

# Obliczenia Naukowe

Kreślenie wykresów w pythonie,  
Czyli o bibliotece matplotlib

Bartek Wilczyński  
[bartek@mimuw.edu.pl](mailto:bartek@mimuw.edu.pl)

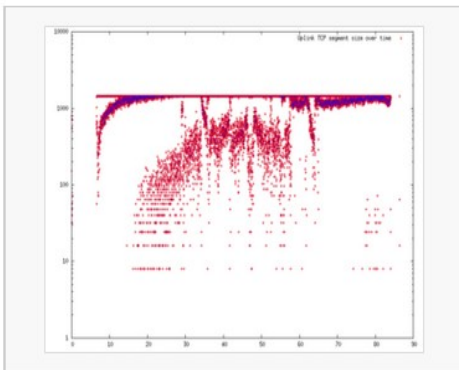
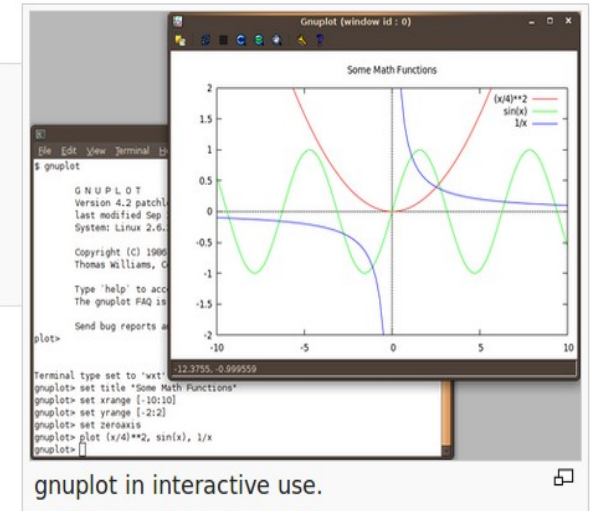
11. marca 2019

# GNUplot – jedno z pierwszych narzędzi, wcale nie GNU, ale opensource

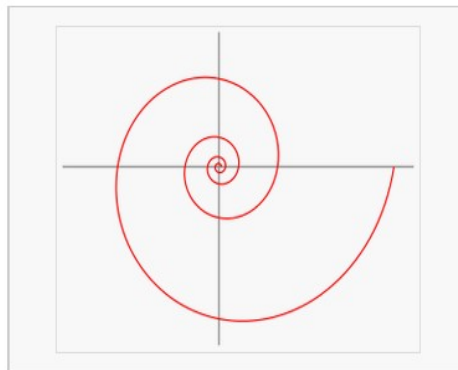
The gnuplot core code is programmed in [C](#). Modular subsystems for output via Qt, wxWidgets, and LaTeX/TikZ/ConTeXt are written in C++ and lua.

The code below creates the graph to the right.

```
set title "Some math functions"
set xrange [-10:10]
set yrange [-2:2]
set zeroaxis
plot (x/4)**2, sin(x), 1/x
```



A [scatter plot](#) of samples from a text file.



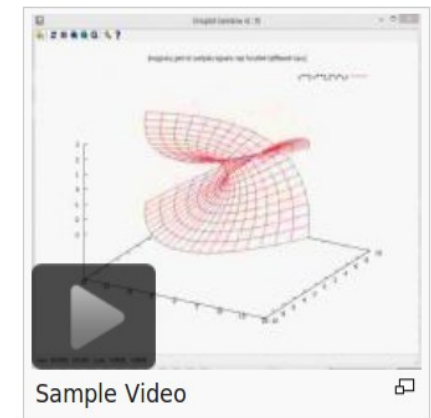
A logarithmic spiral.

The name of this program was originally chosen to avoid conflicts with a program called "newplot", and was originally a compromise between "llamaplot" and "nplot".<sup>[4]</sup>

## Distribution terms [\[ edit \]](#)

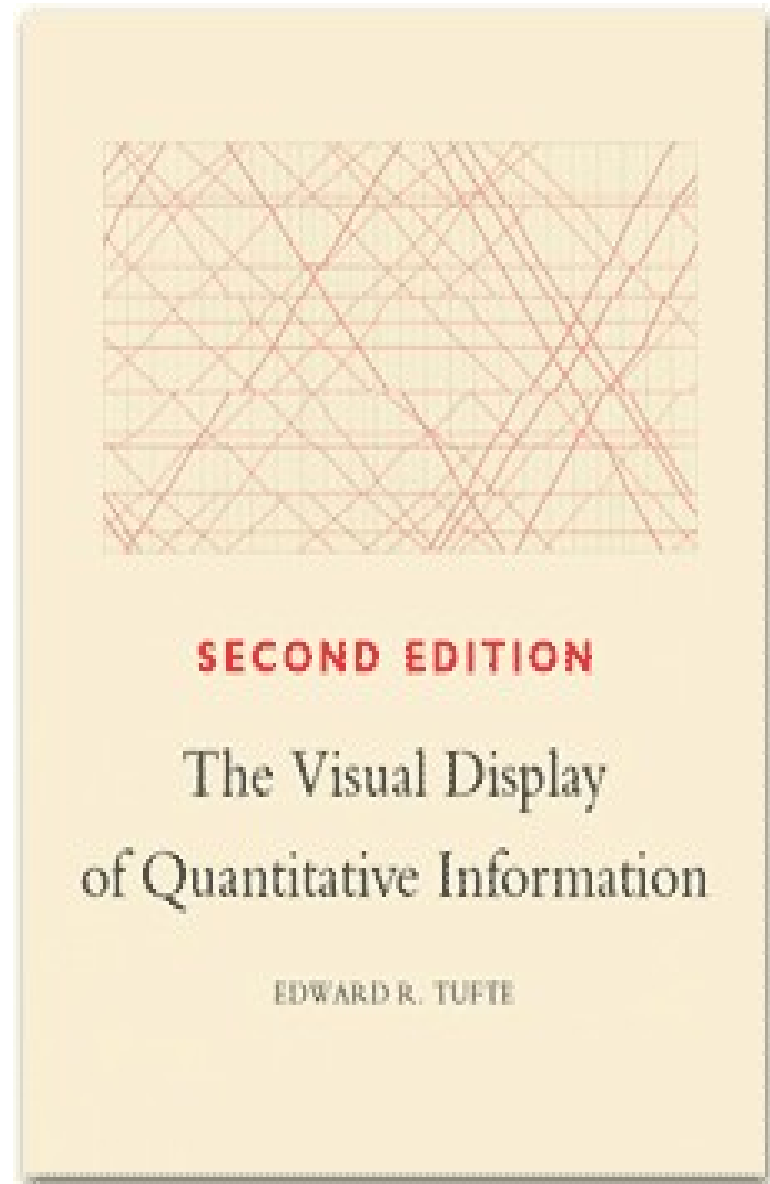
Despite gnuplot's name, it is not named after, part of or related to the [GNU Project](#), nor does it use the [GNU General Public License](#). It was named as part of a compromise by the original authors, punning on *gnu* (the animal) and *newplot*.<sup>[5]</sup>

Official source code to gnuplot is freely redistributable, but modified versions thereof are not. The gnuplot license instead recommends distribution of patches against official releases, optionally accompanied by officially released source code. Binaries may be distributed along with the unmodified source code and any patches applied thereto. Contact information must be supplied with derived works for [technical support](#) for the modified



# Edward R. Tufte – wizualizacja musi być estetyczna i nie kłamać

- Opublikował The Visual Display of Quantitative Information w 1993 (ponad 20 “dodruków”, dwa wydania)
- Spopularyzował pojęcia “lie factor” i “data-ink ratio”
- Zwalcza generyczne wykresy z Excela i powerpointa (np. Pie chart)



# Lie Factor

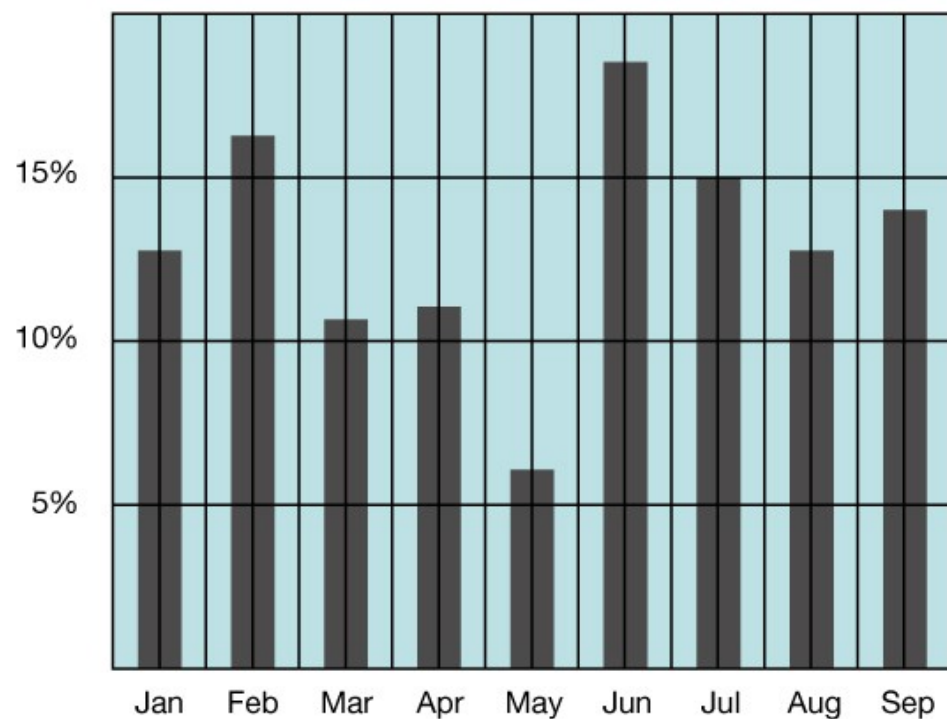
- Współczynnik tego jak bardzo proporcje na wykresie różnią się od rzeczywistych proporcji w danych
- Tutaj np. Mamy rzeczywisty spadek z 8023 na 6212, gdy infografika przedstawia mniej więcej 6-krotne zmniejszenie “lekarza”



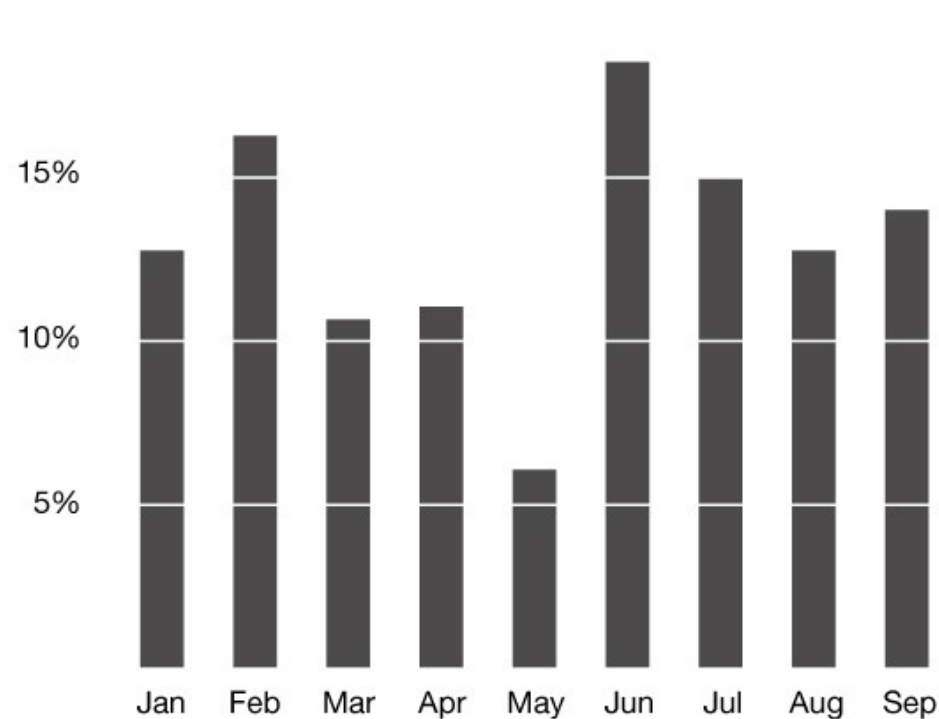
# Data-ink ratio

- Stosunek tuszu koniecznego do przedstawienia danych na wykresie do tuszu faktycznie użytego

LOW INK-DATA RATIO



HIGH INK-DATA RATIO



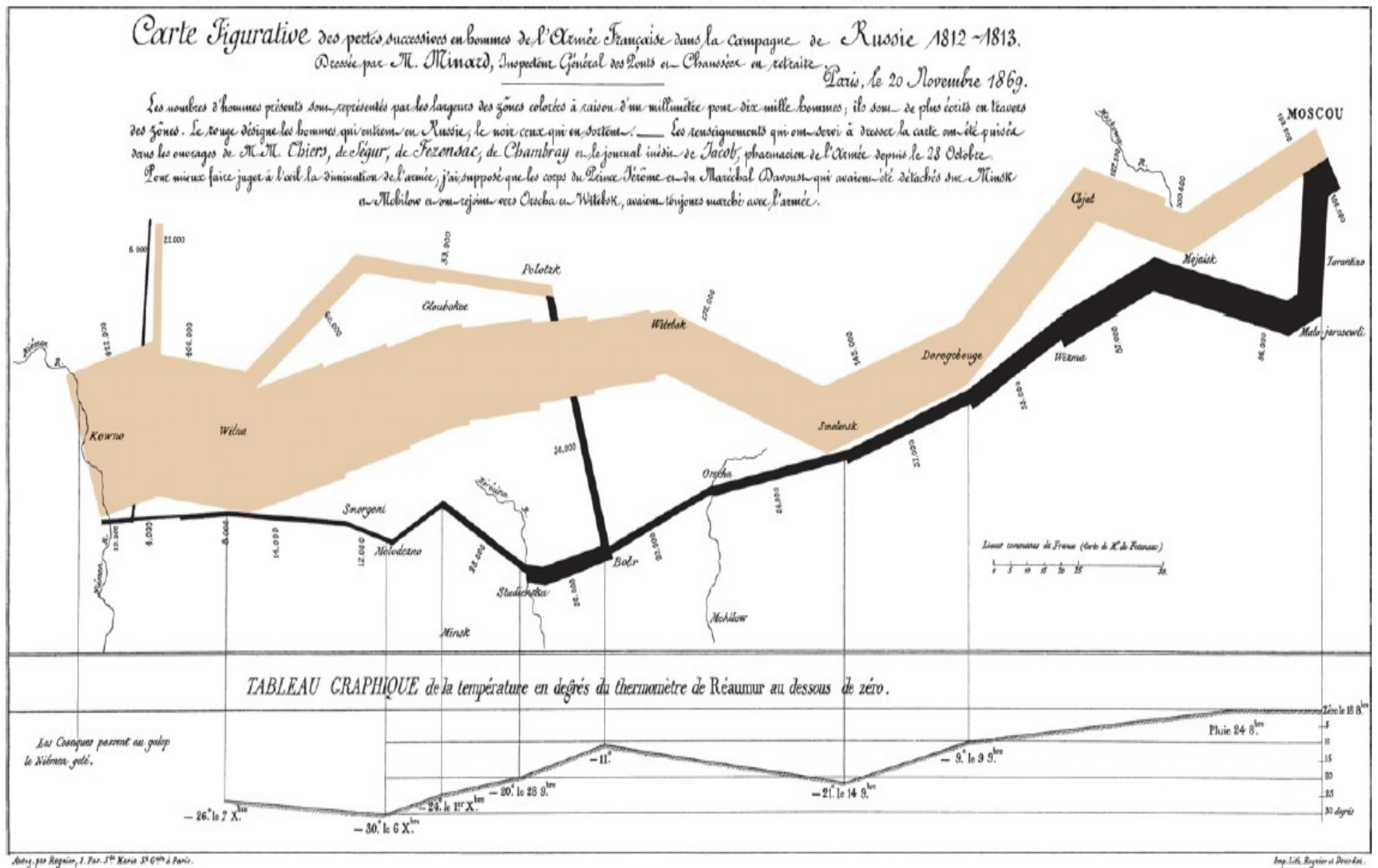
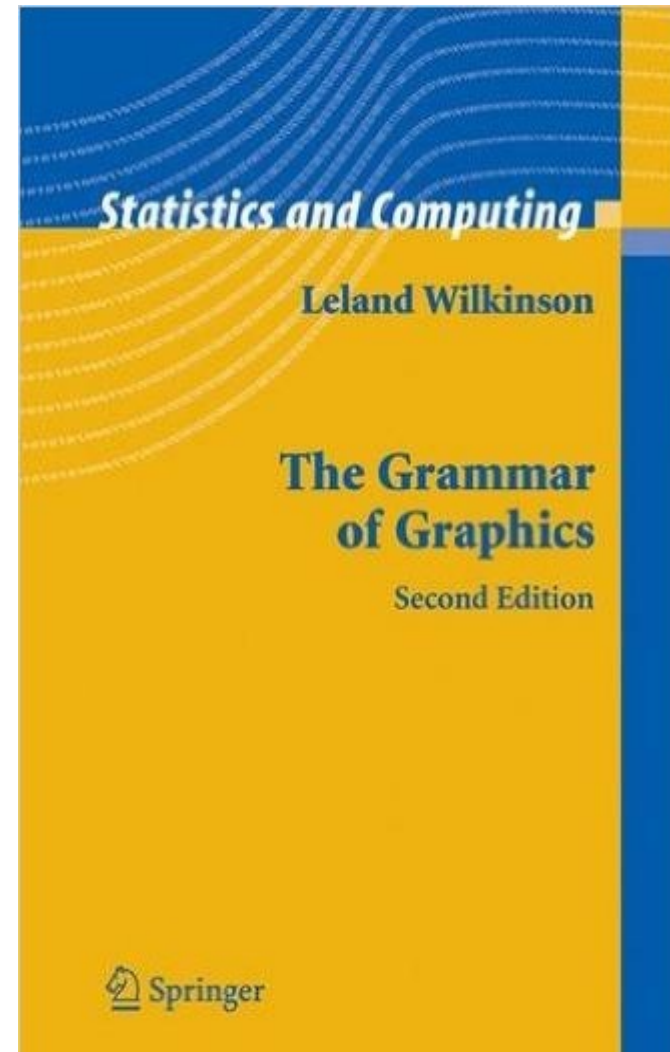


Figure 11. “Carte figurative des pertes successives en hommes de l’Armée Française dans la campagne de Russie 1812–1813” by Charles Joseph Minard. Public domain image from <http://en.wikipedia.org/wiki/File:Minard.png>.



# L. Wilkinson – the Grammar of graphics

- Tutaj nieco inna historia,
- Pracownik uniwersytetu I firm komercyjnych (m.in. SPSS)
- Autor książki the grammar of graphics (1999), i wzorowanego na niej systemu nVizN
- Rozpoznawalny tylko w kręgach specjalistów



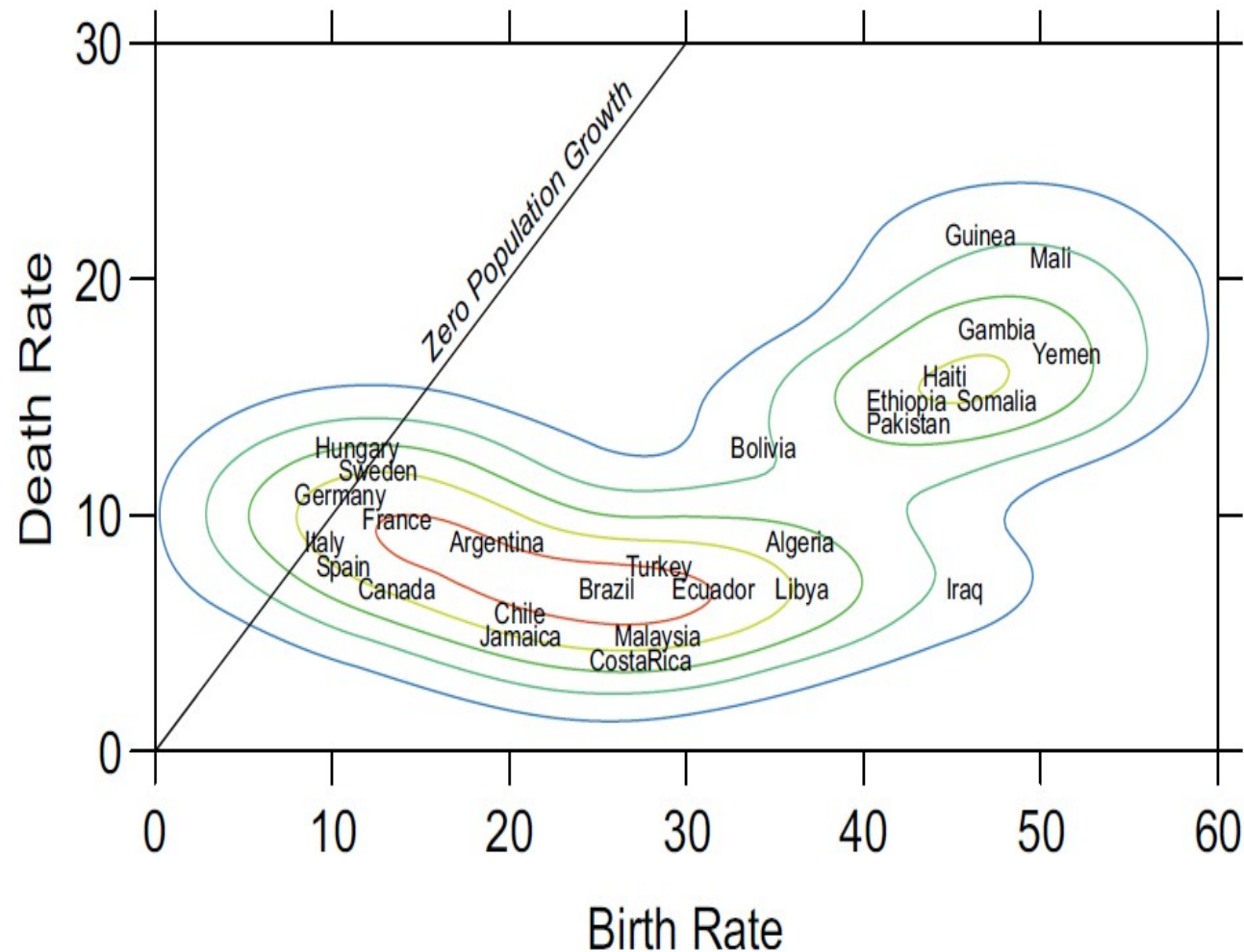
ELEMENT: *point(position(birth\*death), size(0), label(country))*

ELEMENT: *contour(position(smooth.density.kernel.epanechnikov.joint(birth\*death)), color.hue())*

GUIDE: *form.line(position((0,0),(30,30)), label("Zero Population Growth"))*

GUIDE: *axis(dim(1), label("Birth Rate"))*

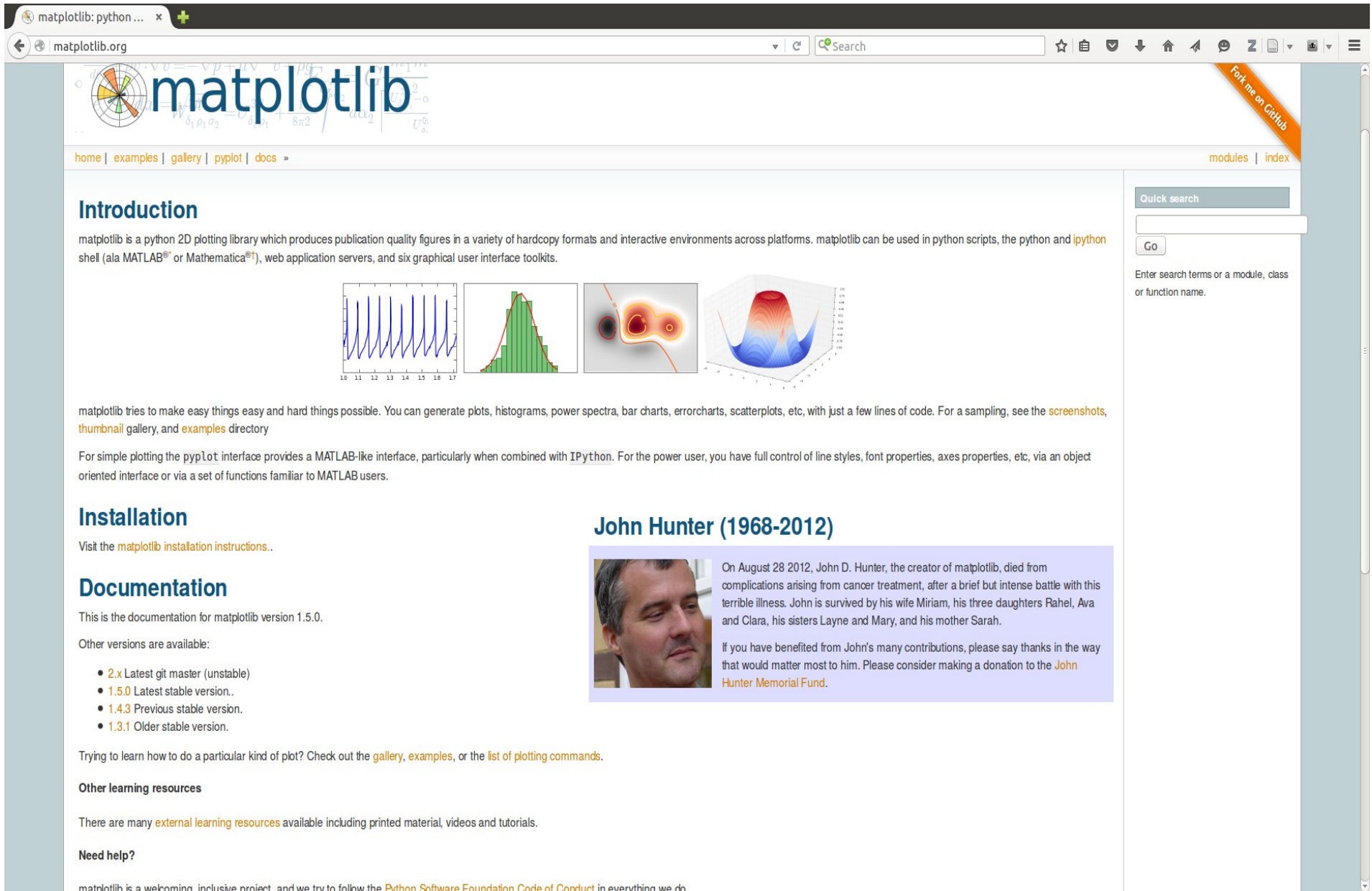
GUIDE: *axis(dim(2), label("Death Rate"))*



**Figure 1.1** Plot of death rates against birth rates for selected countries



# Matplotlib – od matlaba dla python'a do standardu wykresów



The screenshot shows the matplotlib.org website. The browser address bar displays 'matplotlib: python ...' and 'matplotlib.org'. The website header features the matplotlib logo, navigation links (home, examples, gallery, pyplot, docs), and a search bar. The main content area is titled 'Introduction' and describes matplotlib as a python 2D plotting library. It includes a row of four example plots: a line plot with multiple series, a histogram, a contour plot, and a 3D surface plot. Below the introduction, there is a section for 'Installation' with a link to installation instructions, a 'Documentation' section for version 1.5.0, and a list of other available versions (2.x, 1.5.0, 1.4.3, 1.3.1). A sidebar on the right contains a 'Quick search' box and a 'Fork me on GitHub' button. At the bottom, there is a section for 'John Hunter (1968-2012)' with a portrait photo and a text block about his passing and a memorial fund.

matplotlib: python ... x

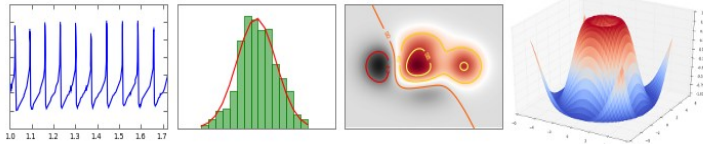
matplotlib.org

home | examples | gallery | pyplot | docs »

modules | index

## Introduction

matplotlib is a python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. matplotlib can be used in python scripts, the python and **ipython** shell (ala MATLAB® or Mathematica®), web application servers, and six graphical user interface toolkits.



matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, errorcharts, scatterplots, etc, with just a few lines of code. For a sampling, see the [screenshots](#), [thumbnail gallery](#), and [examples](#) directory

For simple plotting the `pyplot` interface provides a MATLAB-like interface, particularly when combined with `IPython`. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users.

## Installation

Visit the [matplotlib installation instructions](#).

## Documentation

This is the documentation for matplotlib version 1.5.0.

Other versions are available:

- **2.x** Latest git master (unstable)
- **1.5.0** Latest stable version..
- **1.4.3** Previous stable version.
- **1.3.1** Older stable version.

Trying to learn how to do a particular kind of plot? Check out the [gallery](#), [examples](#), or the [list of plotting commands](#).


### Other learning resources

There are many [external learning resources](#) available including printed material, videos and tutorials.

### Need help?

matplotlib is a welcoming, inclusive project, and we try to follow the [Python Software Foundation Code of Conduct](#) in everything we do.

## John Hunter (1968-2012)



On August 28 2012, John D. Hunter, the creator of matplotlib, died from complications arising from cancer treatment, after a brief but intense battle with this terrible illness. John is survived by his wife Miriam, his three daughters Rahel, Ava and Clara, his sisters Layne and Mary, and his mother Sarah.

If you have benefited from John's many contributions, please say thanks in the way that would matter most to him. Please consider making a donation to the [John Hunter Memorial Fund](#).

Quick search

Go

Enter search terms or a module, class or function name.

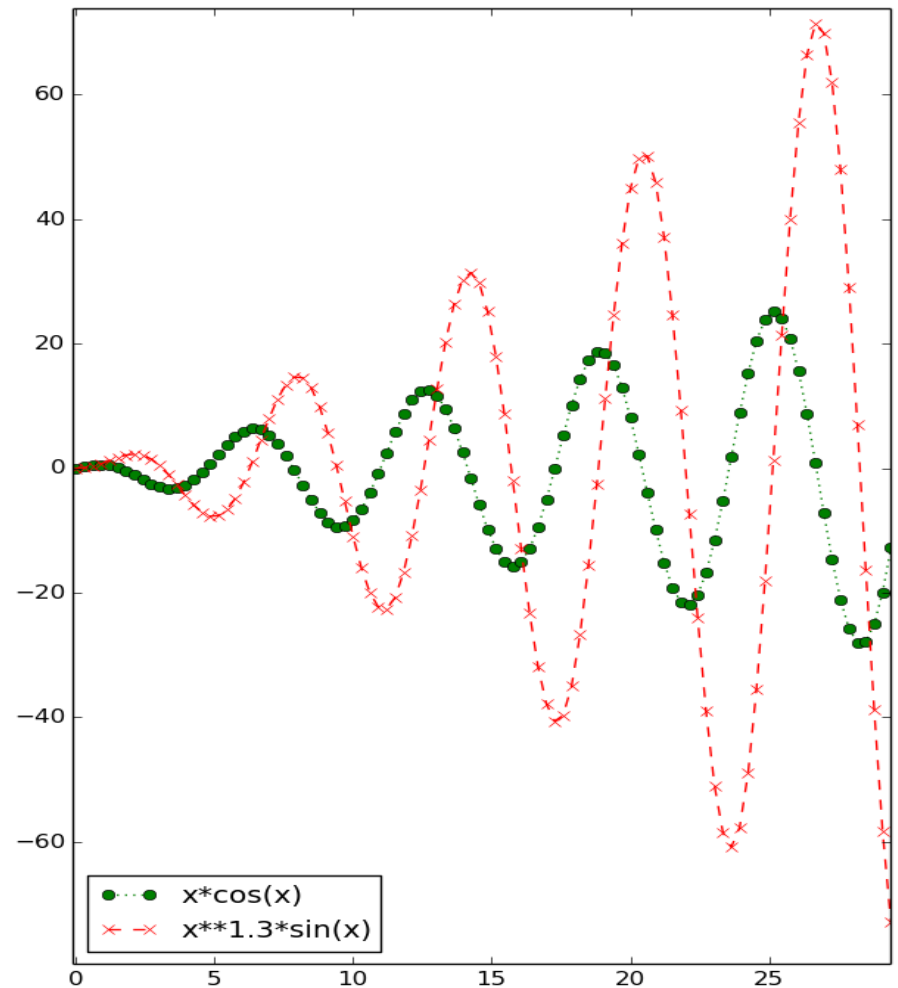
Fork me on GitHub

# Moduł matplotlib, pylab, pyplot

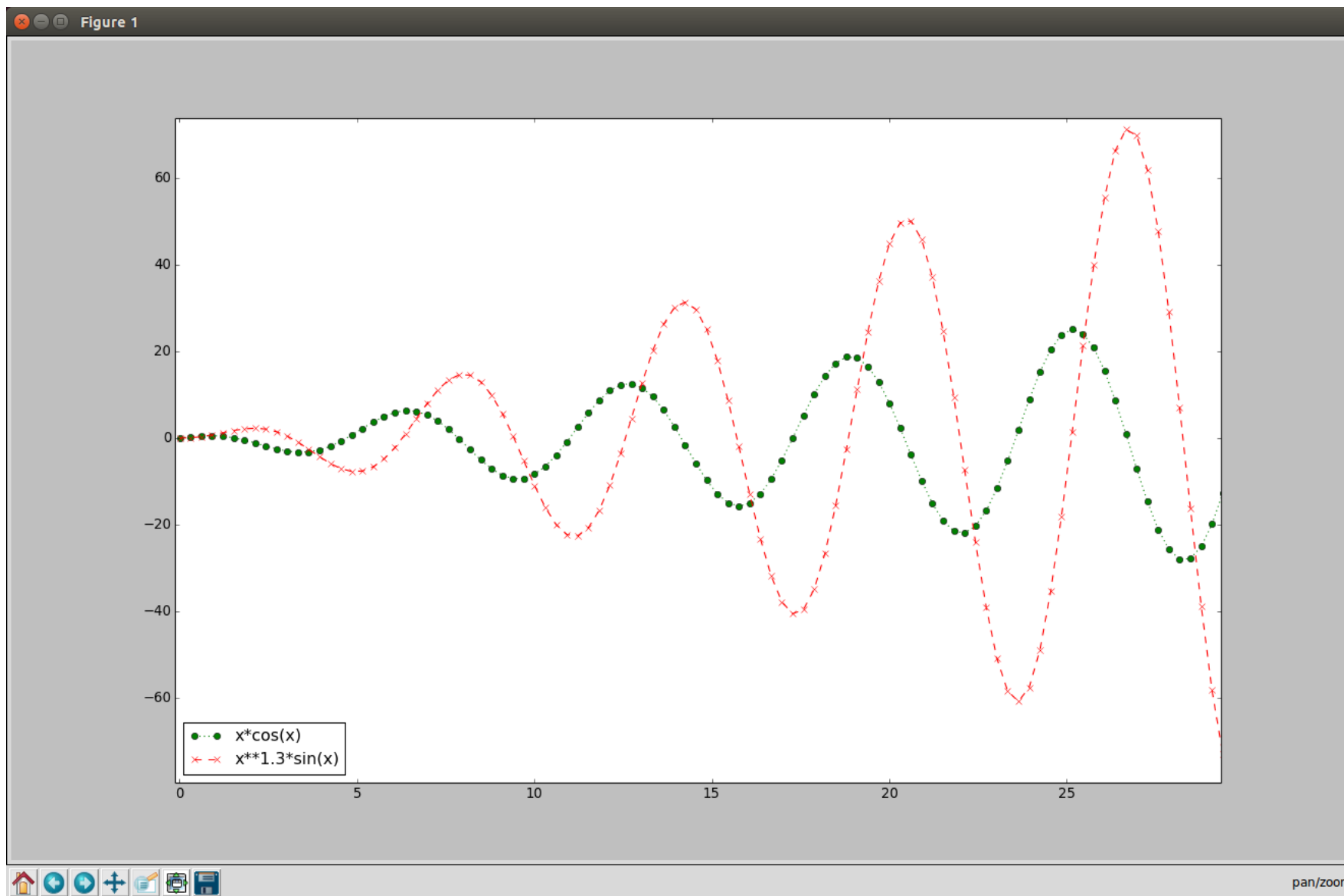
- Możemy importować:
  - `from matplotlib import pylab`
  - `from matplotlib import pyplot`
- Warto zapamiętać:
  - `ipython3 --pylab`

# Pylab.plot(..)

- `Pylab.plot(x, y, "r-x", label="opis")`
- Tworzy wykres funkcji  $y(x)$ , możemy wybrać kolor, typ linii, kształt znacznika
- Wykresy tworzone są w sposób przyrostowy
- Możemy dodać legendę, tytuły i znaczniki na osiach
- `Pylab.show()` odświeża wykres



# Interfejs graficzny

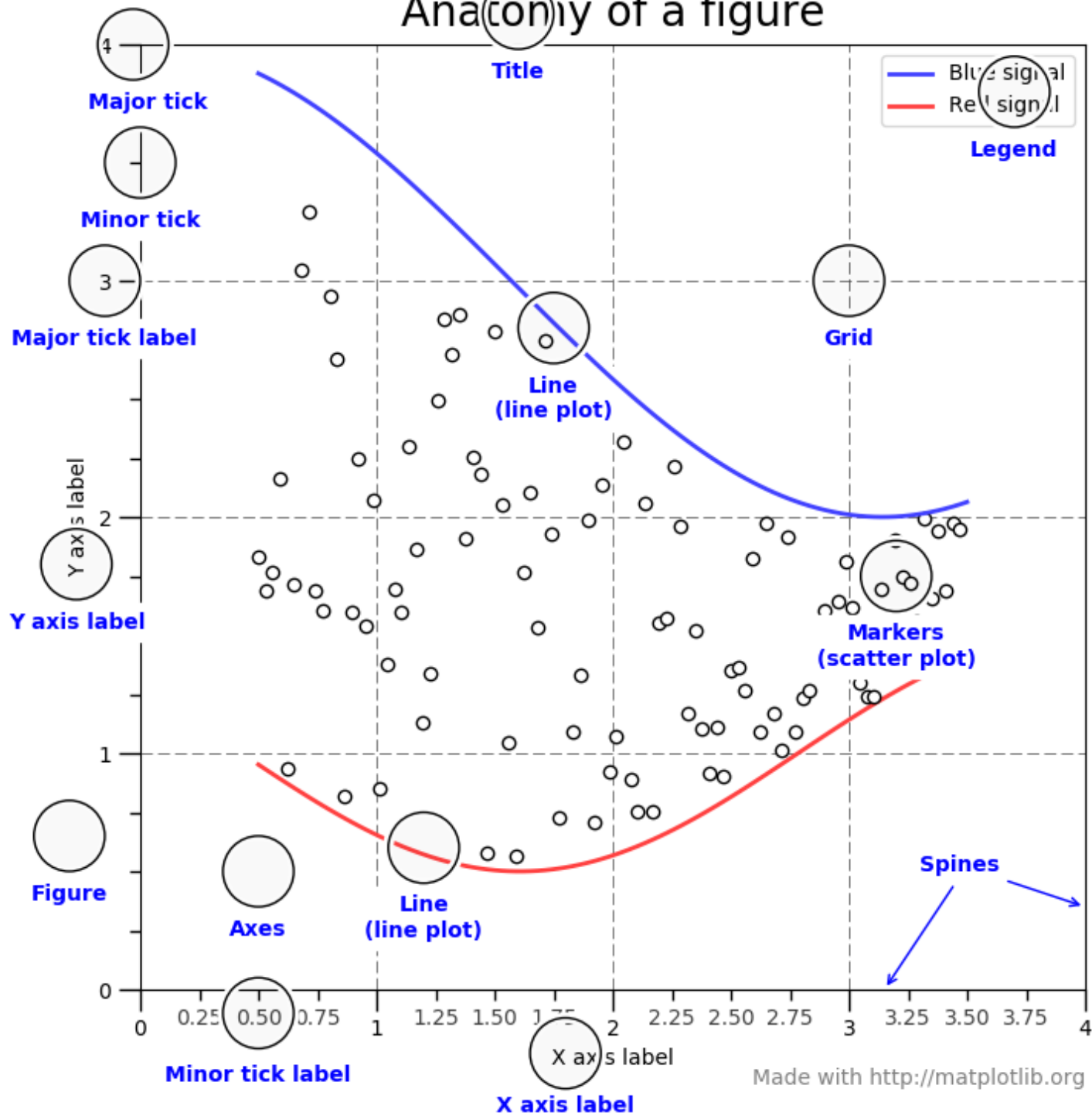


# Zapis do pliku

- Możemy zapisywać do pliku z poziomu interfejsu graficznego
- Pliki graficzne mogą mieć różne typy:
  - Png – bitmapa
  - PDF – format wektorowy, dobry do publikacji
  - Svg – wektorowy edytowalny format (do edycji np. W programie inkscape)
  - Inne (ps, eps, raw, tiff, ...)
- Możemy też użyć wywołania:
  - `pylab.savefig(nazwa)`
- Możemy wybrać typ pliku
- A także rozdzielczość (dpi), w przeciwieństwie do eksportu z interfejsu graficznego

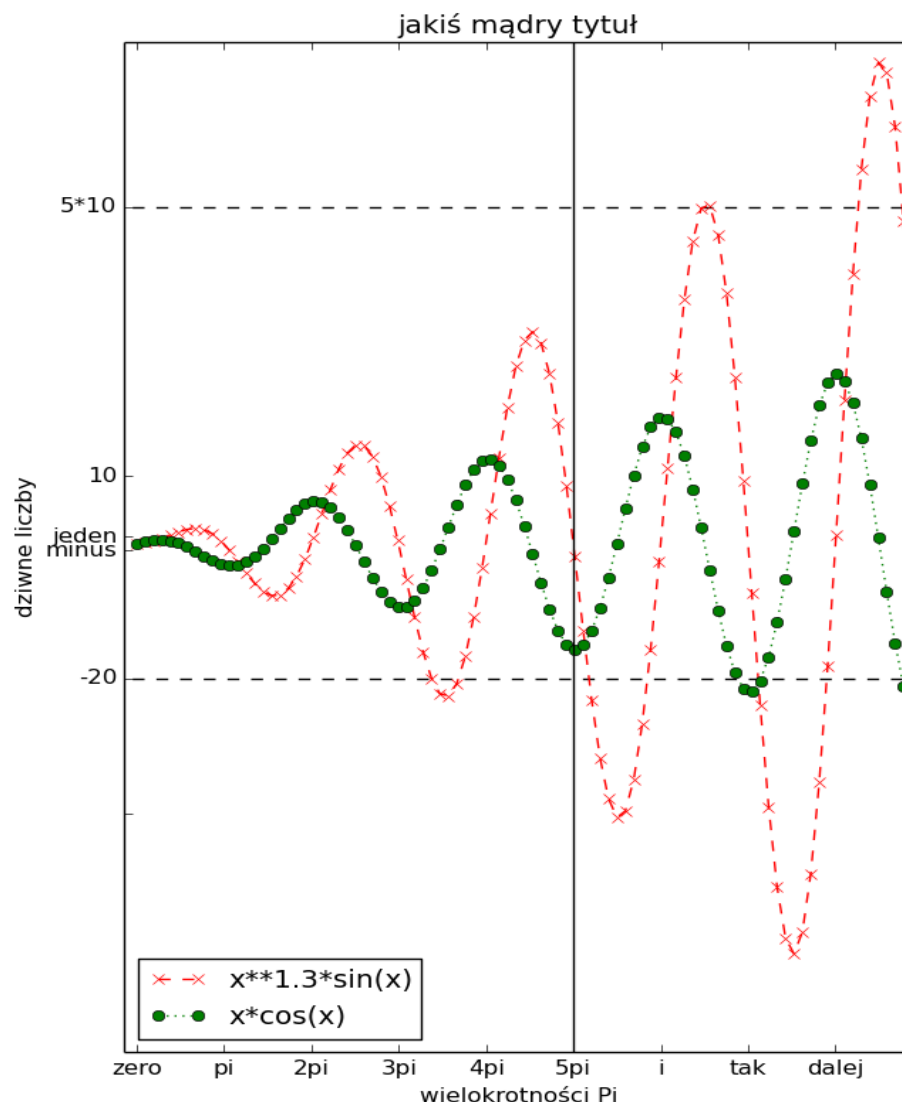


# Anatomy of a figure



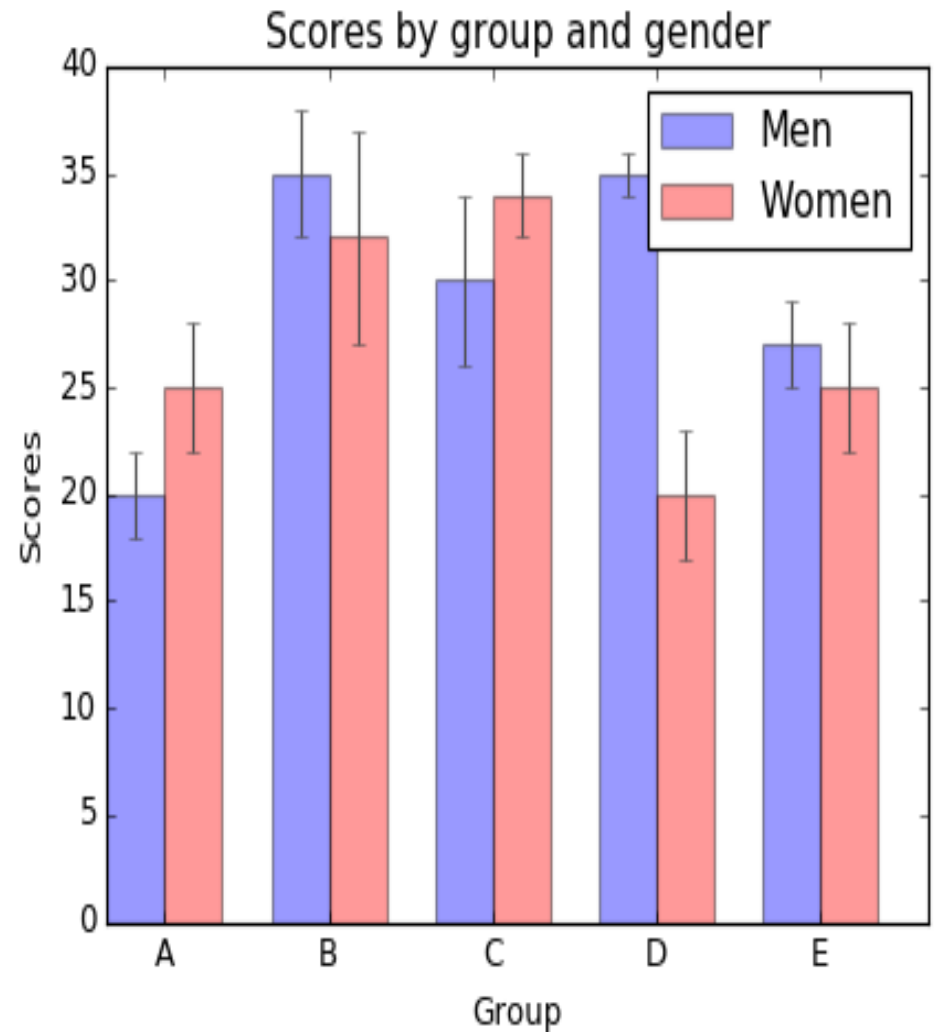
# Figure, axes, ticks, xlim, legend...

- Możemy nasz wykres dekorować:
  - `xticks()`, `yticks()`
  - `xlabel()`, `ylabel()`, `title()`
  - `legend(loc=...)`
  - Możemy dodawać linie bez “label”, aby zaznaczyć konkretne wartości



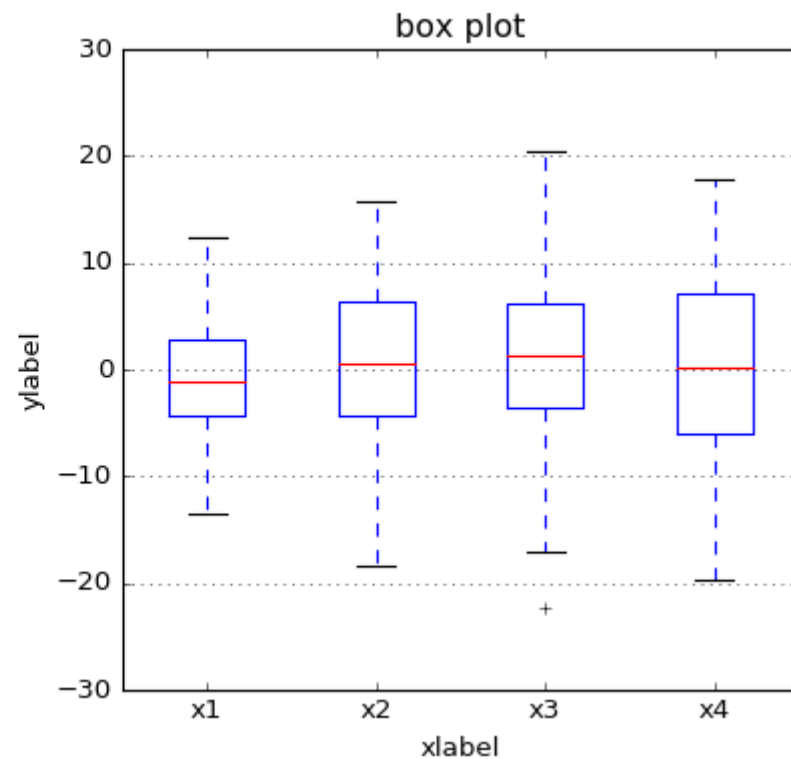
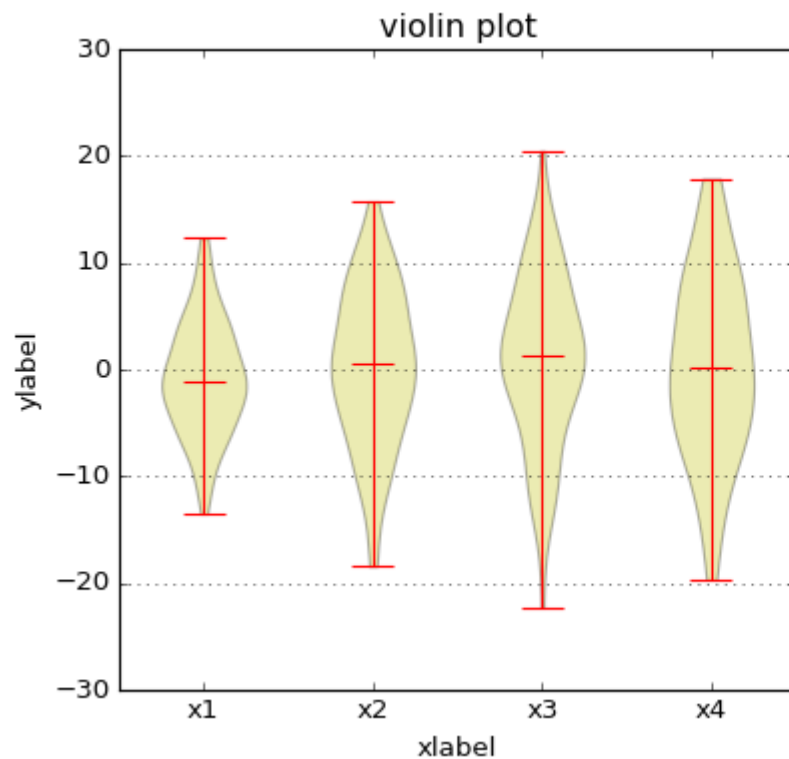
# Wykresy słupkowe

- `bar(left,height)`
- Dodatkowe parametry:
  - `width`
  - `bottom`
  - `yerr`



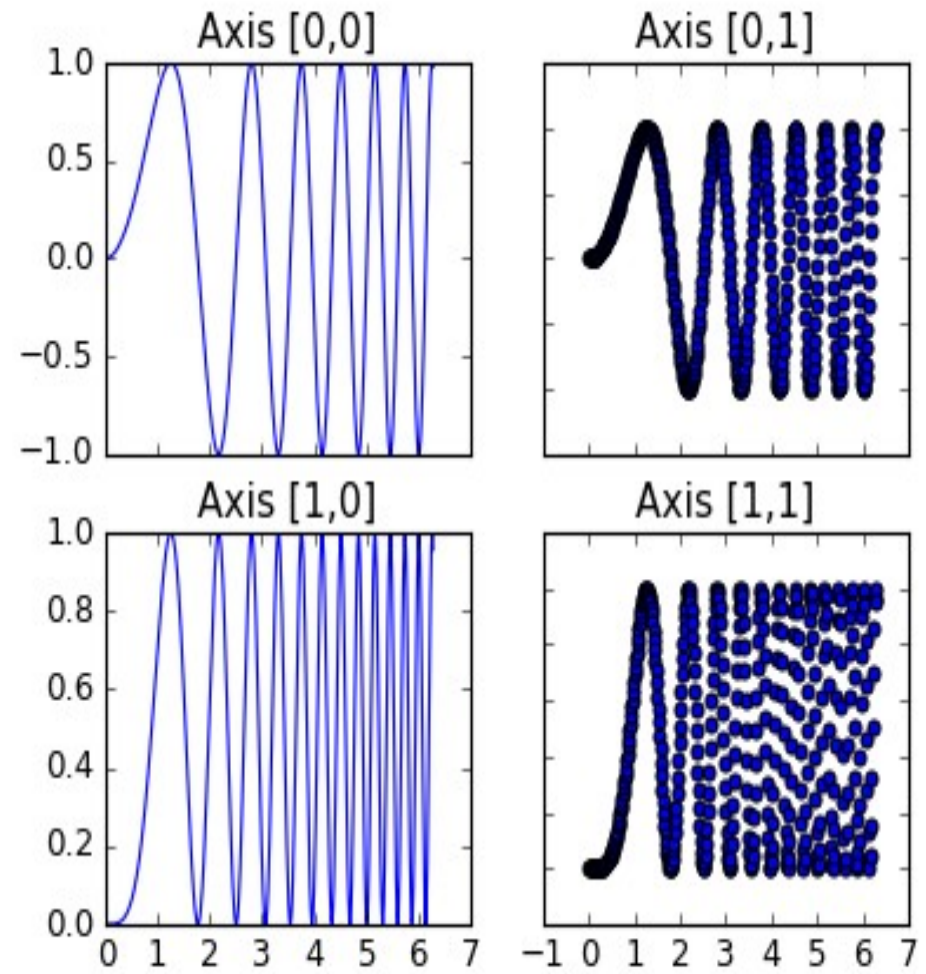
# Wykresy pudełkowe

- `boxplot(xs,ys)`
- Opisuje rozkłady zmiennych przy pomocy mediany, 25, 75 percentyla,  $1.5 \cdot \text{IQR}$



# Podwykresy subplot()

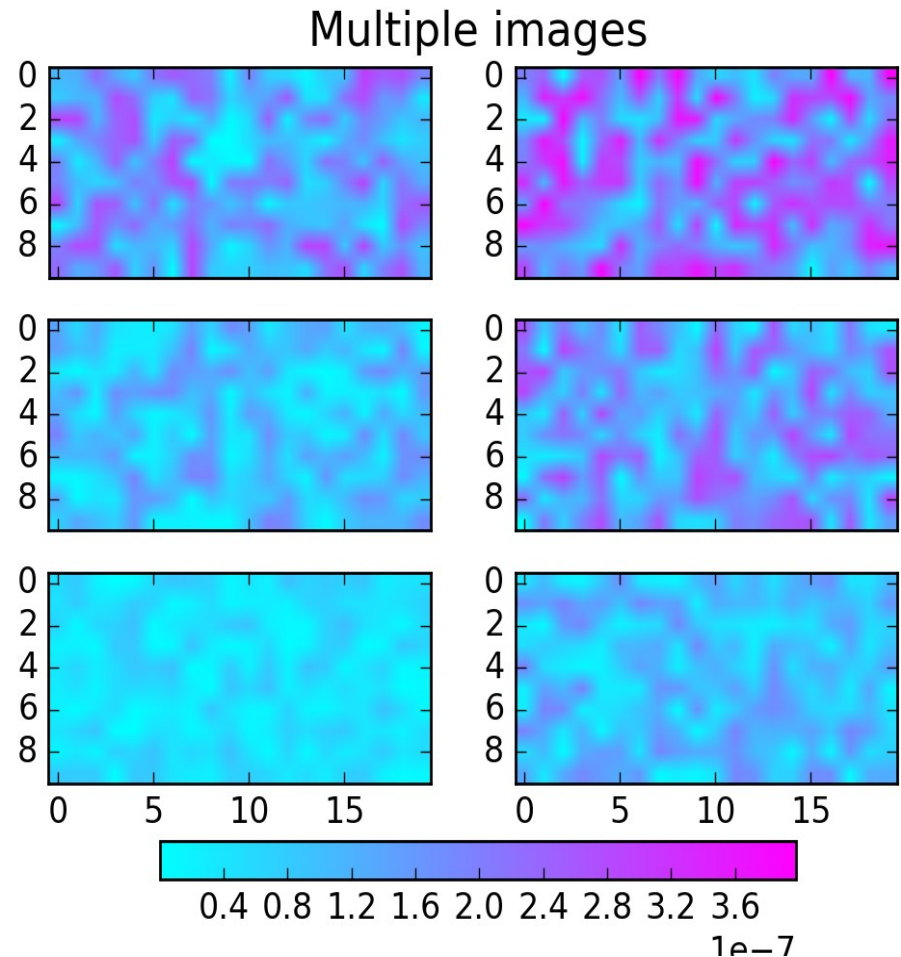
- Możemy tworzyć wiele podwykresów przy pomocy funkcji `subplot(x,y,i)`
- Wykresy są automatycznie układane w wierszach i kolumnach





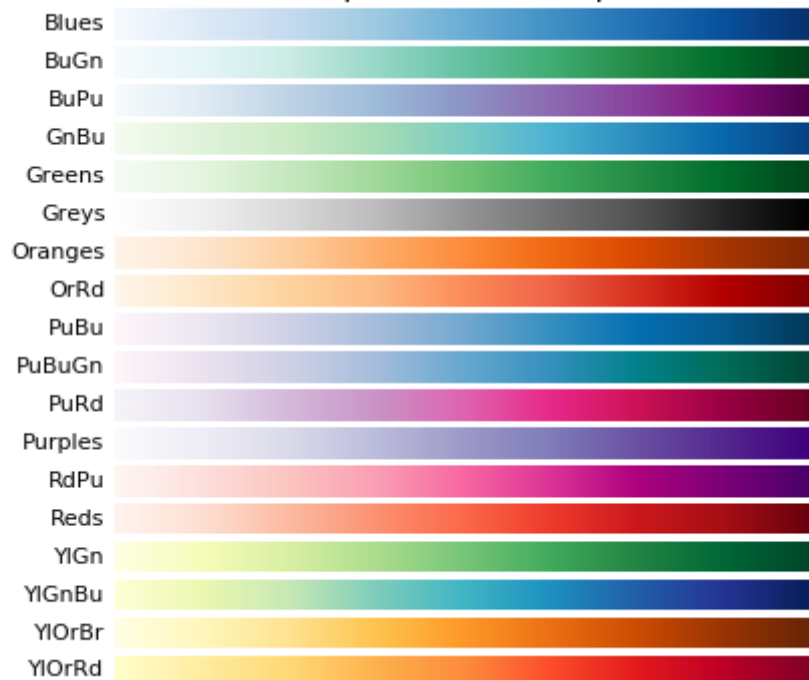
# Mapy ciepła (ang. heatmap)

- `Pylab.imshow(...)`
- Służy do wyświetlania obrazów, albo danych macierzowych
- Obsługuje różne mapy kolorów
- `Colorbar()` pokazuje skalę
- Możemy wybrać mapę z wygładzaniem lub bez

