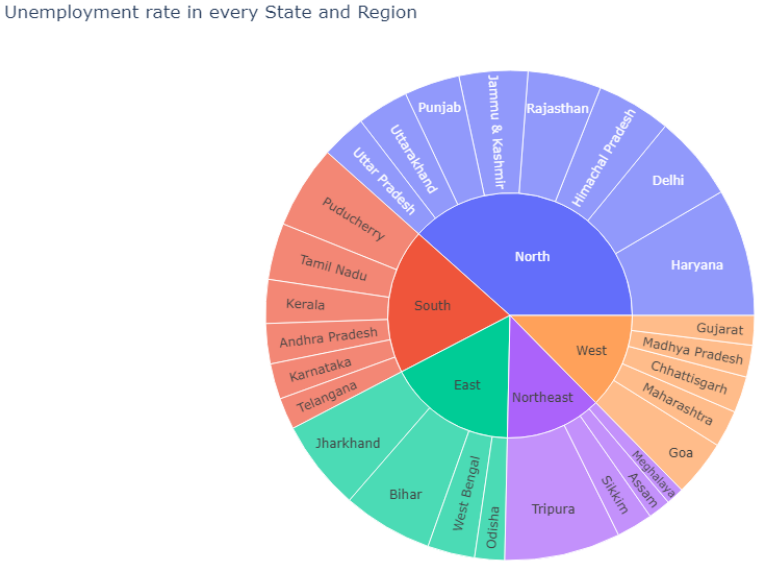


[53]:
```
region = df.groupby(["Region.1"])[["Estimated Unemployment Rate(%)", "Estimated␣
↪Employed", "Estimated Labour Participation Rate(%)"]].mean()
region = pd.DataFrame(region).reset_index()
fig = px.bar(region, x="Region.1", y="Estimated Unemployment Rate(%)",␣
↪color="Region.1", title="Average Unemployment Rate by Region")
fig.update_layout(xaxis={'categoryorder':'total descending'})
fig.show()

# barchart
```



