

# Julia 学习指北

Cheng Jun

2017 年 7 月 27 日

版本: 0.0.1

## 目录

1	说明	2
2	Julia 是什么?	2
3	为什么要关注 Julia?	3
4	怎么学习 Julia?	3
5	进一步了解 Julia	4

## 1 说明

- 编者信息

实证研究小青年，手艺正在提高中。

经管之余，泛读史哲，关注自由与开源动态。

GitHub: [chengjun90](#)

知乎: [Cheng Jun](#)

- 笔记说明

本系列笔记最早是分享在知乎专栏: [Julia 学习笔记](#)。在这里重新梳理和更新。

笔记是在 Jupyter Notebook 编写，导出 Tex 文件，经手动微调后编译成 pdf 文档。

In [2]: # 本系列笔记的 *Julia* 环境

```
versioninfo()
```

Julia Version 0.7.0-DEV.1122

Commit 892768cf52\* (2017-07-25 22:26 UTC)

Platform Info:

OS: Windows (x86\_64-w64-mingw32)

CPU: Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz

WORD\_SIZE: 64

BLAS: libopenblas (USE64BITINT DYNAMIC\_ARCH NO\_AFFINITY Haswell)

LAPACK: libopenblas64\_

LIBM: libopenlibm

LLVM: libLLVM-3.9.1 (ORCJIT, haswell)

Environment:

## 2 Julia 是什么?

Julia is the fastest modern open-source language for data science, machine learning and scientific computing. Julia provides the functionality, ease-of-use and intuitive syntax of R, Python, Matlab, SAS or Stata combined with the speed, capacity and performance of C, C++ or Java.

R、Python 和 Julia 是数据分析中的三个重要工具，在各自己的应用场景中发挥重要的作用。R 对数据分析，尤其是涉及到 `dataframe` 的理解和支持非常好，未来数据分析工具想在这个领域有所

作为必须从 R 中汲取经验和弥补不足，方能出彩。Python 可能在需要统一工具的场景下应用更广，因为 Python 毕竟是通用编程语言（这点不一定对）。在对运行速度有要求的场景下，Julia 的应用会更广。

### 3 为什么要关注 Julia?

从数据分析用户的视角来总结 Julia 的优点就是：

- 开源
- 开发效率高
- 运行效率高
- 商业支持

开源 MIT 协议意味着用户可以自由地安装，不用担心花费问题。在必要的时候可以阅读源代码，了解技术实现的细节，可以根据需要修改源代码。此外，现在数据分析越来越重视可重复性的问题，科学研究的要求之一就是结果可重复。使用 Julia 进行数据分析可以使得我们的数据分析是开放的和可重复的。

开发效率高。新兴语言，专为计算与分析而新设计。Julia 开发之初主要面向科学计算，函数可靠性较高。语法简洁，新手上手容易；支持函数式编程；强大的类型系统；支持元编程技术。

运行效率高。这个特点基本上是 Julia 的主打卖点之一，从目前来看，Julia 语言确实运行很快，在一定程度上解决两种语言的鸿沟。例如，R 语言想进一步提速就需要引入 C/C++。这也是为什么很多排名较高的 R 包基本离不开 C/C++。而作为 Julia 用户，在大多数时候可以用 Julia 来获取较快的速度，而无需依赖于 C/C++。

商业支持。Julia 创始人成立 JuliaComputing，可以提供更好的商业支持。便捷的云平台：Julia 官方提供 JuliaBox，微软的 Azure 支持 JuliaPro。

### 4 怎么学习 Julia?

目前最直接的方式，读 Julia 官方手册，而且是读[最新的手册](#)中的 Manual 部分。

因为目前 Julia 在不断的进化中，所以现有的 Julia 教程和教学资料已经相对陈旧，甚至可能前期积累的代码在新版本的 Julia 都不能正常运行了。

Julia 的第三方包也在不断的进化中，为此，学习和掌握包的使用，最好的方式就是读包的官方手册。

边读边实践边总结，是非常有效的学习方式。根据自己的工作和学习的需要，尝试着用 Julia 做一些实际应用，将会加速学习的进程。纸上得来终觉浅，绝知此事要躬行！

## 5 进一步了解 Julia

- [Why We Created Julia](#)
- [Julia motivation](#)
- [Getting our act together in the data ecosystem](#)