Practical 7 - JosiahTeh IPYNB

December 28, 2021

1 First Name: Josiah

2 Last Name: Teh

[1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import scipy
import matplotlib.pyplot as plt
import statsmodels.formula.api as smf
```

[2]:
```
pd.set_option('display.float_format', lambda x:'%.2f'%x)

gapminder = pd.read_csv('gapminder.csv', low_memory=False)
gapminder.head()
```

[2]:

	country	incomeperperson	alcconsumption	armedforcesrate	\
0	Afghanistan		.03	.5696534	
1	Albania	1914.99655094922	7.29	1.0247361	
2	Algeria	2231.99333515006	.69	2.306817	
3	Andorra	21943.3398976022	10.17		
4	Angola	1381.00426770244	5.57	1.4613288	

	breastcancerper100th	co2emissions	femaleemployrate	hivrate	\
0	26.8	75944000	25.6000003814697		
1	57.4	223747333.333333	42.0999984741211		
2	23.5	2932108666.66667	31.7000007629394	.1	
3					
4	23.1	248358000	69.4000015258789	2	

	internetuserate	lifeexpectancy	oilperperson	polityscore	\
0	3.65412162280064	48.673		0	
1	44.9899469578783	76.918		9	
2	12.5000733055148	73.131	.42009452521537	2	
3	81				
4	9.99995388324075	51.093		-2	

	relectricperperson	suicideper100th	employrate	urbanrate
0		6.68438529968262	55.7000007629394	24.04
1	636.341383366604	7.69932985305786	51.4000015258789	46.72
2	590.509814347428	4.8487696647644	50.5	65.22
3		5.36217880249023		88.92
4	172.999227388199	14.5546770095825	75.6999969482422	56.7

[3]:
```
#setting variables you will be working with to numeric
gapminder['oilperperson'] = pd.
 ↪to_numeric(gapminder['oilperperson'],errors='coerce')
gapminder['relectricperperson'] = pd.
 ↪to_numeric(gapminder['relectricperperson'],errors='coerce')
gapminder['co2emissions'] = pd.
 ↪to_numeric(gapminder['co2emissions'],errors='coerce')
```

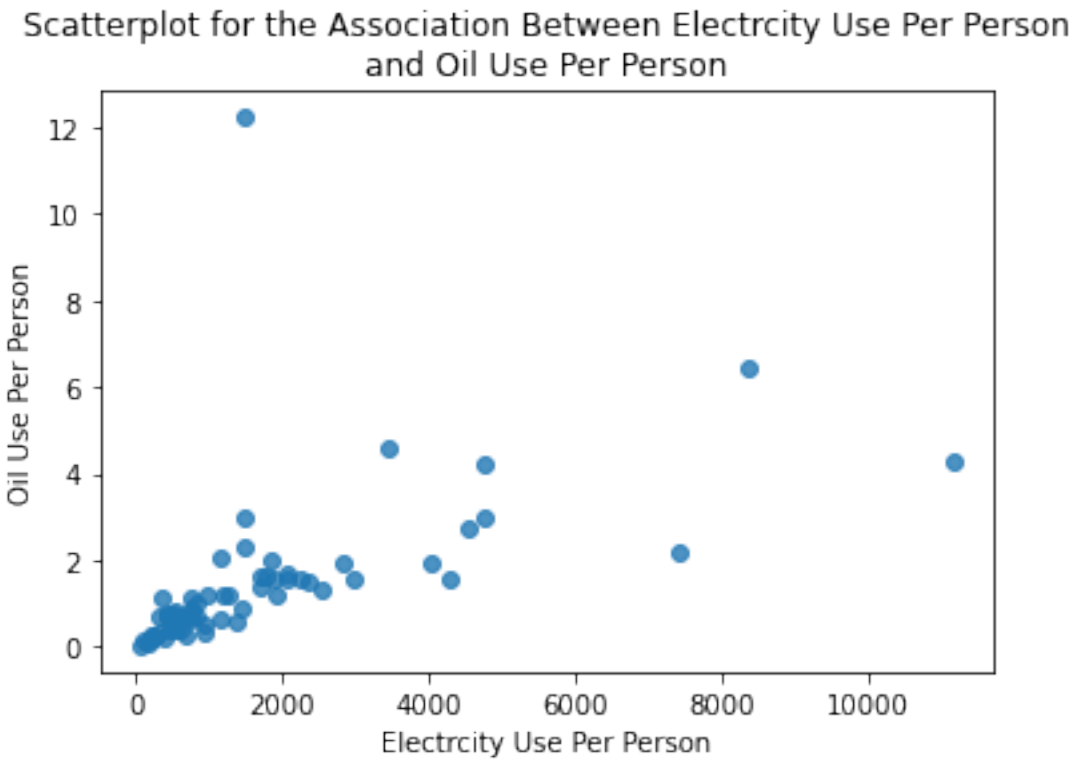
[4]:
```
gapminder_clean=gapminder.dropna()
```

3 Correlation - Scenario 1

4 Scatter plot to show association between relectricperperson (x) and oilperperson (y)

[6]:
```
# hint lecture cell 5
%matplotlib inline
plt.figure()
scat1 = sns.regplot(x='relectricperperson', y='oilperperson', fit_reg=False,␣
 ↪data=gapminder)
plt.xlabel('Electrcity Use Per Person')
plt.ylabel('Oil Use Per Person')
plt.title('Scatterplot for the Association Between Electrcity Use Per Person' +␣
 ↪'\n' + 'and Oil Use Per Person')
```

[6]:
```
Text(0.5, 1.0, 'Scatterplot for the Association Between Electrcity Use Per
Person\nand Oil Use Per Person')
```



5 Pearson correlation - relectricperperson (x) and oilperperson (y)

[8]:
```
# hint lecture cell 6
print ('association between relectricperperson and oilperperson')
print (scipy.stats.pearsonr(gapminder_clean['relectricperperson'],␣
 ↪gapminder_clean['oilperperson'])) #pearson correlation
```

