begin

September 19, 2016

1 Get started with the GDS stack (Windows)

This document shows how you can get started with the tools used in the GDS'16 course. In particular, it walks you through each step of three of the most common tasks you will have to do in order to follow on each computer practical:

1. Start the Jupyter Notebook
2. Download notebook files with the practicals
3. Download data files and access them within the notebook

We assume, you are either on a university computer that has all the required software installed (`Install University Applications -> Scientific -> Geographic Data Science Stack 2016`) or, if you are on a personal computer, you have installed all the software following the instructions (if not, go here).

1.1 Fire up the `jupyter` notebook

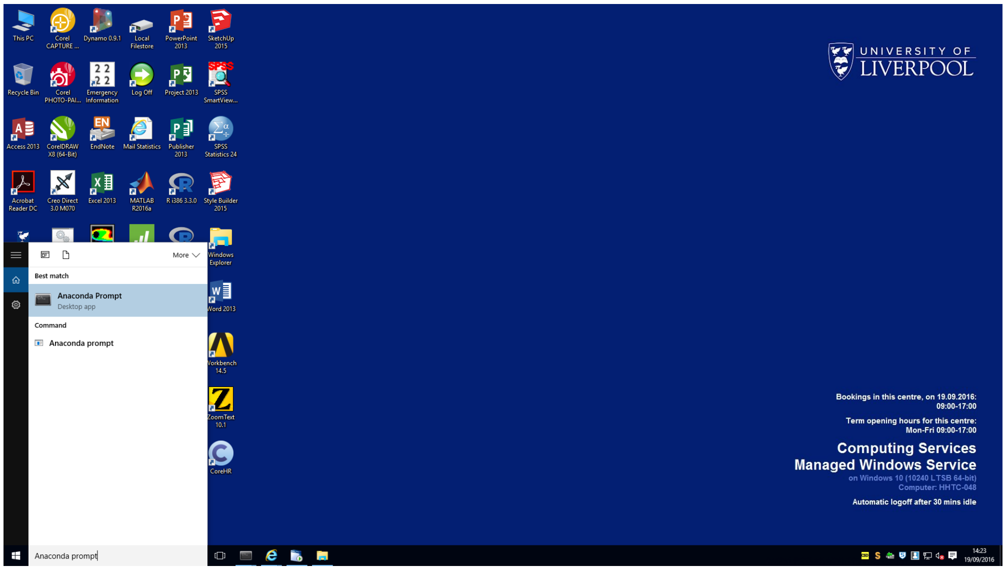
All of the course is based on the **Jupyter notebook**, a computational environment that provides an interface to the Python programming language by allowing to capture in a single document (an `.ipynb` file, or notebook from now on) Python code, its output, and additional text.

Follow these steps to start the notebook app:

- Go to the `Search Windows` box on the bottom-left corner of the screen and type `Anaconda Prompt` (on Windows 7, this can be accessed through the `Start` menu and "`Search Programs and files`"):
- Hit enter and that should open up a window that looks like this:
- Now you need to navigate through the computer file system (just as you would do with the Explorer window, but through commands). The first step is go to the `M:` drive so everything you make is safely saved. Type the following on the prompt:

```
M:
```

which should change the `C:\Windows\system32>` preamble into `M:\`. This means you have changed to the `M` directory.

