healthexp_Dataset

By Hatem Elgenedy

November 1, 2025

[105]:
```
import numpy as np
import seaborn as sns
import pandas as pd
import matplotlib.pyplot as plt
```

[106]:
```
print(sns.get_dataset_names())
```

```
['anagrams', 'anscombe', 'attention', 'brain_networks', 'car_crashes',
'diamonds', 'dots', 'dowjones', 'exercise', 'flights', 'fmri', 'geyser', 'glue',
'healthexp', 'iris', 'mpg', 'penguins', 'planets', 'seaice', 'taxis', 'tips',
'titanic']
```

[107]:
```
df = sns.load_dataset("healthexp")
df.head()
```

[107]:

	Year	Country	Spending_USD	Life_Expectancy
0	1970	Germany	252.311	70.6
1	1970	France	192.143	72.2
2	1970	Great Britain	123.993	71.9
3	1970	Japan	150.437	72.0
4	1970	USA	326.961	70.9

[108]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 274 entries, 0 to 273
Data columns (total 4 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Year             274 non-null    int64
 1   Country          274 non-null    object
 2   Spending_USD     274 non-null    float64
 3   Life_Expectancy  274 non-null    float64
dtypes: float64(2), int64(1), object(1)
memory usage: 8.7+ KB
```

[109]:
```
df.shape
```

[109]: (274, 4)

[110]:
```
df.isnull().sum()
```

[110]:
```
Year              0
Country           0
Spending_USD      0
Life_Expectancy   0
dtype: int64
```

[44]:
```
df.Year
```

[44]:
```
0      1970
1      1970
2      1970
3      1970
4      1970
       …
269    2020
270    2020
271    2020
272    2020
273    2020
Name: Year, Length: 274, dtype: int64
```

[45]:
```
df.Country
```

[45]:
```
0            Germany
1             France
2      Great Britain
3              Japan
4                USA
           …
269          Germany
270           France
271    Great Britain
272            Japan
273              USA
Name: Country, Length: 274, dtype: object
```

[46]:
```
df.Life_Expectancy
```

[46]:
```
0      70.6
1      72.2
2      71.9
3      72.0
4      70.9
       …
269    81.1
270    82.3
```

