Getting started with Python

Installing python is easy. Python is a programming language supported by a large community of people, and with a lot of open source libraries. It is becoming a standard for scientific computing and for data analysis.
A good distribution, free even if from a for-profit company, is **Anaconda**.
You can install following the instructions from here:
https://www.continuum.io/downloads

In order to write effectively Python code, you need an IDE (Integrated Development Environment). One IDE we will use, for sure for the introduction to Python programming, is **Jupyter** (previsouly, **Ipython**). This is a notebook style IDE and comes with the installation of Anaconda. If you need to learn how to use Jupyter, this tutorial is a good starting point.
Other more advanced choices, better for larger code development projects, are **Spider** (again included in the Anaconda distribution), and **PyCharm** from JetBrains, which you can get from here. My recommendation is Jupyter for fast prototyping and simple analysis and PyCharm (or Spider) for bigger projects.
Finally, for introducing python I will use the Iphyton notebooks from RJ Johansson, which you can download from here.
As you notice, the link sends you to **github**, an online free git repo. You don't know what git is? Well, it is a versioning system, which is very useful. Install it from here.
You can learn more about it following one of the many online tutorials (like this one, but you can google to find the best one for you).

Python for scientific computing

Here is a list of notebooks with a step by step introduction to Python and its main libraries. Very much recommended!!

If you prefer video tutorials, here is a nice selection of introductory courses.

- **Numpy:** foundamental library for scientific computing
  https://youtu.be/QUT1VHiLmmI

- **Matplotlib:** have fun in representing your data and you results
  - First plots: https://youtu.be/UO98lJQ3QGI
  - Histograms: https://youtu.be/XDv6T4a0RNc
  - Scatter plots: https://youtu.be/zZZ_RCwp49g
  - Subplots: https://youtu.be/XFZRVnP-MTU

- **Scipy:** built on top of Numpy library

https://nbviewer.jupyter.org/github/jrjohansson/scientific-python-lectures/blob/master/Lecture-3-Scipy.ipynb
Useful topics are:

  - Integration: https://www.youtube.com/watch?v=4grhQ5Y_MWo
  - Optimization
  - Statistics: https://www.youtube.com/watch?v=cg63hLYyuhI

Final advice

The libraries above contain a huge number of tools for statistics and scientific computing in general. **Do not reinvent the wheel.** Get used to read the manuals and to learn how to use the optimized functions provided by such libraries.

During the tutoring sessions I will present the most important tools and functions present in such libraries. In the meantime, get familiar with coding in Python.