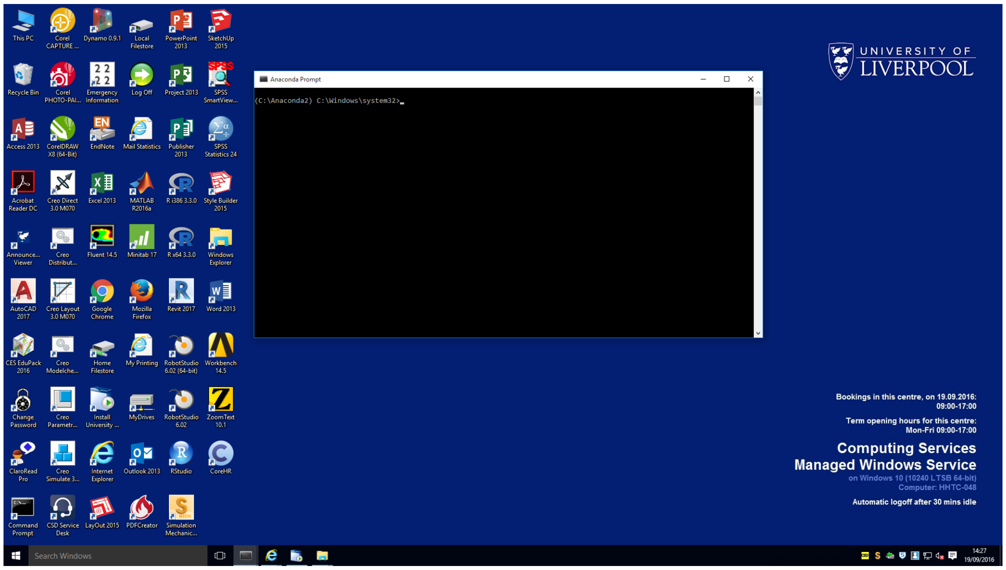


Bookings in this centre, on 19.09.2016:
09:00-17:00

Term opening hours for this centre:
Mon-Fri 09:00-17:00

Computing Services
Managed Windows Service
on Windows 10 (10240 LTSB 64-bit)
Computer: HHTC-048

Automatic logoff after 30 mins idle



```
(C:\Anaconda2) C:\Windows\system32>_
```

Bookings in this centre, on 19.09.2016:
09:00-17:00

Term opening hours for this centre:
Mon-Fri 09:00-17:00

Computing Services
Managed Windows Service
on Windows 10 (10240 LTSB 64-bit)
Computer: HHTC-048

Automatic logoff after 30 mins idle

- Next step is activating the `gds` environment. For that, type the following command:

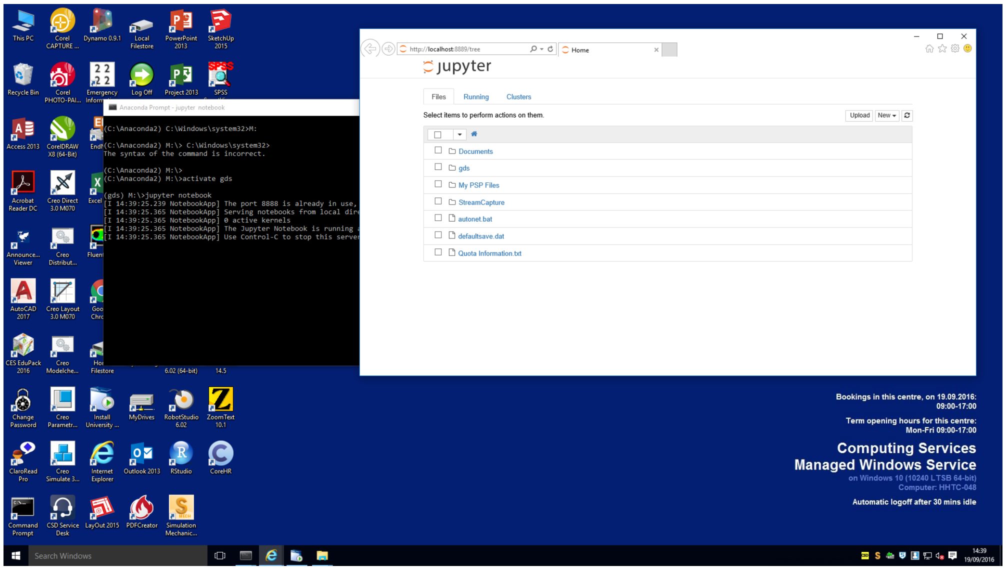
```
activate gds
```

This should set at the beginning of each prompt line the text `(gds)`.

- At this point, you are ready to fire up the notebook! Type:

```
jupyter notebook
```

And the browser should start and open a page that looks like this:



```
(C:\Anaconda2) C:\Windows\system32>M:

(C:\Anaconda2) M:\> C:\Windows\system32>
The syntax of the command is incorrect.

(C:\Anaconda2) M:\>
(C:\Anaconda2) M:\>activate gds

(gds) M:\>jupyter notebook
[I 14:39:25.239 NotebookApp] The port 8888 is already in use,
[I 14:39:25.365 NotebookApp] Serving notebooks from local dire
[I 14:39:25.365 NotebookApp] 0 active kernels
[I 14:39:25.365 NotebookApp] The Jupyter Notebook is running a
[I 14:39:25.365 NotebookApp] Use Control-C to stop this server
```

Select items to perform actions on them.

- Documents
- gds
- My PSP Files
- StreamCapture
- autonet.bat
- defaultsave.dat
- Quota Information.txt

Bookings in this centre, on 19.09.2016:
09:00-17:00

Term opening hours for this centre:
Mon-Fri 09:00-17:00

Computing Services
Managed Windows Service
on Windows 10 (10240 LTSB 64-bit)
Computer: HHTC-048

Automatic logoff after 30 mins idle

Congrats, you are good to go!!!

