eda-practice

May 8, 2024

[1]:
```
print("Practicing EDA")
```

```
Practicing EDA
```

[3]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[4]:
```
df = pd.read_csv("vaccination-data.csv")
```

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

[5]:

	COUNTRY	ISO3	WHO_REGION	DATA_SOURCE	DATE_UPDATED	TOTAL_VACCINATIONS	PERSONS_VACCINATED_1PLUS_DOSE	TOTAL_VACCINATIONS_PER100	PERSONS_VACCINATED_1PLUS_DOSE_PER100	PERSONS_LAST_DOSE	PERSONS_LAST_DOSE_PER100	VACCINES_USED
0	Bhutan	BTN	SEARO	REPORTING	2022-10-30	2011426.0	699116.0	261.0	91.0	677669.0	88.0	NaN
1	Namibia	NAM	AFRO	REPORTING	2023-11-12	1005937.0	629767.0	40.0	25.0	550978.0	22.0	NaN
2	Iran (Islamic Republic of)	IRN	EMRO	REPORTING	2023-11-26	155461757.0	65199831.0	185.0	78.0	58585264.0	70.0	NaN
3	Kenya	KEN	AFRO	REPORTING	2023-04-02	23750431.0	14494372.0	44.0	27.0			
4	Greenland	GRL	EURO	REPORTING	NaN	NaN	NaN	NaN	NaN			

```
3         11090440.0                      21.0            NaN
4                NaN                       NaN            NaN

  FIRST_VACCINE_DATE  NUMBER_VACCINES_TYPES_USED  PERSONS_BOOSTER_ADD_DOSE  \
0         2021-03-27                         NaN                  634641.0
1         2021-03-19                         NaN                  298560.0
2         2021-02-09                         NaN                31352288.0
3         2021-03-05                         NaN                 2000636.0
4                NaN                         NaN                       NaN

   PERSONS_BOOSTER_ADD_DOSE_PER100
0                             82.0
1                             12.0
2                             37.0
3                              4.0
4                              NaN
```

[6]:
```
df.columns
```

[6]:
```
Index(['COUNTRY', 'ISO3', 'WHO_REGION', 'DATA_SOURCE', 'DATE_UPDATED',
       'TOTAL_VACCINATIONS', 'PERSONS_VACCINATED_1PLUS_DOSE',
       'TOTAL_VACCINATIONS_PER100', 'PERSONS_VACCINATED_1PLUS_DOSE_PER100',
       'PERSONS_LAST_DOSE', 'PERSONS_LAST_DOSE_PER100', 'VACCINES_USED',
       'FIRST_VACCINE_DATE', 'NUMBER_VACCINES_TYPES_USED',
       'PERSONS_BOOSTER_ADD_DOSE', 'PERSONS_BOOSTER_ADD_DOSE_PER100'],
      dtype='object')
```

[7]:
```
df.isna().sum()
```

[7]:
```
COUNTRY                                 0
ISO3                                    0
WHO_REGION                              4
DATA_SOURCE                             0
DATE_UPDATED                            7
TOTAL_VACCINATIONS                      6
PERSONS_VACCINATED_1PLUS_DOSE           6
TOTAL_VACCINATIONS_PER100               8
PERSONS_VACCINATED_1PLUS_DOSE_PER100    8
PERSONS_LAST_DOSE                       6
PERSONS_LAST_DOSE_PER100                8
VACCINES_USED                         210
FIRST_VACCINE_DATE                     14
NUMBER_VACCINES_TYPES_USED            210
PERSONS_BOOSTER_ADD_DOSE               20
PERSONS_BOOSTER_ADD_DOSE_PER100        31
dtype: int64
```

[8]:
```
df['WHO_REGION']
```

[8]:
```
0      SEARO
1       AFRO
2       EMRO
3       AFRO
4       EURO
       …
205     AFRO
206      NaN
207     AFRO
208     AFRO
209     EMRO
Name: WHO_REGION, Length: 210, dtype: object
```

[11]:
```
common_region = df["WHO_REGION"].mode()[0]
df['WHO_REGION'].fillna(common_region,inplace = True)
```

[12]:
```
df['DATE_UPDATED']
```

[12]:
```
0      2022-10-30
1      2023-11-12
2      2023-11-26
3      2023-04-02
4             NaN
          …
205    2023-10-15
206           NaN
207    2023-02-19
208    2022-07-24
209    2023-05-21
Name: DATE_UPDATED, Length: 210, dtype: object
```

[13]:
```
df.dropna(subset=['DATE_UPDATED'],inplace=True)
```

[14]:
```
df.head()
```

[14]:

	COUNTRY	ISO3	WHO_REGION	DATA_SOURCE	DATE_UPDATED	\
0	Bhutan	BTN	SEARO	REPORTING	2022-10-30	
1	Namibia	NAM	AFRO	REPORTING	2023-11-12	
2	Iran (Islamic Republic of)	IRN	EMRO	REPORTING	2023-11-26	
3	Kenya	KEN	AFRO	REPORTING	2023-04-02	
5	Comoros	COM	AFRO	REPORTING	2022-10-02	

	TOTAL_VACCINATIONS	PERSONS_VACCINATED_1PLUS_DOSE	\
0	2011426.0	699116.0	
1	1005937.0	629767.0	

```
2         155461757.0                    65199831.0
3          23750431.0                    14494372.0
5            835021.0                      438825.0

   TOTAL_VACCINATIONS_PER100  PERSONS_VACCINATED_1PLUS_DOSE_PER100  \
0                      261.0                                 91.0
1                       40.0                                 25.0
2                      185.0                                 78.0
3                       44.0                                 27.0
5                       96.0                                 50.0

   PERSONS_LAST_DOSE  PERSONS_LAST_DOSE_PER100  VACCINES_USED  \
0           677669.0                      88.0            NaN
1           550978.0                      22.0            NaN
2         58585264.0                      70.0            NaN
3         11090440.0                      21.0            NaN
5           397080.0                      46.0            NaN

  FIRST_VACCINE_DATE  NUMBER_VACCINES_TYPES_USED  PERSONS_BOOSTER_ADD_DOSE  \
0         2021-03-27                         NaN                  634641.0
1         2021-03-19                         NaN                  298560.0
2         2021-02-09                         NaN                31352288.0
3         2021-03-05                         NaN                 2000636.0
5         2021-04-10                         NaN                       NaN

   PERSONS_BOOSTER_ADD_DOSE_PER100
0                             82.0
1                             12.0
2                             37.0
3                              4.0
5                              NaN
```

[18]:
```
df.shape
```

[18]:
```
(203, 16)
```

[19]:
```
df.isna().sum() / len(df)
```

[19]:
```
COUNTRY                                0.000000
ISO3                                   0.000000
WHO_REGION                             0.000000
DATA_SOURCE                            0.000000
DATE_UPDATED                           0.000000
TOTAL_VACCINATIONS                     0.000000
PERSONS_VACCINATED_1PLUS_DOSE          0.000000
TOTAL_VACCINATIONS_PER100              0.004926
PERSONS_VACCINATED_1PLUS_DOSE_PER100   0.004926
```

```
PERSONS_LAST_DOSE                       0.000000
PERSONS_LAST_DOSE_PER100                0.004926
VACCINES_USED                           1.000000
FIRST_VACCINE_DATE                      0.034483
NUMBER_VACCINES_TYPES_USED              1.000000
PERSONS_BOOSTER_ADD_DOSE                0.068966
PERSONS_BOOSTER_ADD_DOSE_PER100         0.118227
dtype: float64
```

[24]:
```
df.drop(['VACCINES_USED','NUMBER_VACCINES_TYPES_USED'], axis=1, inplace=True)
```

[25]:
```
df.isna().sum() / len(df)
```

[25]:
```
COUNTRY                                 0.000000
ISO3                                    0.000000
WHO_REGION                              0.000000
DATA_SOURCE                             0.000000
DATE_UPDATED                            0.000000
TOTAL_VACCINATIONS                      0.000000
PERSONS_VACCINATED_1PLUS_DOSE           0.000000
TOTAL_VACCINATIONS_PER100               0.004926
PERSONS_VACCINATED_1PLUS_DOSE_PER100    0.004926
PERSONS_LAST_DOSE                       0.000000
PERSONS_LAST_DOSE_PER100                0.004926
FIRST_VACCINE_DATE                      0.034483
PERSONS_BOOSTER_ADD_DOSE                0.068966
PERSONS_BOOSTER_ADD_DOSE_PER100         0.118227
dtype: float64
```

[30]:
```
m1=df['TOTAL_VACCINATIONS_PER100'].mean()
df['TOTAL_VACCINATIONS_PER100'].fillna(m1,inplace=True)
```

[33]:
```
m2=df['PERSONS_VACCINATED_1PLUS_DOSE_PER100'].mean()
df['PERSONS_VACCINATED_1PLUS_DOSE_PER100'].fillna(m2,inplace=True)
```

[35]:
```
m3=df['PERSONS_LAST_DOSE_PER100'].mean()
df['PERSONS_LAST_DOSE_PER100'].fillna(m3,inplace=True)
```

[42]:
```
m4=df['PERSONS_BOOSTER_ADD_DOSE_PER100'].mean()
df['PERSONS_BOOSTER_ADD_DOSE_PER100'].fillna(m4,inplace=True)
```