```
271    80.4
272    84.7
273    77.0
Name: Life_Expectancy, Length: 274, dtype: float64
```

[47]:
```
df.Spending_USD
```

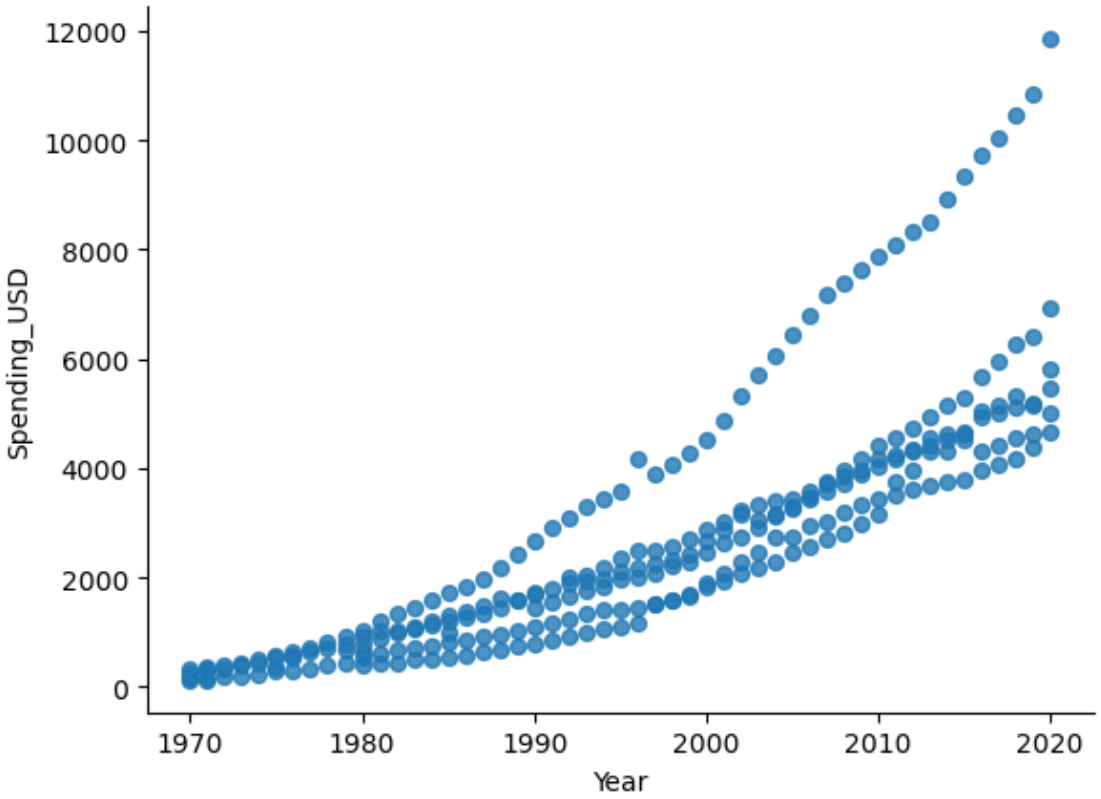
[47]:
```
0        252.311
1        192.143
2        123.993
3        150.437
4        326.961
          …
269     6938.983
270     5468.418
271     5018.700
272     4665.641
273    11859.179
Name: Spending_USD, Length: 274, dtype: float64
```

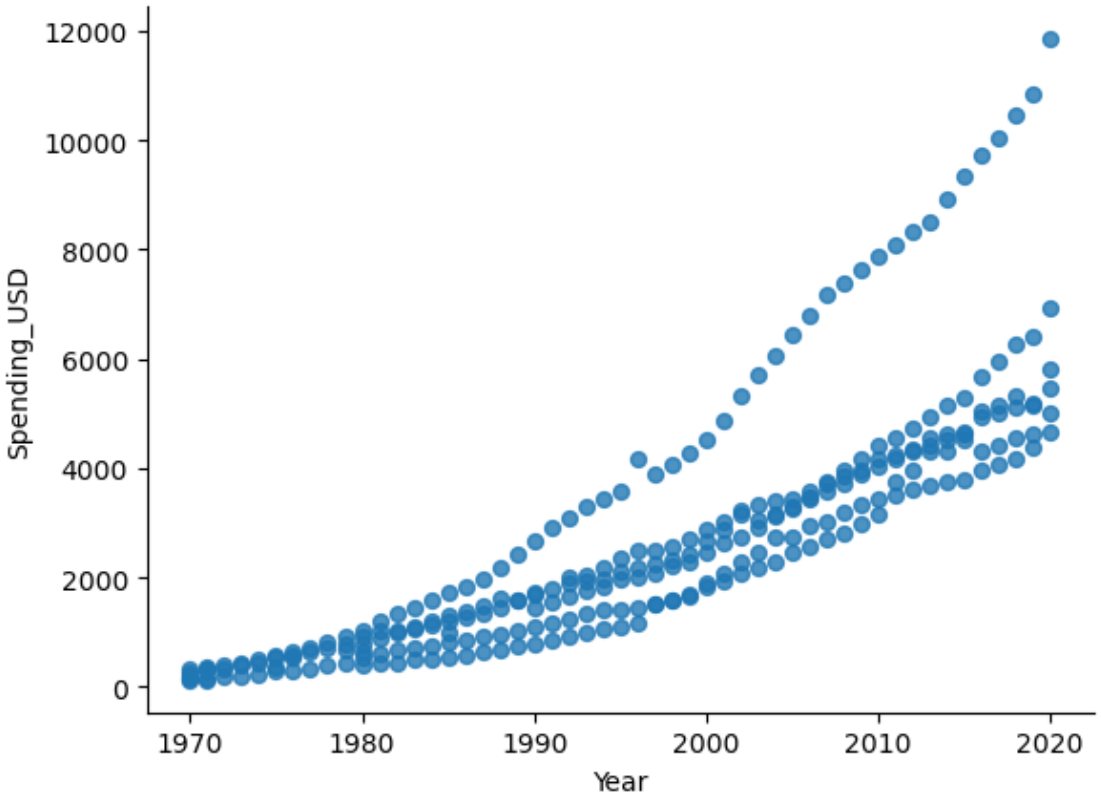
[48]:
```
df.describe()
```

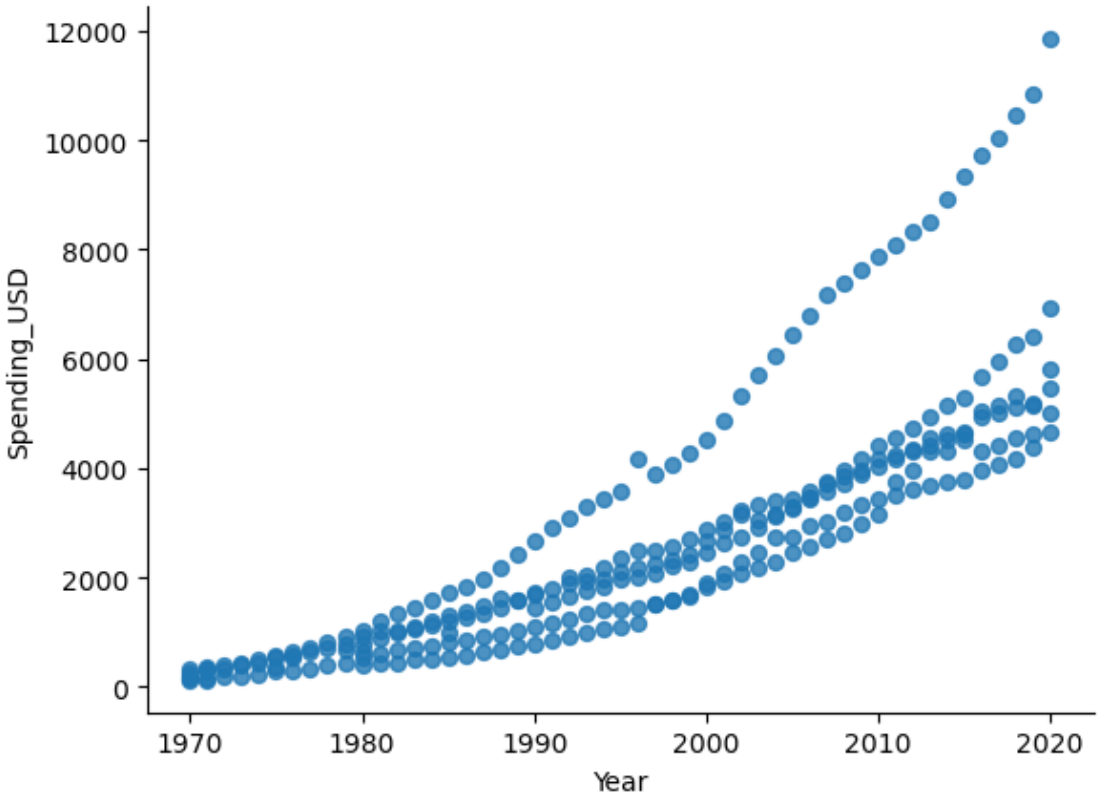
[48]:

	Year	Spending_USD	Life_Expectancy
count	274.000000	274.000000	274.000000
mean	1996.992701	2789.338905	77.909489
std	14.180933	2194.939785	3.276263
min	1970.000000	123.993000	70.600000
25%	1985.250000	1038.357000	75.525000
50%	1998.000000	2295.578000	78.100000
75%	2009.000000	4055.610000	80.575000
max	2020.000000	11859.179000	84.700000

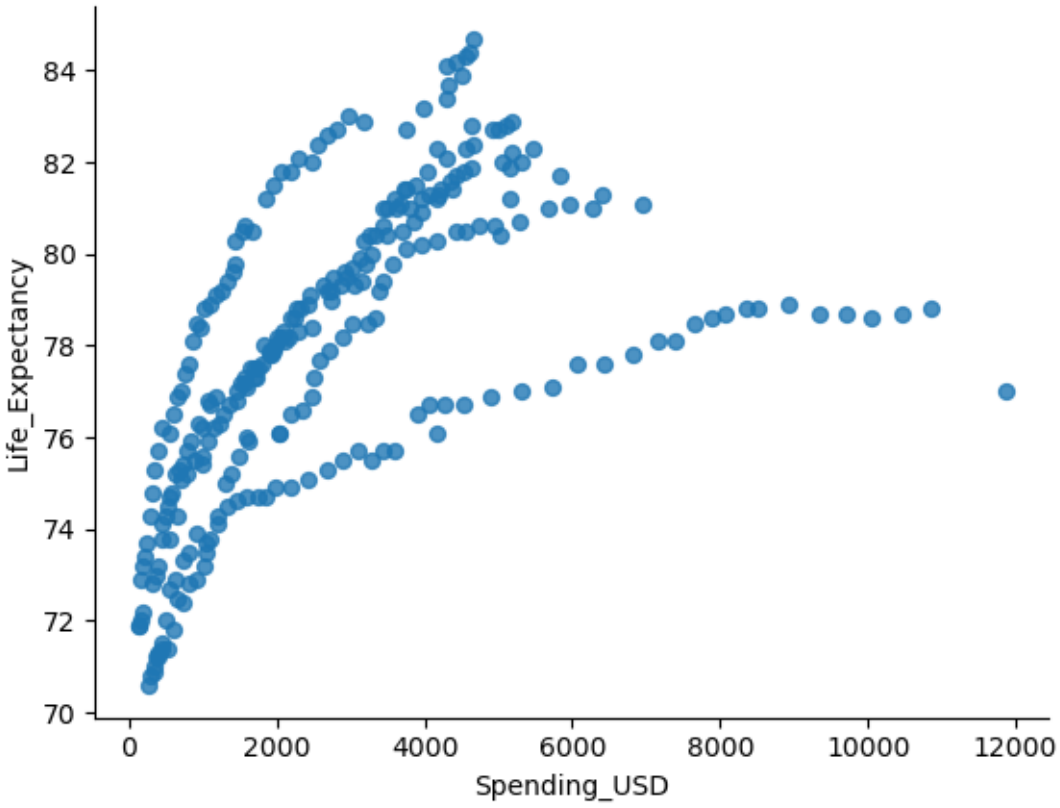
[111]:
```
from matplotlib import pyplot as plt
df.plot(kind='scatter', x='Year', y='Spending_USD', s=32, alpha=.8)
plt.gca().spines[['top', 'right',]].set_visible(False)
plt.show()
```


12000
10000
8000
6000
4000
2000
0
Spending_USD
1970
1980
1990
2000
2010
2020
Year

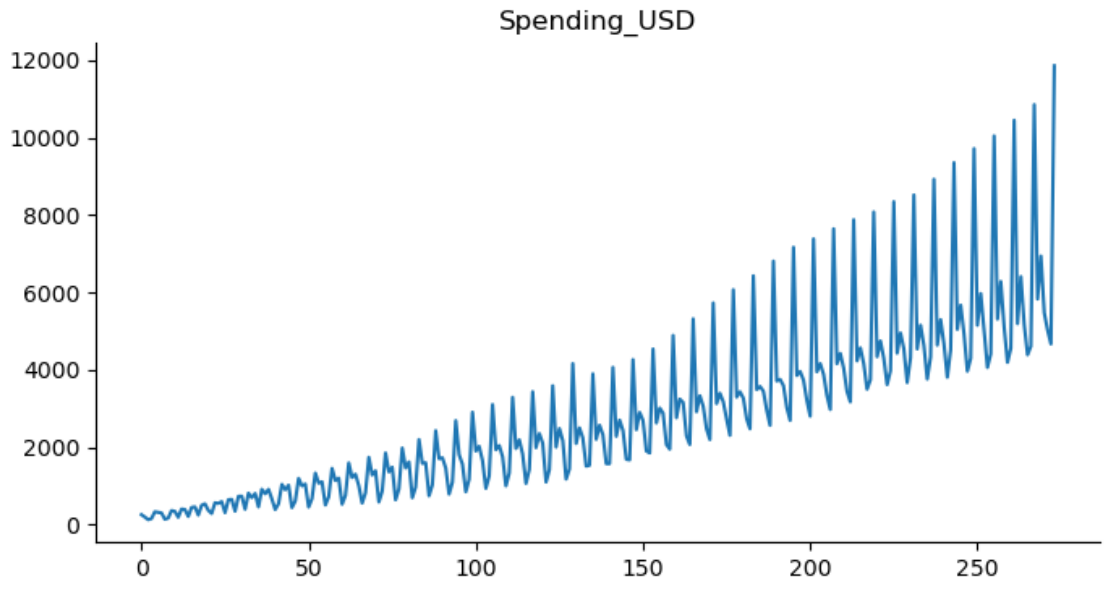

12000
10000
8000
6000
4000
2000
0
Spending_USD
1970
1980
1990
2000
2010
2020
Year



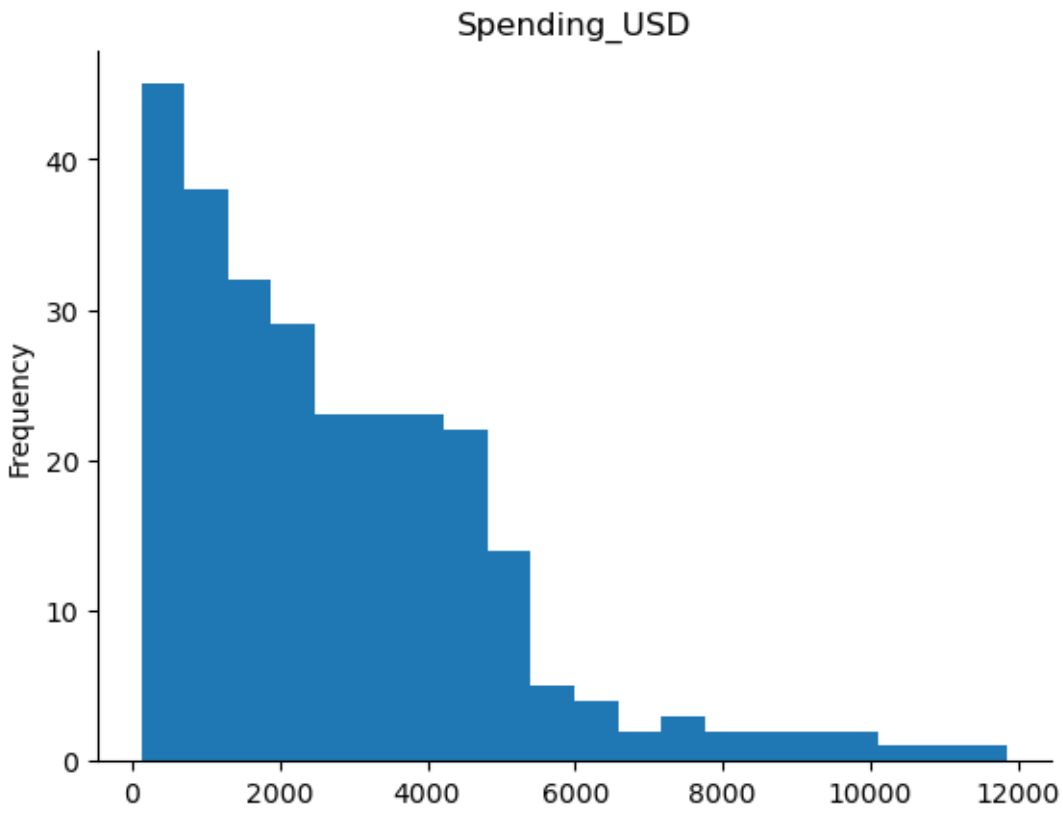
[112]:
```
from matplotlib import pyplot as plt
df.plot(kind='scatter', x='Spending_USD', y='Life_Expectancy', s=32, alpha=.8)
plt.gca().spines[['top', 'right',]].set_visible(False)
plt.show()
```



[113]:
```
from matplotlib import pyplot as plt
df['Spending_USD'].plot(kind='line', figsize=(8, 4), title='Spending_USD')
plt.gca().spines[['top', 'right']].set_visible(False)
plt.show()
```



[114]:
```
from matplotlib import pyplot as plt
df['Spending_USD'].plot(kind='hist', bins=20, title='Spending_USD')
plt.gca().spines[['top', 'right',]].set_visible(False)
plt.show()
```



[115]:
```
df.Spending_USD.value_counts()
```

[115]:
```
Spending_USD
252.311      1
3264.574     1
3444.855     1
3567.061     1
3486.621     1
             ..
842.797      1
1166.430     1
2901.589     1
1897.456     1
11859.179    1
Name: count, Length: 274, dtype: int64
```

[117]:
```
df.Life_Expectancy.value_counts()
```

[117]:
```
Life_Expectancy
78.8    6
81.0    6
```