1.2 Download course files

Almost every file you will need for this course can be accessed through the course website:

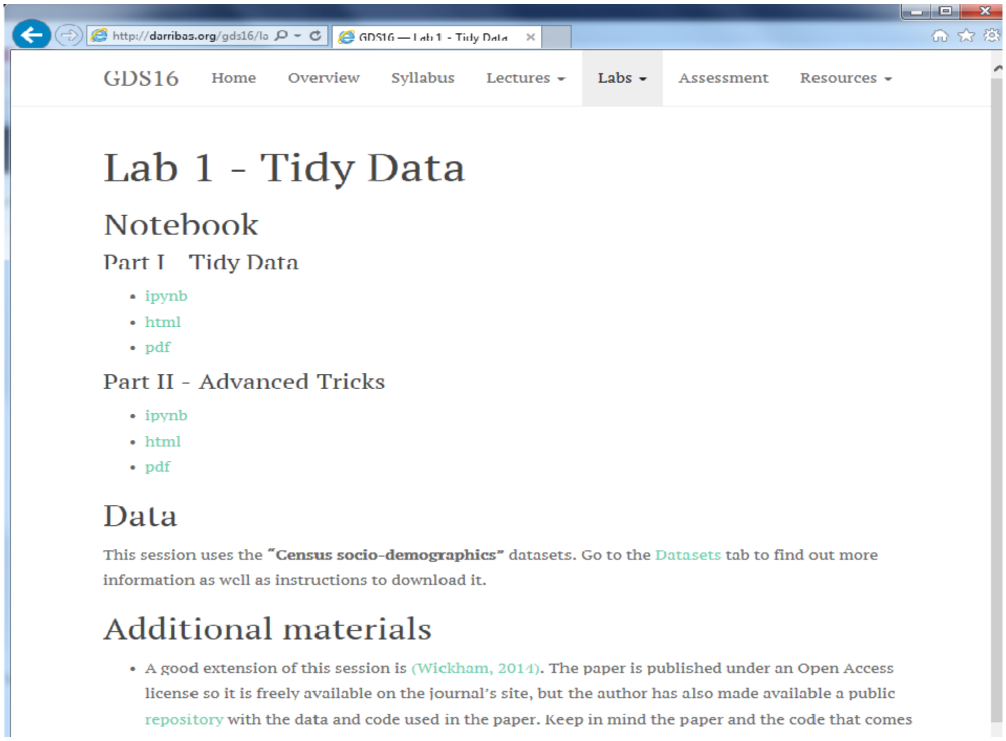
darribas.org/gds16

To demonstrate how to effectively access the files needed for the computer practicals, we will use the example of the first practical. Here are the steps to follow:

- Head to the practical's page:

- In this case, we are interested in the first part. The file you want to download is the `ipynb`, which contains the notebook. Go ahead and right-click with the mouse on top of the link.

And save it somewhere **WITHIN the `M` DRIVE**. This is important for two reasons: a) you have started your Jupyter session there and b) it is the only way to ensure the file stays safely backed up and protected.



Lab 1 - Tidy Data

Notebook

Part I Tidy Data

- ipynb
- html
- pdf

Part II - Advanced Tricks

- ipynb
- html
- pdf

Data

This session uses the "**Census socio-demographics**" datasets. Go to the Datasets tab to find out more information as well as instructions to download it.

Additional materials

- A good extension of this session is (Wickham, 2014). The paper is published under an Open Access license so it is freely available on the journal's site, but the author has also made available a public repository with the data and code used in the paper. Keep in mind the paper and the code that comes

- At this point, you can go back to you Jupyter Notebook session:

And navigate within it to the folder where you have placed the notebook file.

- Click on the file and that should open a new page that looks like this:

At this point, you are all set to hack some Python code!

1.3 Read files into the notebook

Finally, once you have the notebook for a practical ready, you will want to download the dataset of interest and be able to access it within the notebook. The download part is just as any other linked file from the internet; the accessing it through the notebook is a bit trickier but, once you've got around doing it once, it is always the same process.

Let us use the dataset in the first lab as an example. To download it, you can access it on this link:

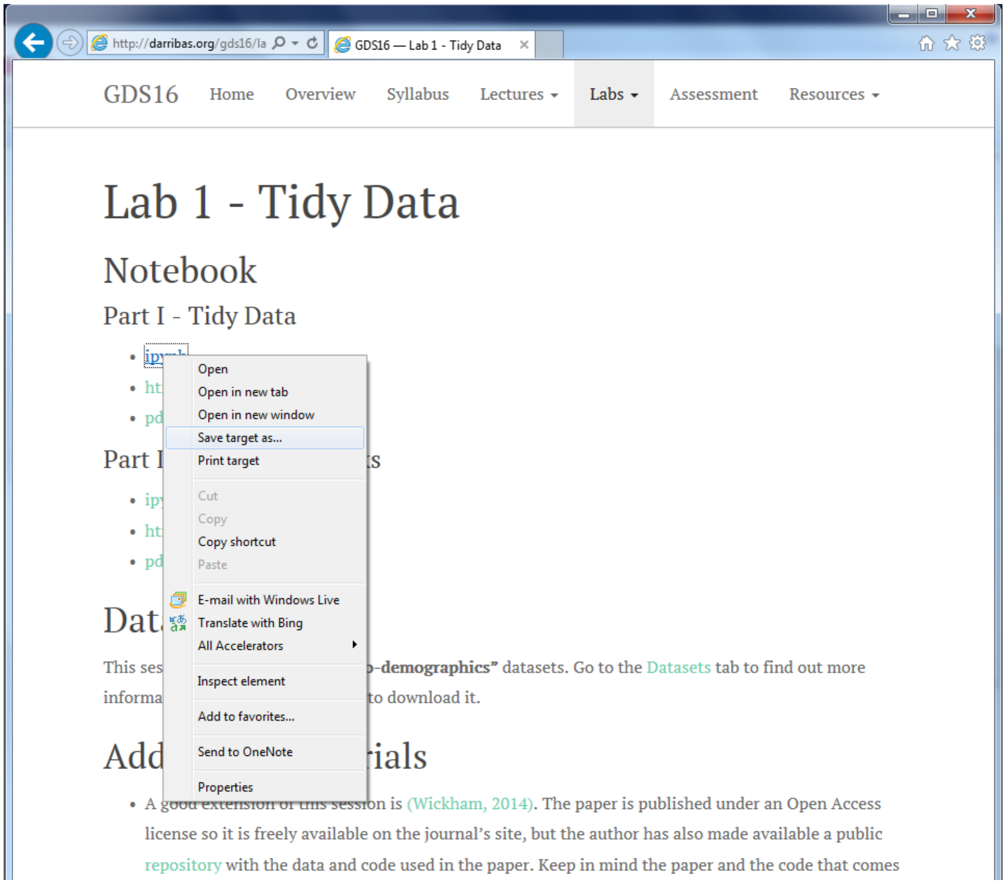
http://darribas.org/gds16/content/labs/data/liv_pop.csv

NOTE This is explained also in the data section of the course website.

You can download in the same way as we did the notebook before: righ-click on the link –> "Save target as. . . ". Same as before, place it somewhere **within your `M` drive**.

Now, depending on where you have saved the file, you will do one thing or another of the following:

A - **If you have placed the dataset in the same folder as the notebook**, all you need to do is just use the name of the file:



Lab 1 - Tidy Data

Notebook

Part I - Tidy Data

- ipynb
- ht
- pd

Part I   s

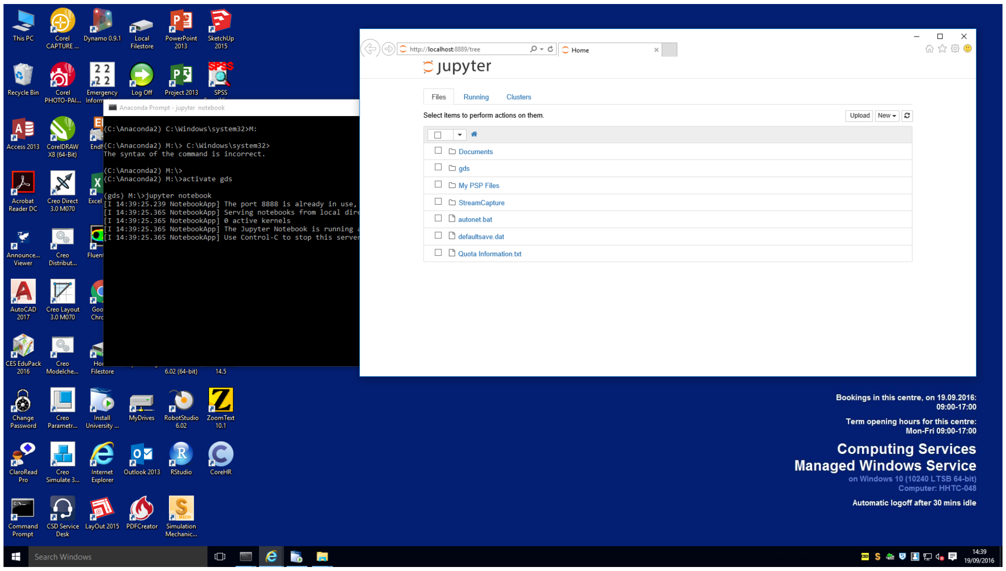
- ipy
- ht
- pd

Dat

This ses   -demographics” datasets. Go to the Datasets tab to find out more informa   to download it.

Add   ials

- A good extension of this session is (Wickham, 2014). The paper is published under an Open Access license so it is freely available on the journal’s site, but the author has also made available a public repository with the data and code used in the paper. Keep in mind the paper and the code that comes


Anaconda Prompt - jupyter notebook
http://localhost:8889/tree
Home
jupyter
Files
Running
Clusters
Upload
New

```
(C:\Anaconda2) C:\Windows\system32>M:

(C:\Anaconda2) M:\> C:\Windows\system32>
The syntax of the command is incorrect.

(C:\Anaconda2) M:\>
(C:\Anaconda2) M:\>activate gds

(gds) M:\>jupyter notebook
[I 14:39:25.239 NotebookApp] The port 8888 is already in use,
[I 14:39:25.365 NotebookApp] Serving notebooks from local dire
[I 14:39:25.365 NotebookApp] 0 active kernels
[I 14:39:25.365 NotebookApp] The Jupyter Notebook is running a
[I 14:39:25.365 NotebookApp] Use Control-C to stop this server
```

Select items to perform actions on them.

- □ Documents
- □ gds
- □ My PSP Files
- □ StreamCapture
- □ autonet.bat
- □ defaultsave.dat
- □ Quota Information.txt

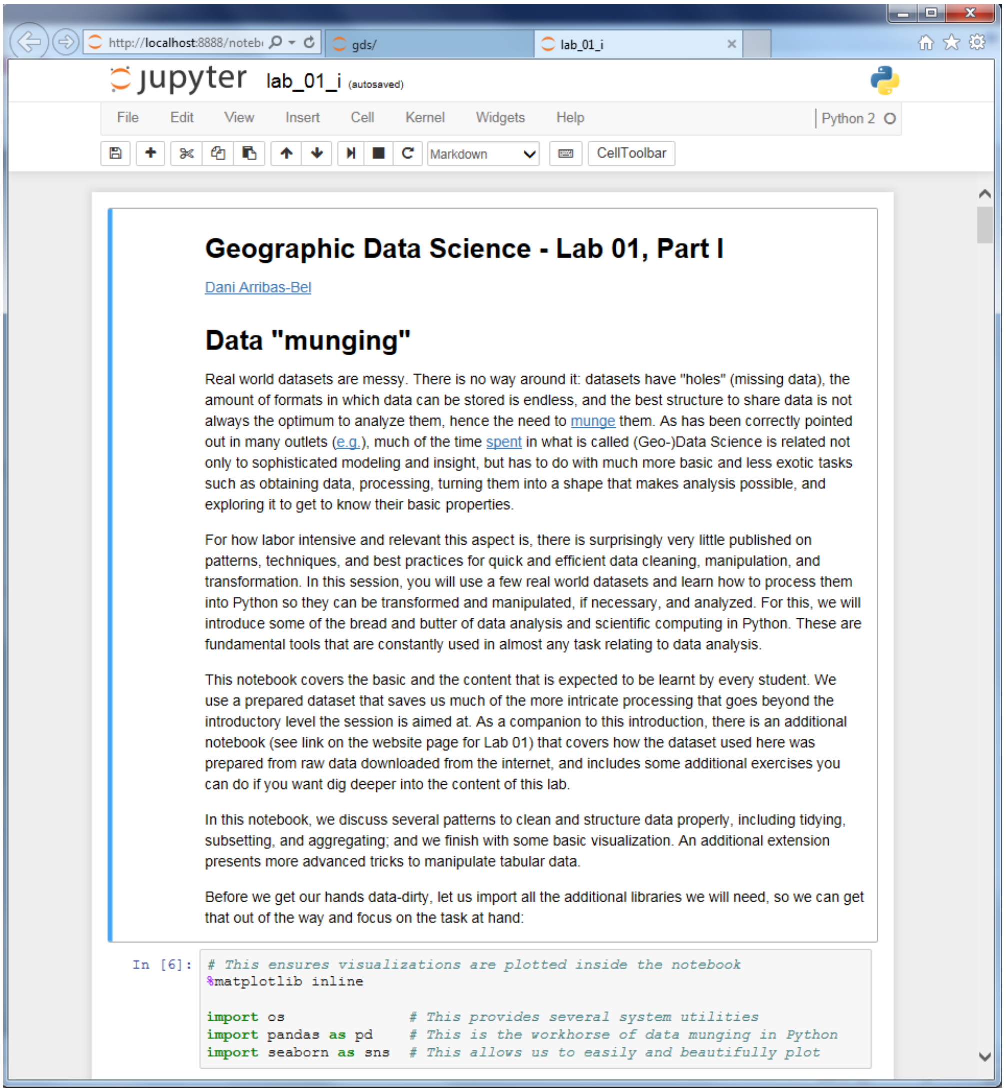
Bookings in this centre, on 19.09.2016:
09:00-17:00

Term opening hours for this centre:
Mon-Fri 09:00-17:00

Computing Services
Managed Windows Service
on Windows 10 (10240 LTSB 64-bit)
Computer: HHTC-048

Automatic logoff after 30 mins idle

14:39
19/09/2016


http://localhost:8888/notebo
gds/
lab_01_i
jupyter lab_01_i (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Python 2
Markdown
CellToolbar

Geographic Data Science - Lab 01, Part I

Dani Arribas-Bel

Data "munging"

Real world datasets are messy. There is no way around it: datasets have "holes" (missing data), the amount of formats in which data can be stored is endless, and the best structure to share data is not always the optimum to analyze them, hence the need to munge them. As has been correctly pointed out in many outlets (e.g.), much of the time spent in what is called (Geo-)Data Science is related not only to sophisticated modeling and insight, but has to do with much more basic and less exotic tasks such as obtaining data, processing, turning them into a shape that makes analysis possible, and exploring it to get to know their basic properties.

For how labor intensive and relevant this aspect is, there is surprisingly very little published on patterns, techniques, and best practices for quick and efficient data cleaning, manipulation, and transformation. In this session, you will use a few real world datasets and learn how to process them into Python so they can be transformed and manipulated, if necessary, and analyzed. For this, we will introduce some of the bread and butter of data analysis and scientific computing in Python. These are fundamental tools that are constantly used in almost any task relating to data analysis.

This notebook covers the basic and the content that is expected to be learnt by every student. We use a prepared dataset that saves us much of the more intricate processing that goes beyond the introductory level the session is aimed at. As a companion to this introduction, there is an additional notebook (see link on the website page for Lab 01) that covers how the dataset used here was prepared from raw data downloaded from the internet, and includes some additional exercises you can do if you want dig deeper into the content of this lab.

