qm_coursewrk1_DuccioAiazzi_v2

November 1, 2015

```
In [1]: import IPython.core.display as di

        # This line will hide code by default when the notebook is exported as HTML
        di.display_html('<script>jQuery(function() {if (jQuery("body.notebook_app").length == 0) { jQue

        # This line will add a button to toggle visibility of code blocks, for use with the HTML export
        di.display_html('''<button onclick="jQuery('.input_area').toggle(); jQuery('.prompt').toggle();
```

```
In [2]: import matplotlib.pyplot as plt
        import matplotlib
        matplotlib.style.use('ggplot')


        import patsy
        import pandas as pd
        #import statsmodels as sm
        #import statsmodels.api as sm
        import numpy as np
        import statsmodels.api as sm
        from statsmodels.formula.api import ols
        import csv
        import os

        pd.set_option('display.mpl_style', 'default') # Make the graphs a bit prettier
        figsize(10, 5)
```

```
In [6]: os.chdir('/Users/duccioa/CLOUD/C07_UCL_SmartCities/QuantitativeMethods/qm_coursewrk1')
```

1 Introduction

Based on the dataset Data_for_Coursework_1_Countries.csv and within the context of the course Quantitative Methods at UCL

2 Analysis

```
In [7]: countries = pd.read_csv('countries.csv', sep = ',', encoding = 'latin1')
        countries.columns = ['X', 'Year', 'CountryCode', 'CountryName',
                             'Population', 'GDP', 'FoodImports', 'FuelImports']
```

In this project I aim at investigating the determinant of food import based on the available dataset, which contains data for 190 countries about Gross Domestic Product (GDP), food import and population for the year 2005.

```
In [8]: countries[:3]

Out[8]:    X  Year CountryCode  CountryName  Population           GDP  FoodImports  \
        0  1  2005         ABW        Aruba      100031  1.160240e+12     97166150
        1  2  2005         AFG  Afghanistan    24860855  6.275076e+09    528341972
        2  3  2005         AGO       Angola    16544376  2.823370e+10   1075607744

           FuelImports
        0     32335285
        1    461521897
        2     48218538
```

```
In [9]: countries['Population_log'] = log(countries['Population'])
        countries['FoodImports_log'] = log(countries['FoodImports'])
        countries['GDP_log'] = log(countries['GDP'])
        countries['GDP_pc'] = countries['GDP']/countries['Population']
        countries['FoodImports_pc'] = countries['FoodImports']/countries['Population']
        countries['GDP_pc_log'] = log(countries['GDP_pc'])
        countries['FoodImports_pc_log'] = log(countries['FoodImports_pc'])
```

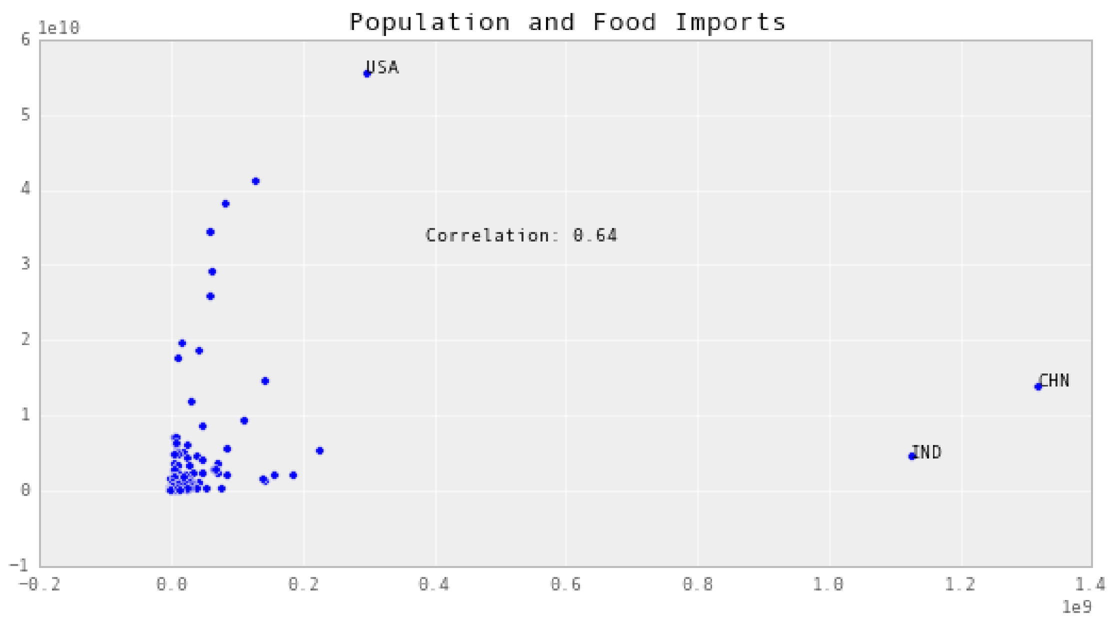
Both plots present patterns of correlation with three big outliers: USA, India and China which have respectivly very high GDP, population, GDP and population. GDP have a very strong correlation (0.92) with food import and the ln-ln plot strongly suggests a power law relation with values very concentrated on the regression line (the ln-ln plot with population shows higher variation than with GDP).