[43]:
```
df.isna().sum() / len(df)
```

[43]:
```
COUNTRY                                 0.000000
ISO3                                    0.000000
WHO_REGION                              0.000000
DATA_SOURCE                             0.000000
```

```
DATE_UPDATED                          0.000000
TOTAL_VACCINATIONS                    0.000000
PERSONS_VACCINATED_1PLUS_DOSE         0.000000
TOTAL_VACCINATIONS_PER100             0.000000
PERSONS_VACCINATED_1PLUS_DOSE_PER100  0.000000
PERSONS_LAST_DOSE                     0.000000
PERSONS_LAST_DOSE_PER100              0.000000
FIRST_VACCINE_DATE                    0.034483
PERSONS_BOOSTER_ADD_DOSE              0.068966
PERSONS_BOOSTER_ADD_DOSE_PER100       0.000000
dtype: float64
```

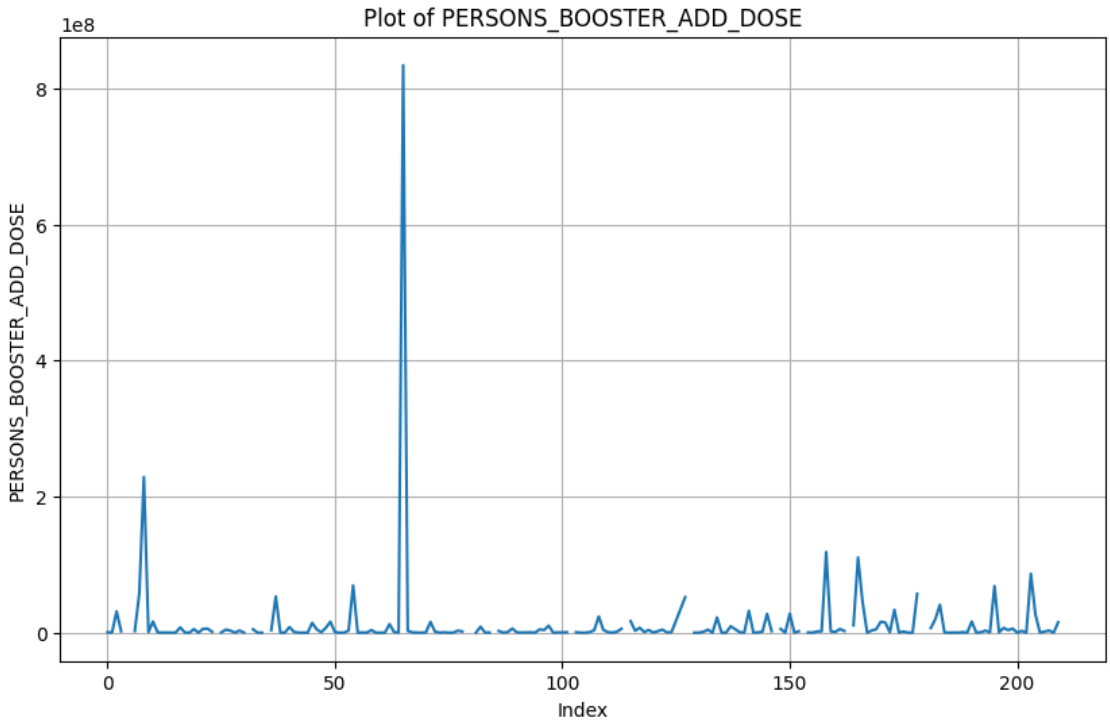
[44]:
```
df.iloc[:,-3:]   # to view the last three
```

[44]:
```
    FIRST_VACCINE_DATE  PERSONS_BOOSTER_ADD_DOSE  \
0           2021-03-27                634641.0
1           2021-03-19                298560.0
2           2021-02-09              31352288.0
3           2021-03-05               2000636.0
5           2021-04-10                     NaN
..                 …                       …
204         2020-12-30              26573833.0
205         2021-04-02                 76359.0
207         2021-03-01               3138712.0
208         2021-02-12                  4597.0
209         2021-01-24              15217352.0

     PERSONS_BOOSTER_ADD_DOSE_PER100
0                          82.000000
1                          12.000000
2                          37.000000
3                           4.000000
5                          32.681564
..                               …
204                        56.000000
205                         4.000000
207                        12.000000
208                        32.681564
209                        15.000000

[203 rows x 3 columns]
```

[50]:
```
df['PERSONS_BOOSTER_ADD_DOSE'].median()
```

[50]:
```
902114.0
```

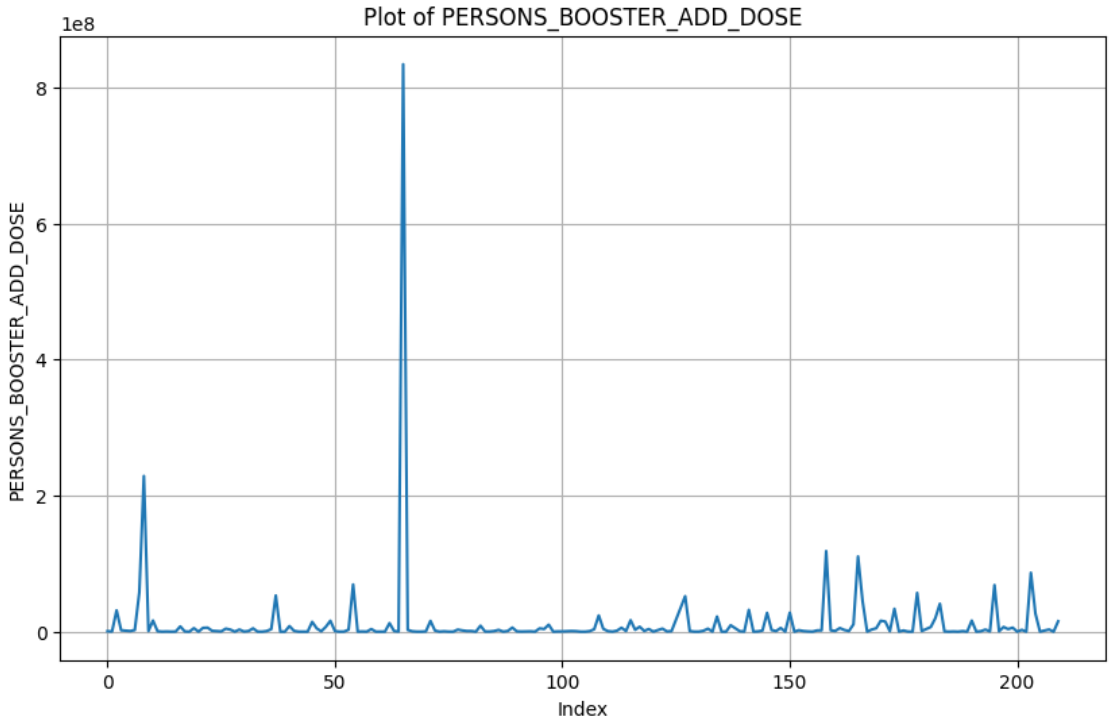
[48]:
```
plt.figure(figsize=(10, 6))
plt.plot(df['PERSONS_BOOSTER_ADD_DOSE'])
plt.title('Plot of PERSONS_BOOSTER_ADD_DOSE')
plt.xlabel('Index')
plt.ylabel('PERSONS_BOOSTER_ADD_DOSE')
plt.grid(True)
plt.show()
```



[51]:
```
median_value = df['PERSONS_BOOSTER_ADD_DOSE'].median()
df['PERSONS_BOOSTER_ADD_DOSE'].fillna(median_value, inplace=True)
```

[52]:
```
# after cleaning
```

[53]:
```
plt.figure(figsize=(10, 6))
plt.plot(df['PERSONS_BOOSTER_ADD_DOSE'])
plt.title('Plot of PERSONS_BOOSTER_ADD_DOSE')
plt.xlabel('Index')
plt.ylabel('PERSONS_BOOSTER_ADD_DOSE')
plt.grid(True)
plt.show()
```



[54]:
```
df['FIRST_VACCINE_DATE'].fillna(method='ffill', inplace=True)  # Forward fill␣
↪missing dates
```

[55]:
```
df.isna().sum()
```

[55]:
```
COUNTRY                              0
ISO3                                 0
WHO_REGION                           0
DATA_SOURCE                          0
DATE_UPDATED                         0
TOTAL_VACCINATIONS                   0
PERSONS_VACCINATED_1PLUS_DOSE        0
TOTAL_VACCINATIONS_PER100            0
PERSONS_VACCINATED_1PLUS_DOSE_PER100 0
PERSONS_LAST_DOSE                    0
PERSONS_LAST_DOSE_PER100             0
FIRST_VACCINE_DATE                   0
PERSONS_BOOSTER_ADD_DOSE             0
PERSONS_BOOSTER_ADD_DOSE_PER100      0
dtype: int64
```

[56]:
```
# performing EDA on data
```

[57]:
```
df.columns
```

[57]:
```
Index(['COUNTRY', 'ISO3', 'WHO_REGION', 'DATA_SOURCE', 'DATE_UPDATED',
       'TOTAL_VACCINATIONS', 'PERSONS_VACCINATED_1PLUS_DOSE',
       'TOTAL_VACCINATIONS_PER100', 'PERSONS_VACCINATED_1PLUS_DOSE_PER100',
       'PERSONS_LAST_DOSE', 'PERSONS_LAST_DOSE_PER100', 'FIRST_VACCINE_DATE',
       'PERSONS_BOOSTER_ADD_DOSE', 'PERSONS_BOOSTER_ADD_DOSE_PER100'],
      dtype='object')
```