```
75.7    5
81.2    5
78.6    5
       ..
77.7    1
79.9    1
79.0    1
80.0    1
84.7    1
Name: count, Length: 118, dtype: int64
```

[55]:
```
!pip install ydata-profiling
```

```
Requirement already satisfied: ydata-profiling in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (4.17.0)
Requirement already satisfied: scipy<1.16,>=1.4.1 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (1.13.1)
Requirement already satisfied: pandas!=1.4.0,<3.0,>1.1 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (2.3.1)
Requirement already satisfied: matplotlib<=3.10,>=3.5 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (3.10.0)
Requirement already satisfied: pydantic>=2 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ydata-profiling) (2.11.7)
Requirement already satisfied: PyYAML<6.1,>=5.0.0 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (6.0.2)
Requirement already satisfied: jinja2<3.2,>=2.11.1 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (3.1.6)
Requirement already satisfied: visions<0.8.2,>=0.7.5 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
visions[type_image_path]<0.8.2,>=0.7.5->ydata-profiling) (0.8.1)
Requirement already satisfied: numpy<2.2,>=1.16.0 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (1.26.4)
Requirement already satisfied: minify-html>=0.15.0 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (0.18.1)
Requirement already satisfied: filetype>=1.0.0 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ydata-profiling) (1.2.0)
Requirement already satisfied: phik<0.13,>=0.11.1 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (0.12.5)
Requirement already satisfied: requests<3,>=2.24.0 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
```

```
profiling) (2.32.4)
Requirement already satisfied: tqdm<5,>=4.48.2 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ydata-profiling) (4.67.1)
Requirement already satisfied: seaborn<0.14,>=0.10.1 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (0.13.2)
Requirement already satisfied: multimethod<2,>=1.4 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (1.12)
Requirement already satisfied: statsmodels<1,>=0.13.2 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (0.14.5)
Requirement already satisfied: typeguard<5,>=3 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ydata-profiling) (4.4.4)
Requirement already satisfied: imagehash==4.3.1 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ydata-profiling) (4.3.1)
Requirement already satisfied: wordcloud>=1.9.3 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ydata-profiling) (1.9.4)
Requirement already satisfied: dacite>=1.8 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ydata-profiling) (1.9.2)
Requirement already satisfied: numba<=0.61,>=0.56.0 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ydata-
profiling) (0.61.0)
Requirement already satisfied: PyWavelets in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from imagehash==4.3.1->ydata-profiling)
(1.9.0)
Requirement already satisfied: pillow in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from imagehash==4.3.1->ydata-profiling)
(11.3.0)
Requirement already satisfied: MarkupSafe>=2.0 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from jinja2<3.2,>=2.11.1->ydata-profiling)
(3.0.2)
Requirement already satisfied: contourpy>=1.0.1 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from matplotlib<=3.10,>=3.5->ydata-profiling)
(1.3.1)
Requirement already satisfied: cycler>=0.10 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from matplotlib<=3.10,>=3.5->ydata-profiling)
(0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
matplotlib<=3.10,>=3.5->ydata-profiling) (4.55.3)
Requirement already satisfied: kiwisolver>=1.3.1 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
matplotlib<=3.10,>=3.5->ydata-profiling) (1.4.8)
Requirement already satisfied: packaging>=20.0 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from matplotlib<=3.10,>=3.5->ydata-profiling)
(24.2)
Requirement already satisfied: pyparsing>=2.3.1 in /opt/anaconda3/envs/anaconda-
```

```
nlp/lib/python3.11/site-packages (from matplotlib<=3.10,>=3.5->ydata-profiling)
(3.2.0)
Requirement already satisfied: python-dateutil>=2.7 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
matplotlib<=3.10,>=3.5->ydata-profiling) (2.9.0.post0)
Requirement already satisfied: llvmlite<0.45,>=0.44.0dev0 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
numba<=0.61,>=0.56.0->ydata-profiling) (0.44.0)
Requirement already satisfied: pytz>=2020.1 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from pandas!=1.4.0,<3.0,>1.1->ydata-profiling)
(2025.2)
Requirement already satisfied: tzdata>=2022.7 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from pandas!=1.4.0,<3.0,>1.1->ydata-profiling)
(2025.2)
Requirement already satisfied: joblib>=0.14.1 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from phik<0.13,>=0.11.1->ydata-profiling)
(1.5.1)
Requirement already satisfied: charset_normalizer<4,>=2 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
requests<3,>=2.24.0->ydata-profiling) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from requests<3,>=2.24.0->ydata-profiling)
(3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
requests<3,>=2.24.0->ydata-profiling) (2.5.0)
Requirement already satisfied: certifi>=2017.4.17 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
requests<3,>=2.24.0->ydata-profiling) (2025.8.3)
Requirement already satisfied: patsy>=0.5.6 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from statsmodels<1,>=0.13.2->ydata-profiling)
(1.0.2)
Requirement already satisfied: typing_extensions>=4.14.0 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
typeguard<5,>=3->ydata-profiling) (4.15.0)
Requirement already satisfied: attrs>=19.3.0 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from
visions<0.8.2,>=0.7.5->visions[type_image_path]<0.8.2,>=0.7.5->ydata-profiling)
(24.3.0)
Requirement already satisfied: networkx>=2.4 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from
visions<0.8.2,>=0.7.5->visions[type_image_path]<0.8.2,>=0.7.5->ydata-profiling)
(3.4.2)
Requirement already satisfied: puremagic in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from
visions<0.8.2,>=0.7.5->visions[type_image_path]<0.8.2,>=0.7.5->ydata-profiling)
(1.30)
Requirement already satisfied: annotated-types>=0.6.0 in
```

