COMPARISON BETWEEN PYTHON, R AND JULIA LANGUAGE IN DATA SCIENCE

Sreelakshmi Menon[1]; Dr. Anjana S Chandran[2]
[1]*Scholar, SCMS, Cochin, Kerala, India, imca-38@scmsgroup.org*
[2]*Assistant Professor, SCMS, Cochin, Kerala, India, anjana@scmsgroup.org*

Abstract

Data science is the field of study that combines domain expertise, programming skills, and knowledge of math and statistics to extract meaningful insights from data. Data science practitioners apply machine learning algorithms to numbers, text, images, video, audio, and more to produce artificial intelligence (AI) systems that perform tasks which ordinarily require human intelligence. In turn, these systems generate insights that analysts and business users translate into tangible business value. All three languages, Python, R, and Julia are dynamically typed, have a command line interface for the interpreter, and come with great number of additional and useful libraries to support scientific and technical computing. Conveniently, these languages also offer great solutions for easy plotting and visualizations.

Keywords: Data Science, Python, R, Julia

1. Introduction

Data science is a multidisciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from structured and unstructured data. Data science is related to data mining and big data.

Data science is a "concept to unify statistics, data analysis, machine learning and their related methods" in order to "understand and analyze actual phenomena" with data. It employs techniques and theories drawn from many fields within the context of mathematics, statistics, computer science, and information science. Turing award winner Jim Gray imagined data science as a "fourth paradigm" of science (empirical, theoretical, computational and data-driven) and asserted that "everything about science is changing because of the impact of information technology" and the data deluge. In 2015, the American Statistical Association identified database management, statistics and machine learning, and distributed and parallel systems as the three emerging foundational professional communities.[1]

2. Overview

2.1. Python

Python is a suitable language for both learning and real world programming. Python is a powerful high-level, object-oriented programming language created by Guido van Rossum. Python is a general-purpose, high-level programming language which is widely used in the recent times[2] [3][4]. Its design philosophy emphasizes code readability, and its syntax allows programmers to express concepts in fewer lines of code than would be possible in languages such as C [5]. The language constructs enable the user to write clear programs on both a small and large scale [6]. The most important feature in Python being it supports multiple programming paradigms, including object-oriented, imperative and functional programming or procedural styles. Python supports a dynamic type system and automatic memory management and has a large and comprehensive standard library. Python interpreters are available for many operating systems.

Python is a well-designed language that can be used for real world programming. Python is a very high-level, dynamic, object-oriented, general purpose programming language that uses interpreter and can be used in a vast domain of applications. Python was designed to be easy to understand and use. Python is termed as a very user-friendly and beginner-friendly language in the recent times. Python has gained popularity for being a beginner-friendly language, and it has replaced Java as the most popular introductory language. As a dynamically typed language, Python is really flexible. Furthermore, Python is also more forgiving of errors, so you'll still be able to compile and run your program until you hit the problematic part. Python is a flexible, simple coding programming language. This language can support different styles of programming including structural and object-oriented. Other styles can be used, too. Python is very flexible, because of its ability to use modular components that were designed in other programming languages. For example, you can write a program in C++ and import it to python as a module. Then add something else to it (for example design a GUI for it)[7].

2.2. R

The development of R was inspired by S with some programming influences from Scheme [8]. Two professors introduced the language to assist students with a more intuitive language, specifically lexical scoping which eliminates the necessity for global defining of variables [9]. Although the history R can find a foundation in FORTRAN, R is its own language. R is an interpreted language with code directly executed rather than compiled. Using a compiler, programmers can write interfaces for C, C++, and FORTRAN for efficiency. R is part of the GNU Project, which focused on free software allowing users the ability to run, redistribute, and improve the program [10]. Although initially criticized, R upgraded quickly with collaboration from around the globe. Since R is open source, the target audience is any user interested in statistical computing. R can be installed using Unix, Windows, or Mac. R is available for download via the Comprehensive R Archive Network (CRAN). The master site is in Austria; however, mirrored sites throughout the world distribute the load on the network. In addition to the software, the CRAN hosts provide supporting documentation and libraries with add-on packages. The open-source add-on packages, which are groups of functions developed by other users, are available on CRAN. As on January 27, 2017, the CRAN hosted more the 10,000 packages which does not include packages from other vendors [11]. Although no warranties are given by R for any packages on CRAN, all the package contributions are reviewed by the CRAN team. Some packages in the libraries may restrict commercial use although the same packages may be openly available for education and research. RStudio is an IDE using packages (knitr and rmarkdown) to develop composed documents with the code and output from the R language. In addition, RStudio is an editor for LaTeX which is a markup language to produce high quality documents. A 2011 poll rated RStudio as the most used IDE with only the basic R console more frequently used [12].

