



Single Day Event

Sanjeev SR, *Accenture Life Sciences R&D*

phuse.global



"JULIA IN A NUTSHELL" AS AN OPEN SOURCE

AGENDA

- 01 Abstract & Introduction**
- 02 julia language and its features**
- 03 Usage of julia and its statistics**
- 04 Comparison**
- 05 Pumas & adoption of julia in pharmaceutical industry, key highlights**
- 06 julia- Getting Started**
- 07 julia- Tutorials**
- 08 References**



ABSTRACT & INTRODUCTION

With a growing number of companies in the pharmaceutical industry, there is no surprise when it comes to adopting open-source tools and methods to potentially move faster, while performing exceptionally well in the clinical submissions.

julia is a free and an open-source language with multi-model functionality and object-oriented programming. It provides ease and expressiveness for high-level numerical computing, like R & Python. It supports general programming and works with almost all the databases.

julia has been downloaded over 25 million times and which has also registered over 5,000 Julia packages for community use. These include various mathematical libraries, data manipulation tools and packages for general purpose computing. In addition, we can easily use libraries from Python/R, C, C++, and Java.

Its compilation is different from the other languages like Python or R. It is easy to understand, let's learn how to use it and that will be as quick as learning C.





FEATURES OF JULIA

Created in 2009 by a four-person team and unveiled to the public in 2012, **julia** is meant to address the shortcomings in Python and other languages and applications used for scientific computing and data processing.

COMPILED, NOT INTERPRETED

- For faster runtime performance, **julia** is Just-In-Time (JIT) **compiled using the LLVM compiler framework**. At its best, **julia** can approach **the speed of 'C'**.

STRAIGHTFORWARD SYNTAX

- **julia's Syntax is similar to Python** but also expressive and powerful

DYNAMIC TYPING & STATIC TYPING

- Write a **function that accepts integers without specifying the length** or signing of the integer. We can even do without typing entirely if it isn't needed in a particular context

FULL-FEATURED DEBUGGER

- **julia** introduced a **debugger suite, which executes code in a local REPL** and allows to step through results, inspect variables and add breakpoints in code

INTERACTIVE

- Similar to what Python offers, **julia includes REPL (Read-Eval-Print-Loop), interactive command line**

CAN CALL PYTHON, R, C & FORTRAN LIBRARIES

- **julia** can **interface directly with external libraries** and can be shared vice versa

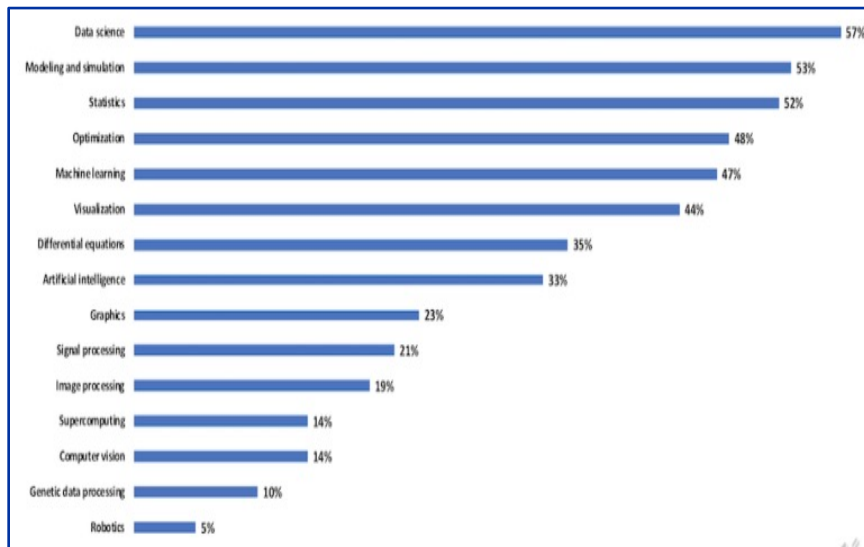


STATISTICS OF JULIA USAGE

In the Open-Source **julia**, the number of downloads has spiked to 87 percent (2020) which is more than 2019's 24 million downloads. It is interesting to know in **julia** more than 50% of the users effectively utilized this who are from Data Science, Modeling & Simulation, and Statistics.

Julia adoption accelerated at a rapid pace in 2020.