```
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
pydantic>=2->ydata-profiling) (0.6.0)
Requirement already satisfied: pydantic-core==2.33.2 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
pydantic>=2->ydata-profiling) (2.33.2)
Requirement already satisfied: typing-inspection>=0.4.0 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
pydantic>=2->ydata-profiling) (0.4.0)
Requirement already satisfied: six>=1.5 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from python-
dateutil>=2.7->matplotlib<=3.10,>=3.5->ydata-profiling) (1.17.0)
```

[56]:
```
from ydata_profiling import ProfileReport
```

[57]:
```
Profile = ProfileReport(df, title = "healthexp Report", explorative = True)
```

[58]:
```
pip install ipywidgets
```

```
Requirement already satisfied: ipywidgets in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (8.1.8)
Requirement already satisfied: comm>=0.1.3 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ipywidgets) (0.2.1)
Requirement already satisfied: ipython>=6.1.0 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ipywidgets) (9.1.0)
Requirement already satisfied: traitlets>=4.3.1 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ipywidgets) (5.14.3)
Requirement already satisfied: widgetsnbextension~=4.0.14 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ipywidgets)
(4.0.15)
Requirement already satisfied: jupyterlab_widgets~=3.0.15 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from ipywidgets)
(3.0.16)
Requirement already satisfied: decorator in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ipython>=6.1.0->ipywidgets) (5.1.1)
Requirement already satisfied: ipython-pygments-lexers in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
ipython>=6.1.0->ipywidgets) (1.1.1)
Requirement already satisfied: jedi>=0.16 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ipython>=6.1.0->ipywidgets) (0.19.2)
Requirement already satisfied: matplotlib-inline in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
ipython>=6.1.0->ipywidgets) (0.1.6)
Requirement already satisfied: pexpect>4.3 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ipython>=6.1.0->ipywidgets) (4.9.0)
Requirement already satisfied: prompt_toolkit<3.1.0,>=3.0.41 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
ipython>=6.1.0->ipywidgets) (3.0.43)
Requirement already satisfied: pygments>=2.4.0 in /opt/anaconda3/envs/anaconda-
```

```
nlp/lib/python3.11/site-packages (from ipython>=6.1.0->ipywidgets) (2.19.1)
Requirement already satisfied: stack_data in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from ipython>=6.1.0->ipywidgets) (0.2.0)
Requirement already satisfied: typing_extensions>=4.6 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
ipython>=6.1.0->ipywidgets) (4.15.0)
Requirement already satisfied: wcwidth in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from
prompt_toolkit<3.1.0,>=3.0.41->ipython>=6.1.0->ipywidgets) (0.2.13)
Requirement already satisfied: parso<0.9.0,>=0.8.4 in
/opt/anaconda3/envs/anaconda-nlp/lib/python3.11/site-packages (from
jedi>=0.16->ipython>=6.1.0->ipywidgets) (0.8.4)
Requirement already satisfied: ptyprocess>=0.5 in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from pexpect>4.3->ipython>=6.1.0->ipywidgets)
(0.7.0)
Requirement already satisfied: executing in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from stack_data->ipython>=6.1.0->ipywidgets)
(0.8.3)
Requirement already satisfied: asttokens in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from stack_data->ipython>=6.1.0->ipywidgets)
(3.0.0)
Requirement already satisfied: pure-eval in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (from stack_data->ipython>=6.1.0->ipywidgets)
(0.2.2)
Note: you may need to restart the kernel to use updated packages.
```

[59]:
```
Profile.to_notebook_iframe()
```

```
Summarize dataset:   0%|          | 0/5 [00:00<?, ?it/s]

100%|      | 4/4 [00:00<00:00, 28777.39it/s]

Generate report structure:   0%|          | 0/1 [00:00<?, ?it/s]

Render HTML:   0%|          | 0/1 [00:00<?, ?it/s]

<IPython.core.display.HTML object>
```

[60]:
```
Profile.to_file("healthexp Report.html")
```

```
Export report to file:   0%|          | 0/1 [00:00<?, ?it/s]
```

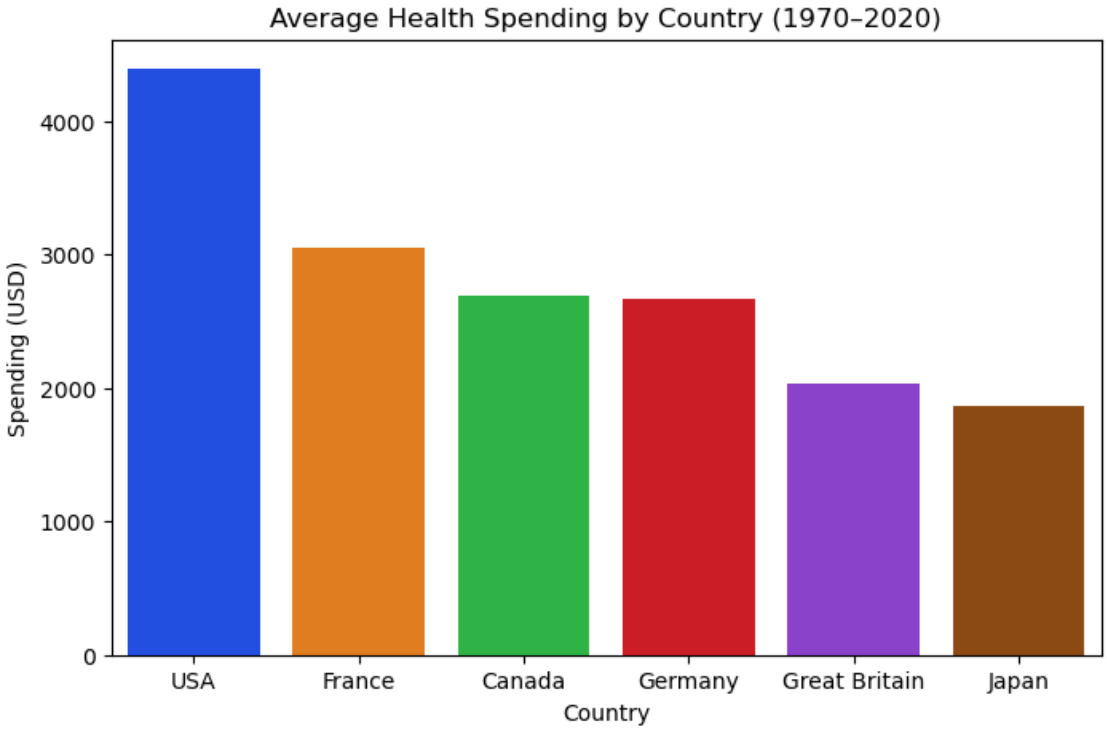
[119]:
```
country_summary = df.groupby('Country')[['Spending_USD', 'Life_Expectancy']].
  ↪mean().sort_values('Spending_USD', ascending=False)
print(country_summary)
```

```
              Spending_USD  Life_Expectancy
Country
USA            4388.570529        75.843137
France         3045.145057        79.565714
Canada         2685.778341        78.706818
```