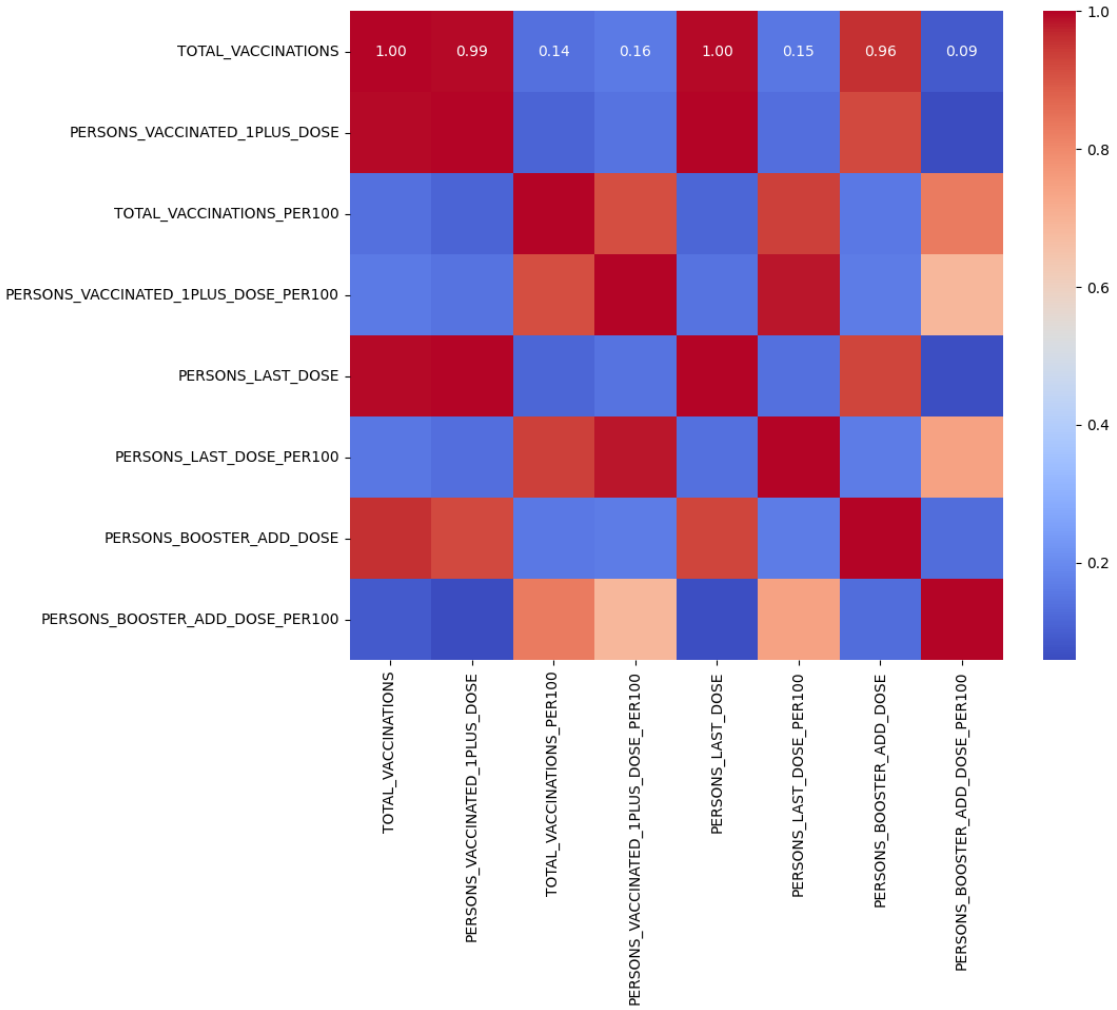
[65]:
```
df.head()
```

[65]:

	COUNTRY	ISO3	WHO_REGION	DATA_SOURCE	DATE_UPDATED	TOTAL_VACCINATIONS	PERSONS_VACCINATED_1PLUS_DOSE	TOTAL_VACCINATIONS_PER100	PERSONS_VACCINATED_1PLUS_DOSE_PER100	PERSONS_LAST_DOSE	PERSONS_LAST_DOSE_PER100	FIRST_VACCINE_DATE	PERSONS_BOOSTER_ADD_DOSE	PERSONS_BOOSTER_ADD_DOSE_PER100
0	Bhutan	BTN	SEARO	REPORTING	2022-10-30	2011426.0	699116.0	261.0	91.0	677669.0	88.0	2021-03-27	634641.0	82.000000
1	Namibia	NAM	AFRO	REPORTING	2023-11-12	1005937.0	629767.0	40.0	25.0	550978.0	22.0	2021-03-19	298560.0	12.000000
2	Iran (Islamic Republic of)	IRN	EMRO	REPORTING	2023-11-26	155461757.0	65199831.0	185.0	78.0	58585264.0	70.0	2021-02-09	31352288.0	37.000000
3	Kenya	KEN	AFRO	REPORTING	2023-04-02	23750431.0	14494372.0	44.0	27.0	11090440.0	21.0	2021-03-05	2000636.0	4.000000
5	Comoros	COM	AFRO	REPORTING	2022-10-02	835021.0	438825.0	96.0	50.0	397080.0	46.0	2021-04-10	902114.0	32.681564

[92]:
```
numeric_columns =␣
↪df[['TOTAL_VACCINATIONS','PERSONS_VACCINATED_1PLUS_DOSE','TOTAL_VACCINATIONS_PER100',
                    ␣
↪'PERSONS_VACCINATED_1PLUS_DOSE_PER100','PERSONS_LAST_DOSE','PERSONS_LAST_DOSE_PER100',
                    ␣
↪'PERSONS_BOOSTER_ADD_DOSE','PERSONS_BOOSTER_ADD_DOSE_PER100']]
plt.figure(figsize=(10, 8))
sns.heatmap(numeric_columns.corr(),annot=True,fmt='.2f',cmap='coolwarm')
```

[92]: <Axes: >



[69]:
```
numeric_columns.corr()
```

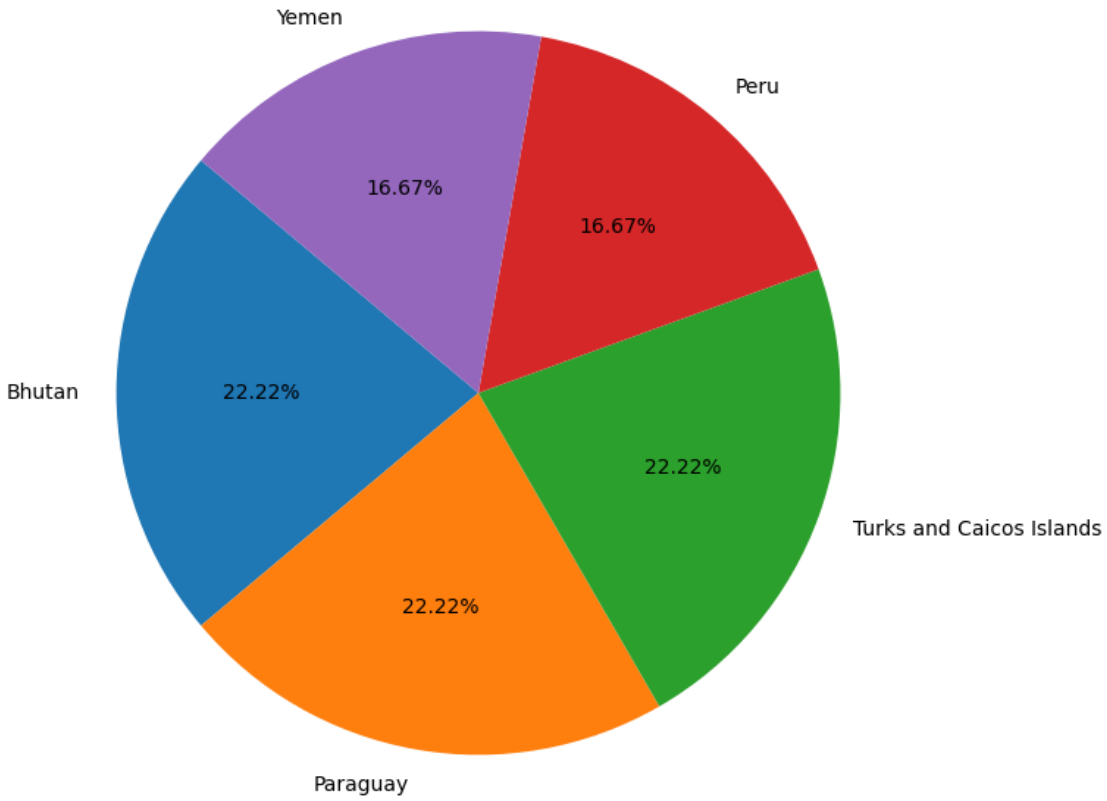
[69]:
```
                                    TOTAL_VACCINATIONS  \
TOTAL_VACCINATIONS                            1.000000
PERSONS_VACCINATED_1PLUS_DOSE                 0.993310
```

```
TOTAL_VACCINATIONS_PER100                    0.137414
PERSONS_VACCINATED_1PLUS_DOSE_PER100         0.158810
PERSONS_LAST_DOSE                            0.995577
PERSONS_LAST_DOSE_PER100                     0.153151
PERSONS_BOOSTER_ADD_DOSE                     0.956369
PERSONS_BOOSTER_ADD_DOSE_PER100              0.089133

                                    PERSONS_VACCINATED_1PLUS_DOSE  \
TOTAL_VACCINATIONS                                       0.993310
PERSONS_VACCINATED_1PLUS_DOSE                            1.000000
TOTAL_VACCINATIONS_PER100                                0.110422
PERSONS_VACCINATED_1PLUS_DOSE_PER100                     0.143877
PERSONS_LAST_DOSE                                        0.999350
PERSONS_LAST_DOSE_PER100                                 0.135412
PERSONS_BOOSTER_ADD_DOSE                                 0.920177
PERSONS_BOOSTER_ADD_DOSE_PER100                          0.058201

                                    TOTAL_VACCINATIONS_PER100  \
TOTAL_VACCINATIONS                                   0.137414
PERSONS_VACCINATED_1PLUS_DOSE                        0.110422
TOTAL_VACCINATIONS_PER100                            1.000000
PERSONS_VACCINATED_1PLUS_DOSE_PER100                 0.913361
PERSONS_LAST_DOSE                                    0.113727
PERSONS_LAST_DOSE_PER100                             0.934663
PERSONS_BOOSTER_ADD_DOSE                             0.154730
PERSONS_BOOSTER_ADD_DOSE_PER100                      0.830581

                                    PERSONS_VACCINATED_1PLUS_DOSE_PER100  \
TOTAL_VACCINATIONS                                                0.158810
PERSONS_VACCINATED_1PLUS_DOSE                                     0.143877
TOTAL_VACCINATIONS_PER100                                         0.913361
PERSONS_VACCINATED_1PLUS_DOSE_PER100                              1.000000
PERSONS_LAST_DOSE                                                 0.144470
PERSONS_LAST_DOSE_PER100                                          0.982525
PERSONS_BOOSTER_ADD_DOSE                                          0.163304
PERSONS_BOOSTER_ADD_DOSE_PER100                                   0.690379

                                    PERSONS_LAST_DOSE  \
TOTAL_VACCINATIONS                           0.995577
PERSONS_VACCINATED_1PLUS_DOSE                0.999350
TOTAL_VACCINATIONS_PER100                    0.113727
PERSONS_VACCINATED_1PLUS_DOSE_PER100         0.144470
PERSONS_LAST_DOSE                            1.000000
PERSONS_LAST_DOSE_PER100                     0.138416
PERSONS_BOOSTER_ADD_DOSE                     0.929985
PERSONS_BOOSTER_ADD_DOSE_PER100              0.064565
```