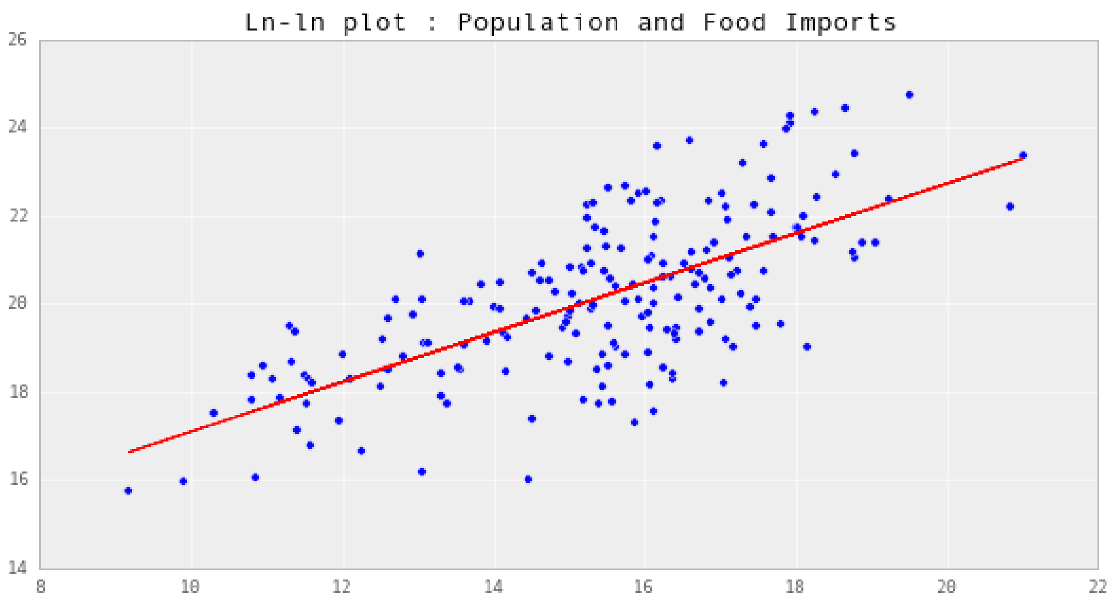
```
In [10]: coords = countries[countries['CountryCode'] == 'CHN']
         coords = coords.append(countries[countries['CountryCode'] == 'IND'])
         coords = coords.append(countries[countries['CountryCode'] == 'USA'])
```

```
In [11]: #POPULATION AND FOOD IMPORT : analysis and plot
         fig, ax = plt.subplots()
         ax.scatter(countries['Population'], countries['FoodImports'])
         ax.annotate(coords['CountryCode'][179],
                     xy = (coords['Population'][179], coords['FoodImports'][179]))
         ax.annotate(coords['CountryCode'][34],
                     xy = (coords['Population'][34], coords['FoodImports'][34]))
         ax.annotate(coords['CountryCode'][82],
                     xy = (coords['Population'][82], coords['FoodImports'][82]))
         ax.annotate('Correlation: ' +
                     str(round(countries['Population'].corr(countries['FoodImports'], method='spearman'
                     xy = (coords['Population'][179]*1.3, coords['FoodImports'][179]*0.6))
         plt.title('Population and Food Imports')
         #Log plot
         fig, ax = plt.subplots()
         ax.scatter(countries['Population_log'], countries['FoodImports_log'])
         plt.title('Ln-ln plot : Population and Food Imports')

         #Regression line
         mod_pop = ols(formula='FoodImports_log ~ Population_log', data=countries)
         res_pop = mod_pop.fit()
         par_pop = res_pop.params
         ax.plot(countries['Population_log'], par_pop[0] +
                 par_pop[1]*countries['Population_log'], color = 'red')

Out[11]: [<matplotlib.lines.Line2D at 0x11108f850>]
```