```
Germany          2667.280200         76.726000
Great Britain    2034.192465         77.620930
Japan            1860.257902         79.554902
```

[121]:
```
plt.figure(figsize=(8,5))
sns.barplot(data=country_summary.reset_index(), x='Country', y='Spending_USD',␣
 ↪palette='bright')
plt.title('Average Health Spending by Country (1970-2020)')
plt.ylabel('Spending (USD)')
plt.show()
```

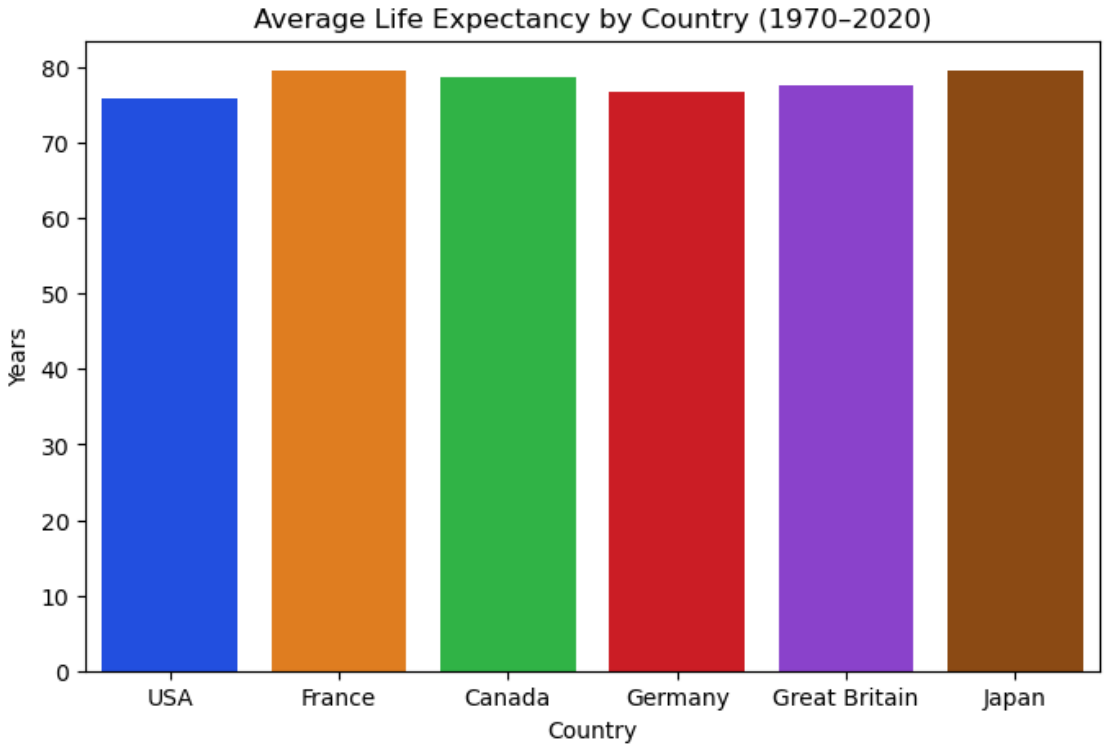
```
/var/folders/r9/c6cjksp1313g4v86v8y03ppw0000gn/T/ipykernel_81099/1376839844.py:2
: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  sns.barplot(data=country_summary.reset_index(), x='Country', y='Spending_USD',
palette='bright')
```



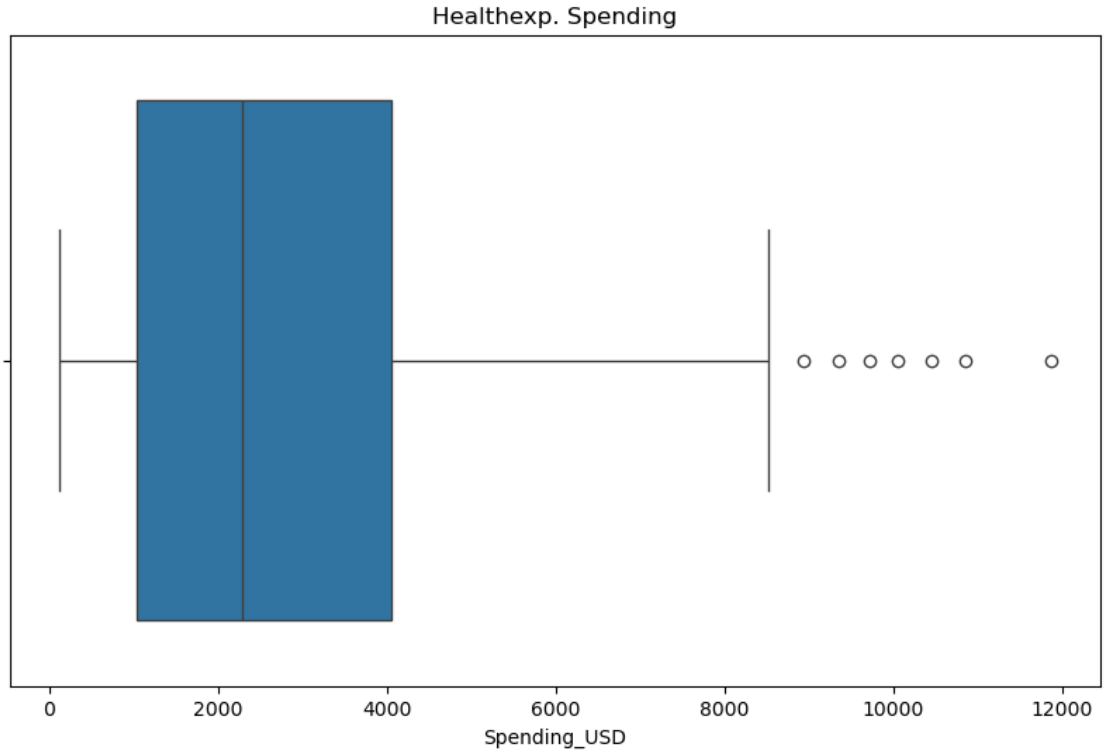
[123]:
```
plt.figure(figsize=(8,5))
sns.barplot(data=country_summary.reset_index(), x='Country',␣
 ↪y='Life_Expectancy', palette='bright')
```

```
plt.title('Average Life Expectancy by Country (1970-2020)')
plt.ylabel('Years')
plt.show()
```

```
/var/folders/r9/c6cjksp1313g4v86v8y03ppw0000gn/T/ipykernel_81099/778442369.py:2:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  sns.barplot(data=country_summary.reset_index(), x='Country',
y='Life_Expectancy', palette='bright')
```



[61]:
```
%matplotlib inline
fig = plt.figure(figsize = (10,6))
ax = fig.subplots()

sns.boxplot(x = df["Spending_USD"] , ax = ax)
plt.title("Healthexp. Spending")
plt.xlabel("Spending_USD")
plt.show()
```



[62]:
```
import matplotlib.dates as mdates
fig,ax = plt.subplots(figsize = (10,6))
ax.plot(df['Spending_USD'], df['Life_Expectancy'] , color = 'blue')
ax.set_xlabel('Spending_USD')
ax.set_ylabel('Life_Expectancy')
plt.title('Healthexp Spending and Life Expectancy')
plt.tight_layout()
plt.show()
```

[63]:
```
df1 = df.copy()
```

[64]:
```
df1['Spending_USD'].describe()
```

[64]:
```
count      274.000000
mean      2789.338905
std       2194.939785
min        123.993000
25%       1038.357000
50%       2295.578000
75%       4055.610000
max      11859.179000
Name: Spending_USD, dtype: float64
```

[65]:
```
q1 = df1['Spending_USD'].quantile(0.25)
q3 = df1['Spending_USD'].quantile(0.75)
IQR = q3 - q1
print(IQR)
```

```
3017.253
```

[66]:
```
upper_limit = q3 + 1.5 * IQR
lower_limit = q1 - 1.5 * IQR
print(upper_limit)
print(lower_limit)
```

