Python for Analytics and The Role of R

Christian B. Madsen, PhD, Estelle Cormier, PhD, Javier Von Stecher, PhD, Kuo Liu, PhD, Gennady Voronov, PhD, Hans Gu, PhD, Ed Wiley, PhD

Seagate Point of View

Two Popular Open-Source Programming Languages to Consider for Your Data Science Toolkit

R and Python are two very popular open-source programming languages for data analysis. Frequently, users debate as to which tool is more valuable, however both languages offer key features and can be used to complement one another. A common perception is that R offers more depth when it comes to data analysis, data modeling and machine learning, but Python is easier to learn and tends to present graphs in a slightly more polished way.[1,2] Using the interface Python offers for calling R allows users to reap the benefits of both of these powerful, popular tools for data science. Even if you choose not to combine the two, the different ways in which these two languages are valuable make them both important parts of a data science toolkit.

Why Python?

Python is a popular, general-purpose programming language with an emphasis on being readable and allowing programmers to use fewer lines of code to accomplish tasks than in older languages. Libraries such as NumPy, SciPy, and Matplotlib make it useful for scientific computing.

Python is an excellent tool for data analysis for four reasons:

- Open source
- Speed
- Support
- Scope

Python for Analytics and The Role of R

Open Source

Python is free, open source, and is developed using a community-based model. It runs on Windows and Linux environments and can easily be ported to multiple platforms.

Speed

Python is a high-level language, which means it has a number of benefits that accelerate code development. The high-level character of Python makes prototyping ideas and code fast. Another benefit is that Python is relatively easy to learn. That has been ideal for a group where people come from different programming backgrounds (this is common with many data science groups). Finally, but most importantly, there is a great transparency between code and execution. This transparency eases both maintenance of the code (rewriting, finding bugs, etc.) and the process of adding to the code base in a multi-user development environment.

Support

Python is widely used for scientific computing in both academia and industry. As a consequence, a large number of useful analytics libraries are available (and well tested), including packages for numerical computing, data analysis, statistical analysis, visualization and machine learning. All you really need to do in order to get going on a topic is to search online: 'Python + [your analytics approach/tool].' From there, you can begin testing code that offers the analytics you desire and has vast amounts of documentation and examples online to guide you.

Scope

Python supports object-oriented programming and advanced data structures such as lists, tuples, sets, dictionaries and so on. Also matrix operations can be used with the NumPy library and the package pandas supports data frames. Having these abilities within the Python scope helps simplify and speed up data operations.

Why R?

R is a popular programming language for data science. It is based on the S language developed at Bell Labs and is a go-to tool for data scientists. It contains every known statistical method and is widely used by academic statisticians as well.

R is an excellent tool for data analysis for three reasons:

- Open source
- Emphasis on statistical analysis
- Community

Open Source

R is the most comprehensive, open-source data analysis tool in existence. It is free, flexible and evolving with great speed.

Emphasis on Statistical Analysis

R was designed by statisticians with the purpose of facilitating statistical analysis. The large number of packages (5521) and strong statistics support that is available in R allows a user to get sophisticated analyses up and running in very little time. Data scientists may often find that R already contains a package that performs the analysis they are interested in. As such, when working with R, often there is no need to reinvent the wheel. Many of R's pre-packaged libraries are straightforward to use if one's modeling needs are standard. Some of the packages available are:

- Data manipulation: plyr
- Visualization: ggplot2 and animation
- SQL queries: sqldf
- Network client: RCurl
- R-Java interface: rJava
- Geography: maps, RgoogleMaps, ggmap
- Dynamic reporting: knitr
- Parallel computing: Rmpi, snow, multicore, parallel
- Big data analytics: RHadoop, RHIPE, RSpark

Python for Analytics and The Role of R

Community

R is widely used by academic statisticians and has a very active user community. The LinkedIn R user group has more than 30,000 members, sites like Meetup offer more than 2800 groups on the topic, and the language offers a lot of user created documentation that is available publicly online.

Conclusion

Both R and Python are valuable programming languages when it comes to data analysis. The depth of support for statistical analysis in R is unprecedented for a freely available tool and Python integrates better with system environments, real-time data streams and other software tools (including R) than anything else available. As such, the combination of R and Python provides the ultimate tool-kit for today's data scientists.

1 Data Science Wars: Python vs. R - http://inside-bigdata.com/2013/12/09/data-science-wars-python-vs-r/

2 Cleaning Data and Graphing in R and Python - http://www.r-bloggers.com/cleaning-data-and-graphing-in-r-and-python/

www.seagate.com

AMERICAS Seagate Technology LLC 10200 South De Anza Boulevard, Cupertino, California 95014, United States, 408-658-1000
ASIA/PACIFIC Seagate Singapore International Headquarters Pte. Ltd. 7000 Ang Mo Kio Avenue 5, Singapore 569877, 65-6485-3888
EUROPE, MIDDLE EAST AND AFRICA Seagate Technology SAS 16–18, rue du Dôme, 92100 Boulogne-Billancourt, France, 33 1-4186 10 00