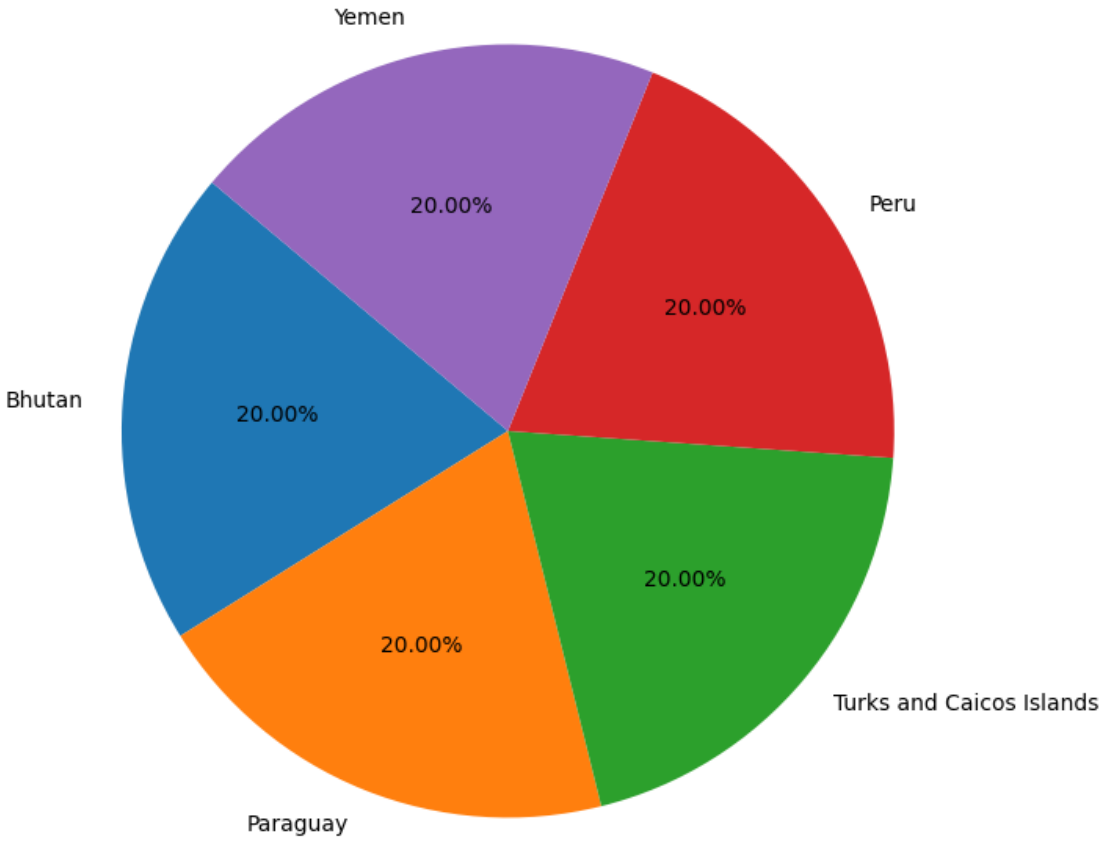
```
                                    PERSONS_LAST_DOSE_PER100  \
TOTAL_VACCINATIONS                                  0.153151
PERSONS_VACCINATED_1PLUS_DOSE                       0.135412
TOTAL_VACCINATIONS_PER100                           0.934663
PERSONS_VACCINATED_1PLUS_DOSE_PER100                0.982525
PERSONS_LAST_DOSE                                   0.138416
PERSONS_LAST_DOSE_PER100                            1.000000
PERSONS_BOOSTER_ADD_DOSE                            0.161698
PERSONS_BOOSTER_ADD_DOSE_PER100                     0.744110

                                    PERSONS_BOOSTER_ADD_DOSE  \
TOTAL_VACCINATIONS                                  0.956369
PERSONS_VACCINATED_1PLUS_DOSE                       0.920177
TOTAL_VACCINATIONS_PER100                           0.154730
PERSONS_VACCINATED_1PLUS_DOSE_PER100                0.163304
PERSONS_LAST_DOSE                                   0.929985
PERSONS_LAST_DOSE_PER100                            0.161698
PERSONS_BOOSTER_ADD_DOSE                            1.000000
PERSONS_BOOSTER_ADD_DOSE_PER100                     0.130992

                                    PERSONS_BOOSTER_ADD_DOSE_PER100
TOTAL_VACCINATIONS                                         0.089133
PERSONS_VACCINATED_1PLUS_DOSE                              0.058201
TOTAL_VACCINATIONS_PER100                                  0.830581
PERSONS_VACCINATED_1PLUS_DOSE_PER100                       0.690379
PERSONS_LAST_DOSE                                          0.064565
PERSONS_LAST_DOSE_PER100                                   0.744110
PERSONS_BOOSTER_ADD_DOSE                                   0.130992
PERSONS_BOOSTER_ADD_DOSE_PER100                            1.000000
```

[98]:
```
top_10_max_values = df['TOTAL_VACCINATIONS_PER100'].nlargest(10)
```

[153]:
```
plt.figure(figsize=(8, 8))
plt.pie(df['TOTAL_VACCINATIONS_PER100'].value_counts().head(5),␣
  ↪labels=df['COUNTRY'].value_counts()[:5].index, startangle=140, autopct="%1.
  ↪2f%%")
plt.show()
```



[152]:
```
if len(df['TOTAL_VACCINATIONS'].value_counts()) >= 5:
    plt.figure(figsize=(8, 8))
    plt.pie(df['TOTAL_VACCINATIONS'].value_counts().head(5),␣
 ↪labels=df['COUNTRY'].value_counts().head(5).index, startangle=140,␣
 ↪autopct="%1.2f%%")
    plt.show()
else:
    print("Not enough unique values in 'TOTAL_VACCINATIONS' column to create␣
 ↪pie chart.")
```