```
8581.4895
-3487.5225
```

[67]:
```
len(df1[df1['Spending_USD'] > upper_limit])
```

[67]:
```
7
```

[68]:
```
df1.shape
```

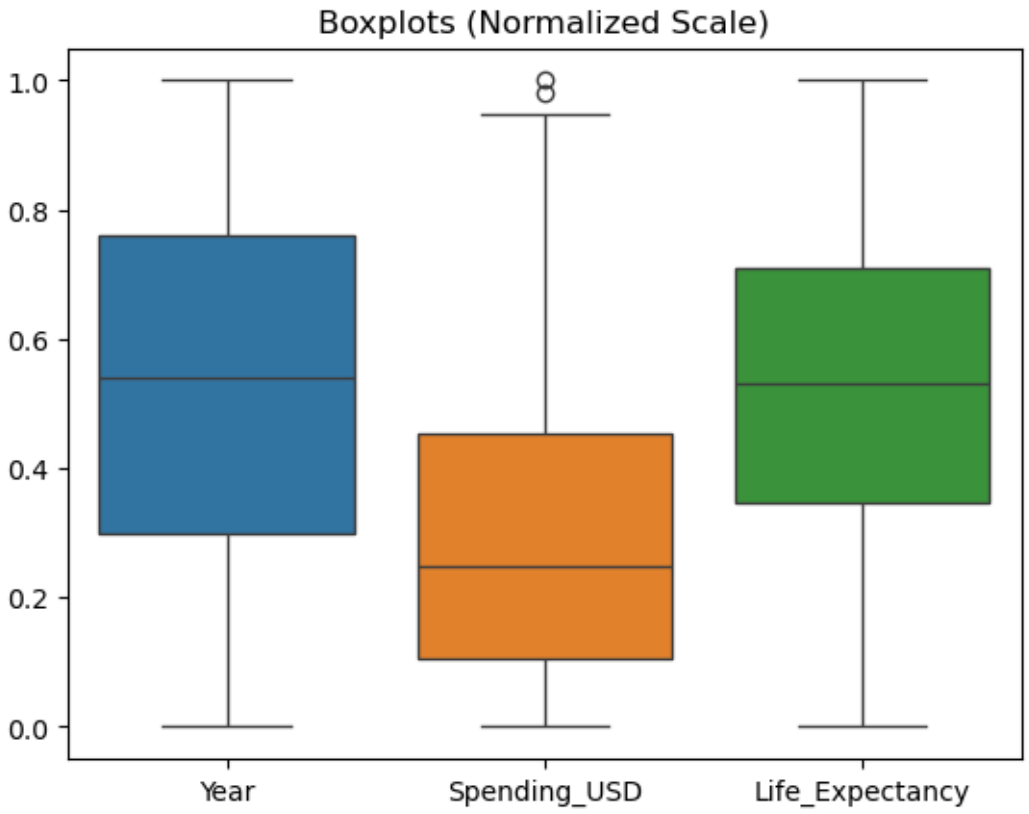
[68]:
```
(274, 4)
```

[69]:
```
df2 = df1[(df['Spending_USD'] > lower_limit) & (df1['Spending_USD'] <␣
↪upper_limit)]
```

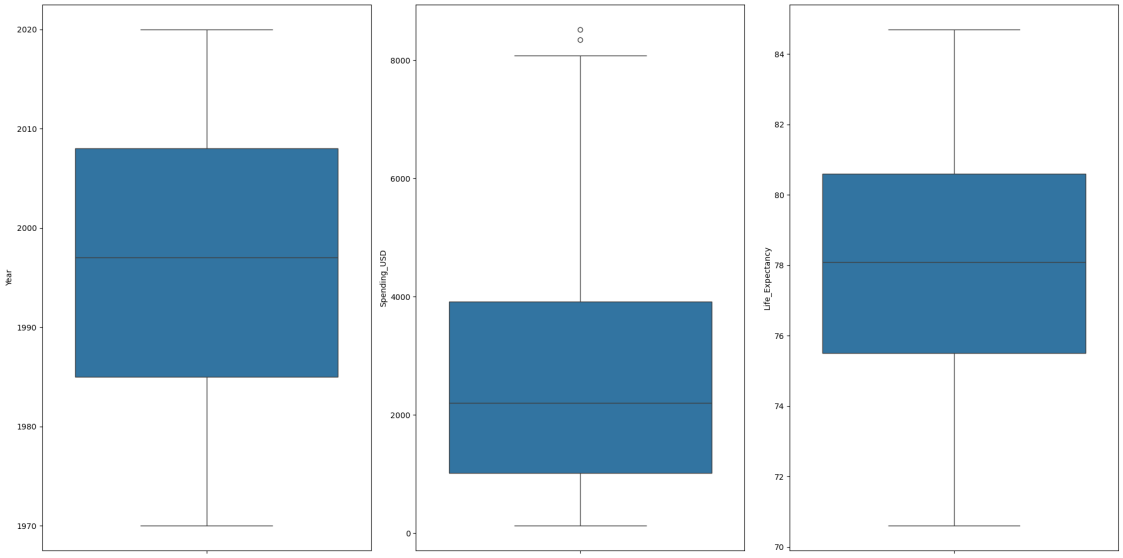
[70]:
```
df2.shape
```

[70]:
```
(267, 4)
```

[71]:
```
from sklearn.preprocessing import MinMaxScaler
scaled = pd.DataFrame(MinMaxScaler().fit_transform(df2[['Year', 'Spending_USD',␣
↪'Life_Expectancy']]),
                      columns=['Year', 'Spending_USD', 'Life_Expectancy'])
sns.boxplot(data=scaled)
plt.title("Boxplots (Normalized Scale)")
plt.show()
```

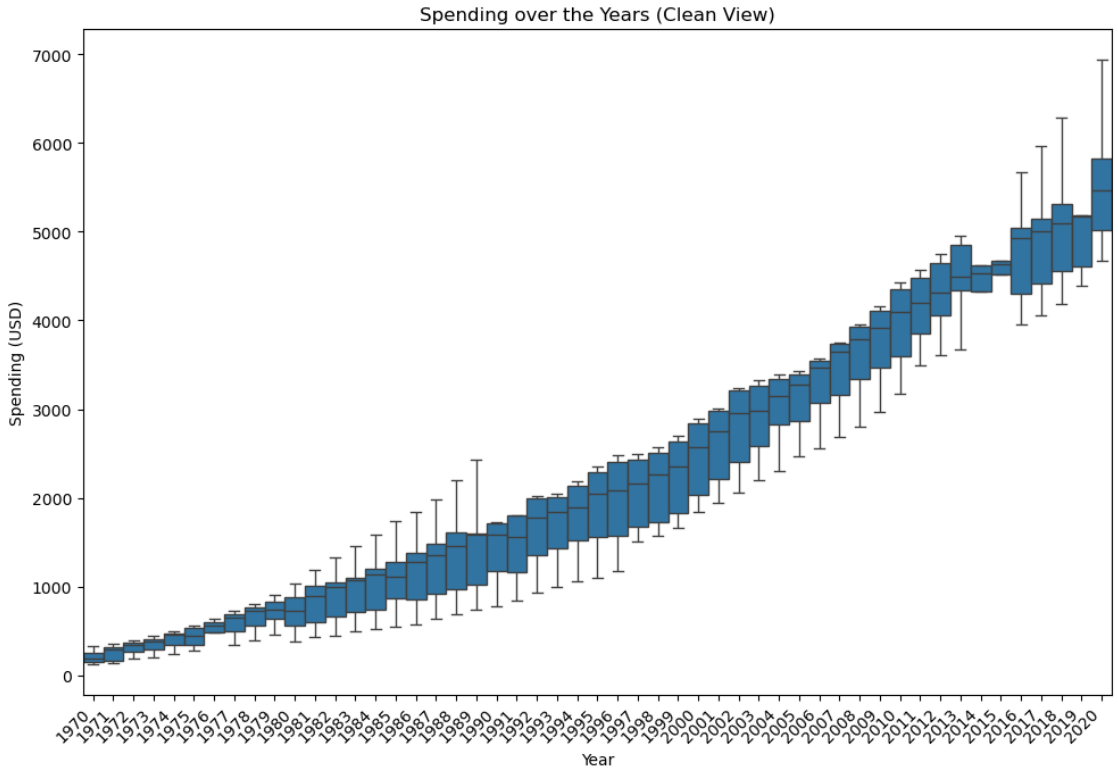


[72]:
```
fig, axes = plt.subplots(1, 3, figsize=(20, 10))
sns.boxplot(y='Year', data=df2, ax=axes[0])
sns.boxplot(y='Spending_USD', data=df2, ax=axes[1])
sns.boxplot(y='Life_Expectancy', data=df2, ax=axes[2])
plt.tight_layout()
plt.show()
```



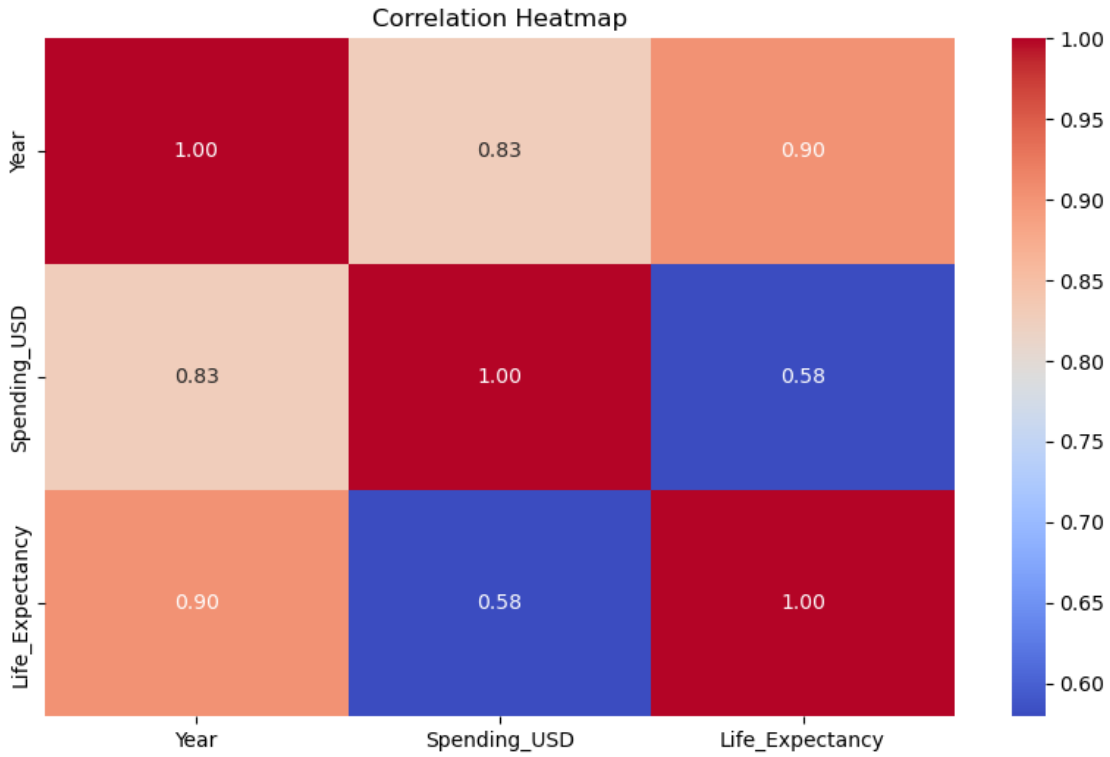
[73]:
```
df2['Year'] = df2['Year'].astype(str)
plt.figure(figsize=(10,7))
sns.boxplot(
    data=df2.sort_values('Year'),
    x='Year',
    y='Spending_USD',
    showfliers=False,
    width=1
)
plt.xticks(rotation=45, ha='right')
plt.title('Spending over the Years (Clean View)')
plt.xlabel('Year')
plt.ylabel('Spending (USD)')
plt.tight_layout()
plt.show()
```

