Julia for Data Science

Tippie Graduate Analytics Workshop
University of Iowa

January 11, 2020

Wecome

Agenda 9:00AM - Noon

Introductory Example

Julia is Fast!

Data Visualization with Gadfly.jl

[break?]

DataFrames & Data Wrangling

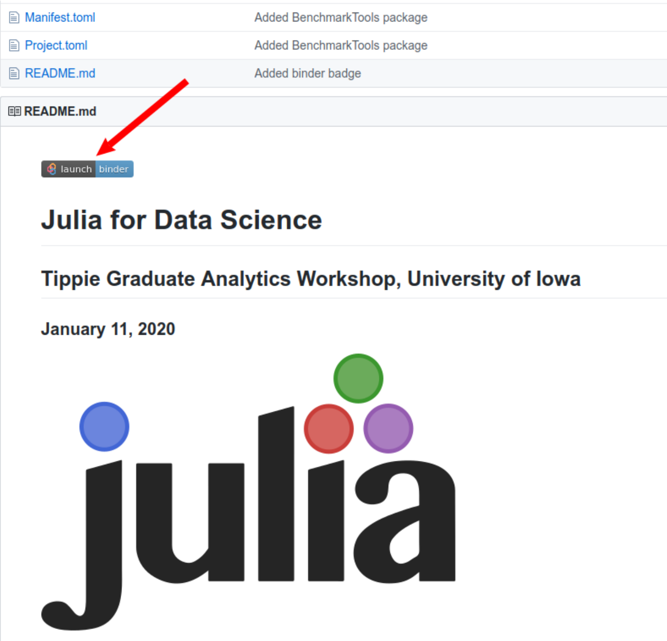
Exploratory Data Analysis (Intro)

Julia Language Features

Julia for Data Science

Everyone please do the following:

1. Nativate to https://github.com/davidbody/julia-workshop-jan-2020
2. Click on the "launch binder" badge in the README.



Manifest.toml	Added BenchmarkTools package
Project.toml	Added BenchmarkTools package
README.md	Added binder badge

README.md

Julia for Data Science

Tippie Graduate Analytics Workshop, University of Iowa

January 11, 2020

Introductory Example

In the Jupyter notebook open the following file:

00-eat-cake-first.ipynb

and follow the instructions.

We'll walk through the steps together.

Try changing one or both of the countries to any of the following:

```
Australia, Austria, Belgium, Canada, Czech Republic, Denmark, Estonia,
Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Israel,
Italy, Luxembourg, Netherlands, Norway, Poland, Slovakia, Slovenia,
Spain, Sweden, Switzerland, United Kingdom, United States
```

Welcome

Email: davidbody@bigcreek.com

Twitter: @david_body

Welcome

Workshop goals

- Introduce Julia
- Explore some data science workflows
- Overview of what makes Julia special
- Have fun

Welcome

Introductions

- Your name
- Why are you here?
- One other thing about you

Julia for Data Science

Agenda 9:00AM - Noon

~~Introductory Example~~

Julia is Fast!

Data Visualization with Gadfly.jl

[break?]

DataFrames & Data Wrangling

Exploratory Data Analysis (Intro)

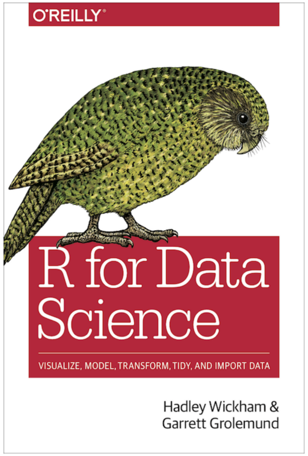
Julia Language Features

Logistics

- Breaks
- Rest rooms
- Lunch

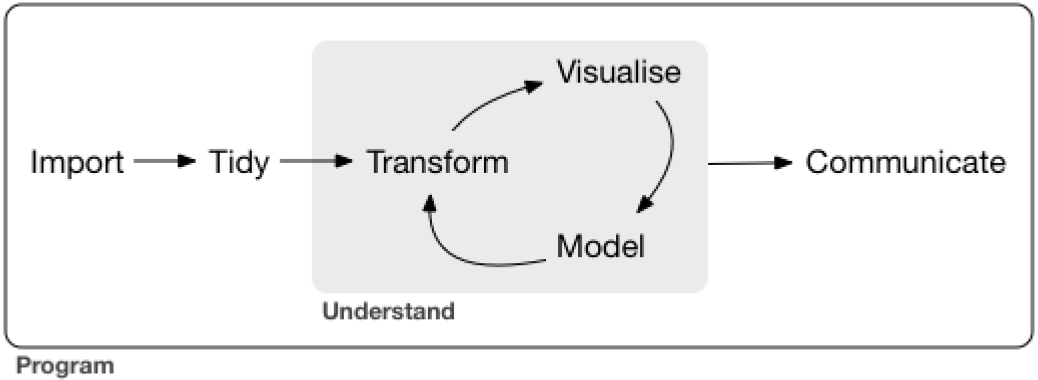
Julia for Data Science

This workshop is based in part on the execellent book *R for Data Science* by Hadley Wickham and Garrett Grolemund.



