MinorProject

July 2, 2023

1 Minor Project

2 Internship Roll No.- 23368

3 Enr No. - 11601172022

4 Name - Shatakshi Bansal

5 Branch - CSE-AI

6 Date of Submission - 2nd July 2023

7 39. Global Air Pollution

Air Pollution is contamination of the indoor or outdoor environment by any chemical, physical or biological agent that modifies the natural characteristics of the atmosphere. Pollutants of major public health concern include particulate matter, carbon monoxide, ozone, nitrogen dioxide and sulfur dioxide. Outdoor and indoor air pollution cause respiratory and other diseases .

Dataset : https://www.kaggle.com/datasets/hasibalmuzdadid/global-air-pollution-dataset

[1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:
```
df = pd.read_csv("dataset.csv")
```

8 Exploratory Data Analysis

8.0.1 QUESTION 1

Show if there are any columns with missing values with their count.

[3]:
```
missing_values = df.isnull().sum()
print(missing_values[missing_values>0])
```

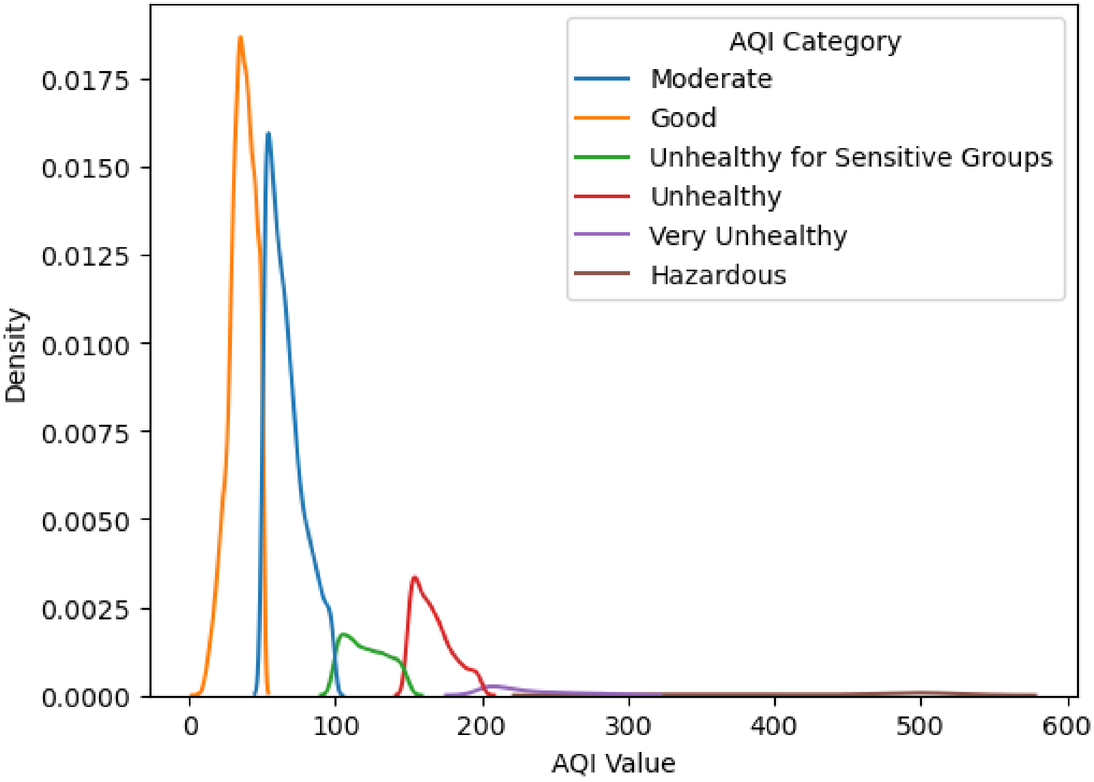
```
Country    427
City         1
dtype: int64
```

Here, we can see that City and Country has some missing values.

8.0.2 QUESTION 2

Plot the distplot of ‘AQI Value’ vs ‘AQI Category’.(kind-‘kde’)

[4]:
```
sns.kdeplot(data = df, x = 'AQI Value', hue = 'AQI Category')
plt.show()
```

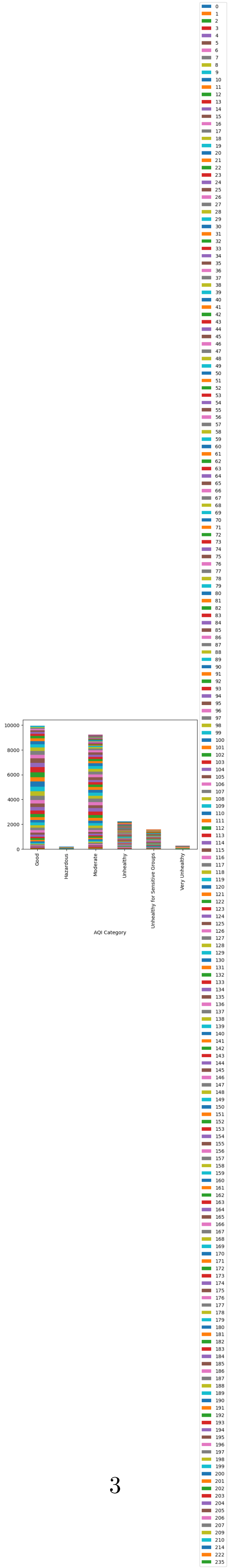


Maximum density has an AQI value below hundred which is categorised as Moderate.