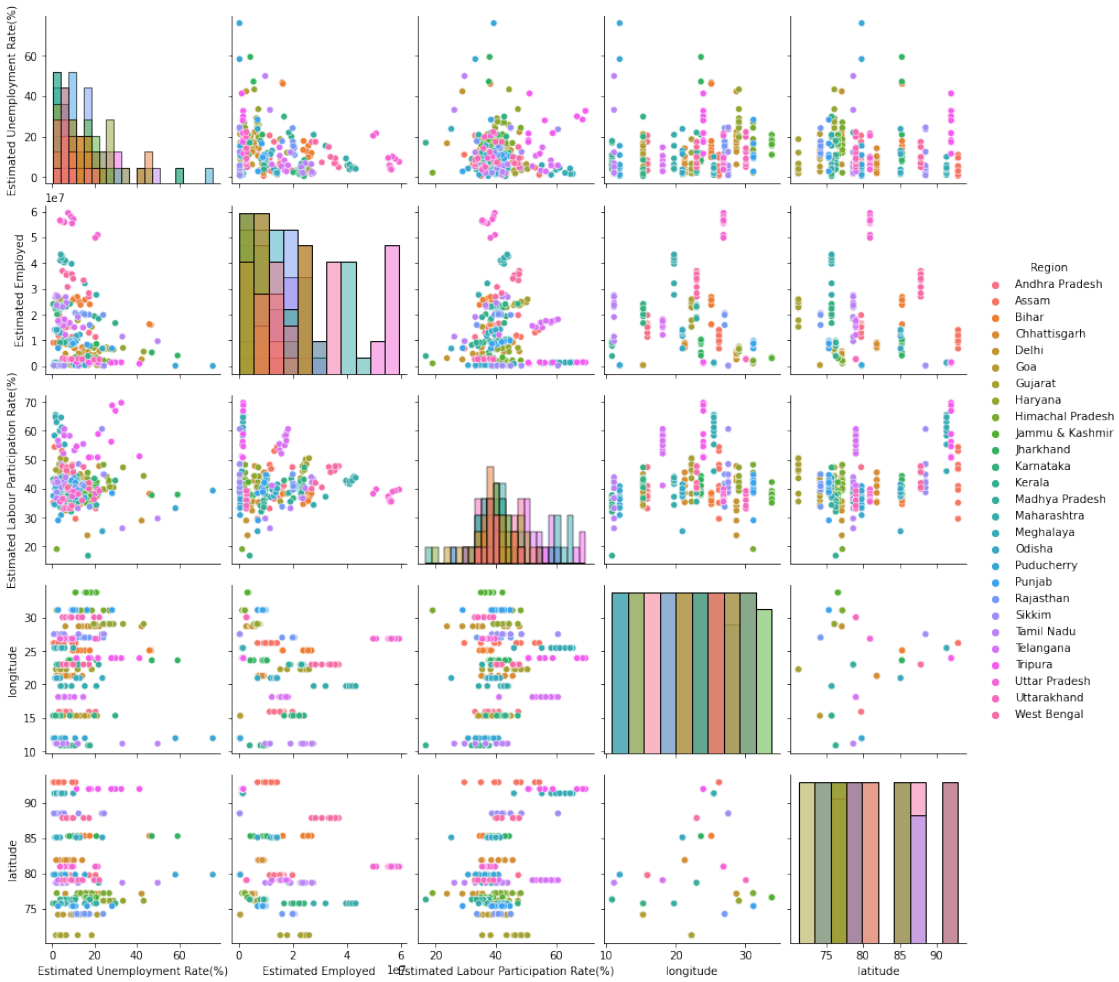
[54]:
```
unemployment = df[["Region", "Region.1", "Estimated Unemployment Rate(%)"]]
fig = px.sunburst(unemployment, path=['Region.1','Region'], values='Estimated␣
 ↪Unemployment Rate(%)',
                  title='Unemployment rate in every State and Region',␣
 ↪height=650)
fig.show()
```

```
/opt/conda/envs/anaconda-2022.05-py39/lib/python3.9/site-
packages/plotly/express/_core.py:1637: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a
future version. Use pandas.concat instead.

/opt/conda/envs/anaconda-2022.05-py39/lib/python3.9/site-
packages/plotly/express/_core.py:1637: FutureWarning:

The frame.append method is deprecated and will be removed from pandas in a
future version. Use pandas.concat instead.
```



[55]:
```
sns.pairplot(df,hue="Region",diag_kind="hist")     #pairplots using seaborn
```

[55]: <seaborn.axisgrid.PairGrid at 0x7f86082c47c0>



[56]:
```
df = df.drop(columns='Date', axis=1)
df = df.drop(columns='Frequency', axis=1)

# print the resulting dataset
print(df)
```

```
            Region  Estimated Unemployment Rate(%)  Estimated Employed  \
0    Andhra Pradesh                            5.48            16635535
1    Andhra Pradesh                            5.83            16545652
2    Andhra Pradesh                            5.79            15881197
3    Andhra Pradesh                           20.51            11336911
4    Andhra Pradesh                           17.43            12988845
..              ...                             ...                 ...
262     West Bengal                            7.29            30726310
263     West Bengal                            6.83            35372506
264     West Bengal                           14.87            33298644
265     West Bengal                            9.35            35707239
```

```
266     West Bengal                                9.98            33962549

     Estimated Labour Participation Rate(%) Region.1  longitude  latitude
0                                    41.02    South    15.9129    79.740
1                                    40.90    South    15.9129    79.740
2                                    39.18    South    15.9129    79.740
3                                    33.10    South    15.9129    79.740
4                                    36.46    South    15.9129    79.740
..                                     …      …        …       …
262                                  40.39     East    22.9868    87.855
263                                  46.17     East    22.9868    87.855
264                                  47.48     East    22.9868    87.855
265                                  47.73     East    22.9868    87.855
266                                  45.63     East    22.9868    87.855

[267 rows x 7 columns]
```