	Total Cumulative as of Jan 1, 2020	Total Cumulative as of Jan 1, 2021	Change
Number of Julia Downloads (JuliaLang + Docker + JuliaPro)	12,950,630	24,205,141	+87%
Number of Julia Packages	2,787	4,809	+73%
GitHub stars (Julia language repo + registered packages)	99,830	161,774	+62%
YouTube views (Julia language channel)	1,562,223	3,320,915	+113%
Published citations of Julia: A Fast Dynamic Language for Technical Computing (2012) + Julia: A Fresh Approach to Numerical Computing (2017)	1,680	2,531	+51%
Discourse posts	137,399	211,888	+54%
TIOBE Index Rank	#47	#23	+24



Continued...

#	Python	R	julia
1	Python is an open-source programming language. Python's codes are simpler and easier to learn	R is an open-source language available as per GNU General Public License	It's an open-source language with an MIT license, and it's free
2	Python is an object-oriented language; therefore, it focuses on objects. However, Python can support both procedure-oriented and object-oriented programming	R, slightly similar to the real object-oriented language like Java, but it is not vastly used	julia supports both objected-oriented and functional programming, moreover, the language is good for interactive use
3	Python is a dynamically-typed language, the type for a variable is dedicated at runtime.	R is a dynamically-typed language	julia is a dynamically-typed language however, it also offers you the ability to specify types. Developers can create different types of hierarchies, which enables them to handle variables of specific types
4	Python comes with a large standard library, and it has already written codes for many functions. It has extensive libraries	R has a few libraries (ggplot, caret etc.)	Has limited libraries but it can access that of C and Fortran to handle plots. Can also use Python packages
5	Simple and readable code structures makes Python easier to learn for beginners	It is not easy to learn for beginners	julia uses a high-level syntax, making It easy for developers of all backgrounds to Learn.
6	Python is fast but slower in comparison to C and has very large communities.	R is slower when compared to C. Limited communities.	julia is faster than Python but close to C. Limited communities.

PUMAS OVERVIEW & JULIA INVOLVEMENT



- **Pumas is a comprehensive platform for pharmaceutical modeling and simulation**, providing a single tool for the entire drug development pipeline. It is used for simulation and estimation of quantitative pre-clinical and clinical pharmacological models (Quantitative Systems Pharmacology – QSP)
- Pumas can also perform non-compartmental analysis (NCA), bioequivalence and in vitro-in vivo correlation (IVIVC) analysis providing a common toolset to perform all analysis parallelly for clinical drug development and clinical trial simulation. **Pumas is massively scalable with an inherent ability to run on GPUs and on any hosted or private cloud computing setups in conjunction with JuliaTeam and JuliaRun.**

*Pumas software was developed through partnership with experts at 'Julia Computing' utilizing **julia** programming language.*



ADOPTION OF JULIA IN PHARMACEUTICAL INDUSTRY

Below are a few organizations across Pharma currently utilizing **julia** on End-to-End drug development, pharmacometrics, Quantitative Systems Pharmacology (QSP) and more, to increase productivity gains, scalability and cost savings.



CASE STUDY 1 - ADOPTION OF JULIA IN PHARMACEUTICAL INDUSTRY



CASE STUDY-1

To increase **R&D productivity and reduce cost**, **PFIZER** uses *model-informed drug development (MIDD) techniques* like quantitative systems pharmacology (QSP) to model the effect of their therapies in patients before testing them in the clinical trials.

QSP modeling assists in identifying issues with efficacy and safety before commencing the next phase of trials. This technique has been employed to help predict trial failures and potentially save valuable resources by allowing the company to focus on therapies with the highest probabilities of clinical success.

OUTCOMES

- Simulations of novel pharmacology of a treatment for heart failure sped up by **175x faster using GPUs in julia**
- Simulations of a **model of cardiac metabolism were accelerated by 26x using Julia on single CPUs**
- Multithreading the same cardiac model with julia increased **the simulation speed up to 115x**

Continued...



CASE STUDY -2 ADOPTION OF JULIA IN PHARMACEUTICAL INDUSTRY



United Therapeutics is a biotech company that develops treatments for rare diseases, including diseases affecting the lungs. They **use JuliaHub in the cloud to build a computational model of the lung for drug development.**

The computational biology group at United Therapeutics combines models at different scales to **predict the effect of drugs on organ-level clinical outcomes.**

OUTCOMES

- Computational biologists **predicted the effect of a vasodilator drug** by modeling its effect on biochemical reaction networks within smooth muscle cells, and then combined with a CFD model of blood flow to predict clinical outcomes such as blood pressure and vascular resistance.
- The **performance of the JuliaHub was high** along with the ability to easily submit jobs to the cloud – especially for tasks such as parameter estimation, where parallelization across multiple CPUs is found to be efficient

Below are the key highlights of **julia** across the Pharmaceutical industry and Universities.