2.3. Julia

Julia is a high-level dynamic programming language designed to address the requirements of high-performance numerical and scientific computing while also being effective for general purpose programming. Distinctive aspects of Julia's design include having a type system with parametric types in a fully dynamic programming language, and adopting multiple dispatch as its core programming paradigm. It allows for parallel and distributed computing, and direct calling of C and Fortran libraries without glue code. Julia is garbage collected by default, uses eager evaluation, and includes efficient libraries for floating point, linear algebra, random number generation, fast Fourier transforms, and regular expression matching. While a Julia runtime is needed pre-installed, by default for the source code, standalone distribution of pre-packaged Julia programs is also possible. Julia features optional typing, multiple dispatch, and good performance, achieved using type inference and just-in-time (JIT) compilation, implemented using LLVM. It is multi-paradigm, combining features of imperative, functional, and object-oriented programming. Julia provides ease and expressiveness for high-level numerical computing, in the same way as languages such as R, MATLAB, and Python, but also supports general programming. To achieve this, Julia builds upon the lineage of mathematical programming languages, but also borrows much from popular dynamic languages, including Lisp, Perl, Python, Lua, and Ruby. Because Julia's compiler is different from the interpreters used for languages like Python or R, users may find that Julia's performance is unintuitive at first. However, once users understand how Julia works, it's easy to write code that's nearly as fast as C. This workshop is intended to serve as a tutorial for attendees interested in using Julia for their applications, and it will offer the opportunity to engage with one of the developers and users of Julia.

3. Table of Comparison

Aspects	Python	R	Julia
First release	1991	1995	2009
Initial Authors	Guido Van Rossum	Ross Ihaka and Robert Gentleman	Jeff Bezanson, Stefan Karpinski, Viral B. Shah, and Alan Edelman
Current Stable Version	3.7	3.5	1.2
Number of Packages (October 2019)	199,816	15102	~2496
Compiled/Interpreted	Interpreted	Interpreted	Compiled Just-In-Time(JIT)
Main Implementaion Languages	C (CPython)	C and Fortran	Julia
Primitive Data Types	Numbers(Integers,Float), Strings, Boolean	Numeric,Int,Character, Logical	Numbers, Char, Bool
Object-Oriented	Yes	Yes	Selective
Code Structure	Based on Indentation	Free style	Free style
Notebooks/Literate Programming	Jupyter,pweave	Jupyter, R Markdown, swave,knitr	Jupyter,Weave.jl, Literate.jl

4. Conclusion

Julia language is faster than both Python and R and can use Python packages with Pycall. The industry is using it when speed is a bottleneck .R is used a lot in academia, so people with that background or doing academic research usually pick R. Many people pick Python when they also want to integrate it to web apps by using Django or Flask. But the conclusion is that it is hard to pinpoint how the industry are adopting Python and R because both seem to be good at analysis, plotting, creating web applications, etc. It boils down to very specific task. For example, for a quick plot many people prefer R's ggplot2, for web scraping most people prefer Python libraries, and so on.

References

[1]. https://en.wikipedia.org/wiki/Data_science.
[2]. TIOBE Software Index (2011). "TIOBE Programming Community Index Python".
[3]. "Programming Language Trends - O'Reilly Radar". Radar.oreilly.com. 2 August 2006.
[4]. "The RedMonk Programming Language Rankings: January 2011 – tecosystems". Redmonk.com.

[5]. Summerfield, Mark. Rapid GUI Programming with Python and Qt.

[6]. Fraud Detection using Machine Learning Aditya Oza.

[7]. Kuhlman, Dave. "A Python Book: Beginning Python, Advanced Python, and Python Exercises".

[8]. “Python– The Fastest Growing Programming Language” K. R. Srinath.

[9]. R. A. Becker, "R: A brief history of S.," AT&T Laboratories, New Jersey.

[10]. C. A. Gomez Grajales, "Created by statisticians for statisticians: How R took the world of statistics by storm," 19 November 2015. [Online]. Available: http://www.statisticsviews.com/details/feature/8585391/Created-by-statisticians-forstatisticians-How-R-took-the-world-of-statistics-by.html. [Accessed 11 November 2017].

[11]. D. Smith, "CRAN now has 10,000 R packages. Here's how to find the ones you need," 27 January 2017. [Online]. Available: http://blog.revolutionanalytics.com/2017/01/cran10000.html. [Accessed 31 October 2017].

[12]. "R GUIs you frequently use," April 2011. [Online]. Available: https://www.kdnuggets.com/polls/2011/r-gui-used.html.. [Accessed 31 October 2017].

[13]. K. A. Renzulli, C. Weisser and M. Leonhardt, "The 21 Most Valuable Career Skills Now," Time.com, 16 May 2016.

[14]. Alan Edelman, “Julia Introduction”, 2015 IEEE International Parallel and Distributed Processing Symposium Workshop.