# First Name: Last Name:

In [1]:

**import** pandas **as** pd **import** numpy **as** np **import** seaborn **as** sns **import** scipy

**import** matplotlib.pyplot **as** plt

In [2]:

pd.set\_option('display.float\_format', **lambda** x:'%.2f'**%**x)

gapminder = pd.read\_csv('gapminder.csv', low\_memory=**False**) gapminder.head()

Out[2]:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **country** | **incomeperperson** | **alcconsumption** | **armedforcesrate** | **breastcancerper100th** | **c** |
| **0** Afghanistan |  | .03 | .5696534 | 26.8 |  |
| **1** Albania | 1914.99655094922 | 7.29 | 1.0247361 | 57.4 | 22374 |
| **2** Algeria | 2231.99333515006 | .69 | 2.306817 | 23.5 | 29321 |
| **3** Andorra | 21943.3398976022 | 10.17 |  |  |  |
| **4** Angola | 1381.00426770244 | 5.57 | 1.4613288 | 23.1 |  |

In [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='coe gapminder['co2emissions'] = pd.to\_numeric(gapminder['co2emissions'],errors='coerce')

In [4]:

gapminder\_clean=gapminder.dropna()

# Correlation - Scenario 1

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

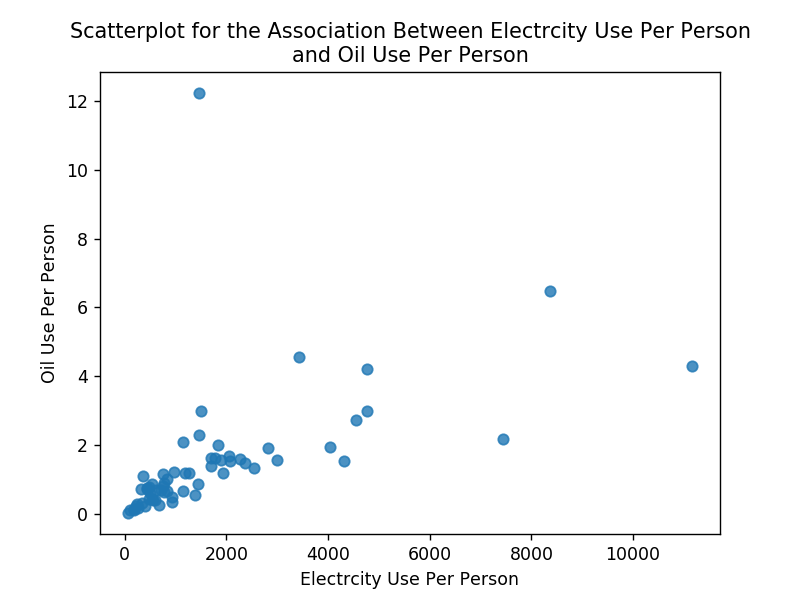
In [28]:

**%**matplotlib notebook

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



Out[28]:

Text(0.5,1,'Scatterplot for the Association Between Electrcity Use Per Perso n\nand Oil Use Per Person')

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

In [6]:

print ('association between relectricperperson and oilperperson')

print (scipy.stats.pearsonr(gapminder\_clean['relectricperperson'], gapminder\_clean['oilperp

association between relectricperperson and oilperperson (0.52493737791598849, 1.0020621767836594e-05)

# Correlation - Scenario 2

**Scatter plot to show association between co2emissions**

# (x) and oilperperson (y)

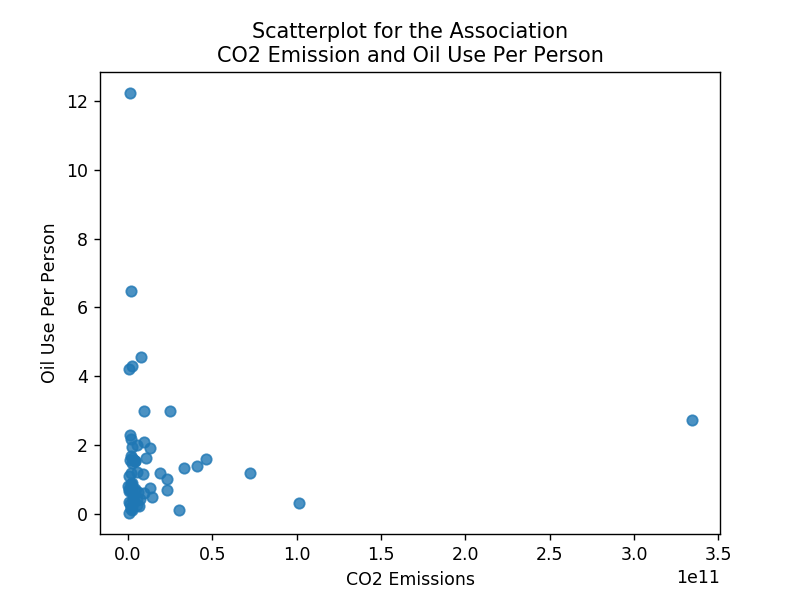
In [29]:

