Creating data visualizations with Python

Workshop @ Budapest BI forum 2013

Almar Klein

A bit of history

- Developed from 1989 by Guido van Rossum
- Adoption by scientists around 1995
- Scipy package around 2001

A bit of history

- Plotting: Matplotlib
- 3D visualization: VTK / Mayavi

Current list of viz toolkits

From http://www.scipy.org/topical-software.html

- matplotlib
- Chaco
- PyQwt
- HippoDraw
- Biggles
- Gnuplot.py
- Graceplot
- OpenCV
- PyChart
- Pygame
- PyNGL
- Veusz

- Mayavi2
- visvis
- Py-OpenDX
- Py-2DX
- IVuPy
- Pyvi
- Mat3D
- S2PLOT
- pyqtgraph

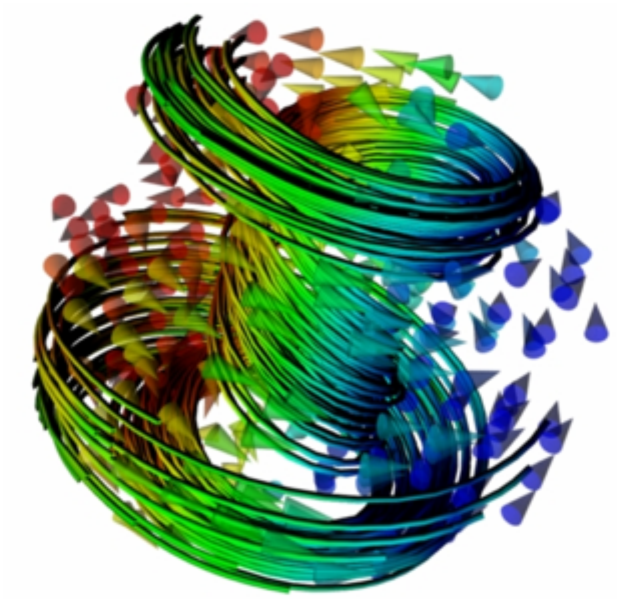
Matplotlib

```
import matplotlib as mpl
import matplotlib.pyplot as plt
from matplotlib.pylab import *
```

- Active development team
- Advanced 2D plotting
- Minor support for 3D
- Vector image output
- Quite slow for interactive use

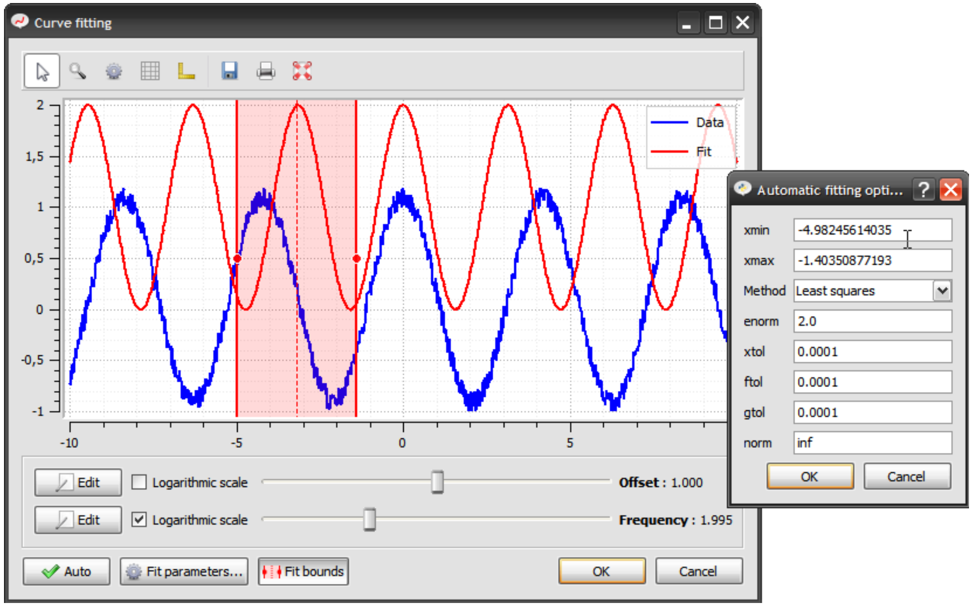
Mayavi2

- Developed by Enthought / Gael Varoquaux
- Based on VTK
- Lots of C++ for bindings
- Therefore hard to maintain
  - Development has more or less stalled



Guiqwt

- Developed by Pierre Raybout
- Qt (Qwt) + Python

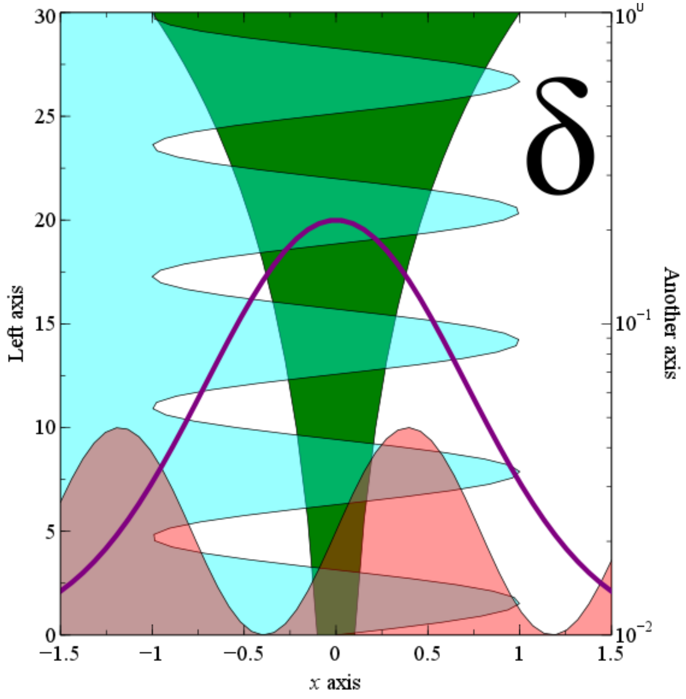


xmin -4.98245614035
xmax -1.40350877193
Method Least squares
enorm 2.0
xtol 0.0001
ftol 0.0001
gtol 0.0001
norm inf

□ Logarithmic scale **Offset** : 1.000
☑ Logarithmic scale **Frequency** : 1.995

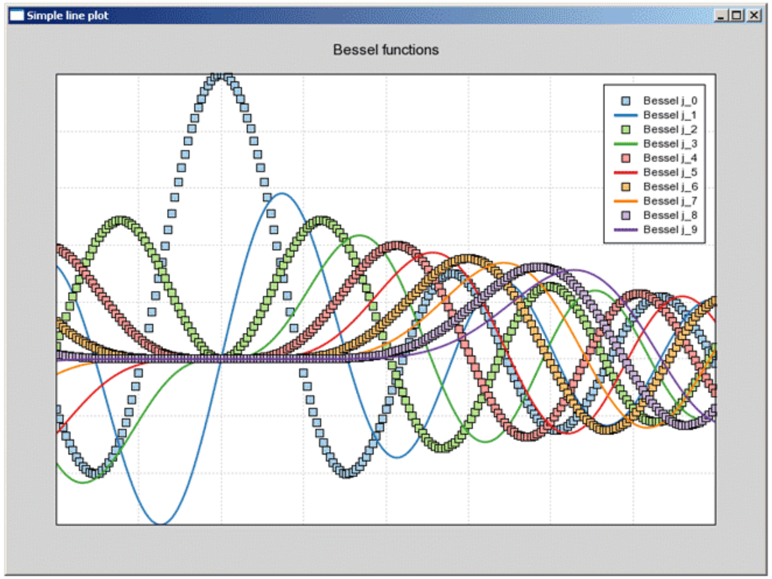
Veusz

- Developed by Jeremy Sanders
- Qt + Python
- Export vector graphics



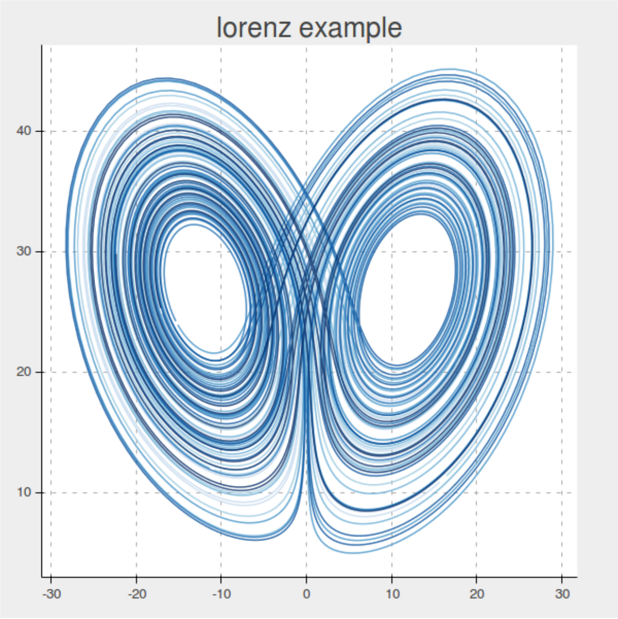
Chaco

- Developed by Enthought / Peter Wang
- 2D plotting
- Plotting applications



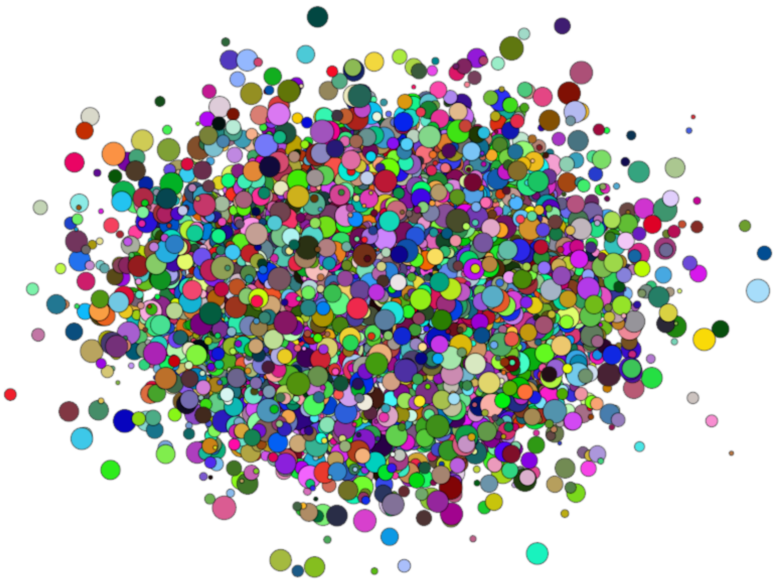
Bokeh

- Developed by Peter Wang
- Young project
- Interactive plotting for big data
- Web backend



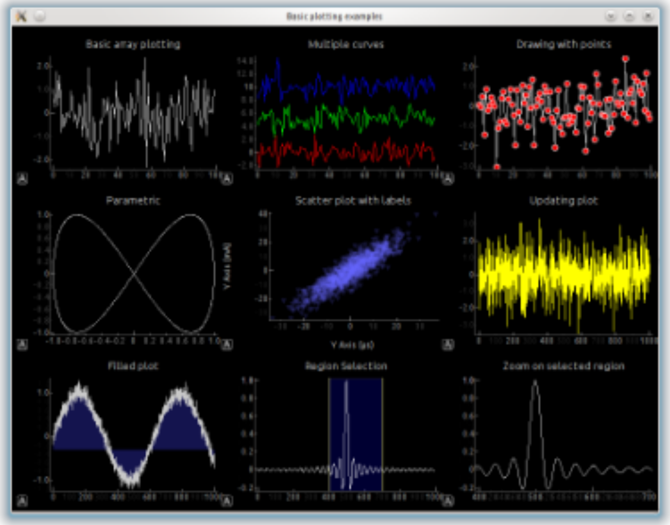
Vispy

- New visualization toolkit
- Will replace Visvis, PyQtGraph, Galry, Glumpy
- Talk tomorrow



PyQtGraph

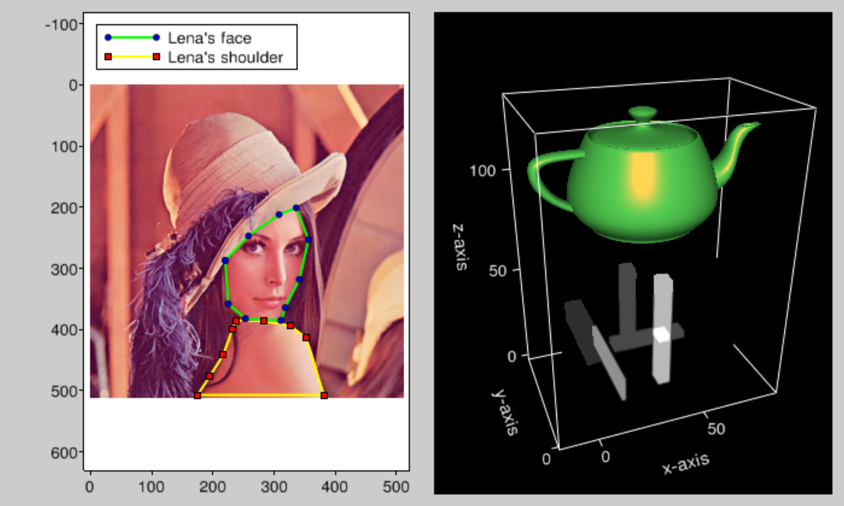
- Developed by Luke Campagnola
- Python + Qt + OpenGL
- Basic plotting
- Plot lots of updating plots in realtime
- 3D visualization
- Data selection and interaction



Visvis

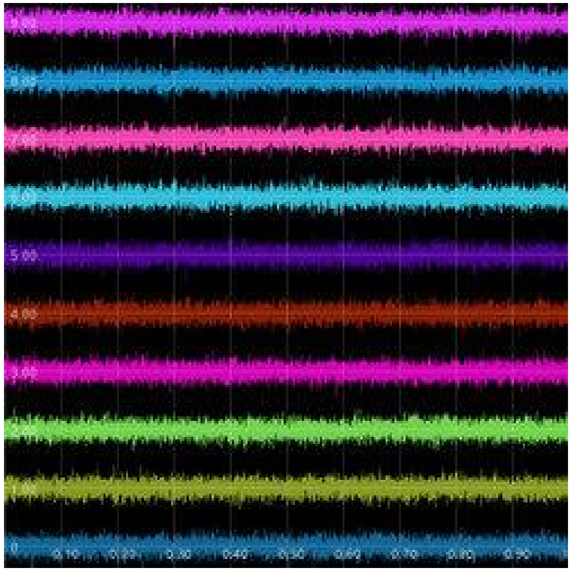
```
import visvis as vv
```

- Developed by Almar Klein
- OpenGL + pure Python
- Basic plotting
- Real time interaction
- Animation
- 3D (e.g. volume rendering)



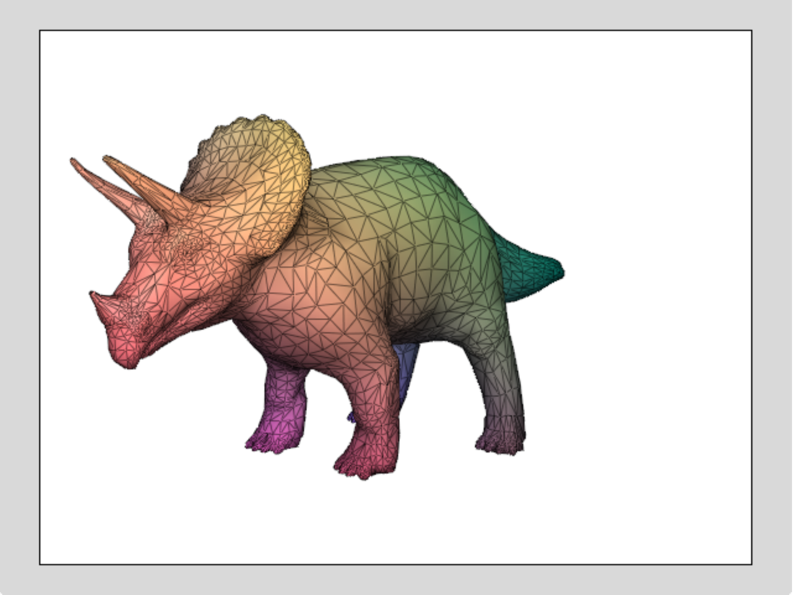
Galry

- Developed by Cyrille Rossant
- High performance
- Even with 100's of live-updating plots



Glumpy

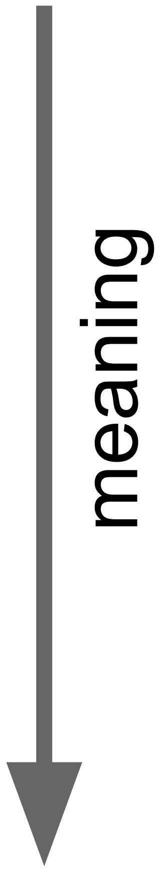
- Developed by Nicolas Rougier
- OpenGL + Python
- Fast and pretty



Data representation

Levels of representation

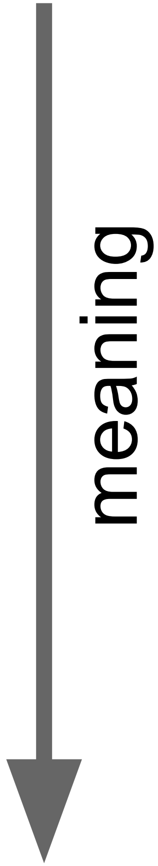
- a collection of bytes
- data types that the language provides:
  array, list, float, int, string, ...
- what the data represents mathematically:
  image, point, surface, motion field, ...
- what the data really represents:
  a sound, heart, lung, demographic data, ...



meaning

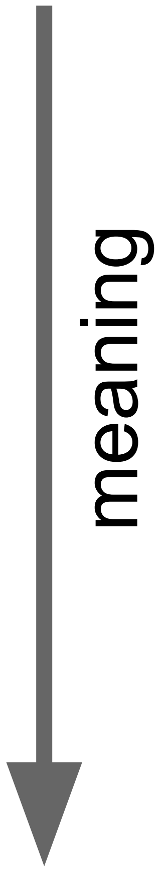
Levels of representation - heart 1

- a collection of bytes
- a 3D numpy array
- a mask (boolean image)
- segmentation of the heart



Levels of representation - heart 2

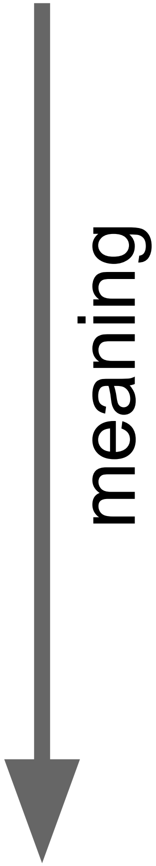
- a collection of bytes
- C-pointer + int
- pointset
- segmentation of the heart



meaning

Levels of representation - heart 3

- a collection of bytes
- 2 java arrays: Nx3 vertex array, Mx3 face array
- a surface (triangulated mesh)
- segmentation of the heart



Different usage of arrays

- 1D signal
- image
- set of points
- a line
- surface mesh (2 or more arrays)
- vector field
- ...

Table of representations

Regular grid	1D signal	2D image	3D image (volume)	4D
Points / vertices	1D points (list of scalars)	2D Points	3D Points	4D Points
Line		Line in 2D (contour)	Line in 3D	Line in 4D
Plane			Plane in 3D Surface	Hyper plane
				?

Other possible representations

- Graph (inter-connected points)
- Vector field
- Motion field volume
- Color image
- Color volume
- ...

The only limit is your imagination ...
... and your sense of style and efficiency.

Example: color image

- **NxMx3 array**
- 3xMxN array
- 3 MxN arrays
- ...

Representation & Visualization

When to use plot()

Regular grid	🟩 1D signal	🟦 2D image	🟦 3D image (volume)	🟦 4D
Points / vertices	🟩 1D points (list of scalars)	🟩 2D Points	🟩 3D Points	🟦 4D Points
Line		🟩 Line in 2D (contour)	🟩 Line in 3D	🟦 Line in 4D
Plane			🟦 Plane in 3D Surface	🟦 Hyper plane
				🟦 ?

When to use imshow()

Regular grid	1D signal	2D image	3D image (volume)	4D
Points / vertices	1D points (list of scalars)	2D Points	3D Points	4D Points
Line		Line in 2D (contour)	Line in 3D	Line in 4D
Plane			Plane in 3D Surface	Hyper plane
				?

When to use volshow()

Regular grid	1D signal	2D image	3D image (volume)	4D
Points / vertices	1D points (list of scalars)	2D Points	3D Points	4D Points
Line		Line in 2D (contour)	Line in 3D	Line in 4D
Plane			Plane in 3D Surface	Hyper plane
				?

When to use mesh()

Regular grid	1D signal	2D image	3D image (volume)	4D
Points / vertices	1D points (list of scalars)	2D Points	3D Points	4D Points
Line		Line in 2D (contour)	Line in 3D	Line in 4D
Plane			Plane in 3D Surface	Hyper plane
				?

Tutorials

Get coding!

Tutorial files on vispy.org/biforum.html