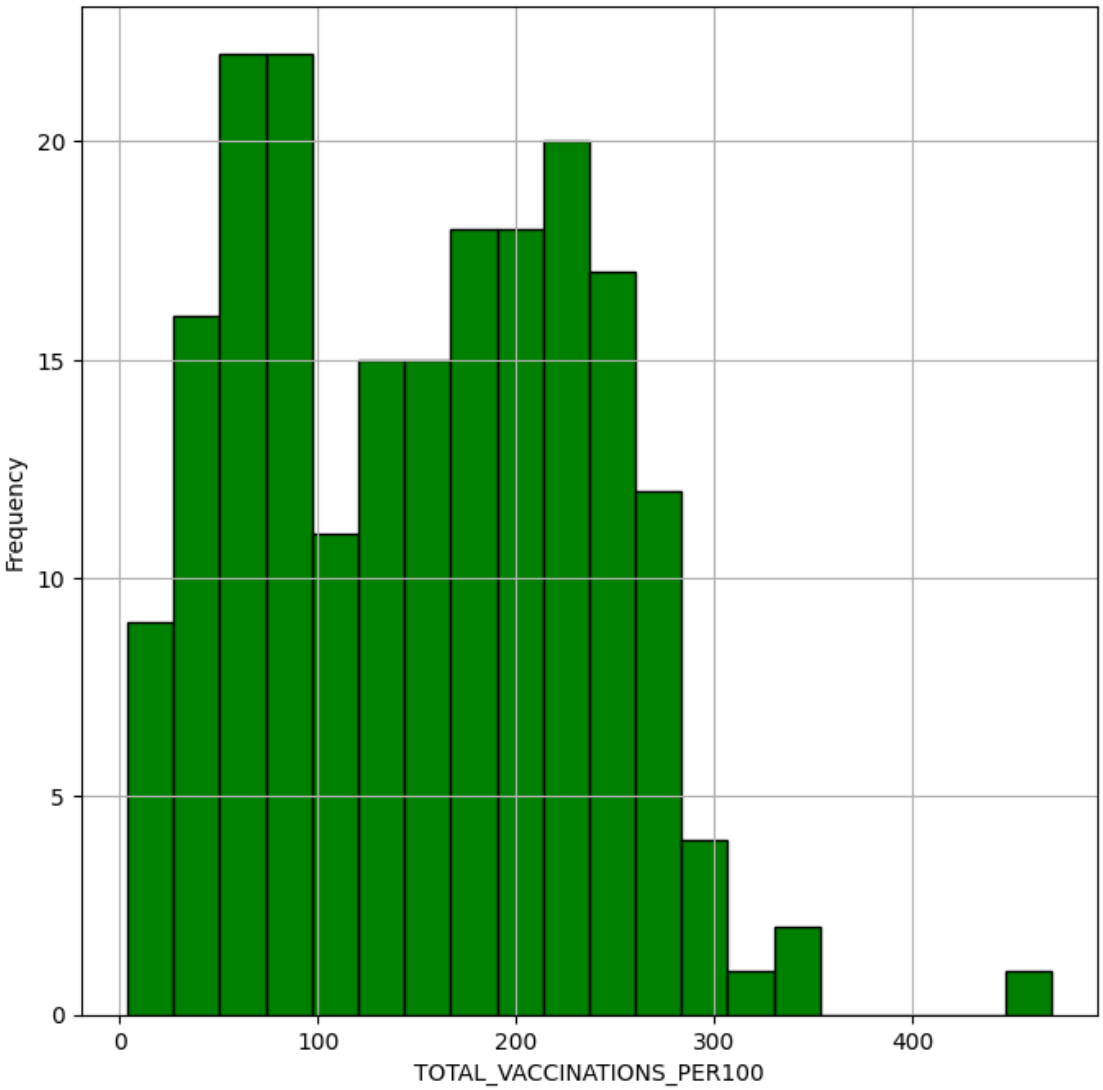
[77]:
```
df.head()
```

[77]:

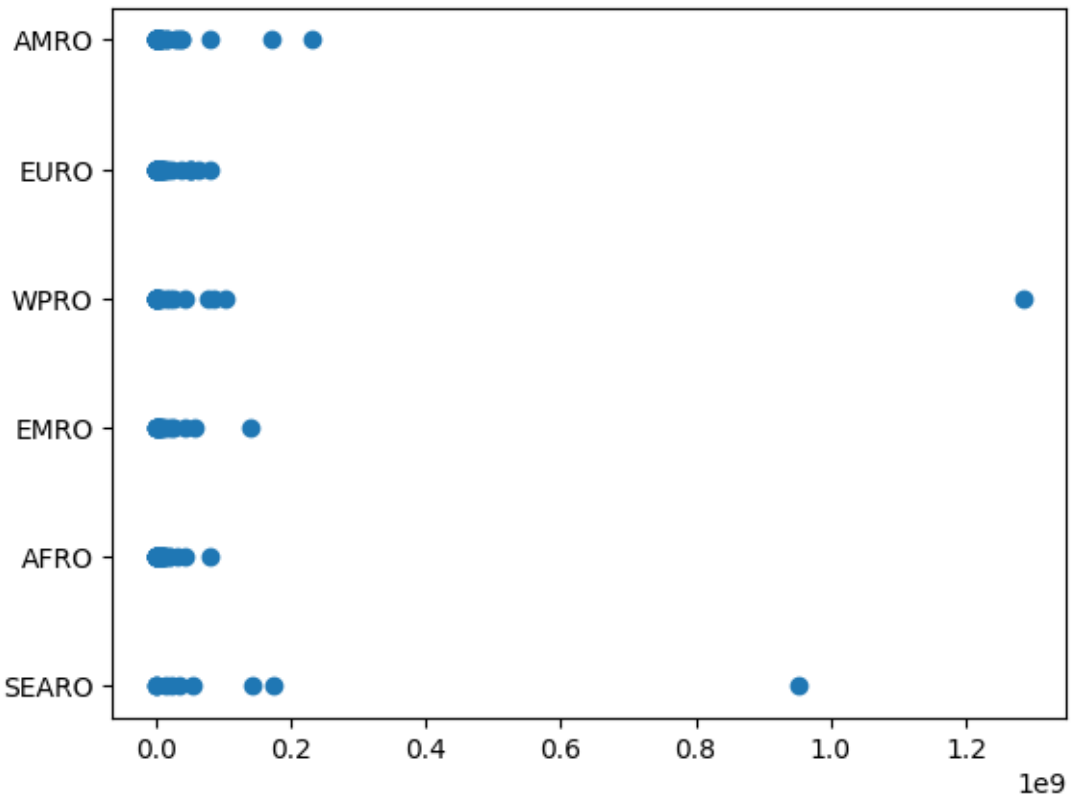
	COUNTRY	ISO3	WHO_REGION	DATA_SOURCE	DATE_UPDATED	TOTAL_VACCINATIONS	PERSONS_VACCINATED_1PLUS_DOSE
0	Bhutan	BTN	SEARO	REPORTING	2022-10-30	2011426.0	699116.0
1	Namibia	NAM	AFRO	REPORTING	2023-11-12	1005937.0	629767.0
2	Iran (Islamic Republic of)	IRN	EMRO	REPORTING	2023-11-26	155461757.0	65199831.0
3	Kenya	KEN	AFRO	REPORTING	2023-04-02	23750431.0	14494372.0
5	Comoros	COM	AFRO	REPORTING	2022-10-02	835021.0	438825.0

```
   TOTAL_VACCINATIONS_PER100  PERSONS_VACCINATED_1PLUS_DOSE_PER100  \
0                      261.0                                 91.0
1                       40.0                                 25.0
2                      185.0                                 78.0
3                       44.0                                 27.0
5                       96.0                                 50.0

   PERSONS_LAST_DOSE  PERSONS_LAST_DOSE_PER100 FIRST_VACCINE_DATE  \
0           677669.0                     88.0         2021-03-27
1           550978.0                     22.0         2021-03-19
2         58585264.0                     70.0         2021-02-09
3         11090440.0                     21.0         2021-03-05
5           397080.0                     46.0         2021-04-10

   PERSONS_BOOSTER_ADD_DOSE  PERSONS_BOOSTER_ADD_DOSE_PER100
0                  634641.0                        82.000000
1                  298560.0                        12.000000
2                31352288.0                        37.000000
3                 2000636.0                         4.000000
5                  902114.0                        32.681564
```

[89]:
```
plt.figure(figsize=(8,8))
plt.hist(df['TOTAL_VACCINATIONS_PER100'],bins=20,color='green',␣
 ↪edgecolor='black')
plt.xlabel('TOTAL_VACCINATIONS_PER100')
plt.ylabel('Frequency')
plt.grid(True)
plt.show()
```



[107]:
```
plt.scatter(df['PERSONS_LAST_DOSE'],y=df['WHO_REGION'])
plt.show()
```



[110]:
```
data = df['PERSONS_LAST_DOSE']
Q1 = np.percentile(data, 25)
Q3 = np.percentile(data, 75)

# Calculate the IQR
IQR = Q3 - Q1

# Define the lower and upper bounds for outliers
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

# Remove outliers
filtered_data = data[(data >= lower_bound) & (data <= upper_bound)]
```

