SPRAWOZDANIE

Zajęcia: Nauka o danych I

Prowadzący: prof. dr hab. Vasyl Martsenyuk

Laboratorium Nr 4 Data 9.11.2024 Temat: "Wizualizacja danych za pomocą biblioteki Matplotlib" Wariant 6	Dawid Klimek Informatyka II stopień, niestacjonarne, 1semestr, gr.1A

1. Polecenie: wariant 6 zadania

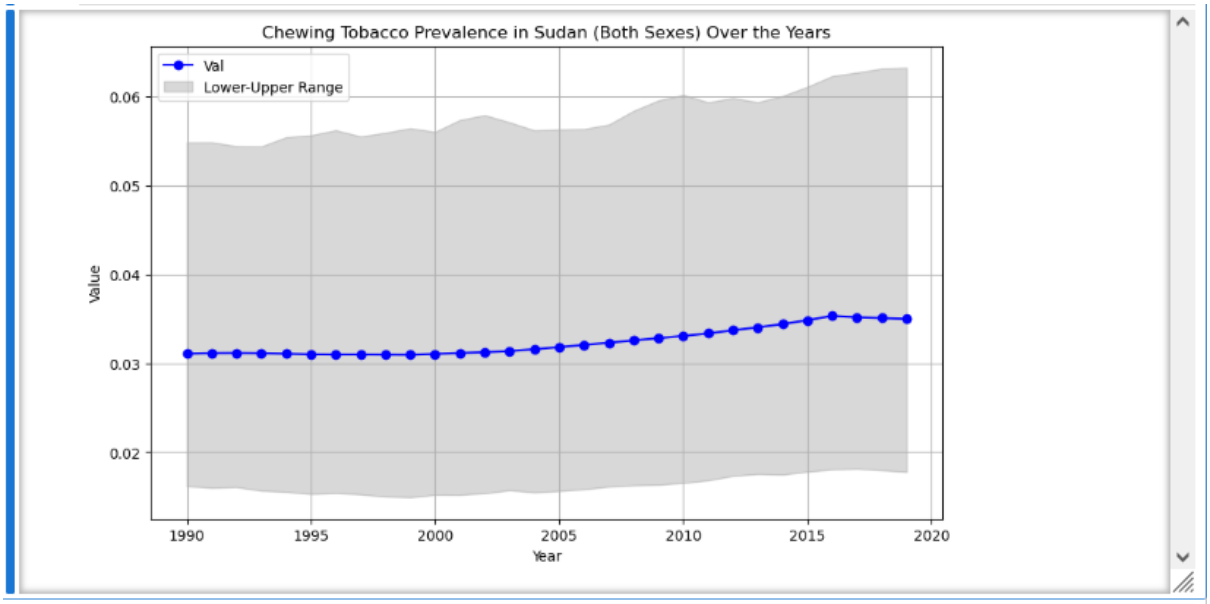
Zadanie dotyczy tworzenia wszystkich możliwych wykresów w celu eksploracji zbioru danych

2. Opis programu opracowanego (kody źródłowe, rzuty ekranu)

[2]:
```
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv('IHME_GBD_2019_CHEWING_TOB_1990_2019_DATA_Y2021M05D27.CSV', encoding='latin1')

print(df.head())
```

```
   measure_id measure_name  location_id location_name  sex_id sex_name  \
0           5   Prevalence            1        Global       1     Male
1           5   Prevalence            1        Global       2   Female
2           5   Prevalence            1        Global       1     Male
3           5   Prevalence            1        Global       2   Female
4           5   Prevalence            1        Global       1     Male

   age_group_id age_group_name  rei_id         rei_name  metric_id  \
0             8       15 to 19     332  Chewing tobacco          3
1             8       15 to 19     332  Chewing tobacco          3
2             8       15 to 19     332  Chewing tobacco          3
3             8       15 to 19     332  Chewing tobacco          3
4             8       15 to 19     332  Chewing tobacco          3

  metric_name  year_id       val     upper     lower
0        Rate     1990  0.038740  0.055586  0.027147
1        Rate     1990  0.011356  0.017594  0.007779
2        Rate     1991  0.039253  0.055838  0.027608
3        Rate     1991  0.011516  0.017807  0.007906
4        Rate     1992  0.039863  0.056448  0.027800
```

