1.022 - Introduction to Network Models

Python – some useful resources

Our course will make use of the python progamming language, which is an easy to learn, powerful programming language. Python is often used for Data Science applications and has a large ecosystem of libraries for machine learning, optimization, etc.

We do not assume that you know all the details about python, some general previous (first) programming experience in high level language (such as, e.g., MATLAB) should be enough to pick up the necessary tools and knowledge throughout this course. A nice feature of python is that there is an enourmous amount of high-quality free resources available online for learning, ranging from online tutorials to video lectures, books and interactive notebooks.

Below, we have provided some information and a few links as a starting point. In addition, we provide some recommendation that may be useful, in particular for the beginning. You do not have to stick to those recommendations however.

Comment   While you are free to use any tool to write / edit your python code, you will have to use python notebooks to hand in your coursework. A short tutorial on python etc. will be given in the first recitation session.

1   Python – Installation and environment

Before you can start you will need of course to have a working python installation on your computer. Installing python is not difficult, and you should be able to set it up on your computer without major problems.

The official python homepage – where you can download python and find a lot of useful documentation – is the following: https://www.python.org/

However, you may find it simpler to install a python environment via a 'package' such as Anaconda, or Canopy.

- https://www.anaconda.com/download

- https://www.enthought.com/product/canopy/

This will install not only python but also a graphical user interface to control your installation and a host of useful packages / toolboxes, some of which we make use of within the course (this includes in particular the matplotlib package for plotting, numpy and scipy for numerical and scientific computing, and networkX for network analysis). We would thus recommend using one of those two installation options – though you are free to use any other variant. There might be a default version of python installed on your computer as

well, if you are using a Max this is almost surely outdated, however, so please ensure to have an up-to-date version of python installed.

Note that there are two versions of python available. Python 2.X and Python 3.X. Both will be fine for the purposes of the course. However, there are some small differences in syntax and conventions (be aware of the differences when it comes to division by an integer!) – so simply decide for one or the other version of python and stick to it throughout the course.

Recommendation If you feel uncomfortable with installing python etc., simply use the Anaconda or Canopy python distributions.

2   Intro to python – some tutorial material

The official python homepage has a complete tutorial on python programming.
https://docs.python.org/3.6/tutorial/
The official documentation is in general a great resource and contain a lot of detailed information.

A guide written for a scientific audience is the following.
https://github.com/jrjohansson/scientific-python-lectures.
To get a general overview over the language have a look at the first 2 chapters. All the material can be downloaded as python notebook as well, so you go through the examples yourself in detail.

Another introduction with many examples drawn from data-science can be found under:
https://github.com/jakevdp/PythonDataScienceHandbook
Again, all the code is available as python notebook.

NetworkX We will make use of networkX throughout the course. NetworkX has a lot of high-level functions to facilitate network analysis, which will make many of your tasks much easier. There is a tutorial that explains most of the basic functionality of networkX here
https://networkx.github.io/documentation/stable/tutorial.html

Recommendation Browse through the first two chapters of the scientific python lectures to get an overview over python. Go through the networkX tutorial to familiarize yourself with networkX.

Some further recommendations for writing code in Python

While there are many ways to write and run programs in python, many people find it easy to work within a more visual environment. If you prefer such a working environment, you may find that the IDE Spyder is a convenient environment for you (see also chapter 1 of the scientific python lectures above).
https://github.com/spyder-ide/spyder.
Spyder is also included in the Anaconda distribution.

Another useful environment to work in are so-called jupyter (or sometimes called ipython) notebooks. These notebooks allow you conveniently to combine text, LaTeX, and code-blocks in a single document, which can be displayed and executed from your browser and be exported to pdf / html. You will make use of these notebooks to write your homework reports (the coding parts). We will provide you with a template (including some pre-filled code) for each homework.

1.022 Introduction to Network Models
Fall 2018