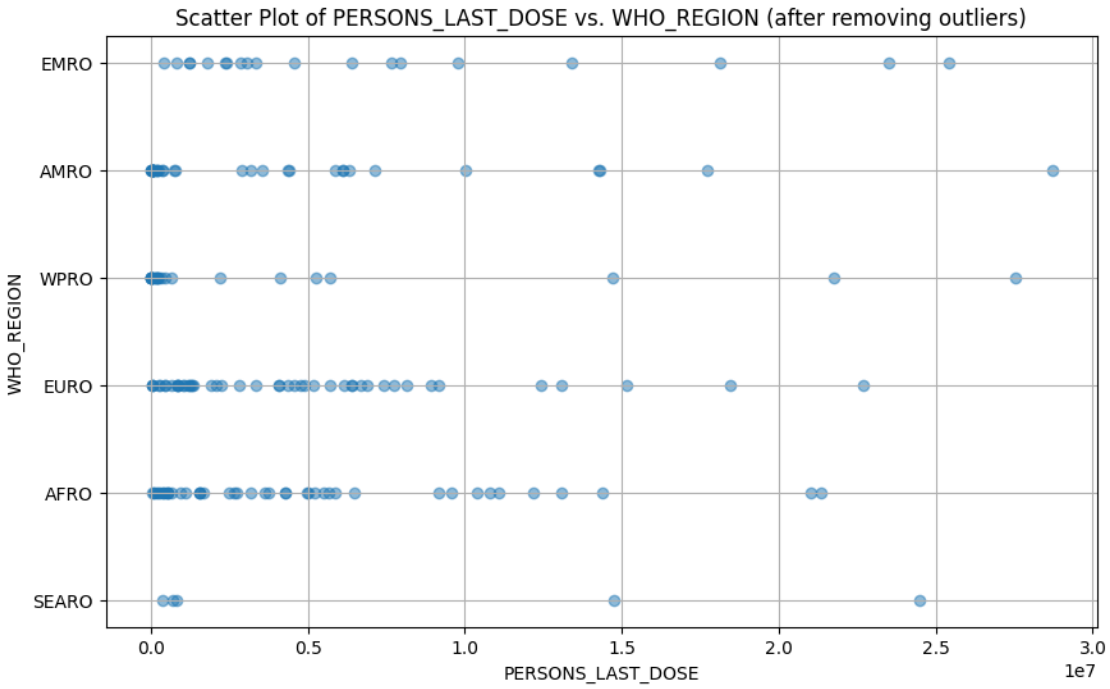
[113]:
```
len(filtered_data)
```

[113]: 174

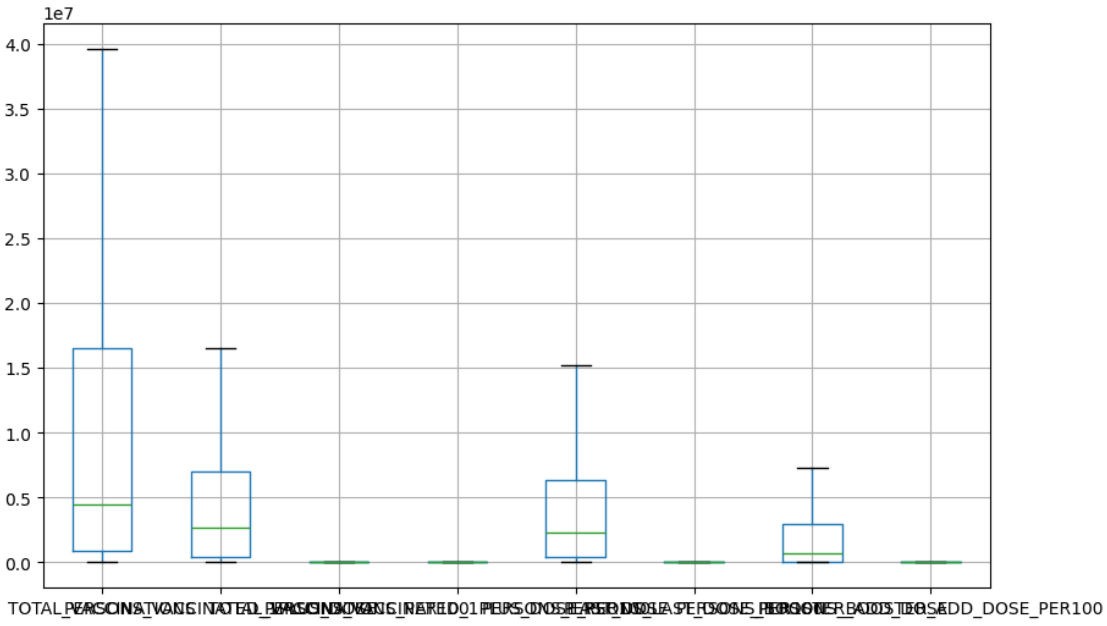
[114]:
```
len(df)
```

[114]: 203

[118]:
```
filtered_df = df[(df['PERSONS_LAST_DOSE'] >= lower_bound) &␣
 ↪(df['PERSONS_LAST_DOSE'] <= upper_bound)]
plt.figure(figsize=(10, 6))
plt.scatter(filtered_df['PERSONS_LAST_DOSE'], filtered_df['WHO_REGION'],␣
 ↪alpha=0.5)
plt.xlabel('PERSONS_LAST_DOSE')
plt.ylabel('WHO_REGION')
plt.title('Scatter Plot of PERSONS_LAST_DOSE vs. WHO_REGION (after removing␣
 ↪outliers)')
plt.grid(True)
plt.show()
```



[123]:
```
plt.figure(figsize=(10, 6))
filtered_df.boxplot(showfliers=False)
plt.show()
```



[124]:
```
df['DATA_SOURCE'].value_counts()
```

[124]:
```
REPORTING    203
Name: DATA_SOURCE, dtype: int64
```

[125]:
```
filtered_df.head()
```

[125]:

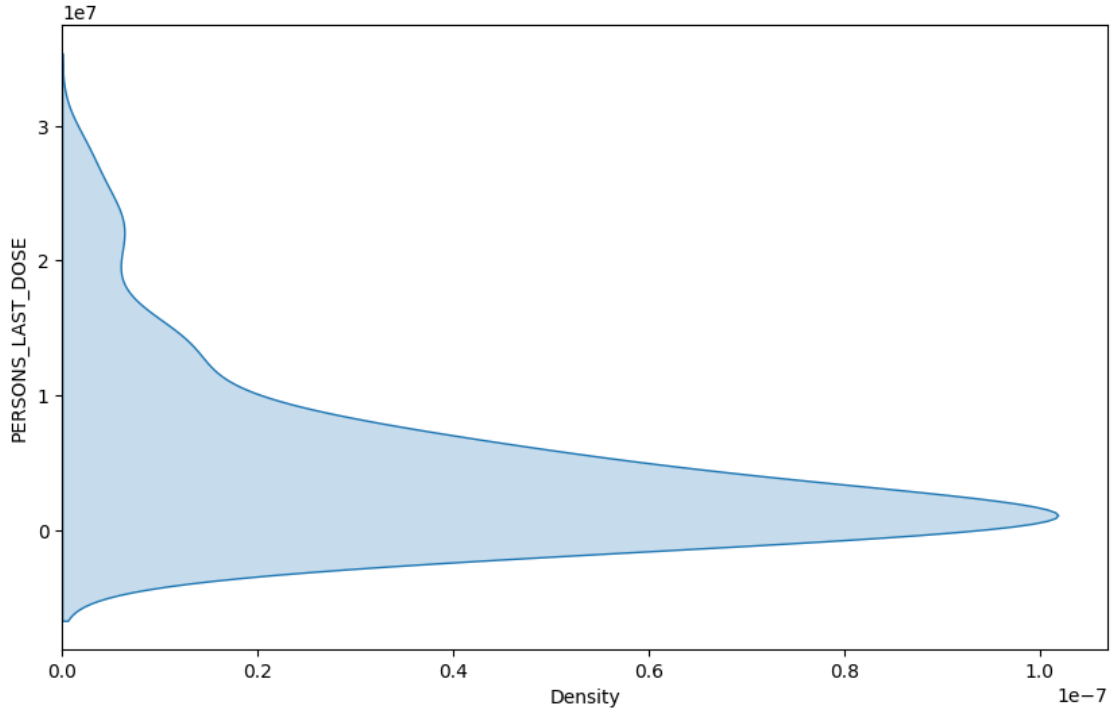
	COUNTRY	ISO3	WHO_REGION	DATA_SOURCE	DATE_UPDATED	TOTAL_VACCINATIONS	PERSONS_VACCINATED_1PLUS_DOSE	TOTAL_VACCINATIONS_PER100	PERSONS_VACCINATED_1PLUS_DOSE_PER100	PERSONS_LAST_DOSE
0	Bhutan	BTN	SEARO	REPORTING	2022-10-30	2011426.0	699116.0	261.0	91.0	677669.0
1	Namibia	NAM	AFRO	REPORTING	2023-11-12	1005937.0	629767.0	40.0	25.0	550978.0
3	Kenya	KEN	AFRO	REPORTING	2023-04-02	23750431.0	14494372.0	44.0	27.0	11090440.0
5	Comoros	COM	AFRO	REPORTING	2022-10-02	835021.0	438825.0	96.0	50.0	397080.0
6	Mozambique	MOZ	AFRO	REPORTING	2023-07-02	34950858.0	22869646.0	112.0		

```
6                                  73.0         21329745.0

   PERSONS_LAST_DOSE_PER100 FIRST_VACCINE_DATE  PERSONS_BOOSTER_ADD_DOSE  \
0                      88.0         2021-03-27                  634641.0
1                      22.0         2021-03-19                  298560.0
3                      21.0         2021-03-05                 2000636.0
5                      46.0         2021-04-10                  902114.0
6                      68.0         2021-03-08                 2323562.0

   PERSONS_BOOSTER_ADD_DOSE_PER100
0                        82.000000
1                        12.000000
3                         4.000000
5                        32.681564
6                         7.000000
```

[135]:
```
plt.figure(figsize=(10, 6))
sns.
  ↪kdeplot(filtered_df['TOTAL_VACCINATIONS'],y=filtered_df['PERSONS_LAST_DOSE'],shade=True)
```

