

Org-mode and julia: an introduction

G. Jay Kerns

February 24, 2013

Contents

1	What you need to get started	1
1.1	Julia	2
1.2	Add on packages	2
1.2.1	Winston	2
1.2.2	Gadfly	2
1.3	ESS	2
1.4	Org-mode	3
2	Getting started	3
3	Plotting with Winston	4
4	Plotting with Gadfly	5
5	Fitting linear models	5

1 What you need to get started

```
(load "~/gitm/projects/ob-julia.el")
(org-babel-julia-initiate-session "*julia*" nil)
```

```
rand(9)
```

```
9-element Float64 Array:
```

```
0.627821
0.786056
0.851449
0.447721
```

```
0.722442
0.447876
0.578059
0.132425
0.338724
```

1.1 Julia

- First install takes the longest, later updates not so bad.
- all the dependencies

1.2 Add on packages

Based on The State of Statistics in Julia by John Myles White.

```
Pkg.add("DataFrames", "Distributions", "MCMC", "Optim", "NHST", "Clustering")
Pkg.add("RDatasets")
```

This one is pretty big.

1.2.1 Winston

```
Pkg.add("Winston")
```

This one has lots of dependencies.

1.2.2 Gadfly

```
Pkg.add("Gadfly")
```

- packages take a lot longer to load than R

1.3 ESS

<http://stat.ethz.ch/ESS/index.php?Section=download>

```
rand(9)
print("hello")
```

9-element Float64 Array:

```
0.294536
0.292205
0.0626337
0.435547
0.830883
0.298874
0.286572
0.539265
0.855514
```

hello

```
(add-to-list 'load-path "~/path/to/ESS/lisp")
(require 'ess-site)
```

1.4 Org-mode

```
(setq inferior-julia-program-name "~/path/to/julia-release-basic")
```

```
(org-babel-do-load-languages
 'org-babel-load-languages
 '((emacs-lisp . t)
  (julia . t)))
```

For inline image display

```
(add-hook 'org-babel-after-execute-hook 'org-display-inline-images)
(add-hook 'org-mode-hook 'org-display-inline-images)
```

2 Getting started

```
[3:50]
x = [3:6]
y = [5:11]
z = [4:7]
x + z
ans
sum(x)
mean(x)
exp(x)
```

```

log(x)
sort(x)
sqrt(x)
diff(x)
x[3]
y[2:4]
typeof(x)
1.0
1.
Inf
-Inf
NaN
true
false
1 + 2im

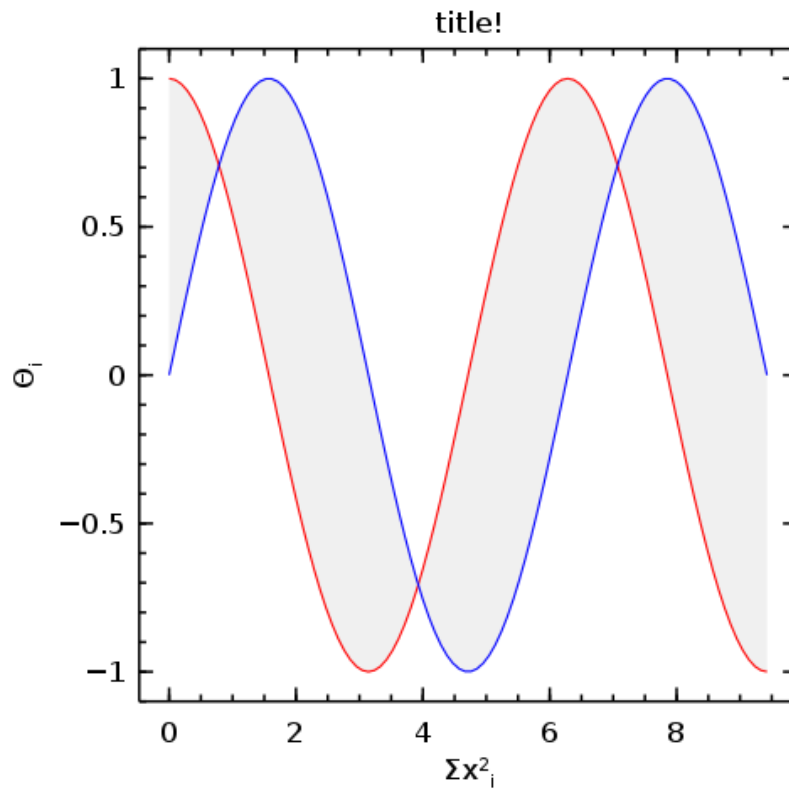
```

3 Plotting with Winston

```

using Winston
x = linspace( 0, 3pi, 100 )
c = cos(x)
s = sin(x)
p = FramedPlot();
setattr(p, "title", "title!")
setattr(p, "xlabel", L"\Sigma x^2_i")
setattr(p, "ylabel", L"\Theta_i")
add(p, FillBetween(x, c, x, s) )
add(p, Curve(x, c, "color", "red") )
add(p, Curve(x, s, "color", "blue") )
file(p, "blah2.png")

```



4 Plotting with Gadfly

```
using RDatasets
using Gadfly
using Compose
iris = data("datasets", "iris")
p = plot(iris, {:x => "Sepal.Length", :y => "Sepal.Width"}, Geom.point);
SVG("iris_plot.svg", 6inch, 4inch)
```

5 Fitting linear models