## Day 2 - Practice Session 1: Hands-on exercises with the EGI Jupyter notebooks

Requirements: For this practise session, we will use the following Python modules:

1. urllib and xlrd to load remote datasets
2. matplotlib to plot datasets

The urllib module in Python allows you access websites via your program. In Python3 this module is slightly different than urllib2 in Python 2, but they are mostly the same. With this module, you can access websites, download data, parse data, modify your headers, and do any GET and POST requests you might need to do.

Here is the first and easiest example of using urllib. First of all we just need to import urllib.requests. From there, we assign the opening of the url to a variable, where we can finally use a .read() command to read the data. The result is a massive mess, but we did indeed read the source code.

|  |
| --- |
| #Used to make requests import urllib.request  filedata = urllib.request.urlopen(‘YOUR\_URL\_HERE’) datatowrite = filedata.read() |

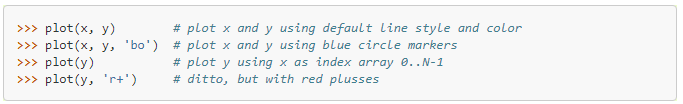
Once you have read your data, define a local file name in your local notebook and store your datasets on it using the .write() method.

|  |
| --- |
| filename = ‘myfile.xls’ with open (filename, ‘wb’) as f:  f.write(datatowrite) |

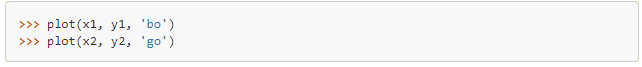
For the Climate Change Knowledge portal, datasets are store in spread-sheet (.xls). To read datasets from spread-sheet we use the xlrd library.

|  |
| --- |
| #Used to read datasets (from spread-sheet) import xlrd  book = xlrd.open\_workflow(‘file containing datasets’) sheet = book.sheets()[0] for item in range(sheet.nrows):  # Do some processing … |

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, [IPython](http://ipython.org/) shells and the [Jupyter](http://jupyter.org/) notebook, web application servers, and for graphical user interface toolkits. Some basic examples are following:

**Plotting y versus x**The coordinates of the points or lines are given by *x,y*.

**Plotting labelled data**There’s a convenient way for plotting objects with labelled data. Instead of giving the data in x and y, you can provide the object in the data parameter and just give the labels for x and y.

**Plotting multiple sets of data**The most straight forward way is just to call [plot](https://matplotlib.org/api/_as_gen/matplotlib.pyplot.plot.html#matplotlib.pyplot.plot) multiple times. Example:

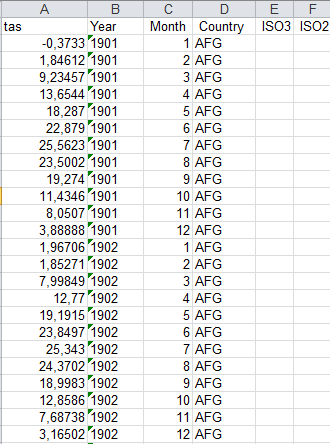
Alternatively, if your data is already a 2d array, you can pass it directly to *x*, *y*. A separate data set will be drawn for every column.

Example: an array a where the first column represents the *x* values and the other columns are the *y* columns.

The third way is to specify multiple sets of *[x], y, [ftm]* groups:

**Exercise 1**:

1. Collect datasets of the region of interest from the [Climate Change Knowledge portal](http://sdwebx.worldbank.org/climateportal/)
   1. Click on the area of the map of interest to get start
   2. Click on the region to view regional data
   3. Click to download historical data
2. Select “Temperature”, “Country” and “Time period” and download datasets to Excel
   1. Typical datasets downloaded from the CCKP:



1. Calculate the *Average Monthly Temperature* using the Jupyter Python kernel.
   1. Plot the results with the matplotlib module.

**Exercise 2**:

1. Repeat the exercise selecting the Average Monthly Rainfall as datasets.
2. Plot the Average Monthly Temperature and Rainfall in the same plot.