450 budding pharmacists trained in **julia** and Pumas in 2020 so far

julia/Pumas has **speed up multiscale models by 10x-30x**



julia/Pumas has allowed scientists to run on GPU

Over **40 pharma firms** have taken up **julia** and Pumas for their workflows

6 academic groups have shifted their educational programs and used **julia/Pumas** along with R in their curriculum



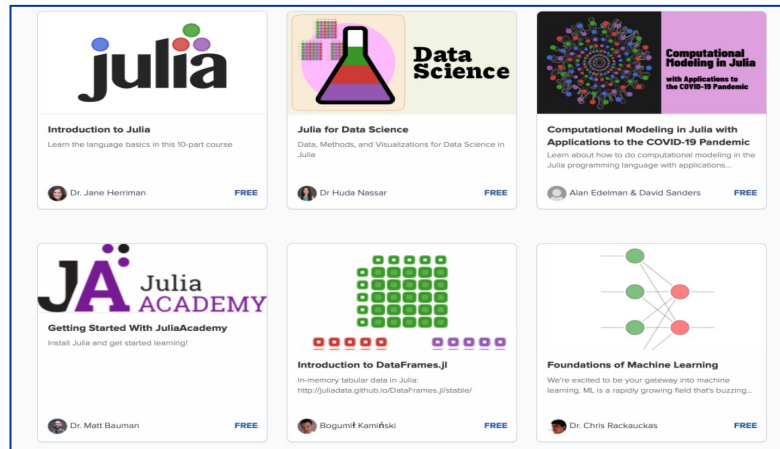
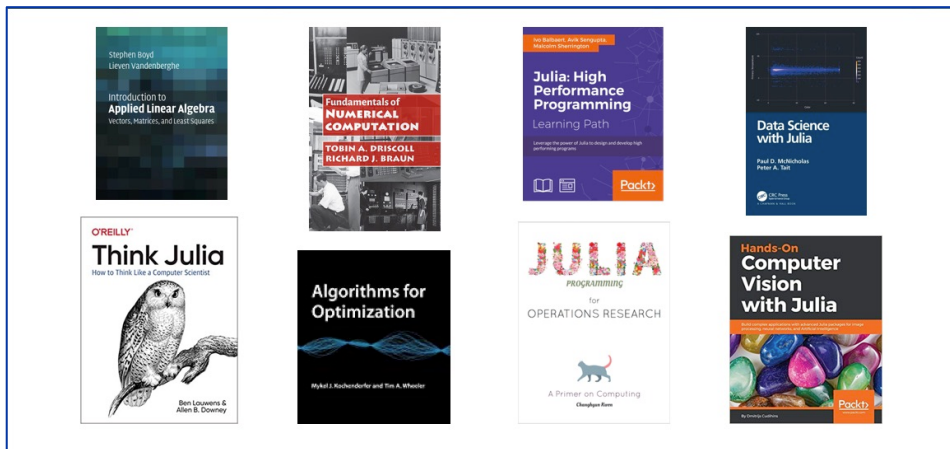
SHOULD WE LEARN JULIA?

- As per the industry, **julia** will be getting more popular as time progresses
- Usage of **julia** will be vital if we start using it. It is very high level and syntax is often similar to Python
- **julia** programming is certainly on the list of things that most of us should be learning as Statistical/Clinical Programmers

GETTING STARTED - JULIA

- **julia** installation is straightforward, whether using precompiled binaries or compiling from the source. Download and install it by following the instructions at <https://julialang.org/downloads/>
- If you are coming to **julia** from one of the following languages, then you should start by reading the section on noteworthy differences from [MATLAB](#), [R](#), [Python](#), [C/C++](#) or [Common Lisp](#). This will help you avoid some common pitfalls since it differs from these languages in many subtle ways

Below are the top viewed books when **julia** was created. There are many courses on it which are freely available on [YouTube](#). You can also sign up through [JuliaAcademy](#) to track your progress, take quizzes, and get certified.