```
association between relectricperperson and oilperperson
(0.5249373779159884, 1.0020621767836635e-05)
```

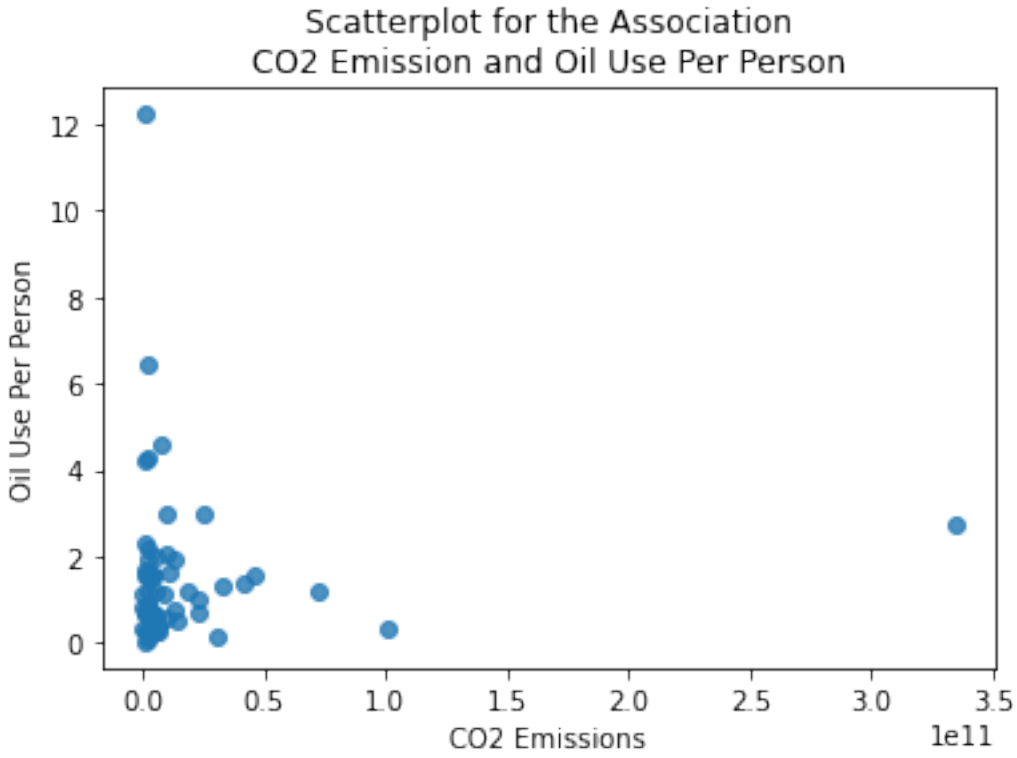
6 Correlation - Scenario 2

7 Scatter plot to show association between co2emissions (x) and oilperperson (y)

[9]:
```
# hint lecture cell 7
%matplotlib inline
plt.figure()
```

```
scat2 = sns.regplot(x='co2emissions', y='oilperperson', fit_reg=False,␣
 ↪data=gapminder)
plt.xlabel('CO2 Emissions')
plt.ylabel('Oil Use Per Person')
plt.title('Scatterplot for the Association' + '\n' + 'CO2 Emission and Oil Use␣
 ↪Per Person')
```

[9]:
```
Text(0.5, 1.0, 'Scatterplot for the Association\nCO2 Emission and Oil Use Per
Person')
```



8 Pearson correlation - co2emissions (x) and oilperperson (y)

[10]:
```
# hint lecture cell 8
print ('association between co2emissions and oilperperson')
print (scipy.stats.pearsonr(gapminder_clean['co2emissions'],␣
 ↪gapminder_clean['oilperperson'])) #pearson correlation
```

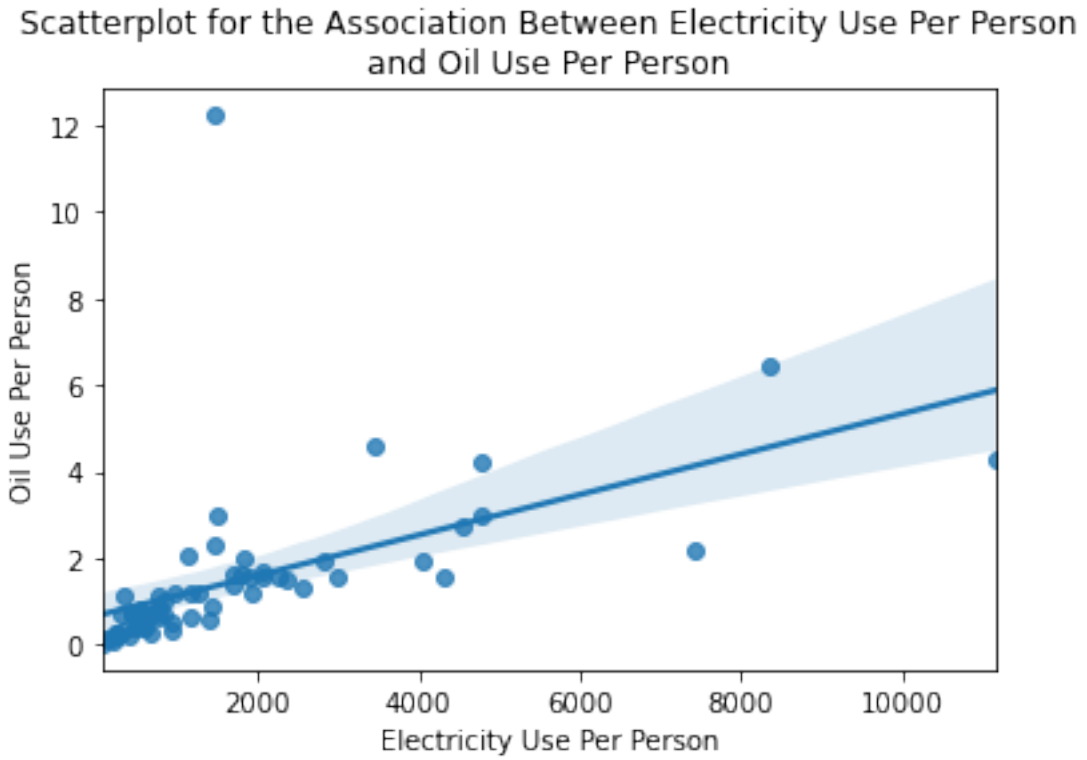
```
association between co2emissions and oilperperson
(0.04444201231228795, 0.7294518840123059)
```

9 Regression - Scenario 3

10 Scatter plot with regression to show relationship between relectricperperson (x) and oilperperson (y) - with regression line

[11]:
```
# hint lecture cell 9
%matplotlib inline
scat1 = sns.regplot(x='relectricperperson', y='oilperperson', fit_reg=True,␣
↪data=gapminder_clean)
plt.xlabel('Electricity Use Per Person')
plt.ylabel('Oil Use Per Person')
plt.title('Scatterplot for the Association Between Electricity Use Per Person'␣
↪+ '\n' + 'and Oil Use Per Person')
```

[11]:
```
Text(0.5, 1.0, 'Scatterplot for the Association Between Electricity Use Per
Person\nand Oil Use Per Person')
```



11 Regression analysis to show association between relectricperperson (x) and oilperperson (y)

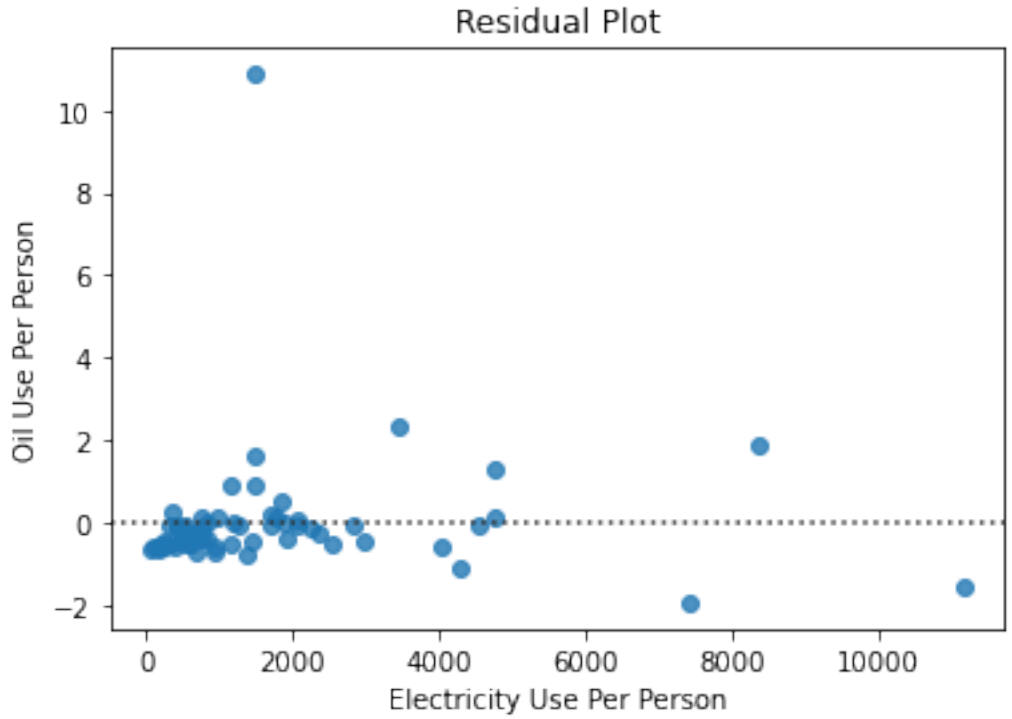
[12]:
```
# hint lecture cell 10
print ("OLS regression model for the association between Electric Use Per␣
 ↪Person and Oil Per Person")
reg1 = smf.ols('oilperperson ~ relectricperperson', data=gapminder_clean).fit()
print (reg1.summary())
```