[57]:
```
from sklearn.preprocessing import LabelEncoder
Numerics=LabelEncoder()

df['Region']=Numerics.fit_transform(df['Region'])
df['Region.1']=Numerics.fit_transform(df['Region.1'])

print("ok")

print(df)
```

```
ok
     Region  Estimated Unemployment Rate(%)  Estimated Employed  \
0         0                            5.48            16635535
1         0                            5.83            16545652
2         0                            5.79            15881197
3         0                           20.51            11336911
4         0                           17.43            12988845
..      …                             …                 …
262      26                            7.29            30726310
263      26                            6.83            35372506
264      26                           14.87            33298644
265      26                            9.35            35707239
266      26                            9.98            33962549

     Estimated Labour Participation Rate(%)  Region.1  longitude  latitude
0                                    41.02         3    15.9129    79.740
1                                    40.90         3    15.9129    79.740
2                                    39.18         3    15.9129    79.740
3                                    33.10         3    15.9129    79.740
4                                    36.46         3    15.9129    79.740
..                                     …       …        …       …
```

```
262                                  40.39         0    22.9868    87.855
263                                  46.17         0    22.9868    87.855
264                                  47.48         0    22.9868    87.855
265                                  47.73         0    22.9868    87.855
266                                  45.63         0    22.9868    87.855

[267 rows x 7 columns]
```

[58]:
```
X = df.drop(['Region'], axis=1)

y = df['Region']
print("okay")
```

```
okay
```

[59]:
```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.33,␣
 ↪random_state = 42)
```

[60]:
```
X_train.shape, X_test.shape
```

[60]:
```
((178, 6), (89, 6))
```

[61]:
```
X_train.dtypes
```

[61]:
```
Estimated Unemployment Rate(%)            float64
Estimated Employed                          int64
Estimated Labour Participation Rate(%)    float64
Region.1                                    int64
longitude                                 float64
latitude                                  float64
dtype: object
```

[62]:
```
#Import Libraries file

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.model_selection import train_test_split #Train Test Split

from sklearn.naive_bayes import GaussianNB   # Naive Bayes Classifier

from sklearn import preprocessing            # Label Encoder

from sklearn.neighbors import KNeighborsClassifier  # KNN Classsifiers
```

[63]:
```
from sklearn.preprocessing import LabelEncoder
Numerics=LabelEncoder()

df['Region']=Numerics.fit_transform(df['Region'])
df['Region.1']=Numerics.fit_transform(df['Region.1'])

print("ok")

print(df)
```

```
ok
     Region  Estimated Unemployment Rate(%)  Estimated Employed  \
0         0                            5.48            16635535
1         0                            5.83            16545652
2         0                            5.79            15881197
3         0                           20.51            11336911
4         0                           17.43            12988845
..      …                             …                 …
262      26                            7.29            30726310
263      26                            6.83            35372506
264      26                           14.87            33298644
265      26                            9.35            35707239
266      26                            9.98            33962549

     Estimated Labour Participation Rate(%)  Region.1  longitude  latitude
0                                     41.02         3    15.9129    79.740
1                                     40.90         3    15.9129    79.740
2                                     39.18         3    15.9129    79.740
3                                     33.10         3    15.9129    79.740
4                                     36.46         3    15.9129    79.740
..                                      …       …         …       …
262                                   40.39         0    22.9868    87.855
263                                   46.17         0    22.9868    87.855
264                                   47.48         0    22.9868    87.855
265                                   47.73         0    22.9868    87.855
266                                   45.63         0    22.9868    87.855

[267 rows x 7 columns]
```

[64]:
```
#Train Test split

x = df[['Estimated Employed','Estimated Labour Participation Rate(%)','Region.
  ↪1','Estimated Unemployment Rate(%)','longitude', 'latitude']]
y = df['Region']

x_train, x_test, y_train, y_test = train_test_split(x,y, test_size=0.30,␣
  ↪random_state=0)
```

```
x_train.shape
```

[64]: (186, 6)