```
/var/folders/r9/c6cjksp1313g4v86v8y03ppw0000gn/T/ipykernel_81099/2800859114.py:1
: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df2['Year'] = df2['Year'].astype(str)
```



[75]:
```
corr = df.corr(numeric_only=True)
plt.figure(figsize=(10,6))
sns.heatmap(corr, annot=True, cmap='coolwarm', fmt='.2f')
plt.title("Correlation Heatmap")
plt.show()
```

```
<Figure size 1000x600 with 0 Axes>
```



[76]:
```
df2.Spending_USD.value_counts()
```

[76]:
```
Spending_USD
252.311     1
6430.757    1
2469.159    1
2194.437    1
5726.538    1
           ..
2684.984    1
1805.209    1
1558.033    1
842.797     1
4665.641    1
Name: count, Length: 267, dtype: int64
```

[77]:
```
df2.columns
```

[77]:
```
Index(['Year', 'Country', 'Spending_USD', 'Life_Expectancy'], dtype='object')
```

[78]:
```
X = df2.drop(columns = ['Spending_USD'])
```

[79]:
```
X.columns
```

[79]:
```
Index(['Year', 'Country', 'Life_Expectancy'], dtype='object')
```

[118]:
```
X = pd.get_dummies(X , drop_first = True).astype('int')
```

[81]:
```
X.head()
```

[81]:

	Life_Expectancy	Year_1971	Year_1972	Year_1973	Year_1974	Year_1975	\
0	70	0	0	0	0	0	
1	72	0	0	0	0	0	
2	71	0	0	0	0	0	
3	72	0	0	0	0	0	
4	70	0	0	0	0	0	

	Year_1976	Year_1977	Year_1978	Year_1979	…	Year_2016	Year_2017	\
0	0	0	0	0	…	0	0	
1	0	0	0	0	…	0	0	
2	0	0	0	0	…	0	0	
3	0	0	0	0	…	0	0	
4	0	0	0	0	…	0	0	

	Year_2018	Year_2019	Year_2020	Country_France	Country_Germany	\
0	0	0	0	0	1	
1	0	0	0	1	0	
2	0	0	0	0	0	
3	0	0	0	0	0	
4	0	0	0	0	0	

	Country_Great Britain	Country_Japan	Country_USA
0	0	0	0
1	0	0	0
2	1	0	0
3	0	1	0
4	0	0	1

[5 rows x 56 columns]

[82]:
```
X.shape
```

[82]:
```
(267, 56)
```

[83]:
```
y = df2[['Life_Expectancy']]
y.columns
```

[83]:
```
Index(['Life_Expectancy'], dtype='object')
```

[84]:
```
y.shape
```

[84]:
```
(267, 1)
```

[85]:
```
%pip install --upgrade cloudpickle
```

```
Requirement already satisfied: cloudpickle in /opt/anaconda3/envs/anaconda-
nlp/lib/python3.11/site-packages (3.1.1)
Note: you may need to restart the kernel to use updated packages.
```

[86]:
```
from sklearn.model_selection import train_test_split as split
```

[87]:
```
X_train, X_test , y_train , y_test = split(X, y , train_size = 0.8 ,␣
↪random_state = 42)
```

[88]:
```
X_train.shape
```

[88]:
```
(213, 56)
```

[89]:
```
y_train.shape
```

[89]:
```
(213, 1)
```

[90]:
```
X_test.shape
```

[90]:
```
(54, 56)
```

[91]:
```
y_test.shape
```

[91]:
```
(54, 1)
```

[92]:
```
y.head()
```

[92]:

	Life_Expectancy
0	70.6
1	72.2
2	71.9
3	72.0
4	70.9

[93]:
```
X.head()
```

[93]:

	Life_Expectancy	Year_1971	Year_1972	Year_1973	Year_1974	Year_1975	\
0	70	0	0	0	0	0	
1	72	0	0	0	0	0	
2	71	0	0	0	0	0	
3	72	0	0	0	0	0	
4	70	0	0	0	0	0	

	Year_1976	Year_1977	Year_1978	Year_1979	…	Year_2016	Year_2017	\
0	0	0	0	0	…	0	0	
1	0	0	0	0	…	0	0	
2	0	0	0	0	…	0	0	

```
3          0          0          0          0  …          0          0
4          0          0          0          0  …          0          0

   Year_2018  Year_2019  Year_2020  Country_France  Country_Germany  \
0          0          0          0               0                1
1          0          0          0               1                0
2          0          0          0               0                0
3          0          0          0               0                0
4          0          0          0               0                0

   Country_Great Britain  Country_Japan  Country_USA
0                      0              0            0
1                      0              0            0
2                      1              0            0
3                      0              1            0
4                      0              0            1

[5 rows x 56 columns]
```

[94]:
```
from sklearn.preprocessing import StandardScaler
```

[95]:
```
sc = StandardScaler()
```

[96]:
```
X_train = sc.fit_transform(X_train)
X_test = sc.fit_transform(X_test)
```

[97]:
```
X_train[:10]
```

[97]:
```
array([[-0.46933953, -0.11952286, -0.09735848, -0.09735848, -0.09735848,
        -0.11952286, -0.11952286, -0.11952286, -0.09735848, -0.11952286,
        -0.17025131, -0.11952286, -0.15504342, -0.15504342, -0.13834289,
        -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.13834289,
        -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.11952286,
        -0.11952286,  6.4498062 , -0.17025131, -0.15504342, -0.15504342,
        -0.13834289, -0.15504342, -0.15504342, -0.15504342, -0.15504342,
        -0.13834289, -0.17025131, -0.13834289, -0.15504342, -0.17025131,
        -0.15504342, -0.15504342, -0.13834289, -0.17025131, -0.13834289,
        -0.11952286, -0.15504342, -0.13834289, -0.13834289, -0.11952286,
        -0.15504342, -0.41271015, -0.45098762,  2.3782872 , -0.49559463,
        -0.45098762],
       [ 1.05924333, -0.11952286, -0.09735848, -0.09735848, -0.09735848,
        -0.11952286, -0.11952286, -0.11952286, -0.09735848, -0.11952286,
        -0.17025131, -0.11952286, -0.15504342, -0.15504342, -0.13834289,
        -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.13834289,
        -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.11952286,
        -0.11952286, -0.15504342, -0.17025131, -0.15504342, -0.15504342,
        -0.13834289, -0.15504342, -0.15504342, -0.15504342, -0.15504342,
```

