EuroScipy 2013 - NumPy and IPython

Valentin Haenel

Freelance Consultant and Software Developer
http://haenel.co

21 Aug 2013

Version: Scipy2013-7-gf08e83f8d1-dirty http://git.io/0zq7dw

Materials

- GitHub Repository: http://git.io/0zq7dw
- Materials ZipFile: http://git.io/chxjqQ

- Anaconda: https://store.continuum.io/

- In case of no network: grab one of the three USB-Keys
  - They read *Bruker*

About The Speaker

- Valentin Haenel from Berlin, Germany
- Freelance software developer and consultant
- Specialise in Git consulting and scientific software tooling

The Scientists Needs

- Acquire data
  - Simulation
  - Experiment
- Manipulate and process that data
- Visualize results
- Communicate results
  - Produce figures for reports or publications
  - Write presentations.

Why Python

- Easy to learn, easy to read, easy to maintain
- Thriving ecosystem of scientific libraries
- Vibrant community
- Numpy and IPython
- Commercial support

The Scientific Python Ecosystem

- Numpy
- IPython
- Scipy
- Matplotlib

- Pandas
- Sympy
- Scikits-Learn
- PyTables
- Cython

About this Tutorial

- **Ipython** (45 Min)
  - Using the IPython notebook
  - Help system, magic functions, aliases and history
- **Numpy** (2:15 Hours)
  - Basic arrays, dtypes and numerical operations
  - Indexing, slicing, reshaping and broadcasting
  - Copies, views and fancy indexing

- The tutorial will feature short bursts of small exercises every 5-10 minutes.
- Some of the tutors from the other tutorials are here to help.
- We can have a break in the middle.

About the Material

- **Ipython**
  - An IPython notebook demonstrating the IPython notebook
  - A demo session of the IPython shell
- **Numpy**
  - Two IPython notebooks
  - (Semi-)Automatically converted from Python Scientific Lecture Notes
  - The generated HTML is included in the GitHub Repository / Zip file and available online

How to Follow

- Grab the IPython notebook, try out the examples, work on the exercises, all from within the notebook.

- Alternatively: view the HTML and copy and paste the examples into an IPython shell or a Python file.

About IPython

- De facto Python interpreter with bells and whistles
- Since 2011: available in the browser as *IPython notebook*:

```
$ ipython notebook --pylab=inline
```

- The URL to access the notebook will be printed
- Numpy and Matplotlib tools are available (pylab mode)
- Plots will be displayed inside the notebook (inline mode)