```
OLS regression model for the association between Electric Use Per Person and Oil
Per Person
                            OLS Regression Results
==============================================================================
Dep. Variable:           oilperperson   R-squared:                       0.276
Model:                            OLS   Adj. R-squared:                  0.264
Method:                 Least Squares   F-statistic:                     23.20
Date:                Tue, 28 Dec 2021   Prob (F-statistic):           1.00e-05
Time:                        01:02:16   Log-Likelihood:                -116.64
No. Observations:                  63   AIC:                             237.3
Df Residuals:                      61   BIC:                             241.6
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
======
                         coef    std err          t      P>|t|      [0.025
0.975]
------------------------------------------------------------------------------
------
Intercept              0.6736      0.259      2.598      0.012       0.155
1.192
relectricperperson     0.0005   9.69e-05      4.817      0.000       0.000
0.001
==============================================================================
Omnibus:                      112.807   Durbin-Watson:                   1.627
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             3834.005
Skew:                           5.613   Prob(JB):                         0.00
Kurtosis:                      39.531   Cond. No.                     3.52e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 3.52e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

12 Residual plot - regression analysis between relectricperperson (x) and oilperperson (y)

[13]:
```
# hint lecture cell 11
%matplotlib inline
scat1 = sns.residplot(x='relectricperperson', y='oilperperson',␣
 ↪data=gapminder_clean)
plt.xlabel('Electricity Use Per Person')
plt.ylabel('Oil Use Per Person')
plt.title('Residual Plot')
```

