Data Mining Project - Sustainability Around The World

Aradhya Mathur and Ozlem Gunes

Simple and Multiple Linear Regression

In [1]:

```
#Importing Libraries
import warnings
warnings.filterwarnings('ignore')
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import scipy.stats as stats
import statsmodels.api as sm
import sklearn.metrics as mt
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import Ridge
```

In [2]:

```
#Reading dataset
factors = pd.read_csv('external_factors.csv')
factors.head()
```

Out[2]:

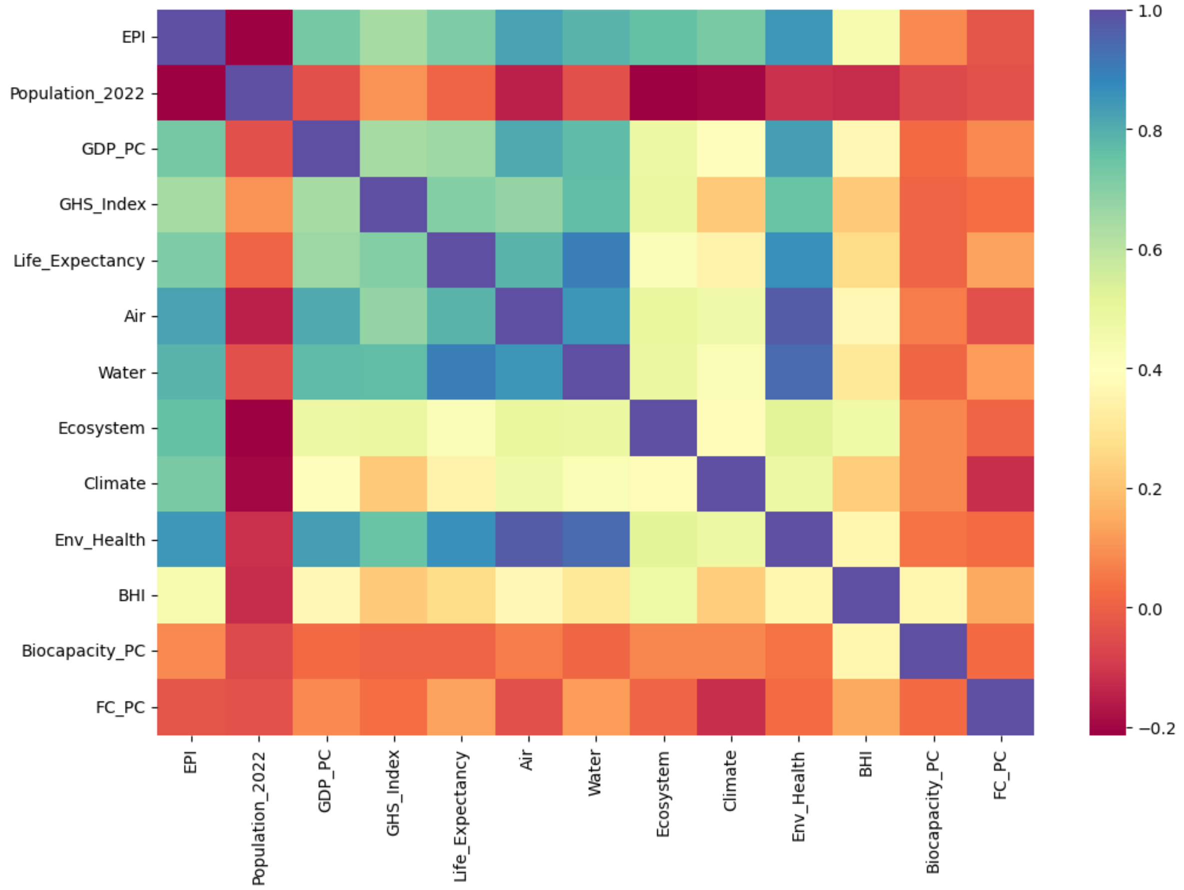
	Country	Region	EPI	Population_2022	GDP_PC	GHS_Index	Life_Expectancy	Air
0	Afghanistan	Southern Asia	43.6	41128771	489.101348	28.8	62.879	15.5
1	Albania	Eastern Europe	47.1	2842321	6424.342465	45.0	76.833	37.5
2	Algeria	Greater Middle East	29.6	44903225	3741.003942	26.2	77.129	39.4
3	Angola	Sub-Saharan Africa	30.5	35588987	2038.467285	29.1	61.929	23.1
4	Antigua and Barbuda	Latin America & Caribbean	52.4	98728	14900.797400	30.0	79.236	56.5

In [3]:

```
#Heatmap for external factors and EPI
plt.figure(figsize = (12,8))
sns.heatmap(factors.corr(), cmap="Spectral")
```

Out[3]:

```
<AxesSubplot:>
```



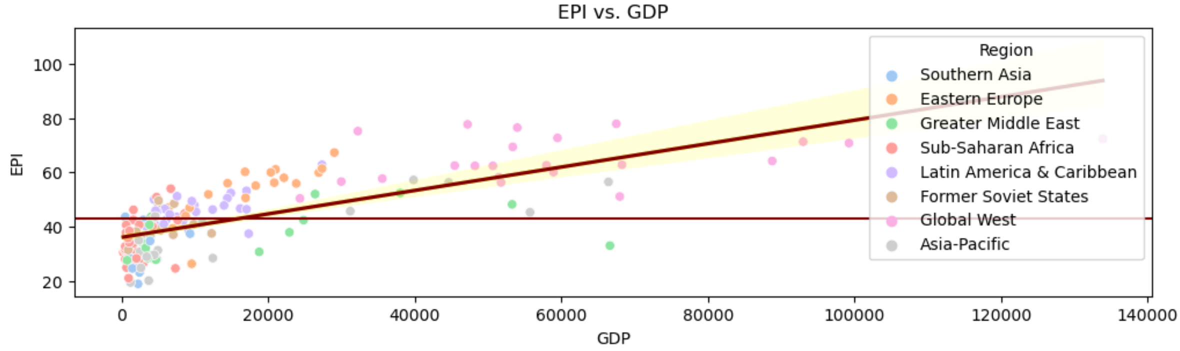
In [4]:

```
#EPI vs GDP
x_train = factors[['GDP_PC']]
y_train = factors['EPI']
x_test = factors[['GDP_PC']]
y_test = factors['EPI']

x_train = sm.add_constant(x_train)
model = sm.OLS(y_train, x_train).fit()
predictions_train = model.predict(x_train)
print_model = model.summary()
print(print_model)
#Plotting
fig = plt.figure(figsize=(12,3))
sns.scatterplot(y=factors.EPI, x=factors.GDP_PC, hue=factors.Region, legend='full'
sns.regplot(data=factors, y=factors.EPI, x=factors.GDP_PC, ax=fig.gca(), scatter =
sns.regplot(data=factors, y=factors.EPI, x=factors.GDP_PC, ax=fig.gca(), scatter =
plt.axhline(factors['EPI'].mean(), color='maroon')
plt.title("EPI vs. GDP")
plt.xlabel('GDP')
plt.ylabel('EPI')
plt.show()
#Correlation
x = factors.EPI
y = factors.GDP_PC
stats.spearmanr(x,y)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    EPI   R-squared:                       0.529
Model:                            OLS   Adj. R-squared:                  0.526
Method:                 Least Squares   F-statistic:                     186.4
Date:                Sun, 11 Dec 2022   Prob (F-statistic):           6.27e-29
Time:                        15:22:12   Log-Likelihood:                -609.76
No. Observations:                 168   AIC:                             1224.
Df Residuals:                     166   BIC:                             1230.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         36.0659      0.870     41.472      0.000      34.349      37.783
GDP_PC         0.0004   3.16e-05     13.651      0.000       0.000       0.000
==============================================================================
Omnibus:                        0.698   Durbin-Watson:                   2.085
Prob(Omnibus):                  0.705   Jarque-Bera (JB):                0.434
Skew:                          -0.107   Prob(JB):                        0.805
Kurtosis:                       3.127   Cond. No.                     3.38e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
[2] The condition number is large, 3.38e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```



Out[4]:
```
SpearmanrResult(correlation=0.7781825131823187, pvalue=2.3129598993934406e-35)
```

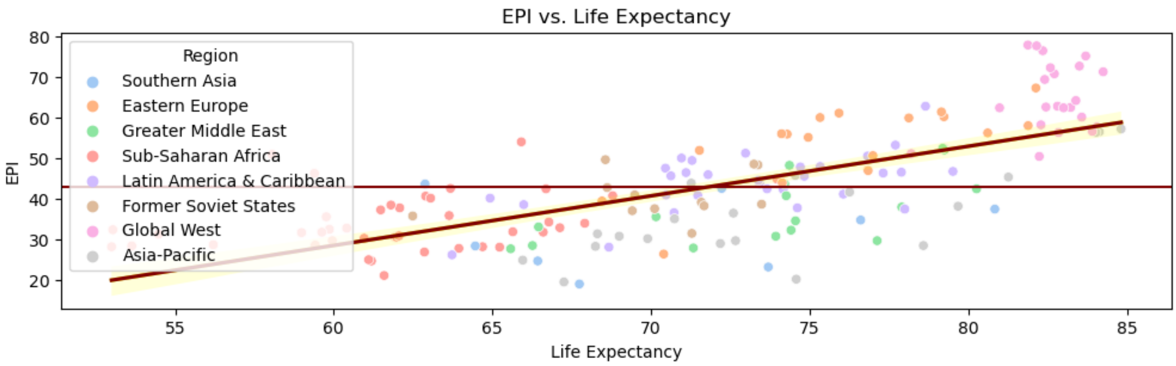
In [5]:
```
#EPI vs Life_Expectancy
x_train = factors[['Life_Expectancy']]
y_train = factors['EPI']
x_test = factors[['Life_Expectancy']]
y_test = factors['EPI']

x_train = sm.add_constant(x_train)
model = sm.OLS(y_train, x_train).fit()
predictions_train = model.predict(x_train)
print_model = model.summary()
print(print_model)
#Plotting
fig = plt.figure(figsize=(12,3))
sns.scatterplot(y=factors.EPI, x=factors.Life_Expectancy, hue=factors.Region, legen
sns.regplot(data=factors, y=factors.EPI, x=factors.Life_Expectancy, ax=fig.gca(), 
sns.regplot(data=factors, y=factors.EPI, x=factors.Life_Expectancy, ax=fig.gca(), 
plt.axhline(factors['EPI'].mean(), color='maroon')
plt.title("EPI vs. Life Expectancy")
plt.xlabel('Life Expectancy')
plt.ylabel('EPI')
plt.show()
#Correlation
```

```
x = factors.EPI
y = factors.Life_Expectancy
stats.spearmanr(x,y)
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                    EPI   R-squared:                       0.509
Model:                            OLS   Adj. R-squared:                  0.506
Method:                 Least Squares   F-statistic:                     172.4
Date:                Sun, 11 Dec 2022   Prob (F-statistic):           1.85e-27
Time:                        15:22:13   Log-Likelihood:                -613.16
No. Observations:                 168   AIC:                             1230.
Df Residuals:                     166   BIC:                             1237.
Df Model:                           1                                         
Covariance Type:            nonrobust                                         
==================================================================================
=
                      coef    std err          t      P>|t|      [0.025      0.97
5]
----------------------------------------------------------------------------------
-
const             -45.0233      6.741     -6.679      0.000     -58.332     -31.71
5
Life_Expectancy     1.2241      0.093     13.128      0.000       1.040       1.40
8
==============================================================================
Omnibus:                        0.361   Durbin-Watson:                   2.045
Prob(Omnibus):                  0.835   Jarque-Bera (JB):                0.127
Skew:                          -0.031   Prob(JB):                        0.938
Kurtosis:                       3.120   Cond. No.                         675.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
```



Out[5]:

```
SpearmanrResult(correlation=0.7238613414141891, pvalue=1.487683365074239e-28)
```

In [6]:

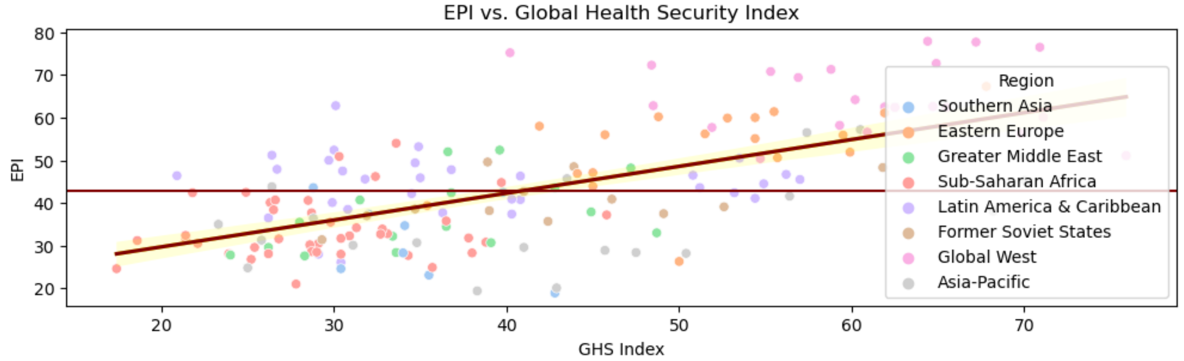
```
#EPI vs GHS Index
x_train = factors[['GHS_Index']]
y_train = factors['EPI']
x_test = factors[['GHS_Index']]
y_test = factors['EPI']

x_train = sm.add_constant(x_train)
model = sm.OLS(y_train, x_train).fit()
predictions_train = model.predict(x_train)
print_model = model.summary()
print(print_model)
#Plotting
fig = plt.figure(figsize=(12,3))
sns.scatterplot(y=factors.EPI, x=factors.GHS_Index, hue=factors.Region, legend='fu
sns.regplot(data=factors, y=factors.EPI, x=factors.GHS_Index, ax=fig.gca(), scatte
```

```
sns.regplot(data=factors, y=factors.EPI, x=factors.GHS_Index, ax=fig.gca(), scatte
plt.axhline(factors['EPI'].mean(), color='maroon')
plt.title("EPI vs. Global Health Security Index")
plt.xlabel('GHS Index')
plt.ylabel('EPI')
plt.show()
#Correlation
x = factors.EPI
y = factors.GHS_Index
stats.spearmanr(x,y)
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                    EPI   R-squared:                       0.410
Model:                            OLS   Adj. R-squared:                  0.407
Method:                 Least Squares   F-statistic:                     115.6
Date:                Sun, 11 Dec 2022   Prob (F-statistic):           8.57e-21
Time:                        15:22:13   Log-Likelihood:                -628.59
No. Observations:                 168   AIC:                             1261.
Df Residuals:                     166   BIC:                             1267.
Df Model:                           1                                         
Covariance Type:            nonrobust                                         
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         17.1410      2.529      6.778      0.000      12.148      22.134
GHS_Index      0.6288      0.058     10.751      0.000       0.513       0.744
==============================================================================
Omnibus:                        1.726   Durbin-Watson:                   2.033
Prob(Omnibus):                  0.422   Jarque-Bera (JB):                1.330
Skew:                           0.188   Prob(JB):                        0.514
Kurtosis:                       3.220   Cond. No.                         138.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
```



Out[6]: SpearmanrResult(correlation=0.6028575893932931, pvalue=5.313985829190582e-18)

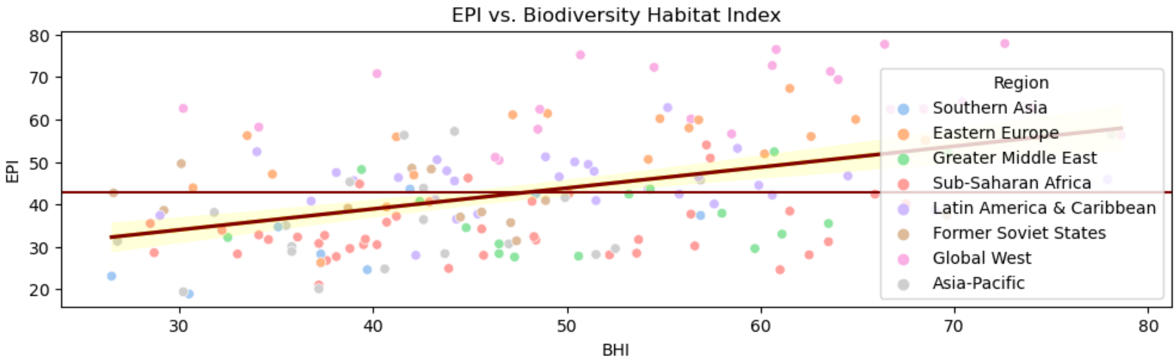
In [7]:
```
#EPI vs BHI
x_train = factors[['BHI']]
y_train = factors['EPI']
x_test = factors[['BHI']]
y_test = factors['EPI']

x_train = sm.add_constant(x_train)
model = sm.OLS(y_train, x_train).fit()
predictions_train = model.predict(x_train)
print_model = model.summary()
print(print_model)
#Plotting
fig = plt.figure(figsize=(12,3))
```

```
sns.scatterplot(y=factors.EPI, x=factors.BHI, hue=factors.Region, legend='full', p
sns.regplot(data=factors, y=factors.EPI, x=factors.BHI, ax=fig.gca(), scatter = Fa
sns.regplot(data=factors, y=factors.EPI, x=factors.BHI, ax=fig.gca(), scatter = Fa
plt.axhline(factors['EPI'].mean(), color='maroon')
plt.title("EPI vs. Biodiversity Habitat Index")
plt.xlabel('BHI')
plt.ylabel('EPI')
plt.show()
#Correlation
x = factors.EPI
y = factors.BHI
stats.spearmanr(x,y)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    EPI   R-squared:                       0.194
Model:                            OLS   Adj. R-squared:                  0.189
Method:                 Least Squares   F-statistic:                     39.93
Date:                Sun, 11 Dec 2022   Prob (F-statistic):           2.33e-09
Time:                        15:22:14   Log-Likelihood:                -654.87
No. Observations:                 168   AIC:                             1314.
Df Residuals:                     166   BIC:                             1320.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         19.2386      3.867      4.976      0.000      11.605      26.872
BHI            0.4923      0.078      6.319      0.000       0.338       0.646
==============================================================================
Omnibus:                        6.392   Durbin-Watson:                   1.829
Prob(Omnibus):                  0.041   Jarque-Bera (JB):                6.463
Skew:                           0.449   Prob(JB):                       0.0395
Kurtosis:                       2.657   Cond. No.                         207.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
```



Out[7]:
```
SpearmanrResult(correlation=0.41239376268679917, pvalue=2.7835796323273694e-08)
```

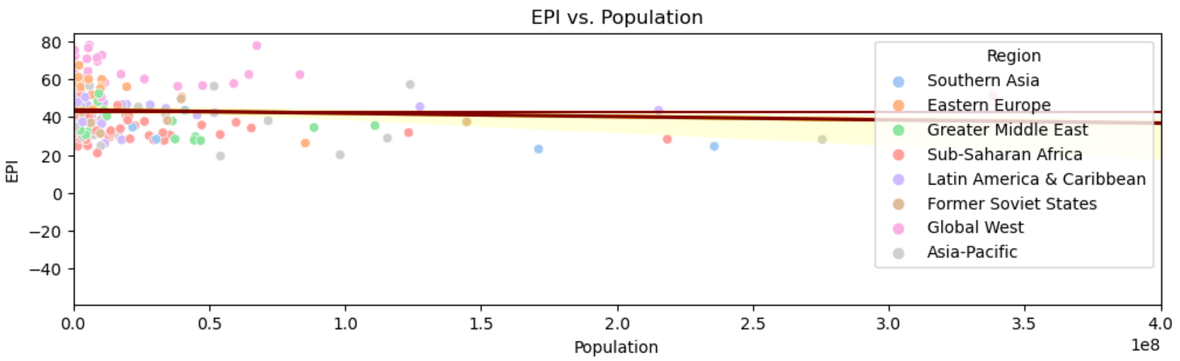
In [8]:
```
#EPI vs Population
x_train = factors[['Population_2022']]
y_train = factors['EPI']
x_test = factors[['Population_2022']]
y_test = factors['EPI']

x_train = sm.add_constant(x_train)
model = sm.OLS(y_train, x_train).fit()
predictions_train = model.predict(x_train)
print_model = model.summary()
print(print_model)
```

```
#Plotting
fig = plt.figure(figsize=(12,3))
sns.scatterplot(y=factors.EPI, x=factors.Population_2022, hue=factors.Region, leger
sns.regplot(data=factors, y=factors.EPI, x=factors.Population_2022, ax=fig.gca(), 
sns.regplot(data=factors, y=factors.EPI, x=factors.Population_2022, ax=fig.gca(), 
plt.axhline(factors['EPI'].mean(), color='maroon')
plt.title("EPI vs. Population")
plt.xlabel('Population')
plt.ylabel('EPI')
plt.xlim(0, 400000000)
plt.show()
#Correlation
x = factors.EPI
y = factors.Population_2022
stats.spearmanr(x,y)
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                    EPI   R-squared:                       0.044
Model:                            OLS   Adj. R-squared:                  0.038
Method:                 Least Squares   F-statistic:                     7.578
Date:                Sun, 11 Dec 2022   Prob (F-statistic):            0.00657
Time:                        15:22:15   Log-Likelihood:                -669.23
No. Observations:                 168   AIC:                             1342.
Df Residuals:                     166   BIC:                             1349.
Df Model:                           1                                         
Covariance Type:            nonrobust                                         
==================================================================================
=
                      coef    std err          t      P>|t|      [0.025      0.97
5]
----------------------------------------------------------------------------------
-
const             43.7600      1.050     41.695      0.000      41.688      45.83
2
Population_2022 -1.743e-08   6.33e-09     -2.753      0.007   -2.99e-08   -4.93e-0
9
==============================================================================
Omnibus:                        9.617   Durbin-Watson:                   1.850
Prob(Omnibus):                  0.008   Jarque-Bera (JB):               10.288
Skew:                           0.592   Prob(JB):                      0.00584
Kurtosis:                       2.738   Cond. No.                     1.72e+08
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
[2] The condition number is large, 1.72e+08. This might indicate that there are
strong multicollinearity or other numerical problems.
```



Out[8]: SpearmanrResult(correlation=-0.2924976763612847, pvalue=0.00011938583848988639)

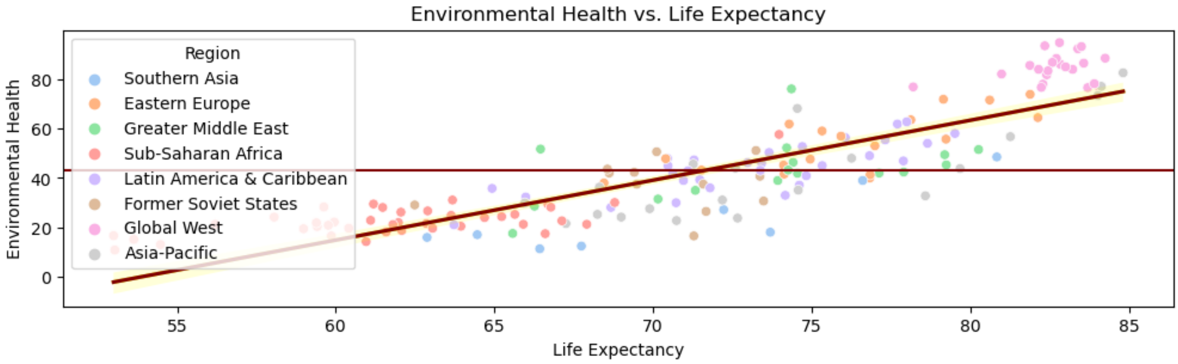
In [9]:

```
#Environmental Health vs Life_Expectancy
x_train = factors[['Life_Expectancy']]
y_train = factors['Env_Health']
x_test = factors[['Life_Expectancy']]
y_test = factors['Env_Health']

x_train = sm.add_constant(x_train)
model = sm.OLS(y_train, x_train).fit()
predictions_train = model.predict(x_train)
print_model = model.summary()
print(print_model)
#Plotting
fig = plt.figure(figsize=(12,3))
sns.scatterplot(y=factors.Env_Health, x=factors.Life_Expectancy, hue=factors.Regio
sns.regplot(data=factors, y=factors.Env_Health, x=factors.Life_Expectancy, ax=fig.
sns.regplot(data=factors, y=factors.Env_Health, x=factors.Life_Expectancy, ax=fig.
plt.axhline(factors['EPI'].mean(), color='maroon')
plt.title("Environmental Health vs. Life Expectancy")
plt.xlabel('Life Expectancy')
plt.ylabel('Environmental Health')
plt.show()
#Correlation
x = factors.Env_Health
y = factors.Life_Expectancy
stats.spearmanr(x,y)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:             Env_Health   R-squared:                       0.738
Model:                            OLS   Adj. R-squared:                  0.736
Method:                 Least Squares   F-statistic:                     466.8
Date:                Sun, 11 Dec 2022   Prob (F-statistic):           4.19e-50
Time:                        15:22:15   Log-Likelihood:                -644.03
No. Observations:                 168   AIC:                             1292.
Df Residuals:                     166   BIC:                             1298.
Df Model:                           1
Covariance Type:            nonrobust
==================================================================================
=
                      coef    std err          t      P>|t|      [0.025      0.97
5]
----------------------------------------------------------------------------------
-
const            -130.3468      8.100    -16.091      0.000    -146.340    -114.35
4
Life_Expectancy     2.4209      0.112     21.605      0.000       2.200       2.64
2
==============================================================================
Omnibus:                        0.994   Durbin-Watson:                   1.836
Prob(Omnibus):                  0.608   Jarque-Bera (JB):                1.006
Skew:                          -0.056   Prob(JB):                        0.605
Kurtosis:                       2.638   Cond. No.                         675.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
```



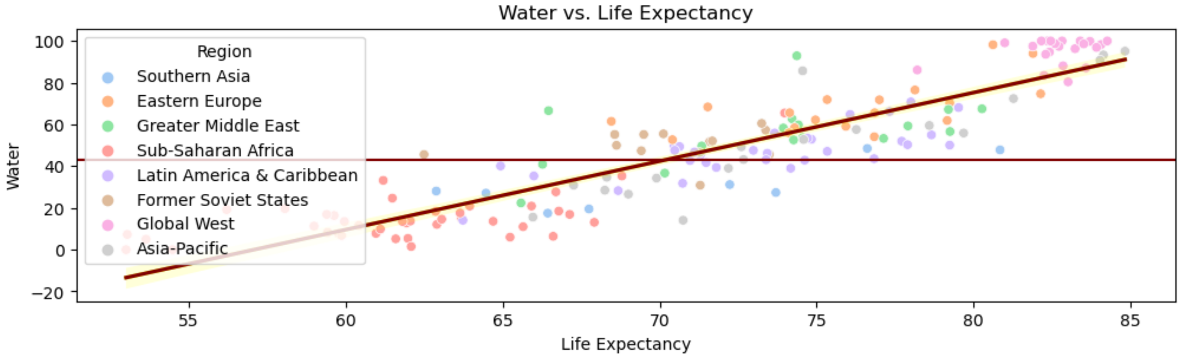
Out[9]: SpearmanrResult(correlation=0.8791698145428337, pvalue=2.547680881811889e-55)

In [10]:
```
#Water vs Life_Expectancy
x_train = factors[['Life_Expectancy']]
y_train = factors['Water']
x_test = factors[['Life_Expectancy']]
y_test = factors['Water']

x_train = sm.add_constant(x_train)
model = sm.OLS(y_train, x_train).fit()
predictions_train = model.predict(x_train)
print_model = model.summary()
print(print_model)
#Plotting
fig = plt.figure(figsize=(12,3))
sns.scatterplot(y=factors.Water, x=factors.Life_Expectancy, hue=factors.Region, le
sns.regplot(data=factors, y=factors.Water, x=factors.Life_Expectancy, ax=fig.gca()
sns.regplot(data=factors, y=factors.Water, x=factors.Life_Expectancy, ax=fig.gca()
plt.axhline(factors['EPI'].mean(), color='maroon')
plt.title("Water vs. Life Expectancy")
plt.xlabel('Life Expectancy')
plt.ylabel('Water')
plt.show()
#Correlation
x = factors.Water
y = factors.Life_Expectancy
stats.spearmanr(x,y)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Water   R-squared:                       0.805
Model:                            OLS   Adj. R-squared:                  0.803
Method:                 Least Squares   F-statistic:                     683.7
Date:                Sun, 11 Dec 2022   Prob (F-statistic):           9.52e-61
Time:                        15:22:16   Log-Likelihood:                -663.17
No. Observations:                 168   AIC:                             1330.
Df Residuals:                     166   BIC:                             1337.
Df Model:                           1
Covariance Type:            nonrobust
==================================================================================
=
                      coef    std err          t      P>|t|      [0.025      0.97
5]
----------------------------------------------------------------------------------
-
const             -187.3841      9.078    -20.642      0.000    -205.307    -169.46
2
Life_Expectancy      3.2833      0.126     26.147      0.000       3.035       3.53
1
==============================================================================
Omnibus:                        0.981   Durbin-Watson:                   1.731
Prob(Omnibus):                  0.612   Jarque-Bera (JB):                0.907
Skew:                           0.179   Prob(JB):                        0.635
Kurtosis:                       2.962   Cond. No.                         675.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
```



Out[10]:

```
SpearmanrResult(correlation=0.8984046821748846, pvalue=3.265359668561759e-61)
```

In [11]:

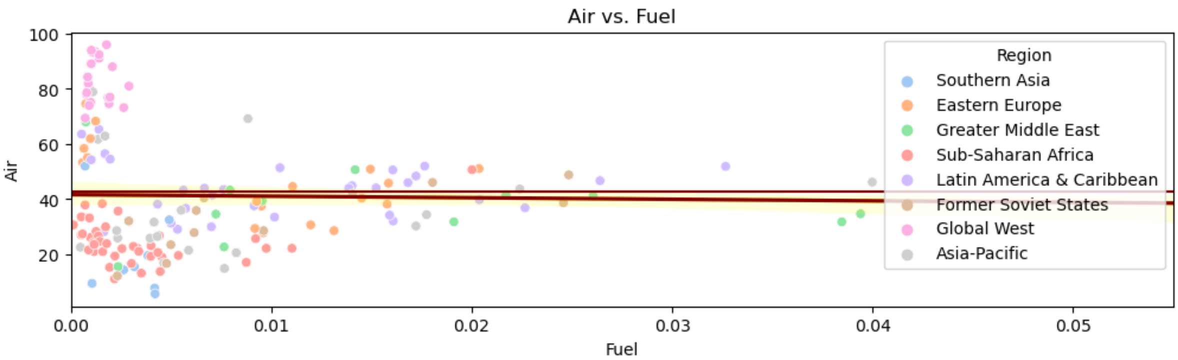
```python
#Fuel_PC vs Air
x_train = factors[['Air']]
y_train = factors['FC_PC']
x_test = factors[['Air']]
y_test = factors['FC_PC']

x_train = sm.add_constant(x_train)
model = sm.OLS(y_train, x_train).fit()
predictions_train = model.predict(x_train)
print_model = model.summary()
print(print_model)
#Plotting
fig = plt.figure(figsize=(12,3))
sns.scatterplot(y=factors.Air, x=factors.FC_PC, hue=factors.Region, legend='full',
sns.regplot(data=factors, y=factors.Air, x=factors.FC_PC, ax=fig.gca(), scatter =
sns.regplot(data=factors, y=factors.Air, x=factors.FC_PC, ax=fig.gca(), scatter =
plt.axhline(factors['EPI'].mean(), color='maroon')
plt.title("Air vs. Fuel")
```

```
plt.xlabel('Fuel')
plt.xlim(0, 0.055)
plt.ylabel('Air')
plt.show()
#Correlation
x = factors.FC_PC
y = factors.Air
stats.spearmanr(x,y)
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                  FC_PC   R-squared:                       0.002
Model:                            OLS   Adj. R-squared:                 -0.004
Method:                 Least Squares   F-statistic:                    0.3127
Date:                Sun, 11 Dec 2022   Prob (F-statistic):              0.577
Time:                        15:22:16   Log-Likelihood:                 451.32
No. Observations:                 168   AIC:                            -898.6
Df Residuals:                     166   BIC:                            -892.4
Df Model:                           1                                         
Covariance Type:            nonrobust                                         
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.0108      0.003      3.884      0.000       0.005       0.016
Air        -3.356e-05      6e-05     -0.559      0.577      -0.000    8.49e-05
==============================================================================
Omnibus:                      185.802   Durbin-Watson:                   2.239
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             4003.966
Skew:                           4.412   Prob(JB):                         0.00
Kurtosis:                      25.229   Cond. No.                         101.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
```



Out[11]:

```
SpearmanrResult(correlation=-0.10478415255054824, pvalue=0.17645787539869248)
```

Multiple Linear Regression

In [12]:

```
#MLR for(Life Expectancy,GHS Index,GDP PC,BHI) and EPI.
x_train = factors[['Life_Expectancy', 'GHS_Index', 'GDP_PC','BHI']]
y_train = factors['EPI']
x_test = factors[['Life_Expectancy', 'GHS_Index', 'GDP_PC','BHI']]
y_test = factors['EPI']
x_train = sm.add_constant(x_train)
model = sm.OLS(y_train, x_train).fit()
predictions_train = model.predict(x_train)
print_model = model.summary()
print(print_model)
x_test = sm.add_constant(x_test, has_constant='add')
```

```
predictions = model.predict(x_test)
mt.r2_score(y_test, predictions)
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                    EPI   R-squared:                       0.666
Model:                            OLS   Adj. R-squared:                  0.657
Method:                 Least Squares   F-statistic:                     81.14
Date:                Sun, 11 Dec 2022   Prob (F-statistic):           9.18e-38
Time:                        15:22:17   Log-Likelihood:                -580.95
No. Observations:                 168   AIC:                             1172.
Df Residuals:                     163   BIC:                             1188.
Df Model:                           4                                         
Covariance Type:            nonrobust                                         
==================================================================================
=
                     coef    std err          t      P>|t|      [0.025      0.97
5]
----------------------------------------------------------------------------------
-
const            -17.9789      7.580     -2.372      0.019     -32.946      -3.01
2
Life_Expectancy    0.5786      0.118      4.890      0.000       0.345       0.81
2
GHS_Index          0.1372      0.066      2.064      0.041       0.006       0.26
8
GDP_PC             0.0002   3.94e-05      5.179      0.000       0.000       0.00
0
BHI                0.2172      0.055      3.984      0.000       0.110       0.32
5
==============================================================================
Omnibus:                        0.240   Durbin-Watson:                   2.089
Prob(Omnibus):                  0.887   Jarque-Bera (JB):                0.199
Skew:                          -0.083   Prob(JB):                        0.905
Kurtosis:                       2.971   Cond. No.                     3.46e+05
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
[2] The condition number is large, 3.46e+05. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Out[12]:

```
0.6656692594960063
```

In [13]:

```
#MLR for (Air quality and Water Quality ) and Life Expectancy
x_train = factors[['Air', 'Water']]
y_train = factors['Life_Expectancy']
x_test = factors[['Air', 'Water']]
y_test = factors['Life_Expectancy']
x_train = sm.add_constant(x_train)
model = sm.OLS(y_train, x_train).fit()
predictions_train = model.predict(x_train)
print_model = model.summary()
print(print_model)
x_test = sm.add_constant(x_test, has_constant='add')
predictions = model.predict(x_test)
mt.r2_score(y_test, predictions)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:        Life_Expectancy   R-squared:                       0.807
Model:                            OLS   Adj. R-squared:                  0.804
Method:                 Least Squares   F-statistic:                     344.6
Date:                Sun, 11 Dec 2022   Prob (F-statistic):           1.22e-59
Time:                        15:22:17   Log-Likelihood:                -444.23
No. Observations:                 168   AIC:                             894.5
Df Residuals:                     165   BIC:                             903.8
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         59.6373      0.577    103.291      0.000      58.497      60.777
Air            0.0323      0.024      1.373      0.172      -0.014      0.079
Water          0.2245      0.018     12.708      0.000       0.190      0.259
==============================================================================
Omnibus:                        3.495   Durbin-Watson:                   1.792
Prob(Omnibus):                  0.174   Jarque-Bera (JB):                3.053
Skew:                          -0.310   Prob(JB):                        0.217
Kurtosis:                       3.226   Cond. No.                         157.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
```

Out[13]:

```
0.8068408628799345
```

Observations

1) Positive strong correlation between EPI and GDP
2) Positive strong correlation between EPI and Life Expectancy
3) Positive correlation between EPI and GHS Index
4) Positive correlation between EPI and Biodiversity Habitat Index
5) Negative correlation between EPI and Population
6) Negative correlation between Fuel Consumption Per Capita and Air Quality
7) Positive strong correlation between Environment Health and Life_Expectancy
8) Positive strong correlation between Water and Life_Expectancy
9) Using Multiple Linear Regression, a strong correlation was found between (Life Expectancy,GHS Index,GDP PC,BHI) and EPI.
10) Using Multiple Linear Regression, a very strong correlation was found between (Air quality and Water Quality) and Life Expectancy