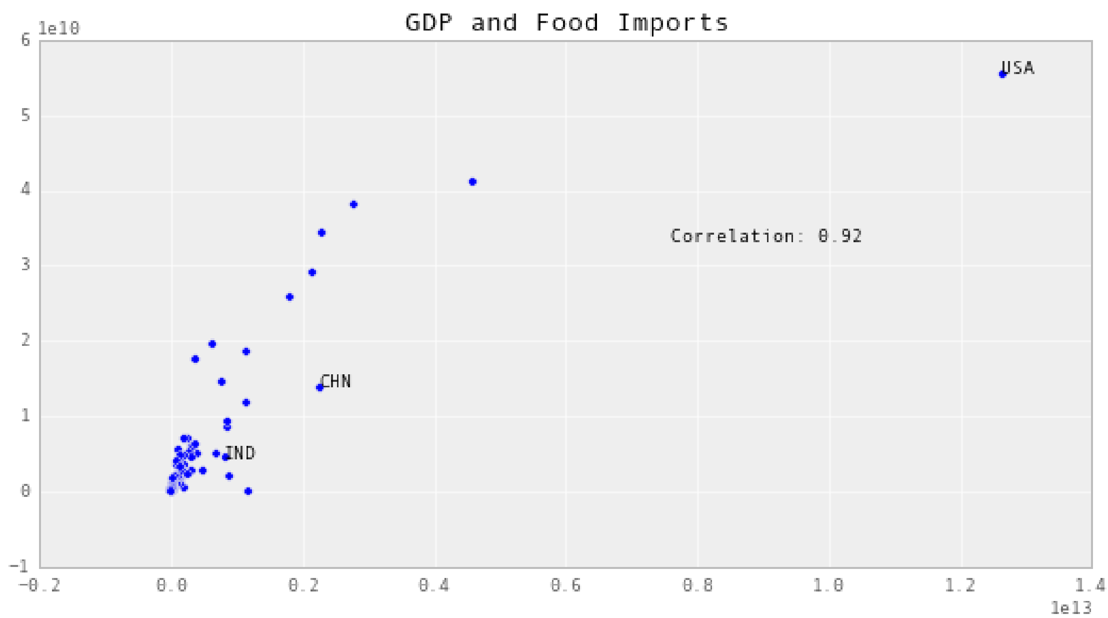


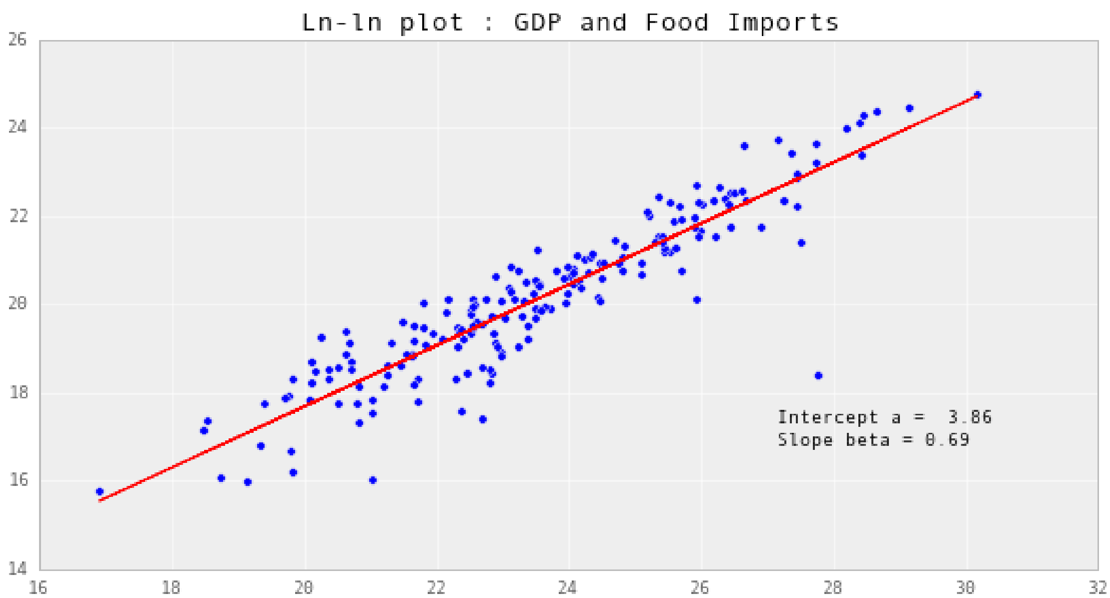


In [76]:
```
##GDP AND FOOD IMPORT : analysis and plot
#Plot
fig, ax = plt.subplots()
ax.scatter(countries['GDP'], countries['FoodImports'])
ax.annotate(coords['CountryCode'][179],
            xy = (coords['GDP'][179], coords['FoodImports'][179]))
ax.annotate(coords['CountryCode'][34],
            xy = (coords['GDP'][34], coords['FoodImports'][34]))
```

```
ax.annotate(coords['CountryCode'][82],
            xy = (coords['GDP'][82], coords['FoodImports'][82]))
ax.annotate('Correlation: ' +
            str(round(countries['GDP'].corr(countries['FoodImports'],
                                            method='spearman'), 2)),
            xy = (coords['GDP'][179]*0.6, coords['FoodImports'][179]*0.6))
plt.title('GDP and Food Imports')
#Log plot
fig, ax = plt.subplots()
ax.scatter(countries['GDP_log'], countries['FoodImports_log'])
plt.title('Ln-ln plot : GDP and Food Imports')
#Regression line
mod_gdp = ols(formula='FoodImports_log ~ GDP_log', data=countries)
res_gdp = mod_gdp.fit()
par_gdp = res_gdp.params
ax.plot(countries['GDP_log'], par_gdp[0] +
        par_gdp[1]*countries['GDP_log'], color = 'red')
ax.annotate('Intercept a =  ' + str(round(par_gdp[0], 2)),
            xy = (coords['GDP_log'][179]*0.9, coords['FoodImports_log'][179]*0.7))
ax.annotate('Slope beta = '  + str(round(par_gdp[1], 2)),
            xy = (coords['GDP_log'][179]*0.9, coords['FoodImports_log'][179]*0.68))
```

```
Out[76]: <matplotlib.text.Annotation at 0x117c8a210>
```





I can write the relation between the outcome Foor Imports (Y) and GDP (X) as following:

$$\left[Y = aX^{\beta}\right] \tag{1}$$

With a = 3.86 and $\alpha = 0.69$ we have an R-squared = 0.84, which is a good fit. I want now to verify whether the equation above holds and population does not influences the Food Imports. If I divide by the population L we can write:

$$\frac{Y}{L} = \frac{aX^{\beta}}{L} = \frac{aX^{\beta}}{L^{\beta}L^{1-\beta}} = a(\frac{X}{L})^{\beta}L^{1-\beta}$$

If I write $y = \frac{Y}{L}$ as the Food Imports per capita and $x = \frac{X}{L}$ as the GDP per capita, I can say that the first equation is equivalent to saying:

$$\left[c = ax^{\beta}L^{1-\beta}\right] \tag{2}$$

This poses a constrain on β which can be verified if we build a linear model of the log of c, y and L as following:

$$ln(c) = \alpha_1 + \alpha_2 ln(y) + \alpha_3 ln(L)$$

If the first equation holds, then $\alpha_2 = 1 - \alpha_3$, which is not the case as shown by running the following multivariate regression model:

```
In [78]: mod = ols(formula='FoodImports_pc_log ~ GDP_pc_log + Population_log', data=countries)
         res = mod.fit()
         print res.params
```

```
Intercept          3.860655
GDP_pc_log         0.674037
Population_log    -0.299762
dtype: float64
```

```
In [86]: mod = ols(formula='FoodImports_pc_log ~ GDP_pc_log + Population_log', data=countries)
         res = mod.fit()
         print res.summary()
```

```
OLS Regression Results
==============================================================================
Dep. Variable:     FoodImports_pc_log   R-squared:                       0.807
Model:                            OLS   Adj. R-squared:                  0.805
Method:                 Least Squares   F-statistic:                     390.9
Date:                Sun, 01 Nov 2015   Prob (F-statistic):           1.60e-67
Time:                        15:54:58   Log-Likelihood:                -207.91
No. Observations:                 190   AIC:                             421.8
Df Residuals:                     187   BIC:                             431.6
Df Model:                           2
Covariance Type:            nonrobust
==================================================================================
                     coef    std err          t      P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------------------------
Intercept          3.8607      0.518      7.456      0.000         2.839     4.882
GDP_pc_log         0.6740      0.032     21.300      0.000         0.612     0.736
Population_log    -0.2998      0.025    -12.010      0.000        -0.349    -0.251
==============================================================================
Omnibus:                       92.805   Durbin-Watson:                   1.841
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              534.208
Skew:                          -1.778   Prob(JB):                    9.96e-117
Kurtosis:                      10.405   Cond. No.                         172.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

```
In [ ]:
```