```
 -0.13834289, -0.17025131, -0.13834289, -0.15504342, -0.17025131,
 -0.15504342, -0.15504342,  7.22841615, -0.17025131, -0.13834289,
 -0.11952286, -0.15504342, -0.13834289, -0.13834289, -0.11952286,
 -0.15504342, -0.41271015, -0.45098762,  2.3782872 , -0.49559463,
 -0.45098762],
[-0.16362295, -0.11952286, -0.09735848, -0.09735848, -0.09735848,
 -0.11952286, -0.11952286, -0.11952286, -0.09735848, -0.11952286,
 -0.17025131, -0.11952286, -0.15504342, -0.15504342, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.13834289,
  6.4498062 , -0.13834289, -0.13834289, -0.13834289, -0.11952286,
 -0.11952286, -0.15504342, -0.17025131, -0.15504342, -0.15504342,
 -0.13834289, -0.15504342, -0.15504342, -0.15504342, -0.15504342,
 -0.13834289, -0.17025131, -0.13834289, -0.15504342, -0.17025131,
 -0.15504342, -0.15504342, -0.13834289, -0.17025131, -0.13834289,
 -0.11952286, -0.15504342, -0.13834289, -0.13834289, -0.11952286,
 -0.15504342, -0.41271015,  2.21735578, -0.42047066, -0.49559463,
 -0.45098762],
[ 0.75352676, -0.11952286, -0.09735848, -0.09735848, -0.09735848,
 -0.11952286, -0.11952286, -0.11952286, -0.09735848, -0.11952286,
 -0.17025131, -0.11952286, -0.15504342, -0.15504342, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.11952286,
 -0.11952286, -0.15504342, -0.17025131, -0.15504342, -0.15504342,
 -0.13834289, -0.15504342, -0.15504342, -0.15504342, -0.15504342,
 -0.13834289, -0.17025131, -0.13834289, -0.15504342, -0.17025131,
 -0.15504342, -0.15504342, -0.13834289, -0.17025131, -0.13834289,
 -0.11952286, -0.15504342, -0.13834289, -0.13834289, -0.11952286,
  6.4498062 , -0.41271015, -0.45098762,  2.3782872 , -0.49559463,
 -0.45098762],
[-0.46933953, -0.11952286, -0.09735848, -0.09735848, -0.09735848,
 -0.11952286, -0.11952286, -0.11952286, -0.09735848, -0.11952286,
 -0.17025131, -0.11952286, -0.15504342, -0.15504342, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289,  7.22841615, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.11952286,
 -0.11952286, -0.15504342, -0.17025131, -0.15504342, -0.15504342,
 -0.13834289, -0.15504342, -0.15504342, -0.15504342, -0.15504342,
 -0.13834289, -0.17025131, -0.13834289, -0.15504342, -0.17025131,
 -0.15504342, -0.15504342, -0.13834289, -0.17025131, -0.13834289,
 -0.11952286, -0.15504342, -0.13834289, -0.13834289, -0.11952286,
 -0.15504342, -0.41271015, -0.45098762, -0.42047066, -0.49559463,
 -0.45098762],
[-1.08077267, -0.11952286, -0.09735848, -0.09735848, -0.09735848,
 -0.11952286,  8.36660027, -0.11952286, -0.09735848, -0.11952286,
 -0.17025131, -0.11952286, -0.15504342, -0.15504342, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.11952286,
 -0.11952286, -0.15504342, -0.17025131, -0.15504342, -0.15504342,
```

```
 -0.13834289, -0.15504342, -0.15504342, -0.15504342, -0.15504342,
 -0.13834289, -0.17025131, -0.13834289, -0.15504342, -0.17025131,
 -0.15504342, -0.15504342, -0.13834289, -0.17025131, -0.13834289,
 -0.11952286, -0.15504342, -0.13834289, -0.13834289, -0.11952286,
 -0.15504342, -0.41271015, -0.45098762, -0.42047066,  2.01777813,
 -0.45098762],
[ 0.14209362, -0.11952286, -0.09735848, -0.09735848, -0.09735848,
 -0.11952286, -0.11952286, -0.11952286, -0.09735848, -0.11952286,
 -0.17025131, -0.11952286, -0.15504342, -0.15504342, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.11952286,
 -0.11952286, -0.15504342,  5.87367006, -0.15504342, -0.15504342,
 -0.13834289, -0.15504342, -0.15504342, -0.15504342, -0.15504342,
 -0.13834289, -0.17025131, -0.13834289, -0.15504342, -0.17025131,
 -0.15504342, -0.15504342, -0.13834289, -0.17025131, -0.13834289,
 -0.11952286, -0.15504342, -0.13834289, -0.13834289, -0.11952286,
 -0.15504342,  2.42300799, -0.45098762, -0.42047066, -0.49559463,
 -0.45098762],
[ 1.36495991, -0.11952286, -0.09735848, -0.09735848, -0.09735848,
 -0.11952286, -0.11952286, -0.11952286, -0.09735848, -0.11952286,
 -0.17025131, -0.11952286, -0.15504342, -0.15504342, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.11952286,
 -0.11952286, -0.15504342, -0.17025131, -0.15504342, -0.15504342,
 -0.13834289, -0.15504342, -0.15504342, -0.15504342,  6.4498062 ,
 -0.13834289, -0.17025131, -0.13834289, -0.15504342, -0.17025131,
 -0.15504342, -0.15504342, -0.13834289, -0.17025131, -0.13834289,
 -0.11952286, -0.15504342, -0.13834289, -0.13834289, -0.11952286,
 -0.15504342, -0.41271015, -0.45098762, -0.42047066,  2.01777813,
 -0.45098762],
[-0.16362295, -0.11952286, -0.09735848, -0.09735848, -0.09735848,
 -0.11952286, -0.11952286, -0.11952286, -0.09735848, -0.11952286,
 -0.17025131, -0.11952286, -0.15504342, -0.15504342, -0.13834289,
  6.4498062 , -0.13834289, -0.13834289, -0.13834289, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.11952286,
 -0.11952286, -0.15504342, -0.17025131, -0.15504342, -0.15504342,
 -0.13834289, -0.15504342, -0.15504342, -0.15504342, -0.15504342,
 -0.13834289, -0.17025131, -0.13834289, -0.15504342, -0.17025131,
 -0.15504342, -0.15504342, -0.13834289, -0.17025131, -0.13834289,
 -0.11952286, -0.15504342, -0.13834289, -0.13834289, -0.11952286,
 -0.15504342, -0.41271015, -0.45098762, -0.42047066,  2.01777813,
 -0.45098762],
[ 0.14209362, -0.11952286, -0.09735848, -0.09735848, -0.09735848,
 -0.11952286, -0.11952286, -0.11952286, -0.09735848, -0.11952286,
 -0.17025131, -0.11952286, -0.15504342, -0.15504342, -0.13834289,
 -0.15504342, -0.13834289,  7.22841615, -0.13834289, -0.13834289,
 -0.15504342, -0.13834289, -0.13834289, -0.13834289, -0.11952286,
```

```
        -0.11952286, -0.15504342, -0.17025131, -0.15504342, -0.15504342,
        -0.13834289, -0.15504342, -0.15504342, -0.15504342, -0.15504342,
        -0.13834289, -0.17025131, -0.13834289, -0.15504342, -0.17025131,
        -0.15504342, -0.15504342, -0.13834289, -0.17025131, -0.13834289,
        -0.11952286, -0.15504342, -0.13834289, -0.13834289, -0.11952286,
        -0.15504342, -0.41271015, -0.45098762, -0.42047066,  2.01777813,
        -0.45098762]])
```

[98]:
```
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.linear_model import Ridge
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import StandardScaler
```

[99]:
```
X_train , X_test , y_train , y_test = train_test_split(X , y  , test_size = 0.2␣
 ↪, random_state = 42)
```

[100]:
```
model = make_pipeline(StandardScaler() , Ridge(alpha = 1.0))
model.fit(X_train , y_train)
```

[100]:
```
Pipeline(steps=[('standardscaler', StandardScaler()), ('ridge', Ridge())])
```

[101]:
```
model.score(X_test , y_test)
```

[101]:
```
0.9934279643647871
```

[102]:
```
corr = df.corr(numeric_only=True)
print("\nCorrelation Matrix:\n", corr)
```

```
Correlation Matrix:
                      Year  Spending_USD  Life_Expectancy
Year              1.000000      0.826273         0.902175
Spending_USD      0.826273      1.000000         0.579430
Life_Expectancy   0.902175      0.579430         1.000000
```

[ ]:
```

```