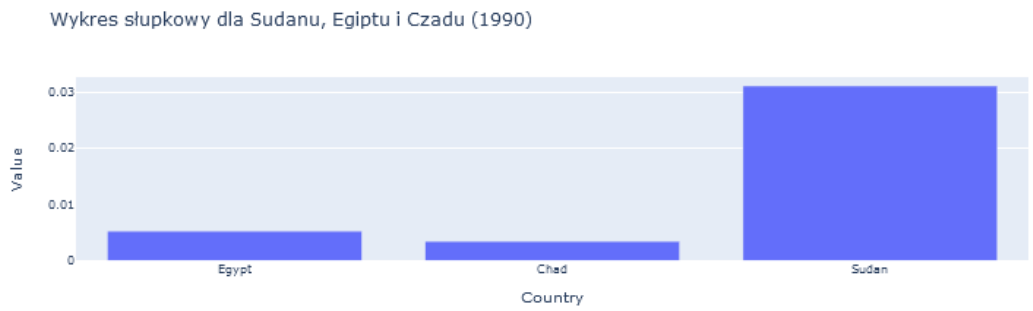
[3]:
```
# Filter data for Sudan and sex_name == 'Both'
sudan_data = df[(df['location_name'] == 'Sudan') & (df['sex_name'] == 'Both')& (df['age_group_name'] == '15 to 19')]

# Extract the relevant columns
years = sudan_data['year_id']
val = sudan_data['val']
lower = sudan_data['lower']
upper = sudan_data['upper']

# Create the plot
plt.figure(figsize=(10, 6))
plt.plot(years, val, label='Val', marker='o', color='blue')
plt.fill_between(years, lower, upper, color='gray', alpha=0.3, label='Lower-Upper Range')

# Add titles and labels
# Add titles and labels
plt.title('Chewing Tobacco Prevalence in Sudan (Both Sexes) Over the Years')
plt.xlabel('Year')
plt.ylabel('Value')
plt.legend()
plt.grid()

# Show the plot
plt.show()
```



[4]:
```
! pip install plotly
```

```
Requirement already satisfied: plotly in c:\users\tomasz 2115\anaconda3\lib\site-packages (5.24.1)
Requirement already satisfied: tenacity>=6.2.0 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from plotly) (8.2.3)
Requirement already satisfied: packaging in c:\users\tomasz 2115\anaconda3\lib\site-packages (from plotly) (24.1)
```

[5]:
```
#wykres słupkowy

import plotly.express as px

# Ensure numeric filtering for year and correct string filtering
data = df[
    (df['location_name'].isin(['Sudan', 'Egypt', 'Chad'])) &
    (df['sex_name'] == 'Both') &
    (df['age_group_name'] == '15 to 19') &
    (df['year_id'] == 1990)  # Note: Use numeric value for 'year_id'
]

# Plot the bar chart
fig = px.bar(
    data_frame=data,
    x='location_name',
    y='val',
    title='Wykres słupkowy dla Sudanu, Egiptu i Czadu (1990)',
    labels={'location_name': 'Country', 'val': 'Value'}
)

fig.show()
```