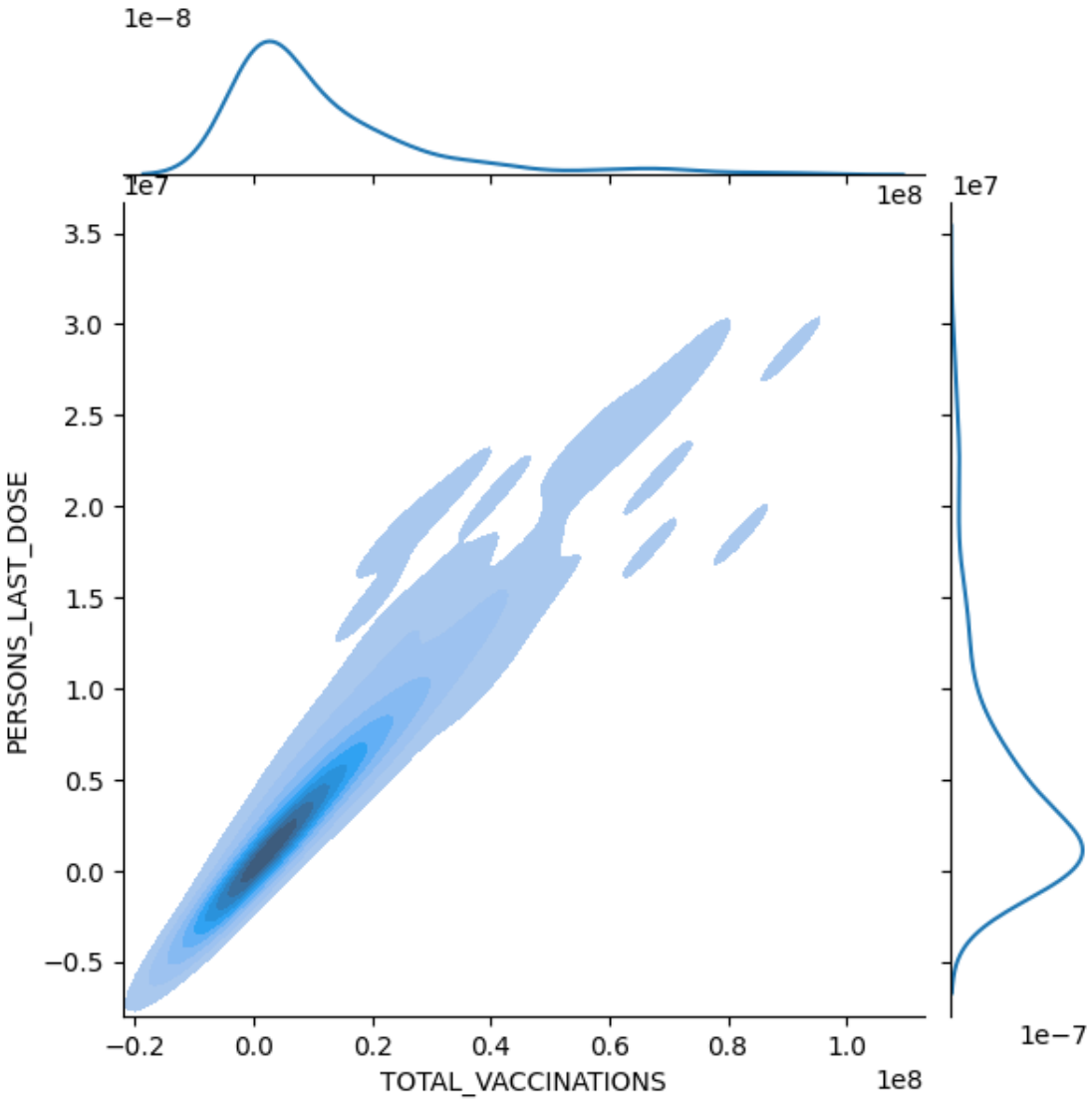
```
C:\Users\Keval Shah\AppData\Local\Temp\ipykernel_13608\4174872694.py:2:
FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(filtered_df['TOTAL_VACCINATIONS'],y=filtered_df['PERSONS_LAST_DOSE
'],shade=True)
```

[135]:
```
<Axes: xlabel='Density', ylabel='PERSONS_LAST_DOSE'>
```

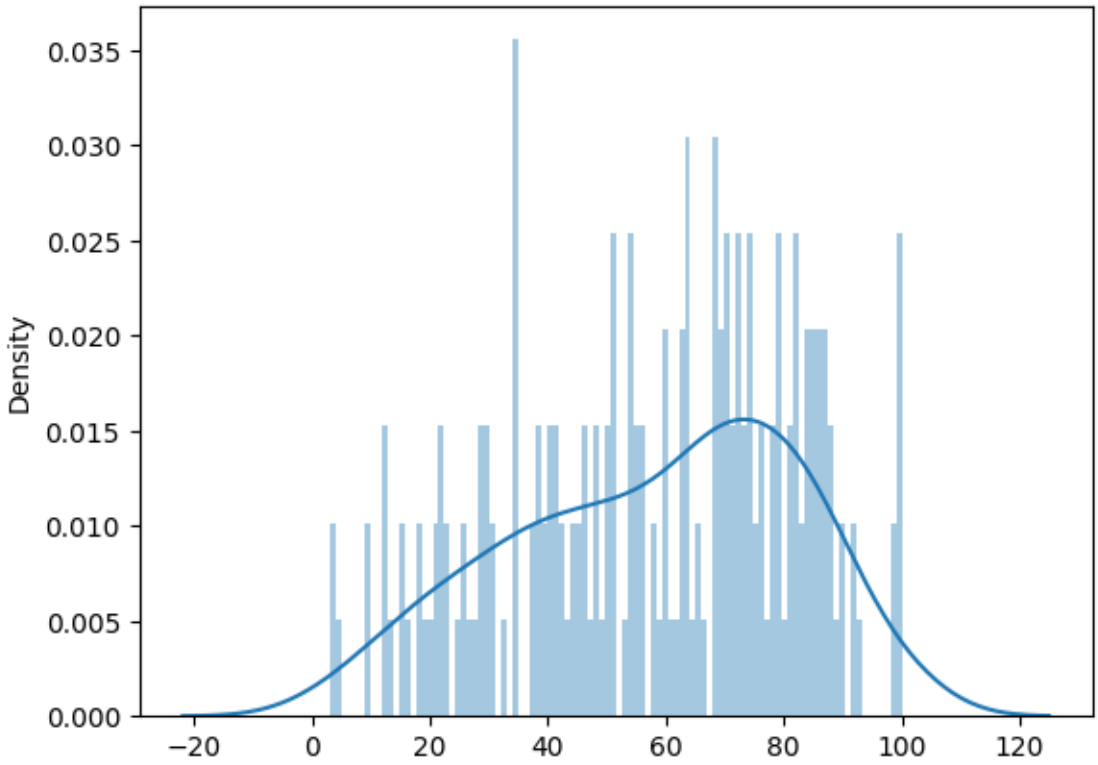


[136]:
```
import warnings
plt.figure(figsize=(10, 6))
sns.jointplot(x=filtered_df['TOTAL_VACCINATIONS'],␣
  ↪y=filtered_df['PERSONS_LAST_DOSE'], kind="kde", shade=True)
warnings.filterwarnings("ignore")
plt.show()
```

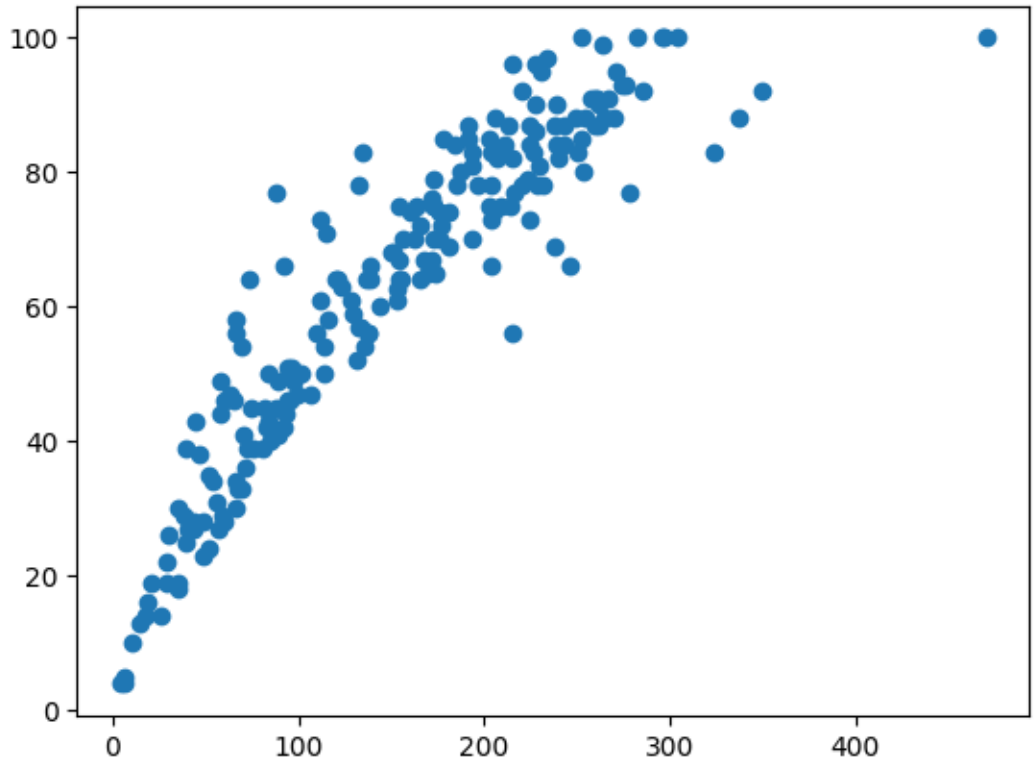
```
C:\ProgramData\anaconda3\lib\site-packages\seaborn\axisgrid.py:1826:
FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  func(x=self.x, y=self.y, **kwargs)
<Figure size 1000x600 with 0 Axes>
```



[137]:
```
sns.distplot(x=df['PERSONS_LAST_DOSE_PER100'],bins=100)
plt.show()
```



[138]:
```
plt.
  ↪scatter(df['TOTAL_VACCINATIONS_PER100'],df['PERSONS_VACCINATED_1PLUS_DOSE_PER100'])
plt.show()
```