[13]:
```
Text(0.5, 1.0, 'Residual Plot')
```



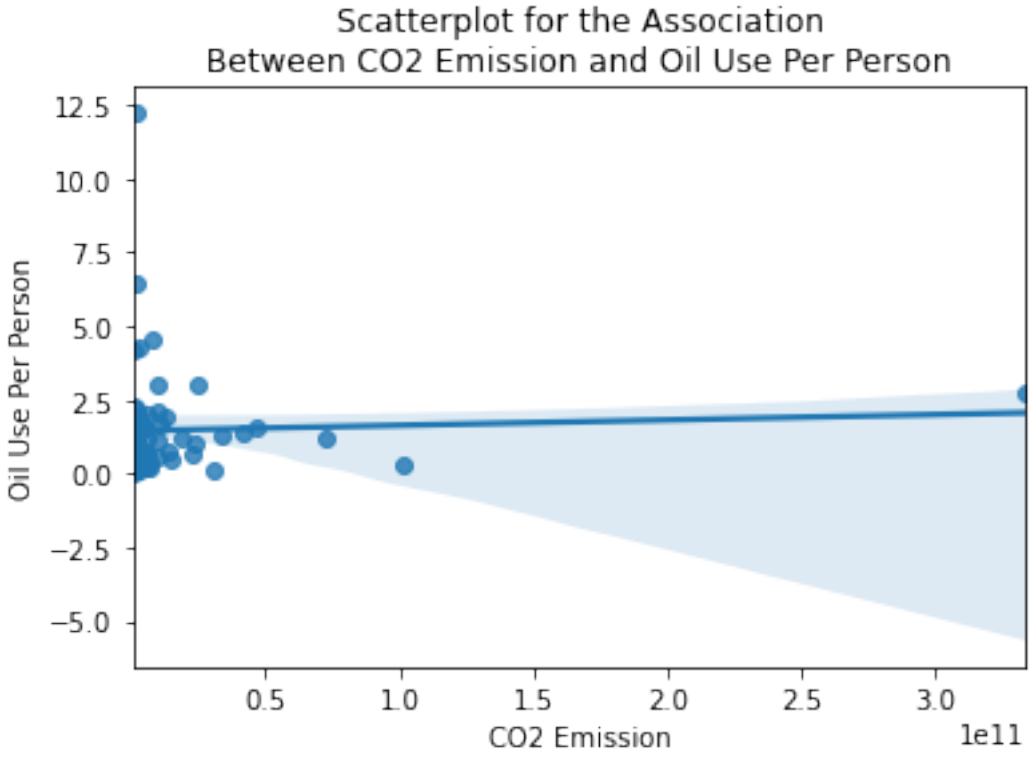
13 # Regression - Scenario 4

14 Scatter plot with regression to show association between co2emissions (x) and oilperperson (y) - with regression line

[14]:
```
# hint lecture cell 12
plt.figure()
scat2 = sns.regplot(x='co2emissions', y='oilperperson', fit_reg=True,␣
 ↪data=gapminder_clean)
```

```
plt.xlabel('CO2 Emission')
plt.ylabel('Oil Use Per Person')
plt.title('Scatterplot for the Association' + '\n' + 'Between CO2 Emission and␣
 ↪Oil Use Per Person')
```

[14]:
```
Text(0.5, 1.0, 'Scatterplot for the Association\nBetween CO2 Emission and Oil
Use Per Person')
```



15 Regression analysis to show association between co2emissions (x) and oilperperson (y)

[16]:
```
# hint lecture cell 13
print ("OLS regression model for the association between CO2 emission and Oil␣
 ↪Use Per Person")
reg1 = smf.ols('oilperperson ~ co2emissions', data=gapminder_clean).fit()
print (reg1.summary())
```

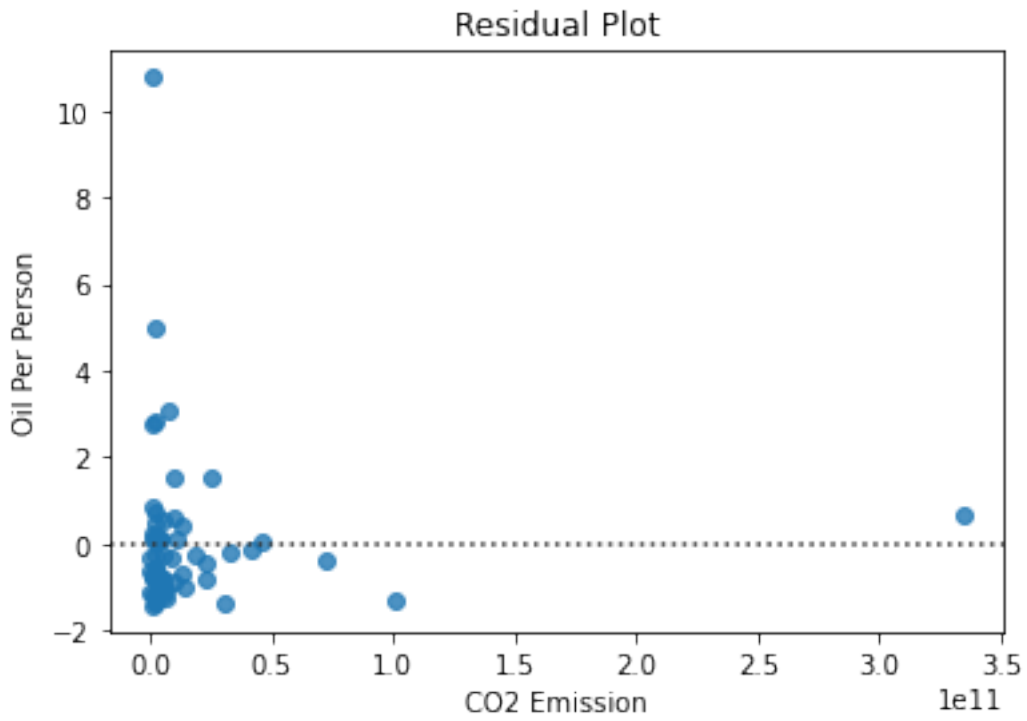
```
OLS regression model for the association between CO2 emission and Oil Use Per
Person
                            OLS Regression Results
==============================================================================
Dep. Variable:           oilperperson   R-squared:                       0.002
```

```
Model:                            OLS   Adj. R-squared:                 -0.014
Method:                 Least Squares   F-statistic:                    0.1207
Date:                Tue, 28 Dec 2021   Prob (F-statistic):             0.729
Time:                        01:04:34   Log-Likelihood:                -126.73
No. Observations:                  63   AIC:                             257.5
Df Residuals:                      61   BIC:                             261.7
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      1.4561      0.245      5.939      0.000       0.966       1.946
co2emissions 1.829e-12   5.26e-12      0.347      0.729    -8.7e-12    1.24e-11
==============================================================================
Omnibus:                       82.847   Durbin-Watson:                   1.727
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1029.853
Skew:                           3.814   Prob(JB):                    2.35e-224
Kurtosis:                      21.279   Cond. No.                     4.93e+10
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 4.93e+10. This might indicate that there are
strong multicollinearity or other numerical problems.
```

16 Residual plot - regression analysis between co2emissions (x) and oilperperson (y)

[17]:
```
# hint lecture cell 14
%matplotlib inline
scat1 = sns.residplot(x='co2emissions', y='oilperperson', data=gapminder_clean)
plt.xlabel('CO2 Emission')
plt.ylabel('Oil Per Person')
plt.title('Residual Plot')
```