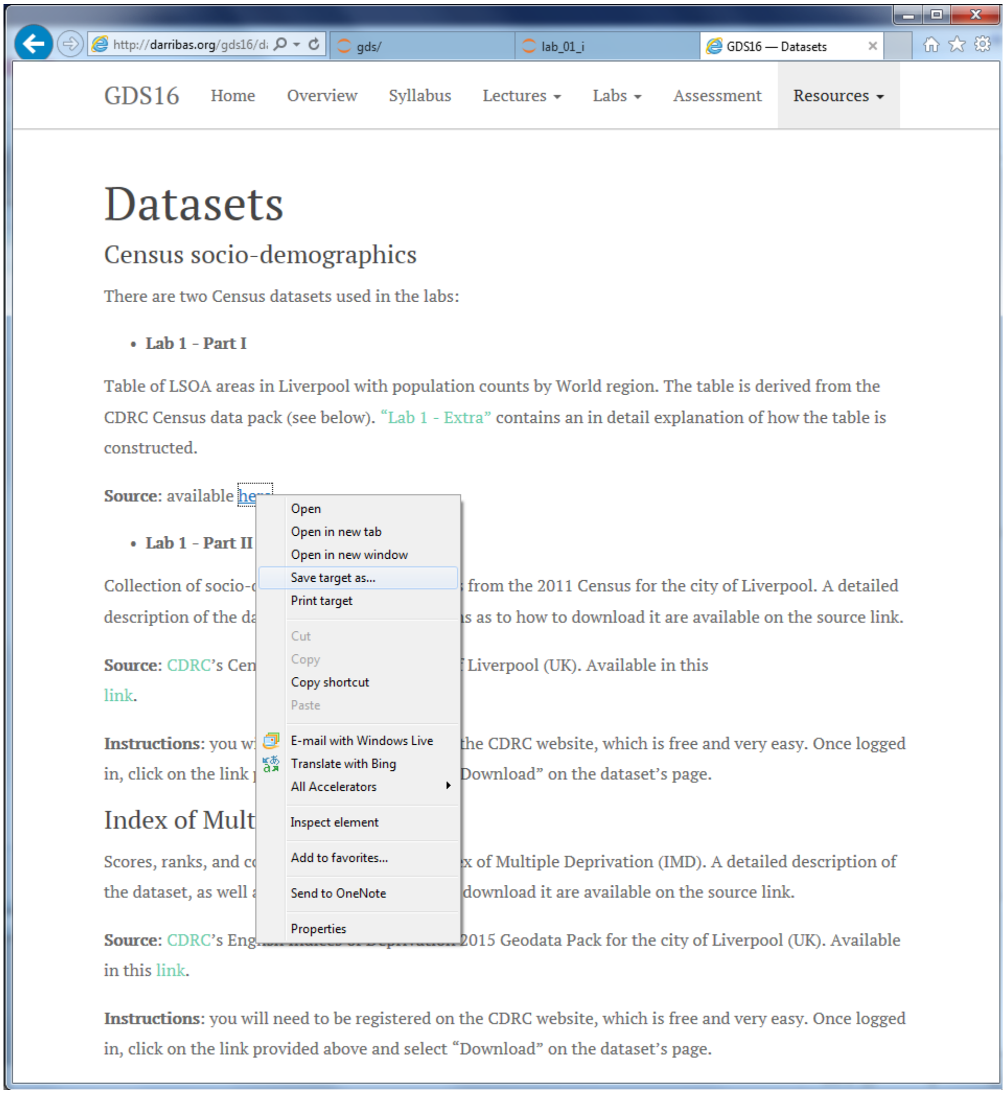
In this notebook, we discuss several patterns to clean and structure data properly, including tidying, subsetting, and aggregating; and we finish with some basic visualization. An additional extension presents more advanced tricks to manipulate tabular data.

Before we get our hands data-dirty, let us import all the additional libraries we will need, so we can get that out of the way and focus on the task at hand:

In [6]:

```
# This ensures visualizations are plotted inside the notebook
%matplotlib inline

import os              # This provides several system utilities
import pandas as pd    # This is the workhorse of data munging in Python
import seaborn as sns  # This allows us to easily and beautifully plot
```



Datasets

Census socio-demographics

There are two Census datasets used in the labs:

- **Lab 1 - Part I**

Table of LSOA areas in Liverpool with population counts by World region. The table is derived from the CDRC Census data pack (see below). "Lab 1 - Extra" contains an in detail explanation of how the table is constructed.

Source: available he

- **Lab 1 - Part II**

Collection of socio-(from the 2011 Census for the city of Liverpool. A detailed description of the da s as to how to download it are available on the source link.

Source: CDRC's Cen Liverpool (UK). Available in this link.

Instructions: you w the CDRC website, which is free and very easy. Once logged in, click on the link Download" on the dataset's page.

Index of Mult

Scores, ranks, and co x of Multiple Deprivation (IMD). A detailed description of the dataset, as well a download it are available on the source link.

Source: CDRC's Eng 2015 Geodata Pack for the city of Liverpool (UK). Available in this link.