[66]:
```
# import Random Forest classifier

from sklearn.ensemble import RandomForestClassifier



# instantiate the classifier

rfc = RandomForestClassifier(random_state=0)



# fit the model

rfc.fit(x_train, y_train)



# Predict the Test set results

y_pred = rfc.predict(x_test)



# Check accuracy score

from sklearn.metrics import accuracy_score

print('Model accuracy score with 10 decision-trees : {0:0.4f}'.␣
 ↪format(accuracy_score(y_test, y_pred)))
```

```
Model accuracy score with 10 decision-trees : 1.0000
```

[67]:
```
# instantiate the classifier with n_estimators = 100

rfc_100 = RandomForestClassifier(n_estimators=100, random_state=0)



# fit the model to the training set

rfc_100.fit(x_train, y_train)
```

```
# Predict on the test set results

y_pred_100 = rfc_100.predict(x_test)



# Check accuracy score

print('Model accuracy score with 100 decision-trees : {0:0.4f}'.␣
 ↪format(accuracy_score(y_test, y_pred_100)))
```

```
Model accuracy score with 100 decision-trees : 1.0000
```

[68]:
```
# create the classifier with n_estimators = 100

clf = RandomForestClassifier(n_estimators=100, random_state=0)



# fit the model to the training set

clf.fit(x_train, y_train)
```

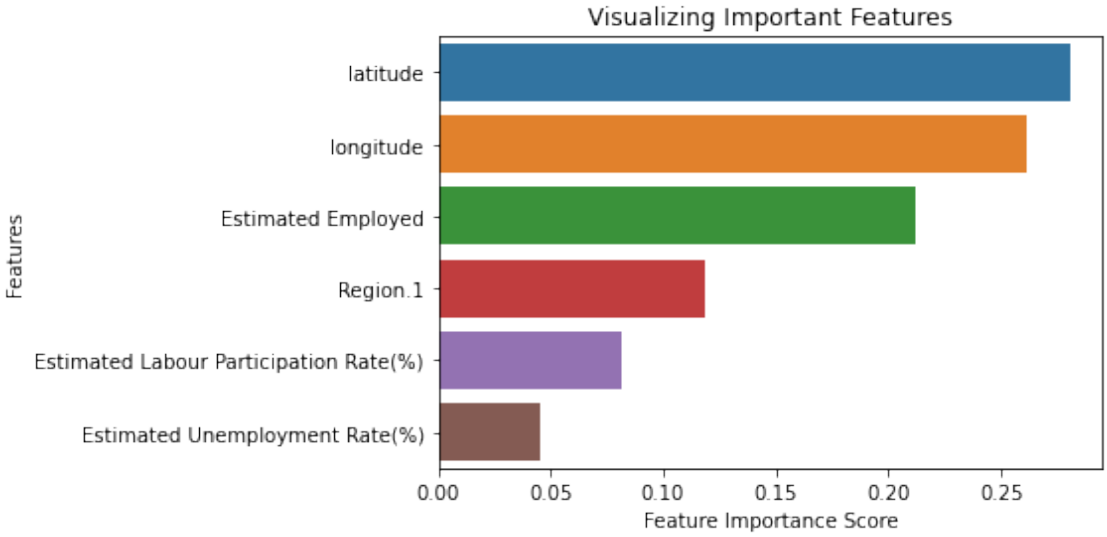
[68]:
```
RandomForestClassifier(random_state=0)
```

[69]:
```
feature_scores = pd.Series(clf.feature_importances_, index=x_train.columns).
 ↪sort_values(ascending=False)

feature_scores
```

[69]:
```
latitude                                0.281118
longitude                               0.261566
Estimated Employed                      0.212318
Region.1                                0.118319
Estimated Labour Participation Rate(%)  0.081323
Estimated Unemployment Rate(%)          0.045356
dtype: float64
```

[70]:
```
# Creating a seaborn bar plot

sns.barplot(x=feature_scores, y=feature_scores.index)
```

```
# Add labels to the graph

plt.xlabel('Feature Importance Score')

plt.ylabel('Features')



# Add title to the graph

plt.title("Visualizing Important Features")



# Visualize the graph

plt.show()
```



[ ]:

[ ]: