julia for research and teaching: first steps

Gary Froyland

What is julia ?

- Created in 2012; see julialang.org and the wikipedia page for details.
- Open-source, high-level (easy to program), high-performance language designed by mathematicians and computer scientists (and now a large scientific community).
- General-purpose language, excelling at scientific computing and data science.
- Mathematical conventions are embedded in clean, math-like syntax; mathematical symbols and Greek letters are also available.
- Won the 2019 SIAM J. H. Wilkinson Prize for Numerical Software.
- >40 million downloads and >8000 registered packages.

Last time

Introduction to Julia’s syntax, illustrating how Julia elegant and math-like expressions can be.
https://www.unsw.edu.au/science/our-schools/maths/engage-with-us/seminars/2022/julia-programming-language

Today’s topics

- matrix factorisations and sparse arrays
- Julia’s package manager
- plotting scalar-valued functions
- polynomial fitting and root finding
- random permutations and sampling
- statistical functions, statistical distributions, sampling and fitting
- creating and running a julia function
- timing and benchmarking a julia function
- reading and writing arrays to file
- reading matlab and R data files; saving and loading julia data files
- introduction to pluto notebooks (a reactive and interactive notebook)
- the sample notebook I used today is available at https://web.maths.unsw.edu.au/~froyland/sample_pluto_notebook.html

Packages from today

Linear algebra and sparse arrays
https://docs.julialang.org/en/v1/stdlib/LinearAlgebra/, https://docs.julialang.org/en/v1/stdlib/SparseArrays/
Plots
https://docs.juliaplots.org/latest/gallery/gr/, https://docs.juliaplots.org/latest/ecosystem/
Polynomials
https://juliamath.github.io/Polynomials.jl/stable/
Random numbers and a small selection of statistics packages
https://docs.julialang.org/en/v1/stdlib/Random/
https://docs.julialang.org/en/v1/stdlib/Statistics/, https://juliastats.org/StatsBase.jl/stable/
https://juliastats.org/Distributions.jl/stable/
Benchmarking
https://juliaci.github.io/BenchmarkTools.jl/stable/
Reading/writing data files
https://docs.julialang.org/en/v1/stdlib/DelimitedFiles/, https://csv.juliadata.org/stable/
https://github.com/JuliaIO/MAT.jl, https://github.com/JuliaData/RData.jl
https://github.com/JuliaIO/JLD.jl, https://juliaio.github.io/JLD2.jl/stable, https://juliaio.github.io/HDF5.jl/stable/

Full lists of packages
https://juliahub.com/ui/Packages
https://juliapackages.com/trending?sort=stars
Julia belongs to all of us – create a new package or add to an existing one.

Introductory and demo pages for julia

- Authored by me:
  - An *introductory guide to Julia for mathematics students*
  - **Pluto notebooks**: a sample demo, statistical distributions, the pendulum ODE, quadrature, dynamical systems

- basic installation/running info
https://julialang.org/learning/getting-started/

- cheatsheets
https://cheatsheets.quantecon.org/julia-cheatsheet.html
https://cheatsheets.quantecon.org/index.html
https://www.juliafordatascience.com/cheat-sheets/

- introduction to Julia
https://enccs.github.io/Julia-for-HPC/
https://enccs.github.io/Julia-for-HPC/syntax-intro/
https://jump.dev/JuMP.jl/stable/tutorials/getting_started/getting_started_with_julia/Getting-started-with-Julia
https://jump.dev/JuMP.jl/stable/tutorials/getting_started/getting_started_with_data_and_plotting/

- many demo **pluto** notebooks used as lecture notes at MIT https://computationalthinking.mit.edu/Fall22/
- **pluto** animations for massive particle simulations – a fancy pluto notebook https://m3g.github.io/2021_FortranCon/

- julia in 100 seconds by Jeff Delaney (Google Developer Expert) https://www.youtube.com/watch?v=JYs_94znYy0