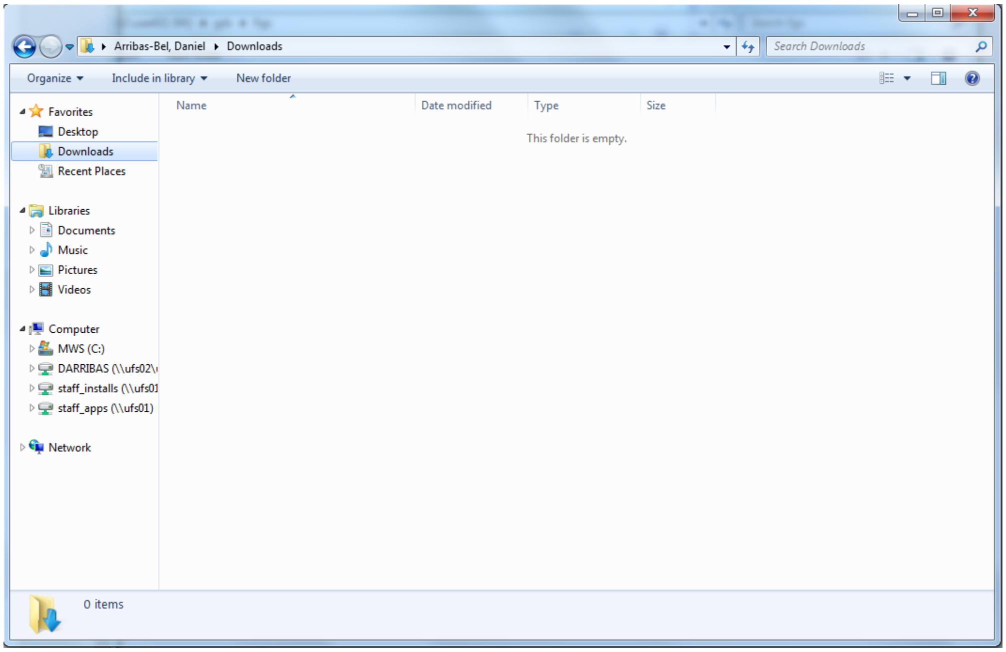
Instructions: you will need to be registered on the CDRC website, which is free and very easy. Once logged in, click on the link provided above and select "Download" on the dataset's page.

```
In [1]: f = 'liv_pop.csv'
```

B - **If you have placed the dataset in a folder *within* the folder where the notebook is**. For example, let us assume that, within the folder where your notebook is, there is a subfolder called `data`. In this case, you will point to the file in this way:

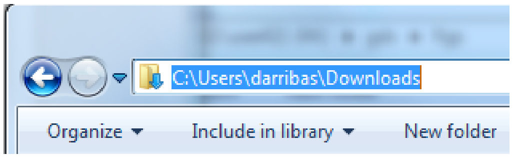
```
In [2]: f = 'data/liv_pop.csv'
```

C - **If you have placed the dataset in a different folder altogether**. In this case, you need to obtain what is called the “full path” of the file. Let us assume you downloaded the data file into the Downloads folder. To obtain the full path, the easiest way is to open a Windows Explorer window:



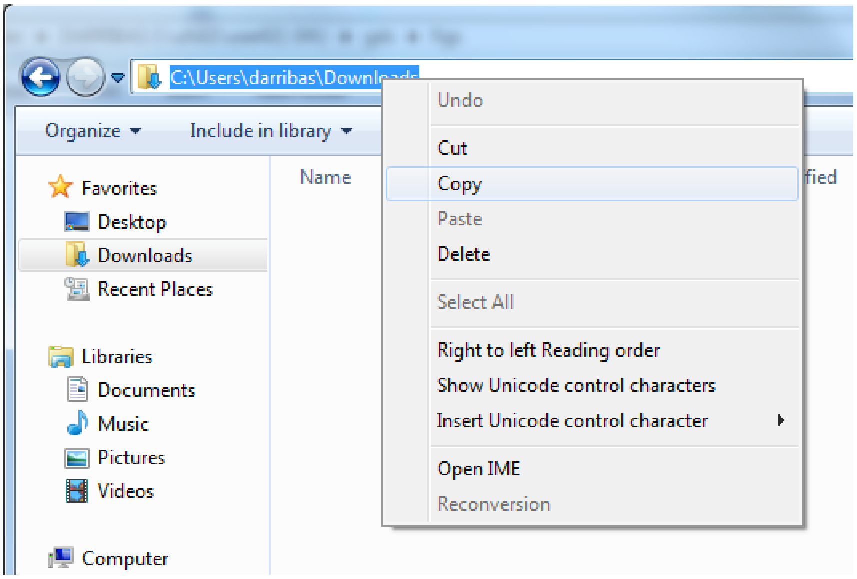
This folder is empty.

- Click on the top bar, where you would usually enter a web address if this was an internet browser. This should change it to something looking approximately like:



C:\Users\darribas\Downloads

- Then right-click on “Desktop” and “Copy”:



C:\Users\darribas\Downloads

- What you have just copied is the path to the folder where the file is, and you can paste this in-lieu of the options above. However, you need to the add the name of the file (just as above):

```
In [3]: f = "C:/Users/darribas/Downloads/liv_pop.csv"
```

Whichever way of the three (A, B, C) you have decided to go, you should now be ready to get on with the practical. Happy hacking!!!