Available free online at https://r4ds.had.co.nz/

Data science workflow



from *R for Data Science*.

Today we'll just scratch the surface of "Transform" and "Visualize".

We will also mostly skip exploratory data analysis so we can focus on more Julia language features.

Julia is Fast!

Riddler Express from FiveThirtyEight

https://fivethirtyeight.com/features/so-you-want-to-tether-your-goat-now-what/

> From Luke Robinson, a serenading stumper:
>
> My daughter really likes to hear me sing “The Unbirthday Song” from “Alice in Wonderland” to her. She also likes to sing it to other people. Obviously, the odds of my being able to sing it to her on any random day are 364 in 365, because I cannot sing it on her birthday. The question is, though, how many random people would she expect to be able to sing it to on any given day before it became more likely than not that she would encounter someone whose birthday it is? In other words, what is the expected length of her singing streak?

Demo - Python vs Julia

Julia is Fast!

Riddler Express from FiveThirtyEight

Aside: The problem is asking two different questions. The minimum number of random people that would make it more likely than not the girl would encounter someone whose birthday it is is the **median**, while the expected length of the singing streak is the **mean** of the distribution of singing streak lengths. We only calculate the **mean**.

Math is faster!

Second Aside: The Geometric distrubution is the probability distribution of the number Y of failures of Bernoulli trials before the first success. The probability mass function for the Geometric distribution is

$$\Pr(Y = k) = (1 - p)^k p$$

for k = 0, 1, 2, 3, …. where p is probability of success for each Bernoulli trial.

The mean of the Geometric distribution is

$$E(Y) = \frac{1 - p}{p}$$

In our case, p is the probability that a random person we encounter has a birthday today, so

$$p = \frac{1}{365}$$

and therefore

$$\begin{aligned} E(Y) &= \frac{1 - \frac{1}{365}}{\frac{1}{365}} \\ &= 365 - 1 \\ &= 364 \end{aligned}$$

Why does Julia sometimes feel slow?

But we've already seen that Julia can *feel* slow when used interactively. Packages can be slow to load, and functions can be slow the first time they are called. What's happening is that Julia code is **compiled "just in time"** and that compilation can take a little time. Julia's maintainers are well aware of the issue of **"compilation latency"** and plan to address it in the future.

In the meantime, everyone will have to decide for themselves if the tradeoffs that Julia makes are worth it for a particular application.

Data Visualization with Gadfly.jl

- Gadfly.jl is a Julia package
- Gadfly.jl is based on `ggplot2`, which is based on a *grammar of graphics*
- We'll cover examples of making plots with Gadfly.jl
- Then it will be your turn to make some plots

Follow along in **notebooks/01-data-visulzation.ipynb** if you want

Your turn

15:00

Data Visualization with Gadfly.jl

your-turn/01-data-visualization.ipynb

DataFrames & Data Wrangling

- DataFrames.jl is a Julia package
- Missing values
- Basic operations:
  - Filtering rows
  - Selecting columns
  - Adding and modifying columns
  - Sorting
  - Performing calculations on all rows by groups of rows
- Query.jl package
- Your turn to work with DataFrames

Follow along in **notebooks/02-data-wrangling.ipynb** if you want

Your turn

15:00

DataFrames & Data Wrangling

your-turn/02-data-wrangling.ipynb

Exploratory data analysis

- Just an introduction to EDA
- Primary example: Anscombe's Quartet

Key take-away: alawys plot your data

Follow along in **notebooks/03-exploratory-data-analysis.ipynb** if you want

Julia Language Features

- Multiple dispatch (parametric polymorphism)
- Dynamic type system ("optional" typing)
- High performance (approaching C, Fortran, etc.)
- Built-in package manager
- Lisp-like macros and metaprogramming
- Interoperability with Python, R, C, Fortran
- Designed for parallel and distributed computing

Follow along in **notebooks/04-julia-language-features.ipynb** if you want

What we covered today

- Initial example using Jupyter Notebooks
- Julia is fast! (so is math)
- Data visualization with Gadfly.jl
- DataFrames & Data Wrangling
- Intro to Exploratory Data Analysis
- Julia Language Features
    - High performance
    - Multiple dispatch
    - "Optional" typing
    - Metaprogramming

Thank you!

Questions?