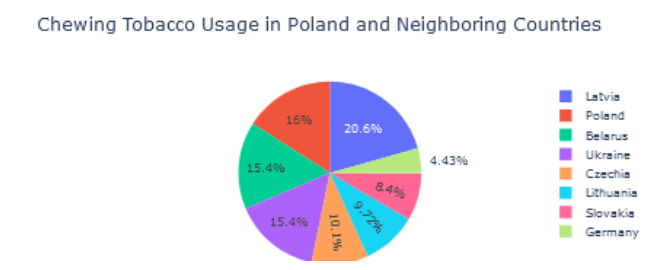


[6]:
```
# Get the unique elements of the 'location_name' column
unique_locations = df['location_name'].unique()

# Print the unique elements
print(unique_locations)
```

```
['Global' 'China' "Democratic People's Republic of Korea"
 'Taiwan (Province of China)' 'Cambodia' 'Indonesia'
 "Lao People's Democratic Republic" 'Malaysia' 'Maldives' 'Myanmar'
 'Philippines' 'Sri Lanka' 'Thailand' 'Timor-Leste' 'Viet Nam' 'Fiji'
 'Kiribati' 'Marshall Islands' 'Micronesia (Federated States of)'
 'Papua New Guinea' 'Samoa' 'Solomon Islands' 'Tonga' 'Vanuatu' 'Armenia'
 'Azerbaijan' 'Georgia' 'Kazakhstan' 'Kyrgyzstan' 'Mongolia' 'Tajikistan'
 'Turkmenistan' 'Uzbekistan' 'Albania' 'Bosnia and Herzegovina' 'Bulgaria'
 'Croatia' 'Czechia' 'Hungary' 'North Macedonia' 'Montenegro' 'Poland'
 'Romania' 'Serbia' 'Slovakia' 'Slovenia' 'Belarus' 'Estonia' 'Latvia'
 'Lithuania' 'Republic of Moldova' 'Russian Federation' 'Ukraine'
 'Brunei Darussalam' 'Japan' 'Republic of Korea' 'Singapore' 'Australia'
 'New Zealand' 'Andorra' 'Austria' 'Belgium' 'Cyprus' 'Denmark' 'Finland'
 'France' 'Germany' 'Greece' 'Iceland' 'Ireland' 'Israel' 'Italy'
 'Luxembourg' 'Malta' 'Netherlands' 'Norway' 'Portugal' 'Spain' 'Sweden'
 'Switzerland' 'United Kingdom' 'Argentina' 'Chile' 'Uruguay' 'Canada'
 'United States of America' 'Antigua and Barbuda' 'Bahamas' 'Barbados'
 'Belize' 'Cuba' 'Dominica' 'Dominican Republic' 'Grenada' 'Guyana'
 'Haiti' 'Jamaica' 'Saint Lucia' 'Saint Vincent and the Grenadines'
 'Suriname' 'Trinidad and Tobago' 'Bolivia (Plurinational State of)'
 'Ecuador' 'Peru' 'Colombia' 'Costa Rica' 'El Salvador' 'Guatemala'
 'Honduras' 'Mexico' 'Nicaragua' 'Panama'
 'Venezuela (Bolivarian Republic of)' 'Brazil' 'Paraguay' 'Algeria'
 'Bahrain' 'Egypt' 'Iran (Islamic Republic of)' 'Iraq' 'Jordan' 'Kuwait'
 'Lebanon' 'Libya' 'Morocco' 'Palestine' 'Oman' 'Qatar' 'Saudi Arabia'
 'Syrian Arab Republic' 'Tunisia' 'Turkey' 'United Arab Emirates' 'Yemen'
 'Afghanistan' 'Bangladesh' 'Bhutan' 'India' 'Nepal' 'Pakistan' 'Angola'
 'Central African Republic' 'Congo' 'Democratic Republic of the Congo'
 'Equatorial Guinea' 'Gabon' 'Burundi' 'Comoros' 'Djibouti' 'Eritrea'
 'Ethiopia' 'Kenya' 'Madagascar' 'Malawi' 'Mauritius' 'Mozambique'
 'Rwanda' 'Seychelles' 'Somalia' 'United Republic of Tanzania' 'Uganda'
 'Zambia' 'Botswana' 'Lesotho' 'Namibia' 'South Africa' 'Eswatini'
 'Zimbabwe' 'Benin' 'Burkina Faso' 'Cameroon' 'Cabo Verde' 'Chad'
 "Côte d'Ivoire" 'Gambia' 'Ghana' 'Guinea' 'Guinea-Bissau' 'Liberia'
 'Mali' 'Mauritania' 'Niger' 'Nigeria' 'Sao Tome and Principe' 'Senegal'
 'Sierra Leone' 'Togo' 'American Samoa' 'Bermuda' 'Cook Islands'
 'Greenland' 'Guam' 'Monaco' 'Nauru' 'Niue' 'Northern Mariana Islands'
 'Palau' 'Puerto Rico' 'Saint Kitts and Nevis' 'San Marino' 'Tokelau'
 'Tuvalu' 'United States Virgin Islands' 'South Sudan' 'Sudan']
```

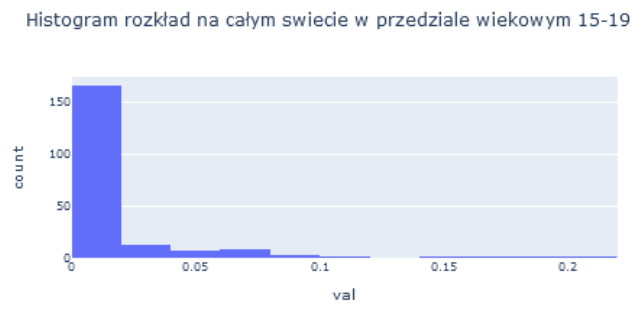
[7]:
```
#Wykres kołowy

data = df[
    (df['location_name'].isin(['Poland','Czechia','Germany','Ukraine','Slovakia','Belarus','Lithuania','Latvia'])) &
    (df['sex_name'] == 'Both') &
    (df['age_group_name'] == '15 to 19') &
    (df['year_id'] == 1990)  # Note: Use numeric value for 'year_id'
]
print(data)

fig = px.pie(data,values='val', names='location_name',title = 'Chewing Tobacco Usage in Poland and Neighboring Countries')
fig.show()
```

```
        measure_id measure_name  location_id location_name  sex_id sex_name  \
63330            5   Prevalence           47       Czechia       3     Both
70170            5   Prevalence           51        Poland       3     Both
75300            5   Prevalence           54      Slovakia       3     Both
78720            5   Prevalence           57       Belarus       3     Both
82140            5   Prevalence           59        Latvia       3     Both
83850            5   Prevalence           60     Lithuania       3     Both
88980            5   Prevalence           63       Ukraine       3     Both
112920           5   Prevalence           81       Germany       3     Both

        age_group_id age_group_name  rei_id         rei_name  metric_id  \
63330              8       15 to 19     332  Chewing tobacco          3
70170              8       15 to 19     332  Chewing tobacco          3
75300              8       15 to 19     332  Chewing tobacco          3
78720              8       15 to 19     332  Chewing tobacco          3
82140              8       15 to 19     332  Chewing tobacco          3
83850              8       15 to 19     332  Chewing tobacco          3
88980              8       15 to 19     332  Chewing tobacco          3
112920             8       15 to 19     332  Chewing tobacco          3

       metric_name  year_id       val     upper     lower
63330         Rate     1990  0.003493  0.005706  0.001823
70170         Rate     1990  0.005542  0.009740  0.003000
75300         Rate     1990  0.002915  0.004822  0.001589
78720         Rate     1990  0.005352  0.009083  0.002938
82140         Rate     1990  0.007153  0.011938  0.003819
83850         Rate     1990  0.003371  0.005831  0.001826
88980         Rate     1990  0.005337  0.009058  0.002940
112920        Rate     1990  0.001537  0.002682  0.000790
```



[8]:
```
#Histogram
data = df[(df['sex_name'] == 'Both') &
    (df['age_group_name'] == '15 to 19') &
    (df['year_id'] == 1990)
]

print(data)

fig = px.histogram(data,
                   x='val',
                   nbins=20,
                   title='Histogram rozkład na całym swiecie w przedziale wiekowym 15-19 lata 1990',
                   labels={'x':'wartości','y':'ilosc krajów'}
                  )
fig.show()
```

