Python for Ecology notes

Tom Purucker

July 26, 2012

Abstract

Python is a high-level scripting language that is becoming increasingly popular for scientific computing. This all-day workshop is designed to introduce the basics of Python programming to ecologists. Some scripting/programming experience is recommended (e.g. familiarity with R).

The morning session will teach the basics of Python. Most of the material is presented within the Python shell and will consist of 15 minute demonstration sections followed by hands-on exercises. This session will touch on computer science basics and how they are implemented in Python. Topics include numbers, operators, lists, modules, functions, namespaces, exceptions, etc.

The afternoon session will venture into the Numpy and Scipy packages for scientific computing. Numpy is a basic package for scientific computing with Python. It adds to Python data arrays that can access to a large library of mathematical functions and operations. It is the basis for all Scipy packages which extends vastly the computational and algorithmic capabilities of Python. SciPy is open-source software for STEM fields and provides many user-friendly and efficient numerical routines such as routines for numerical integration and optimization. We will overview some of the commonly used sub-packages in SciPy, providing coverage of the mathematical models, the software architecture, and some examples and exercises.

Attendees should bring a laptop (Windows, Mac, or Linux) with Python 2.7 installed.

Python is a high-level scripting language that is becoming increasingly popular for scientific computing. This all-day workshop is designed to introduce the basics of Python programming to ecologists. Some scripting/programming experience is recommended (e.g. familiarity with R). The workshop is split into 2 parts: the morning session will teach the basics of python and the afternoon session will delve into the Numpy and Scipy packages.

1 Hour 1

1.1 Python for Ecology

This class is rather large- at over 40. Assuming that there are many here who have very little to no programming experience. Others who have some (particularly R) but want to learn Python and fill in some gaps. I cannot design a class that maximizes the information for all, but I hope there will be plenty on offer today such that everyone goes home being exposed to a lot of new material. The idea of these workshops is a bit of a forced march, you know where you will be and what you are doing for the next few hours and we will move through some programming concepts and some Python particulars.

Logistics. Bathroom location and food schedule. Helpers, Tao and who knows who else. But probably the people in your immediate vicinity. This class is much larger that anticipated, I thought I had capped it at 25 and didn't expect to get that much, but we are well over that. Do not be afraid to ask a neighbor as we go, especially when we get to the problem sections. I have a set amount of material, but we do not have to get through it all.

1.2 Today's approach

Again, designed for those with not much programming experience. The general approach is to lecture for some period of time, with you asking questions as needed, then to challenge you with some problems that you can try for a short period of time. It is not expected or necessary for you to get through all the problem sets each time, in fact, if you consistently do, then maybe you are overqualified for this class.

Being a course for ecologists, the idea is to bring in ecological source material as much as possible to use as examples as we push through different programming concepts.

1.3 Why bother with Python when I have R?

R and Python are both scripting languages, we can work at a command prompt to work with different commands, in R people more and more use RStudio rather than the native R interface. They are both often commonly used to write scripts, collections of commands in a flat text file that can be run all at once.

R is generally acknowledged to be the statistical software leader, Python is more math/physics/engineering. There is a significant area of overlap, the overlapping area in the Venn diagram is growing quickly as Python adds more statistical capabilities and R adds more physics/math type capabilities.

Importantly, it is very easy nowadays to use Python and R together.

At this point, R is probably the easier solution to producing high quality graphics for publications such as journal articles or your dissertation or whatever. However, Python is a high level programming language that is in general more powerful for controlling your computer, for producing applications for other users, for creating scientific web applications, or even for producing apps for your cell phone or for controlling your scientific instruments or for working with ArcGIS.

Of these scientific high level programming languages, Python distinguishes itself by being designed to produce readable code.

1.4 übertool

To give you an idea of the power of Python. I will quickly show you a project we have been working on for less than a year. This is the übertool, an open source EPA project that not only uses Python to run a number of population models and EPA models for regulating pesticides. In addition, Python is working behind the scenes to host the web site and actually serves the HTML (the web pages themselves) via the use of Google App Engine and Amazon Cloud Services.

The übertool is at http://www.ubertool.org.

The chemical exposure models are used to estimate exposure and risk to wildlife from pesticide applications. The results of these models are used to approve or deny applications from chemical companies concerning putting pesticides on the market. In case of approval, they are used to determine the allowable application rates of these pesticides, whether for a farm, a household, or for urban applications (e.g., West Nile Virus). These rates listed on the label are the actual law - the regulations on the allowable use of pesticides - and therefore are a direct connection between models and environmental protection and of course economics.

- Demonstrate THerps.
- Demonstrate PRZM.
- Demonstrate logistic growth.
- Demonstrate Leslie matrix.
- Demonstrate Leslie matrix with logistic dose-response impact on mortality from chemical exposure.

Talk about Tao's talk on the use of chemical and population models in an education session and Robin's talk about dermal exposure which will be used to inform a new model that estimates the exposure of pesticides across the dermal membrane - previously not considered in ecological risk assessment.

These chemical exposure models date to the early 1980s and have been developed over the years with under a variety of technologies. Some were developed in Fortran, others as customized Windows applications, and still others as standalone Excel spreadsheets.

In the near future, the EPA is required to reregister all existing pesticides that have already been approved, and also to run specific pesticide exposure scenarios for all endangered species that are within a defined spraydrift buffer zone for pesticide applications across the entire United States. The number of scenarios are scary-high and are simply not practical with the current set of individual

I'll be quite honest, part of the reason for me giving this workshop is that this is an open source project and we are looking for programmers to help us with this effort and many others like it. Not just producing data analysis and figures for journal articles – which is very very important – but creating scientific applications, which are more and more going to be on the web and on the cloud and on your smart phone, that non-programmers can use to

Questions about how this web page works or about how the EPA uses these models?

1.5 Schedule

- 1- Introduction, Objects, Data types, Assignments, Tracebacks
- 2- Data structures and Arrays
- 3- Control structures

- 4- Functions
- 5-
- 6- Talking to external data sources
- 7- Monte Carlo and Markov chains
- 8- GUI elements

1.6 Getting setup to run Python

You can google ”download python” to get the python installation online.

1.7 The IDLE IDE