[17]:
```
Text(0.5, 1.0, 'Residual Plot')
```



17 Regression with 3 variables

18 Use co2emissionsgrp function to divide/group data into 3 groups

19 Low co2emission (1): min - 1846084167

20 Medium co2emission (2): 1846084168 - 7993752800

21 High co2emission (3): 7993752801 - max

[18]:
```
def co2emissionsgrp  (row):
   if row['co2emissions'] <= 1846084167:
      return 1
   elif row['co2emissions'] <= 7993752800:
      return 2
   elif row['co2emissions'] > 7993752800:
      return 3
```

[19]:
```
gapminder_clean['co2emissionsgrp'] = gapminder_clean.apply (lambda row:␣
↪co2emissionsgrp (row),axis=1)
```

```
<ipython-input-19-8ea3e03abd80>: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
  gapminder_clean['co2emissionsgrp'] = gapminder_clean.apply (lambda row:
co2emissionsgrp (row),axis=1)
```

22 Print the number of countries in each group of CO2 emission

[20]:
```
# hint lecture cell 17
chk1 = gapminder_clean['co2emissionsgrp'].value_counts(sort=False, dropna=False)
print(chk1)
```

```
1    17
2    27
3    19
Name: co2emissionsgrp, dtype: int64
```

23 Divide gapminder_clean into 3 dataframes, each dataframe representing rows of data in low, medium and high CO2 Emission

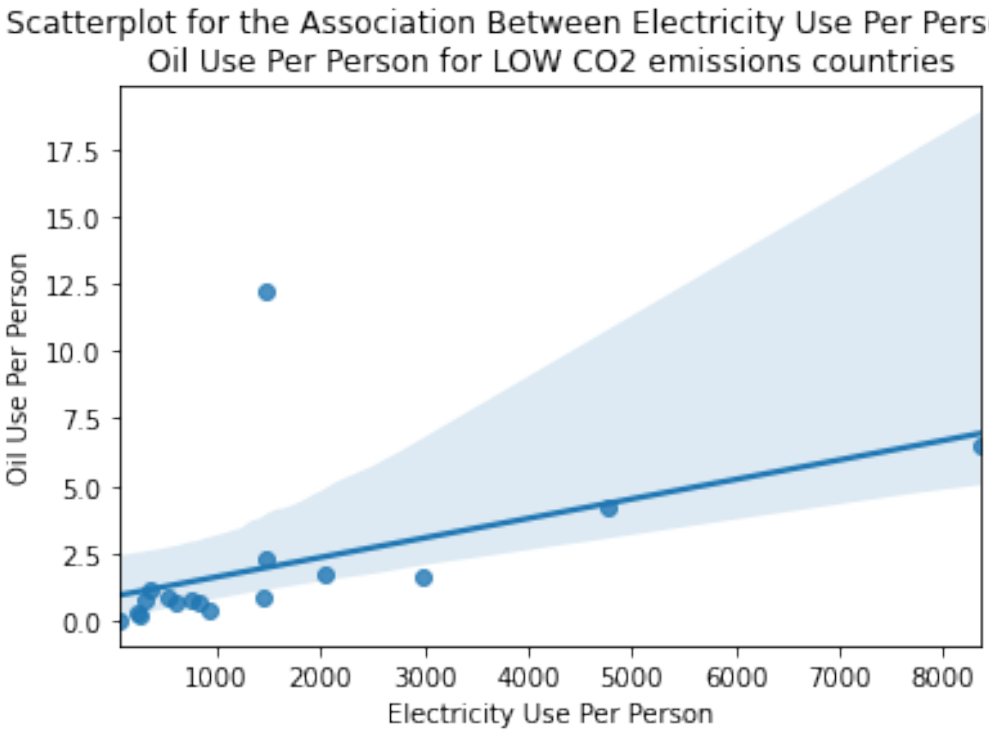
[21]:
```
sub1=gapminder_clean[(gapminder_clean['co2emissionsgrp']== 1)]
sub2=gapminder_clean[(gapminder_clean['co2emissionsgrp']== 2)]
sub3=gapminder_clean[(gapminder_clean['co2emissionsgrp']== 3)]
```

24 Regression - Scenario 5

25 Scatter plot with regression analysis to show association between electricity use per person (x) and oilperperson (y) for low CO2 emission countries

[22]:
```
# hint lecture cell 19
%matplotlib inline
scat1 = sns.regplot(x='relectricperperson', y='oilperperson', data=sub1)
plt.xlabel('Electricity Use Per Person')
plt.ylabel('Oil Use Per Person')
plt.title('Scatterplot for the Association Between Electricity Use Per Person␣
 ↪and' +  '\n' + 'Oil Use Per Person for LOW CO2 emissions countries')
print (scat1)
```

```
AxesSubplot(0.125,0.125;0.775x0.755)
```



26 Regression analysis to show association between electricity use per person (x) and oilperperson (y) for low CO2 emission countries

[23]:
```
# hint lecture cell 20
print ('OLS regression model for the association between Electricty Use Per␣
 ↪Person and Oil Use Per Person for' + '\n' + 'LOW CO2 Emission countries')
reg1 = smf.ols('oilperperson ~ relectricperperson', data=sub1).fit()
print (reg1.summary())
```

```
OLS regression model for the association between Electricty Use Per Person and
Oil Use Per Person for
LOW CO2 Emission countries
                            OLS Regression Results
==============================================================================
Dep. Variable:           oilperperson   R-squared:                       0.244
Model:                            OLS   Adj. R-squared:                  0.194
Method:                 Least Squares   F-statistic:                     4.840
Date:                Tue, 28 Dec 2021   Prob (F-statistic):             0.0439
Time:                        01:07:27   Log-Likelihood:                -40.387
No. Observations:                  17   AIC:                             84.77
Df Residuals:                      15   BIC:                             86.44
```

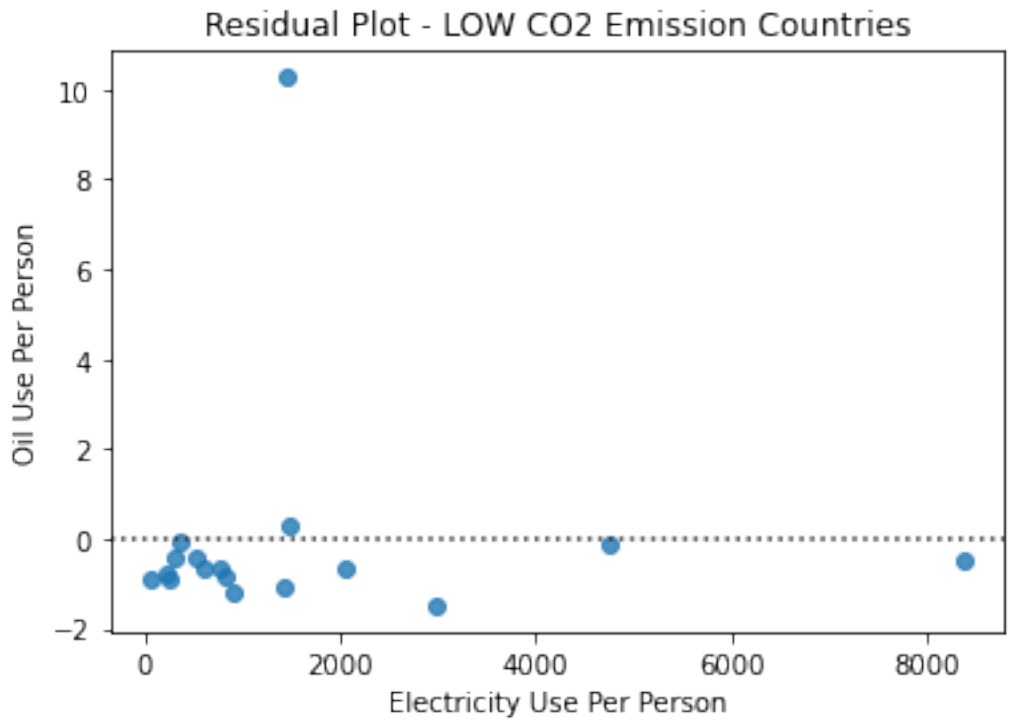
```
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
======
                          coef    std err          t      P>|t|      [0.025
0.975]
------------------------------------------------------------------------------
------
Intercept              0.8962      0.856      1.046      0.312      -0.929
2.722
relectricperperson     0.0007      0.000      2.200      0.044    2.25e-05
0.001
==============================================================================
Omnibus:                       43.166   Durbin-Watson:                   2.057
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              126.442
Skew:                           3.582   Prob(JB):                     3.50e-28
Kurtosis:                      14.278   Cond. No.                     3.32e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 3.32e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

```
C:\Users\Admin\anaconda3\lib\site-packages\scipy\stats\stats.py:1603:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=17
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
```

27 Residual plot - regression analysis between relectricperperson (x) and oilperperson (y) for Low CO2 emission countries

[24]:
```
# hint lecture cell 23
%matplotlib inline
scat1 = sns.residplot(x='relectricperperson', y='oilperperson', data=sub1)
plt.xlabel('Electricity Use Per Person')
plt.ylabel('Oil Use Per Person')
plt.title('Residual Plot - LOW CO2 Emission Countries')
```

[24]:
```
Text(0.5, 1.0, 'Residual Plot - LOW CO2 Emission Countries')
```

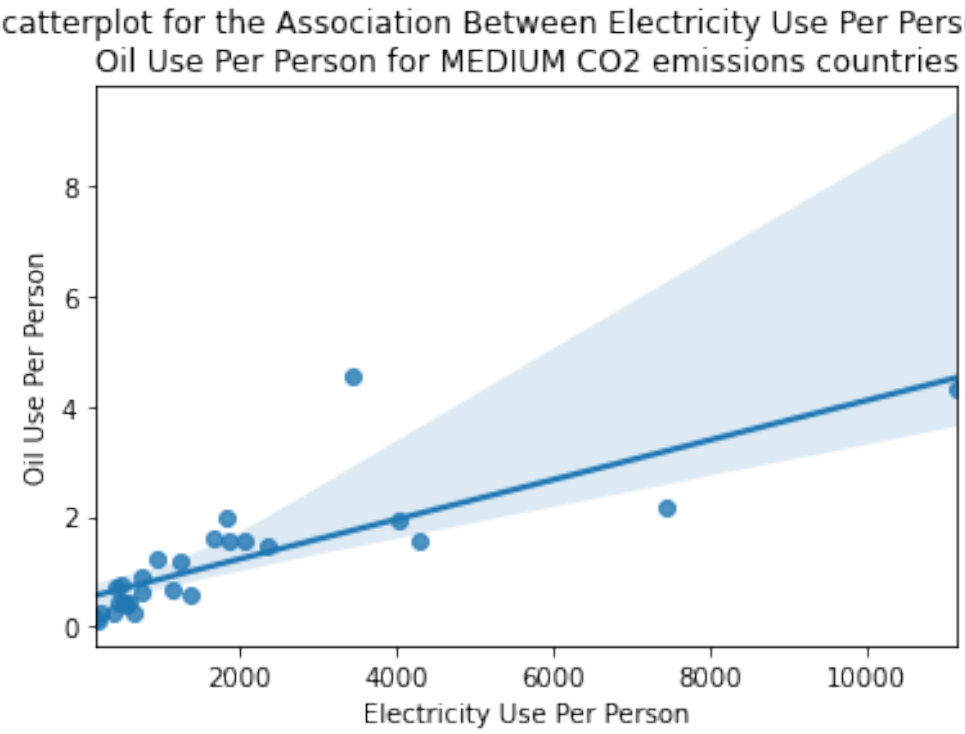


28 Regression - Scenario 6

29 Scatter plot with regression analysis to show association between electricity use per person (x) and oilperperson (y) for medium CO2 emission countries

[25]:
```
# hint lecture cell 24
%matplotlib inline
scat1 = sns.regplot(x='relectricperperson', y='oilperperson', data=sub2)
plt.xlabel('Electricity Use Per Person')
plt.ylabel('Oil Use Per Person')
plt.title('Scatterplot for the Association Between Electricity Use Per Person␣
 ↪and' +  '\n' + 'Oil Use Per Person for MEDIUM CO2 emissions countries')
print (scat1)
```