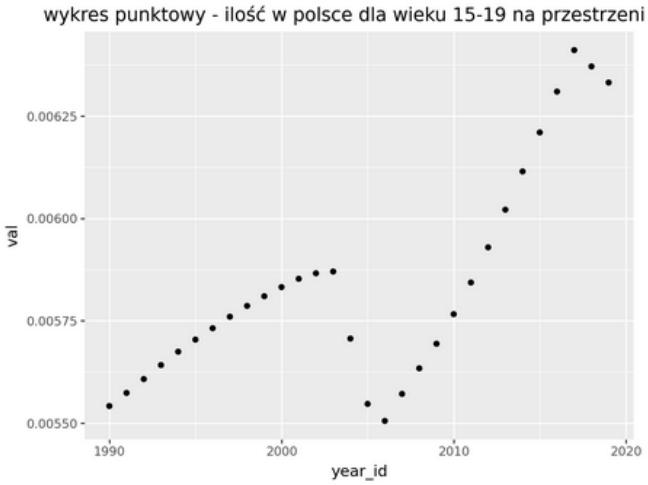
```
       measure_id measure_name  location_id  \
60              5   Prevalence            1
1770            5   Prevalence            6
3480            5   Prevalence            7
5190            5   Prevalence            8
6900            5   Prevalence           10
...           ...          ...          ...
342060          5   Prevalence          413
343770          5   Prevalence          416
345480          5   Prevalence          422
347190          5   Prevalence          435
348900          5   Prevalence          522

                              location_name  sex_id sex_name  age_group_id  \
60                                   Global       3     Both             8
1770                                  China       3     Both             8
3480    Democratic People's Republic of Korea       3     Both             8
5190                Taiwan (Province of China)       3     Both             8
6900                                 Cambodia       3     Both             8
...                                       ...     ...      ...           ...
342060                               Tokelau       3     Both             8
343770                                Tuvalu       3     Both             8
345480          United States Virgin Islands       3     Both             8
347190                           South Sudan       3     Both             8
348900                                 Sudan       3     Both             8

      age_group_name  rei_id         rei_name  metric_id metric_name  \
60           15 to 19     332  Chewing tobacco          3        Rate
1770         15 to 19     332  Chewing tobacco          3        Rate
3480         15 to 19     332  Chewing tobacco          3        Rate
5190         15 to 19     332  Chewing tobacco          3        Rate
6900         15 to 19     332  Chewing tobacco          3        Rate
...               ...     ...              ...        ...         ...
342060       15 to 19     332  Chewing tobacco          3        Rate
343770       15 to 19     332  Chewing tobacco          3        Rate
345480       15 to 19     332  Chewing tobacco          3        Rate
347190       15 to 19     332  Chewing tobacco          3        Rate
348900       15 to 19     332  Chewing tobacco          3        Rate

       year_id       val     upper     lower
60        1990  0.025274  0.033979  0.019034
1770      1990  0.005990  0.010125  0.003250
3480      1990  0.002148  0.003806  0.001134
5190      1990  0.005991  0.010116  0.003320
6900      1990  0.007599  0.012943  0.004069
...        ...       ...       ...       ...
342060    1990  0.071730  0.118773  0.040470
343770    1990  0.070196  0.113344  0.038758
345480    1990  0.002382  0.004043  0.001267
347190    1990  0.013322  0.022557  0.007502
348900    1990  0.031067  0.054843  0.016167

[205 rows x 16 columns]
```



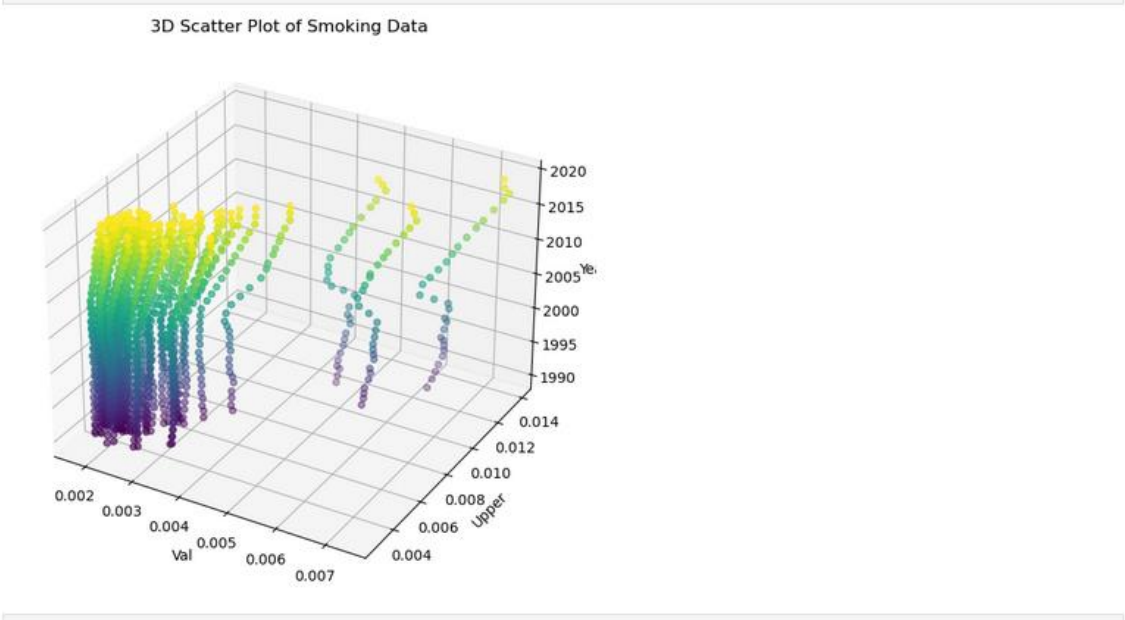
[9]:
```
!pip install plotnine
```

```
Requirement already satisfied: plotnine in c:\users\tomasz 2115\anaconda3\lib\site-packages (0.14.5)
Requirement already satisfied: matplotlib>=3.8.0 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from plotnine) (3.9.2)
Requirement already satisfied: pandas>=2.2.0 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from plotnine) (2.2.2)
Requirement already satisfied: mizani~=0.13.0 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from plotnine) (0.13.1)
Requirement already satisfied: numpy>=1.23.5 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from plotnine) (1.26.4)
Requirement already satisfied: scipy>=1.8.0 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from plotnine) (1.13.1)
Requirement already satisfied: statsmodels>=0.14.0 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from plotnine) (0.14.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from matplotlib>=3.8.0->plotnine) (1.2.0)
Requirement already satisfied: cycler>=0.10 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from matplotlib>=3.8.0->plotnine) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from matplotlib>=3.8.0->plotnine) (4.51.0)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from matplotlib>=3.8.0->plotnine) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from matplotlib>=3.8.0->plotnine) (24.1)
Requirement already satisfied: pillow>=8 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from matplotlib>=3.8.0->plotnine) (10.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from matplotlib>=3.8.0->plotnine) (3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from matplotlib>=3.8.0->plotnine) (2.9.0.post
0)
Requirement already satisfied: tzdata in c:\users\tomasz 2115\anaconda3\lib\site-packages (from mizani~=0.13.0->plotnine) (2023.3)
Requirement already satisfied: pytz>=2020.1 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from pandas>=2.2.0->plotnine) (2024.1)
Requirement already satisfied: patsy>=0.5.6 in c:\users\tomasz 2115\anaconda3\lib\site-packages (from statsmodels>=0.14.0->plotnine) (0.5.6)
Requirement already satisfied: six in c:\users\tomasz 2115\anaconda3\lib\site-packages (from patsy>=0.5.6->statsmodels>=0.14.0->plotnine) (1.16.0)
```

[10]:
```
#Scatter plot

from plotnine import *
data = data = df[
    (df['location_name'].isin(['Poland'])) &
    (df['sex_name'] == 'Both') &
    (df['age_group_name'] == '15 to 19')
]


(ggplot(data) + aes(x='year_id', y='val')+geom_point()+ggtitle("wykres punktowy - ilość w polsce dla wieku 15-19 na przestrzeni lat"))
```



[30]:
```
#wykres 3D
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
# Filtrowanie danych dla 'Prevalence'
df2=df
data = df[(df['measure_name'] == 'Prevalence') &
                  (df['location_name'] == 'Poland')]

x = data['val']  # Oś X
y = data['upper']  # Oś Y
z = data['year_id']  # Oś Z

# Tworzenie wykresu 3D
fig = plt.figure(figsize=(10, 7))
ax = fig.add_subplot(111, projection='3d')

# Rysowanie punktów na wykresie 3D
ax.scatter(x, y, z, c=z, cmap='viridis', marker='o')

# Ustawienia etykiet osi
ax.set_xlabel('Val')
ax.set_ylabel('Upper')
ax.set_zlabel('Year')

# Tytuł wykresu
ax.set_title('3D Scatter Plot of Smoking Data')

# Wyświetlenie wykresu
plt.show()
```



3. Wnioski

Biblioteki Matplotlib, plotly oraz plotnine pozwalają zobrazować posiadane dane jednak należy wziąć po uwagę, że im bardziej skomplikowaną wizualizację chcemy wykonać tym dłużej może ona się tworzyć. Matplotlib pozwala tworzyć najprostsze i najmniej ładne wykresy. Im bardziej zaawansowana biblioteka tym więcej możliwości interakcji z danymi.