The Global Healthcare Data Science Community

Contact Channels

UK +44 1843 609600
office@phuse.global
phuse.global

Social Media

@phusetwitta
/phusebook
/phusetube
/company/phuse

REFERENCES:

<https://www.infoworld.com/article/3241107/julia-vs-python-which-is-best-for-data-science.html>

<https://analyticsindiamag.com/basics-of-julia-programming-language-for-data-scientists/>

<https://juliacomputing.com/case-studies/pfizer/>

<https://juliacomputing.com/>

<https://www.hpcwire.com/2021/01/13/julia-update-adoption-keeps-climbing-is-it-a-python-challenger/>

<https://docs.julialang.org/en/v1/>

<https://www.youtube.com/watch?v=kPcJ4Xqxf8>

<https://www.infoworld.com/article/3241107/julia-vs-python-which-is-best-for-data-science.html>

<https://www.youtube.com/watch?v=kPcJ4Xqxf8>

<https://juliacomputing.com/case-studies/united-therapeutics/>

<https://docs.julialang.org/en/v1/manual/getting-started/>

<https://julialang.org/learning/>

DISCLAIMER: “This document is produced by consultants at Accenture as general guidance. This content is provided for general information purposes and is not intended to be used in place of consultation with our professional advisors. This document refers to marks owned by third parties. All such third-party marks are the property of their respective owners. No sponsorship, endorsement or approval of this content by the owners of such marks is intended, expressed or implied.”

Copyright © 2022 Accenture. All rights reserved.

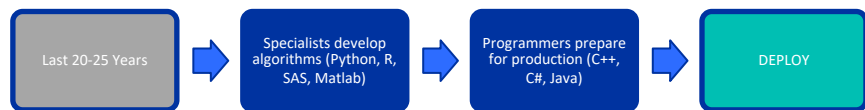


THANK YOU

APPENDIX

ORIGIN OF JULIA AND JOURNEY

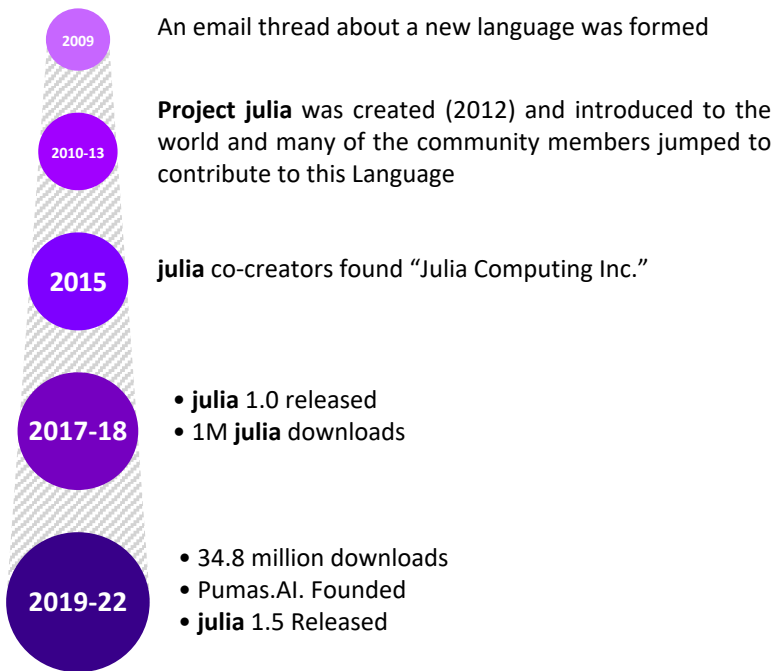
Why was julia created?



COMPRESS THE INNOVATION CYCLE WITH JULIA



JULIA JOURNEY





WHAT IS THE JULIA LANGUAGE?

julia aims to create an unprecedented combination of ease-of-use, power, and efficiency in a single language. In addition to the above details, some advantages of **julia** over comparable systems include:

- Free and open source ([MIT licensed](#))
- **User-defined types** are as fast and compact as built-ins
- Lightweight "green" threading
- Elegant **and extensible conversions** and promotions for numeric and other types
- **Call C functions directly** (no wrappers or special APIs needed)
- Powerful shell-like capabilities for managing other processes
- **Lisp-like macros** and other metaprogramming facilities

USAGE OF JULIA ACROSS THE WORLD

julia is being used across various industries to solve specific problems, enabling 10x to 100x performance improvements, massive scalability, significant cost savings, increase in productivity and decrease in time-to-market. It is being utilized in the below industries/organizations. Also, it is being taught in various universities.

ORGANIZATIONS



UNIVERSITIES





WHAT CAN JULIA PRODUCTS DO FOR PHARMACEUTICAL ANALYTICS?

Use data analytics to accelerate drug discovery and development

Improve efficiency of clinical trials

Target patient population more effectively

Gain insights into patient behavior

Safety and risk management

Marketing analytics and sales analytics