```
AxesSubplot(0.125,0.125;0.775x0.755)
```



[26]:
```
# hint lecture cell 25
print ('OLS regression model for the association between Electricty Use Per␣
 ↪Person and Oil Use Per Person for' + '\n' + 'MEDIUM CO2 Emission countries')
reg1 = smf.ols('oilperperson ~ relectricperperson', data=sub2).fit()
print (reg1.summary())
```

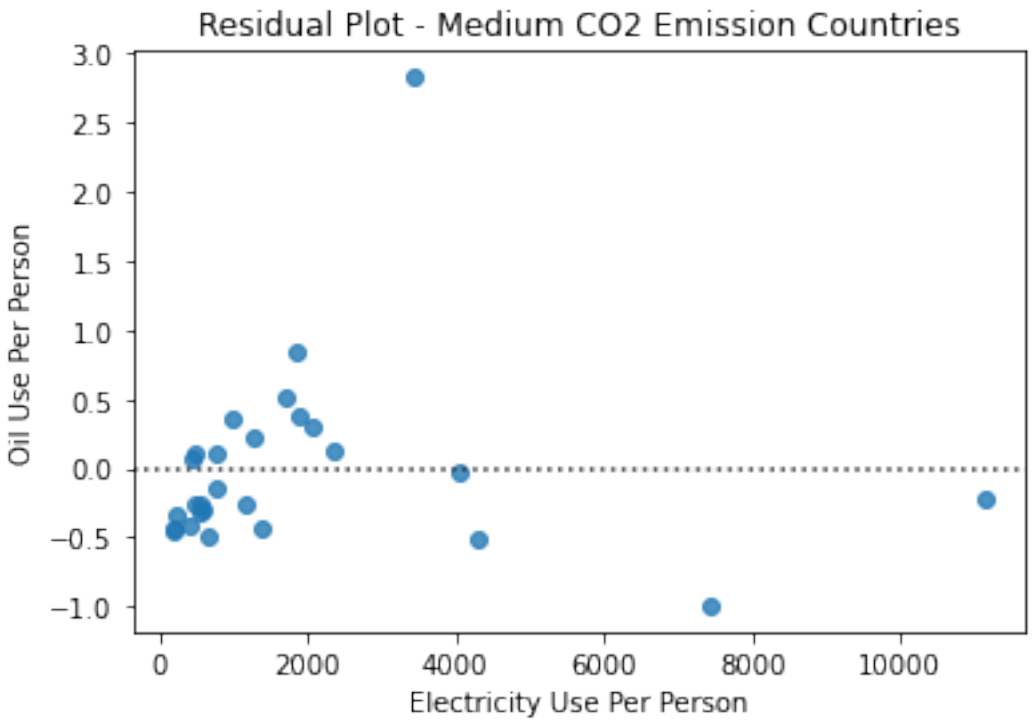
```
OLS regression model for the association between Electricty Use Per Person and
Oil Use Per Person for
MEDIUM CO2 Emission countries
                            OLS Regression Results
==============================================================================
Dep. Variable:           oilperperson   R-squared:                       0.626
Model:                            OLS   Adj. R-squared:                  0.611
Method:                 Least Squares   F-statistic:                     41.89
Date:                Tue, 28 Dec 2021   Prob (F-statistic):           8.88e-07
Time:                        01:09:49   Log-Likelihood:                -27.631
No. Observations:                  27   AIC:                             59.26
Df Residuals:                      25   BIC:                             61.85
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
======
                         coef    std err          t      P>|t|      [0.025
```

```
0.975]
------------------------------------------------------------------------------
------
Intercept              0.5063      0.171      2.958      0.007       0.154
0.859
relectricperperson     0.0004   5.57e-05      6.472      0.000       0.000
0.000
==============================================================================
Omnibus:                       37.330   Durbin-Watson:                   2.273
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              120.141
Skew:                           2.643   Prob(JB):                     8.16e-27
Kurtosis:                      11.880   Cond. No.                     3.91e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 3.91e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

30 Residual plot - regression analysis between relectricperperson (x) and oilperperson (y) for Medium CO2 emission countries

[27]:
```
# hint lecture cell 23
%matplotlib inline
scat1 = sns.residplot(x='relectricperperson', y='oilperperson', data=sub2)
plt.xlabel('Electricity Use Per Person')
plt.ylabel('Oil Use Per Person')
plt.title('Residual Plot - Medium CO2 Emission Countries')
```

[27]:
```
Text(0.5, 1.0, 'Residual Plot - Medium CO2 Emission Countries')
```

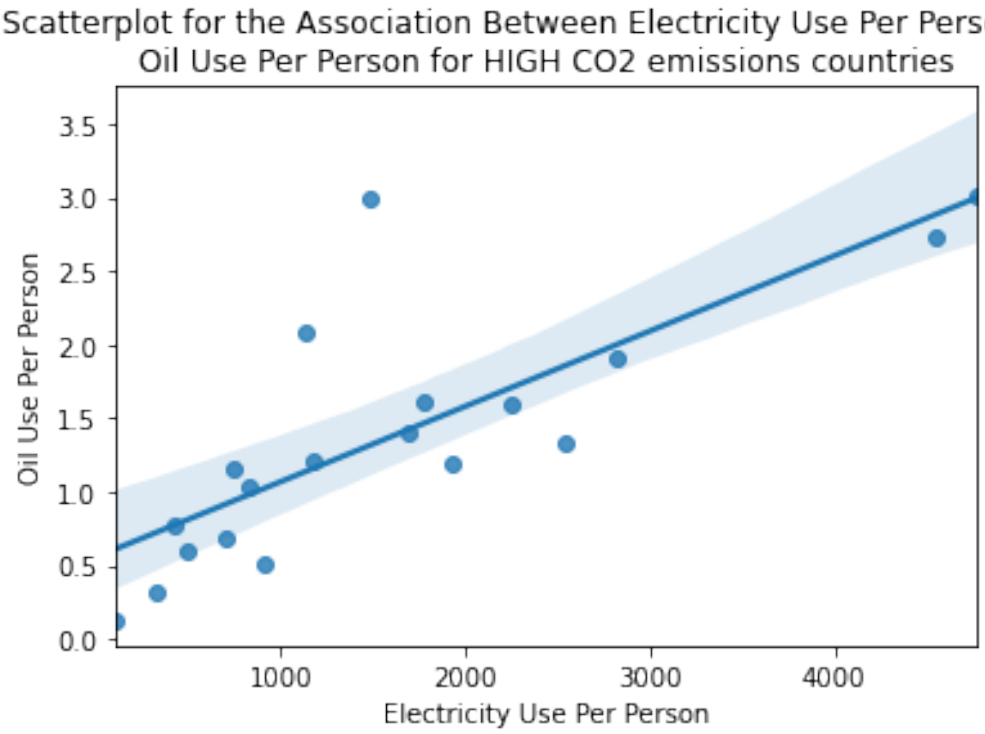


31 Regression - Scenario 7

32 Scatter plot with regression analysis to show association between electricity use per person (x) and oilperperson (y) for high CO2 emission countries

[28]:
```
# hint lecture cell 24
%matplotlib inline
scat1 = sns.regplot(x='relectricperperson', y='oilperperson', data=sub3)
plt.xlabel('Electricity Use Per Person')
plt.ylabel('Oil Use Per Person')
plt.title('Scatterplot for the Association Between Electricity Use Per Person␣
 ↪and' +  '\n' + 'Oil Use Per Person for HIGH CO2 emissions countries')
print (scat1)
```

```
AxesSubplot(0.125,0.125;0.775x0.755)
```



[29]:
```
# hint lecture cell 25
print ('OLS regression model for the association between Electricty Use Per␣
 ↪Person and Oil Use Per Person for' + '\n' + 'HIGH CO2 Emission countries')
reg1 = smf.ols('oilperperson ~ relectricperperson', data=sub3).fit()
print (reg1.summary())
```

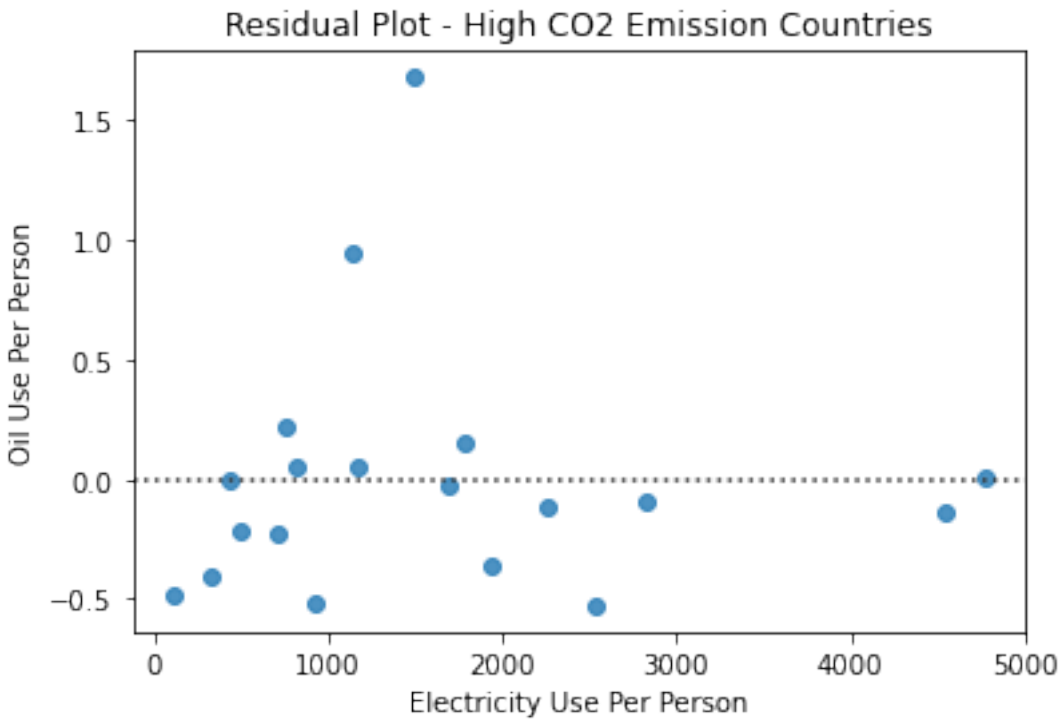
```
OLS regression model for the association between Electricty Use Per Person and
Oil Use Per Person for
HIGH CO2 Emission countries
                            OLS Regression Results
==============================================================================
Dep. Variable:           oilperperson   R-squared:                       0.619
Model:                            OLS   Adj. R-squared:                  0.597
Method:                 Least Squares   F-statistic:                     27.61
Date:                Tue, 28 Dec 2021   Prob (F-statistic):           6.45e-05
Time:                        01:11:25   Log-Likelihood:                -14.302
No. Observations:                  19   AIC:                             32.60
Df Residuals:                      17   BIC:                             34.49
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
======
                         coef    std err          t      P>|t|      [0.025
```

```
0.975]
------------------------------------------------------------------------------
------
Intercept              0.5552      0.201      2.764      0.013       0.131
0.979
relectricperperson     0.0005   9.74e-05      5.255      0.000       0.000
0.001
==============================================================================
Omnibus:                       20.501   Durbin-Watson:                   2.188
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               23.814
Skew:                           1.966   Prob(JB):                     6.74e-06
Kurtosis:                       6.823   Cond. No.                     3.32e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 3.32e+03. This might indicate that there are
strong multicollinearity or other numerical problems.

C:\Users\Admin\anaconda3\lib\site-packages\scipy\stats\stats.py:1603:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=19
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
```

33 Residual plot - regression analysis between relectricperperson (x) and oilperperson (y) for High CO2 emission countries

[30]:
```
# hint lecture cell 23
%matplotlib inline
scat1 = sns.residplot(x='relectricperperson', y='oilperperson', data=sub3)
plt.xlabel('Electricity Use Per Person')
plt.ylabel('Oil Use Per Person')
plt.title('Residual Plot - High CO2 Emission Countries')
```

[30]:
```
Text(0.5, 1.0, 'Residual Plot - High CO2 Emission Countries')
```


Residual Plot - High CO2 Emission Countries
Oil Use Per Person
Electricity Use Per Person
1.5
1.0
0.5
0.0
−0.5
0
1000
2000
3000
4000
5000