**%**matplotlib notebook 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')



Out[29]:

Text(0.5,1,'Scatterplot for the Association\nCO2 Emission and Oil Use Per Pe rson')

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

In [8]:

print ('association between co2emissions and oilperperson')

print (scipy.stats.pearsonr(gapminder\_clean['co2emissions'], gapminder\_clean['oilperperson'

association between co2emissions and oilperperson (0.044442012312287921, 0.72945188401230332)

# Regression - Scenario 3

**Scatter plot with regression to show relationship**

# between relectricperperson (x) and oilperperson (y)

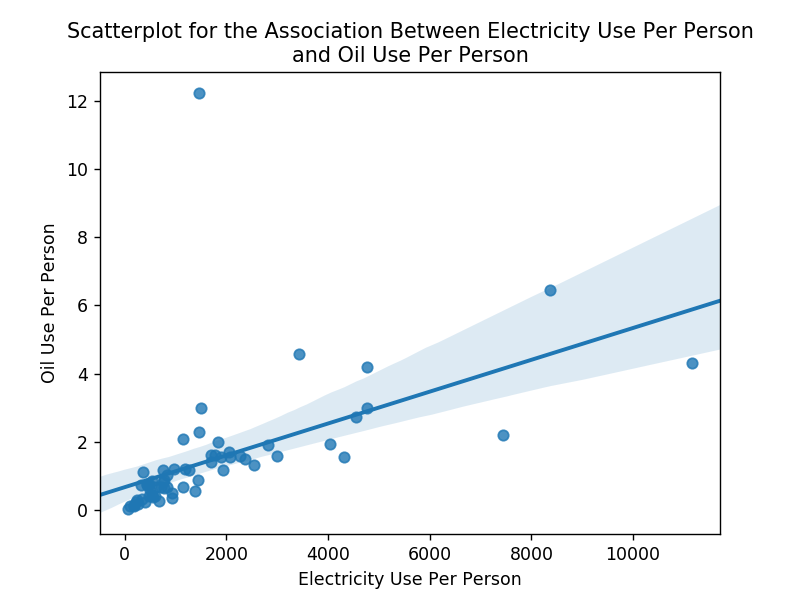
In [30]:

**%**matplotlib notebook

scat1 = sns.regplot(x="relectricperperson", y="oilperperson", fit\_reg=**True**, data=gapminder\_ plt.xlabel('Electricity Use Per Person')

plt.ylabel('Oil Use Per Person')

plt.title('Scatterplot for the Association Between Electricity Use Per Person' **+** '\n' **+** 'an



Out[30]:

Text(0.5,1,'Scatterplot for the Association Between Electricity Use Per Pers on\nand Oil Use Per Person')

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

In [10]:

**import** statsmodels.formula.api **as** smf

print ("OLS regression model for the association between Electric Use Per Person and Oil Pe 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: | 0.2 |
| 76 |  |  |  |
| Model: | OLS | Adj. R-squared: | 0.2 |
| 64 |  |  |  |
| Method: | Least Squares | F-statistic: | 23. |
| 20 |  |  |  |

Date: Fri, 27 Apr 2018 Prob (F-statistic): 1.00e- 05

Time: 15:02:25 Log-Likelihood: -116.

64

No. Observations: 63 AIC: 23

7.3

Df Residuals: 61 BIC: 24

1.6

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: | 1.6 |
| 27 |  |  |  |
| Prob(Omnibus): | 0.000 | Jarque-Bera (JB): | 3834.0 |
| 05 |  |  |  |
| Skew: | 5.613 | Prob(JB): | 0. |
| 00 |  |  |  |
| Kurtosis: | 39.531 | Cond. No. | 3.52e+ |
| 03 |  |  |  |

============================================================================

==

Warnings:

1. Standard Errors assume that the covariance matrix of the errors is corre ctly specified.
2. The condition number is large, 3.52e+03. This might indicate that there are

strong multicollinearity or other numerical problems.

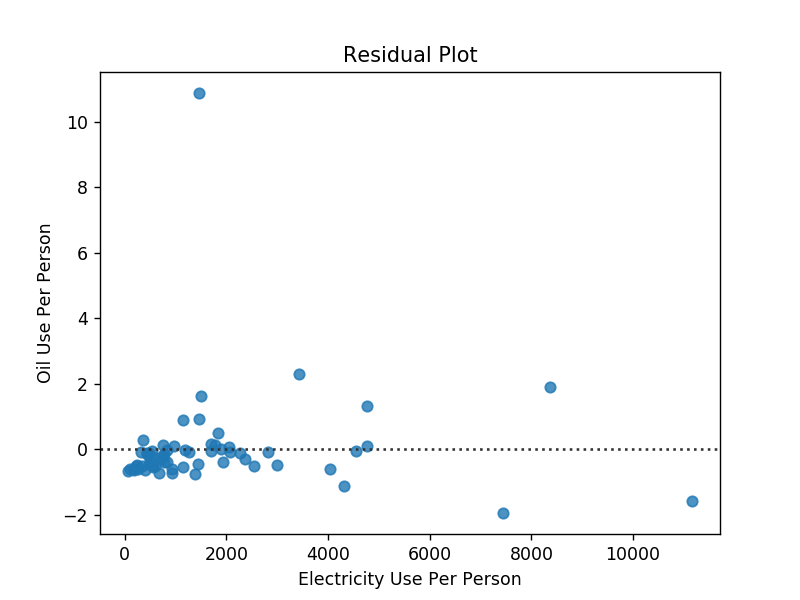
# Residual plot - regression analysis between relectricperperson (x) and oilperperson (y) - if required

In [31]:

**%**matplotlib notebook

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')



Out[31]:

Text(0.5,1,'Residual Plot')

# Regression - Scenario 4

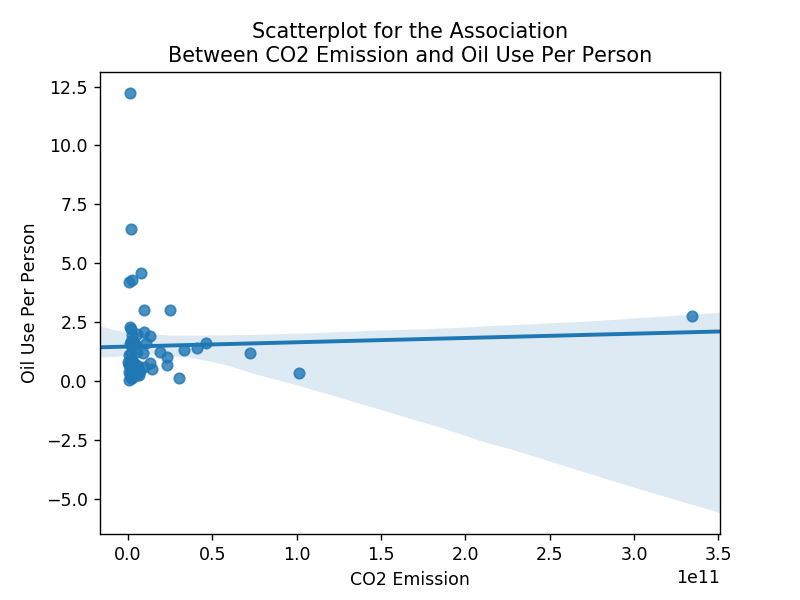
**Scatter plot with regression to show association between co2emissions (x) and oilperperson (y)**

plt.figure()

scat2 = sns.regplot(x="co2emissions", y="oilperperson", fit\_reg=**True**, data=gapminder) plt.xlabel('CO2 Emission')

plt.ylabel('Oil Use Per Person')

plt.title('Scatterplot for the Association' **+** '\n' **+** 'Between CO2 Emission and Oil Use Per



Out[32]:

Text(0.5,1,'Scatterplot for the Association\nBetween CO2 Emission and Oil Us e Per Person')

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

In [13]:

print ("OLS regression model for the association between CO2 emission and Oil Use Per Perso reg1 = smf.ols('oilperperson ~ co2emissions', data=gapminder\_clean).fit()

print (reg1.summary())

OLS regression model for the association between CO2 emission and Oil Use Pe r Person

OLS Regression Results

============================================================================

==

|  |  |  |  |
| --- | --- | --- | --- |
| Dep. Variable: | oilperperson | R-squared: | 0.0 |
| 02 |  |  |  |
| Model: | OLS | Adj. R-squared: | -0.0 |
| 14 |  |  |  |
| Method: | Least Squares | F-statistic: | 0.12 |
| 07 |  |  |  |

Date: Fri, 27 Apr 2018 Prob (F-statistic): 0.7

29

Time: 15:02:25 Log-Likelihood: -126.

73

No. Observations: 63 AIC: 25

7.5

Df Residuals: 61 BIC: 26

1.7

Df Model: 1

Covariance Type: nonrobust

============================================================================

====

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 975] | coef | std err | t | P>|t| | [0.025 | 0. |
|  |  |  |  |  |  |  |
| Intercept 1.946  co2emissions | 1.4561  1.829e-12 | 0.245  5.26e-12 | 5.939  0.347 | 0.000  0.729 | 0.966  -8.7e-12 | 1.24 |
| e-11 |  |  |  |  |  |  |

============================================================================

==

|  |  |  |  |
| --- | --- | --- | --- |
| Omnibus: | 82.847 | Durbin-Watson: | 1.7 |
| 27 |  |  |  |
| Prob(Omnibus): | 0.000 | Jarque-Bera (JB): | 1029.8 |
| 53 |  |  |  |
| Skew: | 3.814 | Prob(JB): | 2.35e-2 |
| 24 |  |  |  |
| Kurtosis: | 21.279 | Cond. No. | 4.93e+ |
| 10 |  |  |  |

============================================================================

==

Warnings:

1. Standard Errors assume that the covariance matrix of the errors is corre ctly specified.
2. The condition number is large, 4.93e+10. This might indicate that there are

strong multicollinearity or other numerical problems.

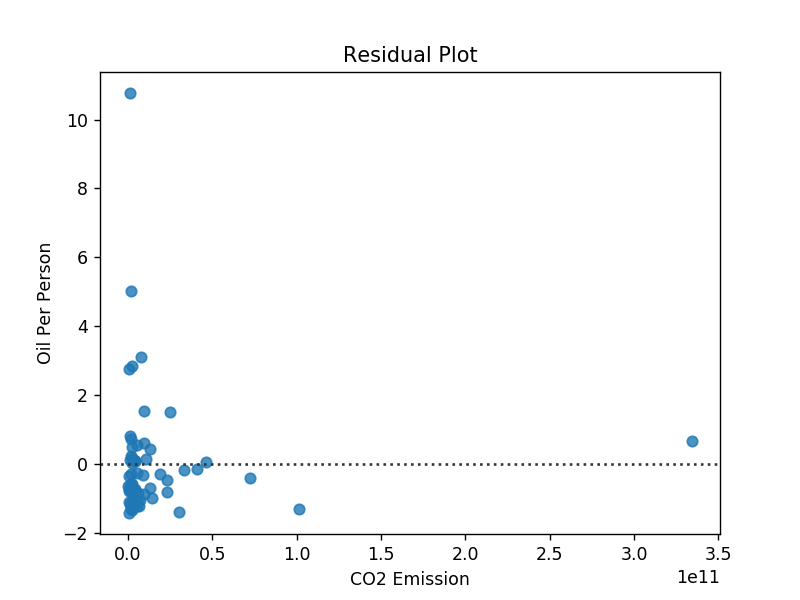
# Residual plot - regression analysis between co2emissions (x) and oilperperson (y) - if required

In [33]:

**%**matplotlib notebook

scat1 = sns.residplot(x="co2emissions", y="oilperperson", data=gapminder\_clean)

plt.xlabel('CO2 Emission') plt.ylabel('Oil Per Person') plt.title('Residual Plot')



Out[33]:

Text(0.5,1,'Residual Plot')

# Regression with 3 variables

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

# Low co2emission (1): min - 1846084167

**Medium co2emission (2): 1846084168 - 7993752800**

# High co2emission (3): 7993752801 - max

In [15]:

**def** co2emissionsgrp (row):

**if** row['co2emissions'] **<**= 1846084167: **return** 1

**elif** row['co2emissions'] **<**= 7993752800: **return** 2

**elif** row['co2emissions'] **>** 7993752800: **return** 3

In [16]:

gapminder\_clean['co2emissionsgrp'] = gapminder\_clean.apply (**lambda** row: co2emissionsgrp (ro

C:\Users\jc443343\AppData\Local\Continuum\anaconda3\lib\site-packages\ipyker nel\_launcher.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: http://pandas.pydata.org/pandas-docs/s table/indexing.html#indexing-view-versus-copy (http://pandas.pydata.org/pand as-docs/stable/indexing.html#indexing-view-versus-copy)](http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy)

"""Entry point for launching an IPython kernel.

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

In [17]:

chk1 = gapminder\_clean['co2emissionsgrp'].value\_counts(sort=**False**, dropna=**False**) print(chk1)

1 17

2 27

3 19

Name: co2emissionsgrp, dtype: int64

# Divide gapminder\_clean into 3 dataframes, each dataframe representing rows of data in low, medium and high CO2 Emission

In [18]:

sub1=gapminder\_clean[(gapminder\_clean['co2emissionsgrp']== 1)]

sub2=gapminder\_clean[(gapminder\_clean['co2emissionsgrp']== 2)]

sub3=gapminder\_clean[(gapminder\_clean['co2emissionsgrp']== 3)]

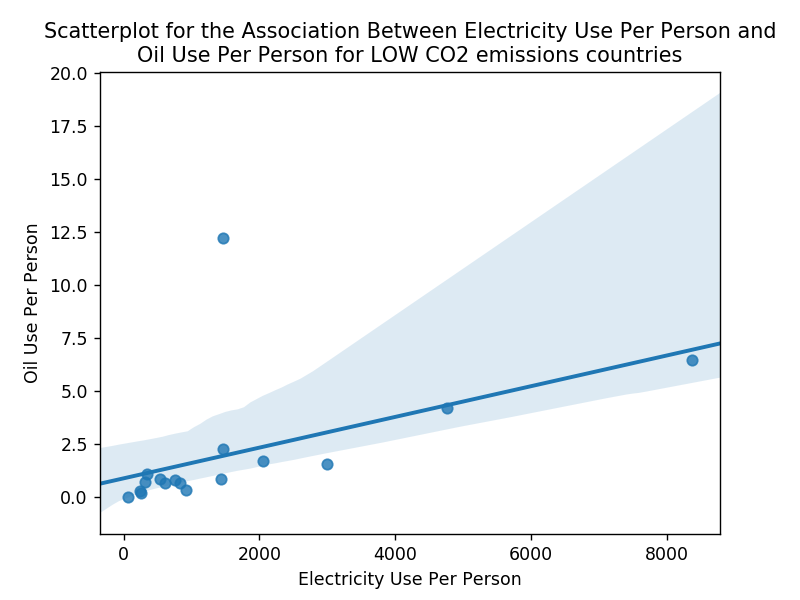
# Regression - Scenario 5

**Scatter plot with regression analysis to show association between electricity use per person (x) and**

# oilperperson (y) for low CO2 emission countries

In [34]:

|  |  |  |
| --- | --- | --- |
| **%**matplotlib notebook  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' print (scat1) | | |
|  |  |  |



AxesSubplot(0.125,0.11;0.775x0.77)

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

In [20]:

|  |  |  |
| --- | --- | --- |
| print ('OLS regression model for the association between Electricty Use Per Person and Oil reg1 = smf.ols('oilperperson ~ relectricperperson', data=sub1).fit()  print (reg1.summary()) | | |
|  |  |  |

OLS regression model for the association between Electricty Use Per Person a nd Oil Use Per Person for

LOW CO2 Emission countries

OLS Regression Results

============================================================================

==

|  |  |  |  |
| --- | --- | --- | --- |
| Dep. Variable: | oilperperson | R-squared: | 0.2 |
| 44 |  |  |  |
| Model: | OLS | Adj. R-squared: | 0.1 |
| 94 |  |  |  |
| Method: | Least Squares | F-statistic: | 4.8 |
| 40 |  |  |  |
| Date: | Fri, 27 Apr 2018 | Prob (F-statistic): | 0.04 |
| 39 |  |  |  |
| Time: | 15:02:26 | Log-Likelihood: | -40.3 |
| 87 |  |  |  |
| No. Observations: | 17 | AIC: | 84. |
| 77 |  |  |  |
| Df Residuals: | 15 | BIC: | 86. |
| 44 |  |  |  |
| Df Model:  Covariance Type: | 1  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: | 2.0 |
| 57 |  |  |  |
| Prob(Omnibus): | 0.000 | Jarque-Bera (JB): | 126.4 |
| 42 |  |  |  |
| Skew: | 3.582 | Prob(JB): | 3.50e- |
| 28 |  |  |  |
| Kurtosis: | 14.278 | Cond. No. | 3.32e+ |
| 03 |  |  |  |

============================================================================

==

Warnings:

1. Standard Errors assume that the covariance matrix of the errors is corre ctly specified.
2. The condition number is large, 3.32e+03. This might indicate that there are

strong multicollinearity or other numerical problems.

C:\Users\jc443343\AppData\Local\Continuum\anaconda3\lib\site-packages\scipy

\stats\stats.py:1334: UserWarning: kurtosistest only valid for n>=20 ... con tinuing anyway, n=17

"anyway, n=%i" % int(n))

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

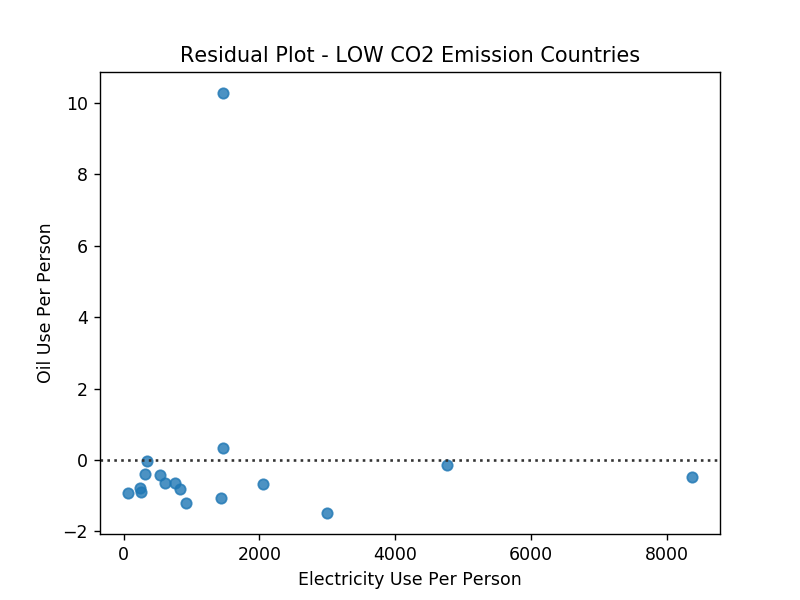
In [35]:

**%**matplotlib notebook

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')



Out[35]:

Text(0.5,1,'Residual Plot - LOW CO2 Emission Countries')

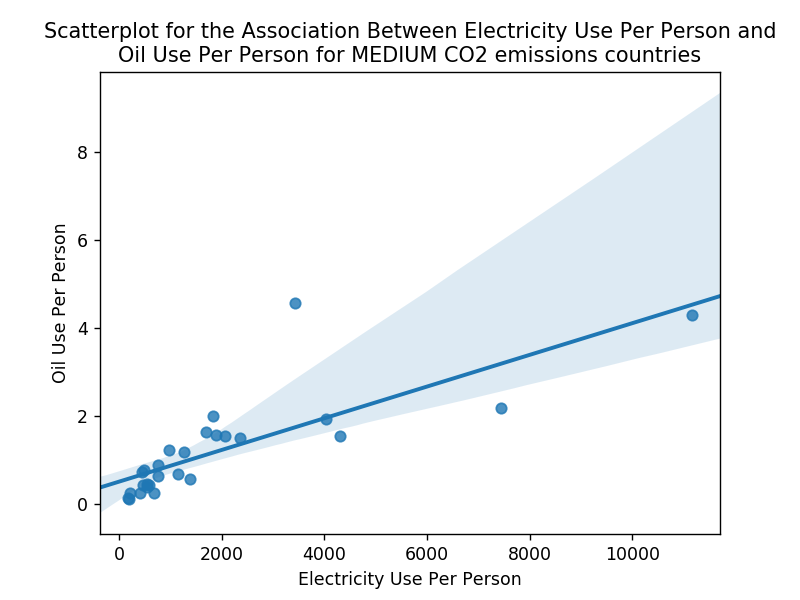
# Regression - Scenario 6

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

# countries

In [36]:

|  |  |  |
| --- | --- | --- |
| **%**matplotlib notebook  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' print (scat1) | | |
|  |  |  |



AxesSubplot(0.125,0.11;0.775x0.77)

In [23]:

|  |  |  |
| --- | --- | --- |
| print ('OLS regression model for the association between Electricty Use Per Person and Oil reg1 = smf.ols('oilperperson ~ relectricperperson', data=sub2).fit()  print (reg1.summary()) | | |
|  |  |  |

OLS regression model for the association between Electricty Use Per Person a nd Oil Use Per Person for

MEDIUM CO2 Emission countries

OLS Regression Results

============================================================================

==

|  |  |  |  |
| --- | --- | --- | --- |
| Dep. Variable: | oilperperson | R-squared: | 0.6 |
| 26 |  |  |  |
| Model: | OLS | Adj. R-squared: | 0.6 |
| 11 |  |  |  |
| Method: | Least Squares | F-statistic: | 41. |
| 89 |  |  |  |
| Date: | Fri, 27 Apr 2018 | Prob (F-statistic): | 8.88e- |
| 07 |  |  |  |
| Time: | 15:02:26 | Log-Likelihood: | -27.6 |
| 31 |  |  |  |
| No. Observations: | 27 | AIC: | 59. |
| 26 |  |  |  |
| Df Residuals: | 25 | BIC: | 61. |
| 85 |  |  |  |
| Df Model:  Covariance Type: | 1  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: | 2.2 |
| 73 |  |  |  |
| Prob(Omnibus): | 0.000 | Jarque-Bera (JB): | 120.1 |
| 41 |  |  |  |
| Skew: | 2.643 | Prob(JB): | 8.16e- |
| 27 |  |  |  |
| Kurtosis: | 11.880 | Cond. No. | 3.91e+ |
| 03 |  |  |  |

============================================================================

==

Warnings:

1. Standard Errors assume that the covariance matrix of the errors is corre ctly specified.
2. The condition number is large, 3.91e+03. This might indicate that there are

strong multicollinearity or other numerical problems.

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

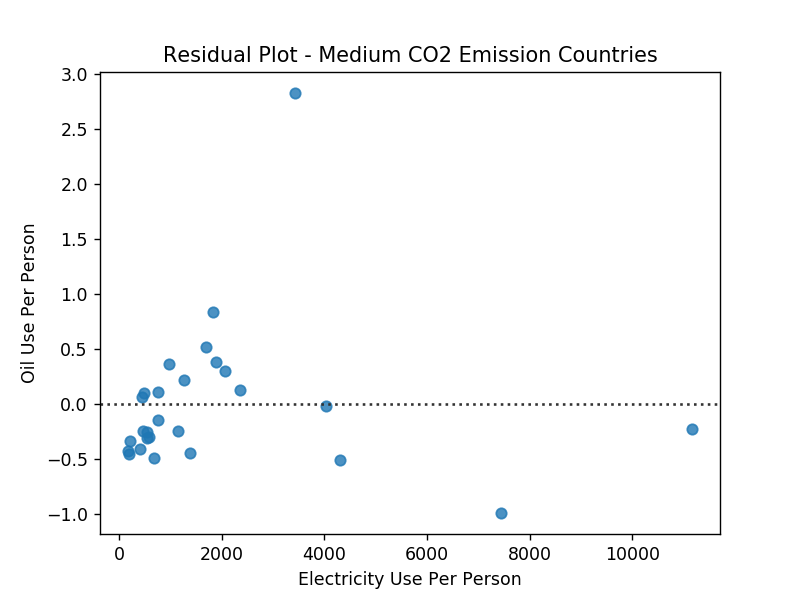
In [37]:

**%**matplotlib notebook

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')



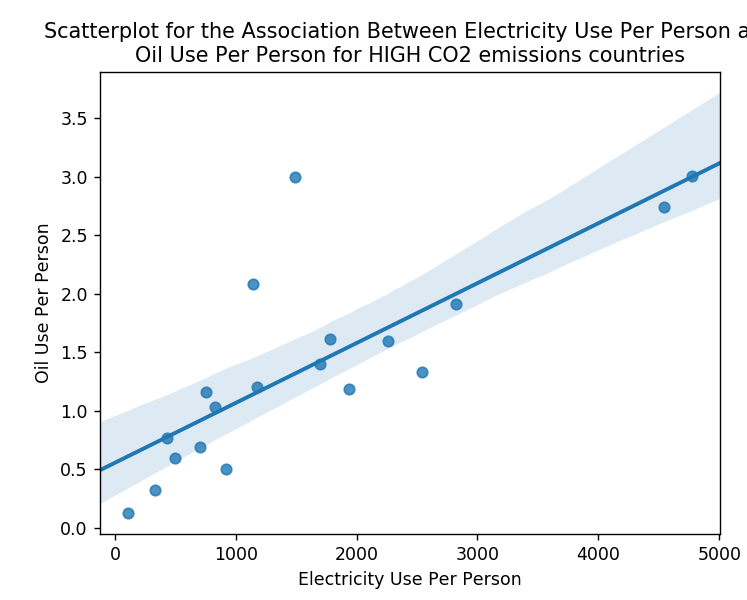
Out[37]:

Text(0.5,1,'Residual Plot - Medium CO2 Emission Countries')

# Regression - Scenario 7

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

|  |  |  |
| --- | --- | --- |
| **%**matplotlib notebook  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' print (scat1) | | |
|  |  |  |



**Figure 1**

AxesSubplot(0.125,0.11;0.775x0.77)

|  |  |  |
| --- | --- | --- |
| print ('OLS regression model for the association between Electricty Use Per Person and Oil  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: | 2 |
| 7.61 |  |  |  |

Date: Fri, 27 Apr 2018 Prob (F-statistic): 6.45

e-05

Time: 15:02:26 Log-Likelihood: -1

4.302

No. Observations: 19 AIC: 3

2.60

Df Residuals: 17 BIC: 3

4.49

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): | 2 |
| 3.814 |  |  |  |
| Skew: | 1.966 | Prob(JB): | 6.74 |
| e-06 |  |  |  |
| Kurtosis: | 6.823 | Cond. No. | 3.32 |
| e+03 |  |  |  |

==========================================================================

====

Warnings:

1. Standard Errors assume that the covariance matrix of the errors is cor rectly specified.
2. The condition number is large, 3.32e+03. This might indicate that ther

strong multicollinearity or other numerical problems.

C:\Users\jc443343\AppData\Local\Continuum\anaconda3\lib\site-packages\scipy

\stats\stats.py:1334: UserWarning: kurtosistest only valid for n>=20 ... con tinuing anyway, n=19

"anyway, n=%i" % int(n))

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

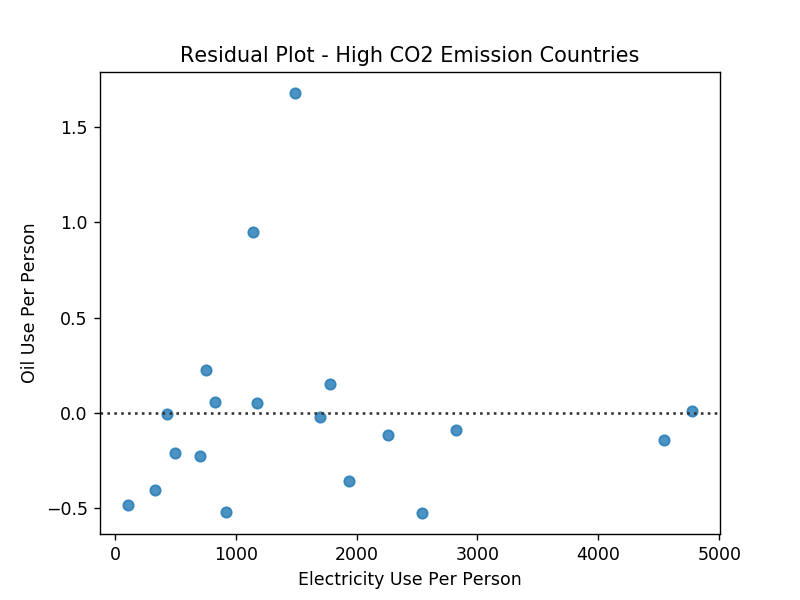
In [27]:

**%**matplotlib notebook

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')



Out[27]:

Text(0.5,1,'Residual Plot - High CO2 Emission Countries')