[154]:
```
def min_max_scaling_df(df, y_min=0, y_max=1):
    scaled_df = df.copy()
    for column in scaled_df.columns:
        x = scaled_df[column]
        x_min = x.min()
        x_max = x.max()
        scaled_df[column] = ((x - x_min) / (x_max - x_min)) * (y_max - y_min) +␣
↪y_min

    return scaled_df
```

[157]:
```
scaled_df = min_max_scaling_df(numeric_columns)
scaled_df.head()
```

[157]:

	TOTAL_VACCINATIONS	PERSONS_VACCINATED_1PLUS_DOSE	\
0	0.000571	0.000529	
1	0.000285	0.000477	
2	0.044203	0.049467	
3	0.006752	0.010996	
5	0.000236	0.000332	

	TOTAL_VACCINATIONS_PER100	PERSONS_VACCINATED_1PLUS_DOSE_PER100	\

```
0                    0.551502                                0.906250
1                    0.077253                                0.218750
2                    0.388412                                0.770833
3                    0.085837                                0.239583
5                    0.197425                                0.479167

   PERSONS_LAST_DOSE  PERSONS_LAST_DOSE_PER100  PERSONS_BOOSTER_ADD_DOSE  \
0           0.000526                  0.876289                  0.000761
1           0.000428                  0.195876                  0.000358
2           0.045609                  0.690722                  0.037590
3           0.008633                  0.185567                  0.002398
5           0.000308                  0.443299                  0.001081

   PERSONS_BOOSTER_ADD_DOSE_PER100
0                         0.987805
1                         0.134146
2                         0.439024
3                         0.036585
5                         0.386361
```

[163]:
```
def z_score(df):
    new_df = pd.DataFrame()  # Create an empty DataFrame to store scaled values

    for column in df:
        x = df[column]
        sd = np.std(x)  # Calculate the standard deviation
        u = np.mean(x)  # Calculate the mean
        new_df[column] = (x - u) / sd  # Calculate Z-score and assign to new␣
↪DataFrame

    return new_df
```

[165]:
```
z_score(numeric_columns)[:5]
```

[165]:
```
   TOTAL_VACCINATIONS  PERSONS_VACCINATED_1PLUS_DOSE  \
0           -0.216587                     -0.223379
1           -0.219938                     -0.223955
2            0.294824                      0.312251
3           -0.144137                     -0.108820
5           -0.220508                     -0.225541

   TOTAL_VACCINATIONS_PER100  PERSONS_VACCINATED_1PLUS_DOSE_PER100  \
0                   1.284928                              1.183532
1                  -1.338643                             -1.586631
2                   0.382705                              0.637894
3                  -1.291158                             -1.502686
5                  -0.673847                             -0.537327
```

```
   PERSONS_LAST_DOSE  PERSONS_LAST_DOSE_PER100  PERSONS_BOOSTER_ADD_DOSE  \
0          -0.215997                  1.254907                 -0.188112
1          -0.217102                 -1.493643                 -0.193511
2           0.289160                  0.505303                  0.305287
3          -0.125161                 -1.535287                 -0.166171
5          -0.218444                 -0.494170                 -0.183816

   PERSONS_BOOSTER_ADD_DOSE_PER100
0                     2.245236e+00
1                    -9.415343e-01
2                     1.965981e-01
3                    -1.305737e+00
5                     3.234767e-16
```

[166]:
```
# feature selection
```

[167]:
```
pearson_corr = df.corr(method='pearson')
pearson_corr
```

[167]:
```
                                    TOTAL_VACCINATIONS  \
TOTAL_VACCINATIONS                            1.000000
PERSONS_VACCINATED_1PLUS_DOSE                 0.993310
TOTAL_VACCINATIONS_PER100                     0.137414
PERSONS_VACCINATED_1PLUS_DOSE_PER100          0.158810
PERSONS_LAST_DOSE                             0.995577
PERSONS_LAST_DOSE_PER100                      0.153151
PERSONS_BOOSTER_ADD_DOSE                      0.956369
PERSONS_BOOSTER_ADD_DOSE_PER100               0.089133

                                    PERSONS_VACCINATED_1PLUS_DOSE  \
TOTAL_VACCINATIONS                                       0.993310
PERSONS_VACCINATED_1PLUS_DOSE                            1.000000
TOTAL_VACCINATIONS_PER100                                0.110422
PERSONS_VACCINATED_1PLUS_DOSE_PER100                     0.143877
PERSONS_LAST_DOSE                                        0.999350
PERSONS_LAST_DOSE_PER100                                 0.135412
PERSONS_BOOSTER_ADD_DOSE                                 0.920177
PERSONS_BOOSTER_ADD_DOSE_PER100                          0.058201

                                    TOTAL_VACCINATIONS_PER100  \
TOTAL_VACCINATIONS                                   0.137414
PERSONS_VACCINATED_1PLUS_DOSE                        0.110422
TOTAL_VACCINATIONS_PER100                            1.000000
PERSONS_VACCINATED_1PLUS_DOSE_PER100                 0.913361
PERSONS_LAST_DOSE                                    0.113727
PERSONS_LAST_DOSE_PER100                             0.934663
```

```
PERSONS_BOOSTER_ADD_DOSE                          0.154730
PERSONS_BOOSTER_ADD_DOSE_PER100                   0.830581

                                  PERSONS_VACCINATED_1PLUS_DOSE_PER100  \
TOTAL_VACCINATIONS                                              0.158810
PERSONS_VACCINATED_1PLUS_DOSE                                   0.143877
TOTAL_VACCINATIONS_PER100                                       0.913361
PERSONS_VACCINATED_1PLUS_DOSE_PER100                            1.000000
PERSONS_LAST_DOSE                                               0.144470
PERSONS_LAST_DOSE_PER100                                        0.982525
PERSONS_BOOSTER_ADD_DOSE                                        0.163304
PERSONS_BOOSTER_ADD_DOSE_PER100                                 0.690379

                                  PERSONS_LAST_DOSE  \
TOTAL_VACCINATIONS                         0.995577
PERSONS_VACCINATED_1PLUS_DOSE              0.999350
TOTAL_VACCINATIONS_PER100                  0.113727
PERSONS_VACCINATED_1PLUS_DOSE_PER100       0.144470
PERSONS_LAST_DOSE                          1.000000
PERSONS_LAST_DOSE_PER100                   0.138416
PERSONS_BOOSTER_ADD_DOSE                   0.929985
PERSONS_BOOSTER_ADD_DOSE_PER100            0.064565

                                  PERSONS_LAST_DOSE_PER100  \
TOTAL_VACCINATIONS                                0.153151
PERSONS_VACCINATED_1PLUS_DOSE                     0.135412
TOTAL_VACCINATIONS_PER100                         0.934663
PERSONS_VACCINATED_1PLUS_DOSE_PER100              0.982525
PERSONS_LAST_DOSE                                 0.138416
PERSONS_LAST_DOSE_PER100                          1.000000
PERSONS_BOOSTER_ADD_DOSE                          0.161698
PERSONS_BOOSTER_ADD_DOSE_PER100                   0.744110

                                  PERSONS_BOOSTER_ADD_DOSE  \
TOTAL_VACCINATIONS                                0.956369
PERSONS_VACCINATED_1PLUS_DOSE                     0.920177
TOTAL_VACCINATIONS_PER100                         0.154730
PERSONS_VACCINATED_1PLUS_DOSE_PER100              0.163304
PERSONS_LAST_DOSE                                 0.929985
PERSONS_LAST_DOSE_PER100                          0.161698
PERSONS_BOOSTER_ADD_DOSE                          1.000000
PERSONS_BOOSTER_ADD_DOSE_PER100                   0.130992

                                  PERSONS_BOOSTER_ADD_DOSE_PER100
TOTAL_VACCINATIONS                                       0.089133
PERSONS_VACCINATED_1PLUS_DOSE                            0.058201
TOTAL_VACCINATIONS_PER100                                0.830581
```

```
PERSONS_VACCINATED_1PLUS_DOSE_PER100                    0.690379
PERSONS_LAST_DOSE                                       0.064565
PERSONS_LAST_DOSE_PER100                                0.744110
PERSONS_BOOSTER_ADD_DOSE                                0.130992
PERSONS_BOOSTER_ADD_DOSE_PER100                         1.000000
```

[170]:
```
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
scaled_data = scaler.fit_transform(numeric_columns)

pca = PCA()
pca.fit(scaled_data)

transformed_data = pca.transform(scaled_data)

explained_variance_ratio = pca.explained_variance_ratio_

print("Explained variance ratio:", explained_variance_ratio)

transformed_df = pd.DataFrame(transformed_data, columns=[f"PC{i+1}" for i in␣
 ↪range(transformed_data.shape[1])])

print(transformed_df.head())
```

```
Explained variance ratio: [5.34278716e-01 3.97961815e-01 4.62039235e-02
1.18567995e-02
 7.74948222e-03 1.79319673e-03 1.26626194e-04 2.94410582e-05]
        PC1       PC2       PC3       PC4       PC5       PC6       PC7  \
0  1.362167 -2.643822  0.937897  0.170006 -0.300025  0.050990 -0.006843
1 -1.933139  1.929535  0.378068 -0.026404  0.060731 -0.049395  0.005927
2  1.002469 -0.350420 -0.262186 -0.024562 -0.077009  0.073070 -0.007781
3 -1.872600  2.124823  0.050388 -0.015853  0.196221  0.013609 -0.007493
4 -0.863275  0.455371  0.379032  0.016708 -0.266847 -0.009913  0.011258

        PC8
0 -0.003159
1 -0.000950
2  0.013274
3  0.008393
4 -0.002899
```

[ ]: