

How to Setup Julia Locally

David Zeng Keegan Go Stephen Boyd

EE263
Stanford University

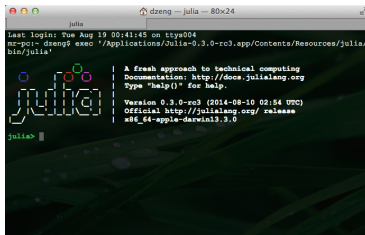
October 1, 2015

Installing Julia

- ▶ download Julia v0.3.11 from <http://julialang.org/downloads/>
- ▶ follow the on-screen instructions to install

The Julia terminal

- ▶ environment to run snippets of Julia code
- ▶ open the Julia application after installation
- ▶ you should see something like this:

A screenshot of a terminal window titled "julia" with a window size of 80x24. The terminal shows the following text:

```
Last login: Tue Aug 19 00:41:45 on tty004
sz-pc:~ dseng$ exec '/Applications/Julia-0.3.0-rc3.app/Contents/Resources/julia/bin/julia'
```

The terminal output displays the Julia logo, which is the word "julia" in a stylized font with colored dots above the letters. To the right of the logo, the following text is displayed:

```
| A fresh approach to technical computing
| Documentation: http://docs.julialang.org
| Type "help()" for help.
|
| Version 0.3.0-rc3 (2014-08-10 02:54 UTC)
| Official http://julialang.org/ release
| x86_64-apple-darwin13.3.0
```

At the bottom left of the terminal, the prompt "julia>" is visible with a cursor.

- ▶ try running a few Julia commands in the terminal

Suppressing output

- ▶ running a command in the Julia terminal will automatically print its output
- ▶ turn off output by ending a line with ;
value = 3
value = 3;

Running Julia scripts

- ▶ the Julia terminal can run files with Julia code
- ▶ use the command `pwd()` to see which folder you are currently in
- ▶ use the command `cd` to change folders
`cd("Documents/ee263")`
- ▶ run a file using the `include` command
`include("testfile.jl")`

Packages

- ▶ code contributed by the community that is not part of the basic installation, *e.g.*, plotting
- ▶ install an official Julia package with the `Pkg.add` function, *e.g.*, to install the plotting package Gadfly, simply specify the name
`Pkg.add("Gadfly")`
- ▶ to use the code in a package, use the `using` command
`using Gadfly`
- ▶ try plotting some points!
`x_values = 0:0.1:10`
`plot(x=x_values, y=sin(x_values), Geom.point)`

Updating packages

Packages are modified by the Julia community from time to time. It's a good idea to update your packages every now and then to make sure you have the latest versions. You can update all packages with one quick command:

```
Pkg.update()
```

Installing LinearLeastSquares

LinearLeastSquares is a package you will be using starting around week 5 of the course

1. ensure you have the most updated list of available packages

```
Pkg.update()
```

2. install LinearLeastSquares

```
Pkg.add("LinearLeastSquares")
```


Configuring the juliarc file

The juliarc file is usually a small file with a few lines of julia code that runs automatically each time you open the Julia application

1. open Julia and run the function: `homedir()`. The output of this function is your home directory
2. open your favorite text editor: Notepad, Sublime, etc.
3. you can make Julia always open in a certain folder, such as your EE263 folder, by typing out a line like the following in the text editor:

```
cd("/Users/dzeng/Documents/ee263")      # Mac  
cd("C:/Users/Keegan/My Documents/ee263") # Windows
```

4. save the file in your home directory as `.juliarc.jl`
5. close and reopen Julia. If you run `pwd()`, it should now return the folder you specified in step 3