Scientific Computing in Economics with Python and Julia

A Workshop at the Federal Reserve Bank of Chicago

Introduction

May 2016

Big thanks to

FEDERAL RESERVE BANK of CHICAGO

and

Introduction

- Agenda
- Personnel
- Topics
- Motivation
- Resources

Agenda

1. Monday morning: Introduction to Julia
2. Monday afternoon: Introduction to Julia, continued
3. Tuesday morning: Introduction to Python
4. Wednesday morning: Dolo
5. Wednesday afternoon: Pandas

Personnel

- Chase Coleman (NYU)
- Matthew McKay (ANU, NYU)
- Viral Shah (Julia Computing)
- John Stachurski (ANU, NYU)
- Pablo Winant (Bank of England)

What's Python?

Modern, high level, open source, general purpose programming language

What's great about Python:

- Beautifully designed
- High productivity
- Extremely widely used
- Great scientific tools

What's Julia?

Modern, high level, open source, scientific programming language

What's great about Julia:

- Great design
- High productivity...
- and high performance!
- Great scientific tools

Which to Choose?

Who will benefit more from Julia?

- Focused on scientific programming
- Write your own algorithms
- Need optimization / high performance

Who will benefit more from Python?

- More diverse coding needs
- Require more stability and high productivity

Wednesday's Sessions

Dolo, by Pablo Winant

- A modeling language
- Solvers
- In Python, but a Julia version is on the way

Pandas, by Matt McKay

- High performance data analysis tool
- Finance industry, AI, data science

Resource

General info:

- `http://quantecon.org/chicago_fed_workshop.html`

Downloads:

- `https://github.com/QuantEcon/ChicagoFed_workshop`

Entry point at the bottom of this page:

- `http://quantecon.org`