8.0.3 QUESTION 3

Plot a stacked bar graph of ‘Ozone AQI Value’ vs ‘AQI Category’.

[5]:
```
grouped_data = df.groupby(['AQI Category', 'Ozone AQI Value']).size().unstack()
grouped_data.plot(kind = 'bar', stacked = True)
plt.legend(loc='center left', bbox_to_anchor=(1, 0.5))
plt.show()
```


10000
8000
6000
4000
2000
0
Good
Hazardous
Moderate
Unhealthy
Unhealthy for Sensitive Groups
Very Unhealthy
AQI Category
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
209
210
214
222
235

8.0.4 QUESTION 4

Show the list of cities without a stated country. Fill the missing columns with ‘Unknown’.

```
[6]: missing_cities = df[df['Country'].isnull()]['City']
     print(missing_cities)
     df['Country'].fillna('Unknown', inplace = True)
```

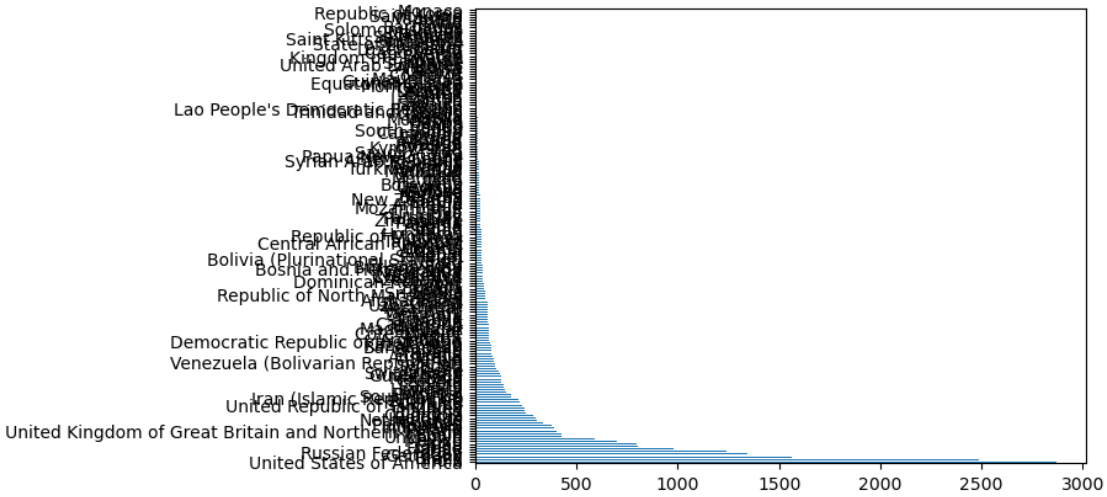
```
535                 Granville
654         Kingston Upon Hull
787              New Waterford
801                 Kingstown
906                  Nanakuli
                 …
22979               Kyaikkami
23115                    Bima
23311               Marapanim
23345                 Calbuco
23420              Sungairaya
Name: City, Length: 427, dtype: object
```

The missing city names here have been filled with the value “Unknown”

8.0.5 QUESTION 5

Plot the most represented countries in this dataset using a horizontal bar graph.

```
[7]: country_count = df['Country'].value_counts()
     country_count.plot(kind = 'barh')
     plt.show()
```



We can see that USA is the most represented country

9 Extension of EDA Task - Classification/Regression

Data Preprocessing

Feature Engineering

Split dataset in train-test (80:20 ratio)

Model selection

Model training

Model evaluation

Fine-tune the Model

Make predictions

[8]:

```
import pandas as pd
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error

# Load the dataset
df = pd.read_csv("dataset.csv")

# Preprocessing
X = df.drop(['PM2.5 AQI Value'], axis=1)
y = df['PM2.5 AQI Value']

categorical_columns = X.select_dtypes(include=['object']).columns
transformer = ColumnTransformer([('one_hot_encoder', OneHotEncoder(),␣
  ↪categorical_columns)], remainder='passthrough')
X_encoded = transformer.fit_transform(X)

# Split dataset in train-test (80:20 ratio)
X_train, X_test, y_train, y_test = train_test_split(X_encoded, y, test_size=0.
  ↪2, random_state=42)

# Model selection
model = RandomForestRegressor()

# Model training
model.fit(X_train, y_train)
```

```
# Model evaluation
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
```

[9]:

```
print("Mean Squared Error: ", mse)
```

```
Mean Squared Error:  20.7930535478372
```

[10]:

```
print(y_pred)
```

```
[ 58.     21.89   47.    … 161.     57.     30.   ]
```