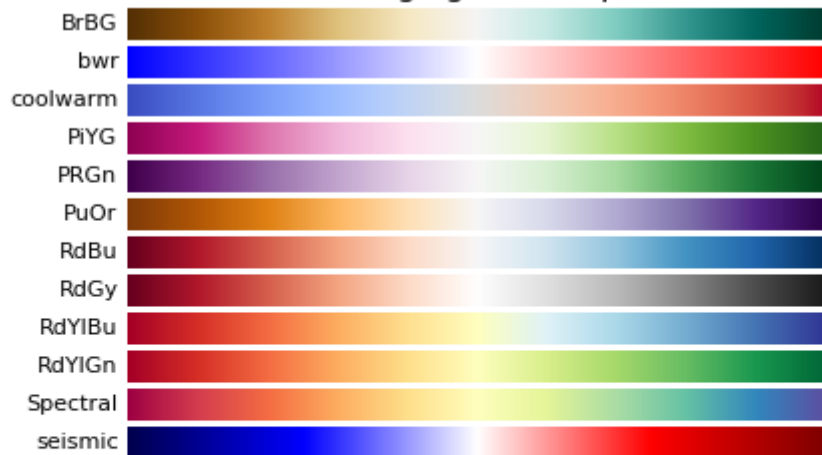


# Skale kolorów - cmap

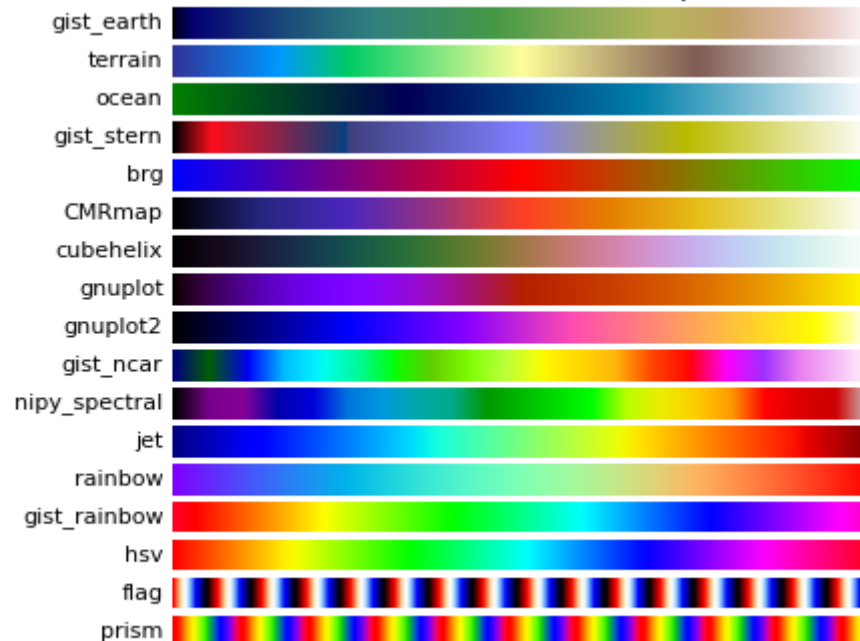
## Sequential colormaps



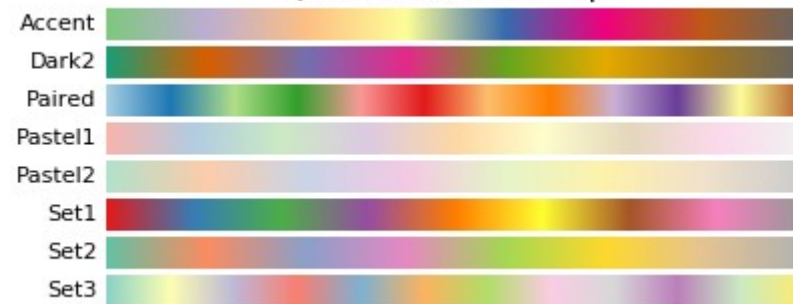
## Diverging colormaps



## Miscellaneous colormaps

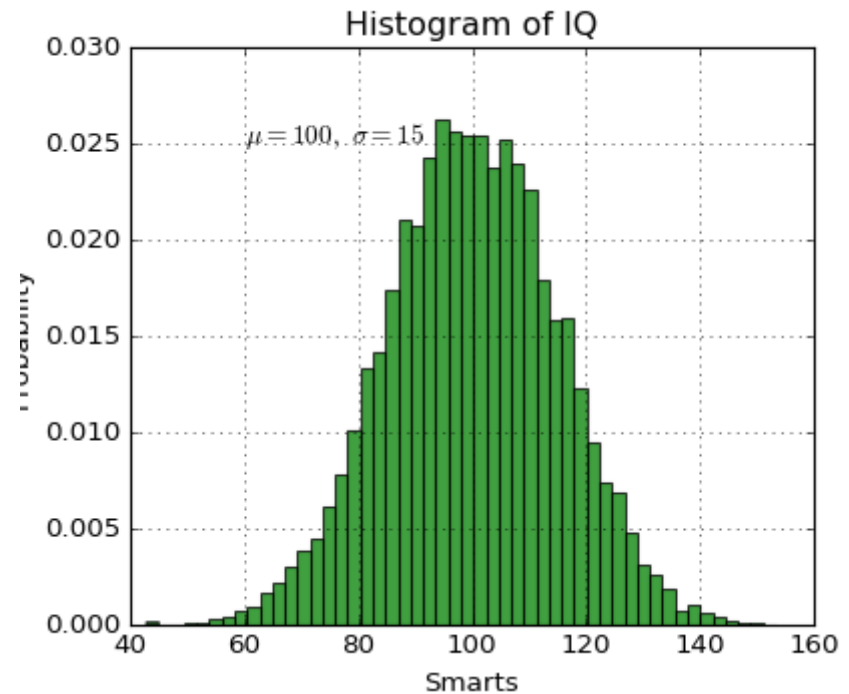


## Qualitative colormaps



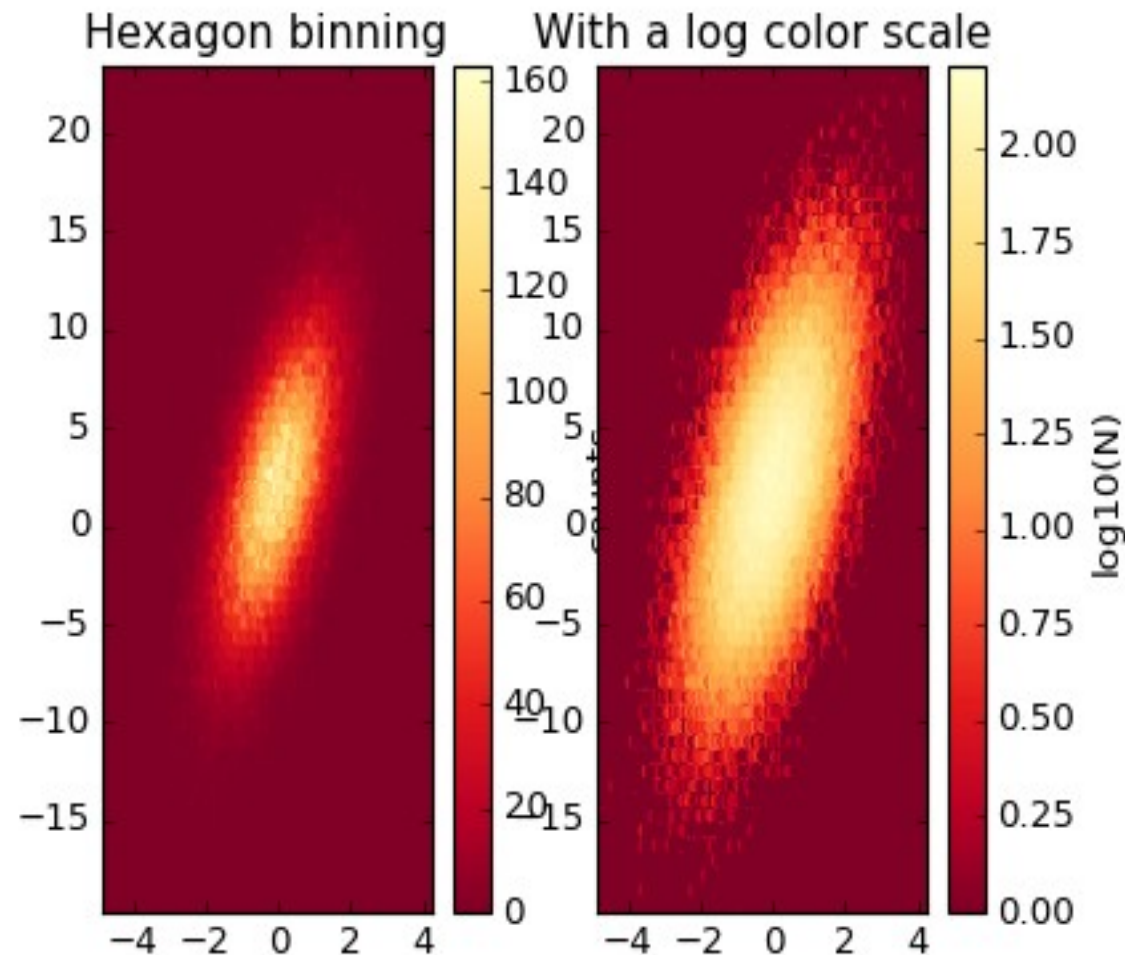
# Histogramy

- Funkcja `hist()` służy do generowania histogramów, czyli zliczeń rozkładów
- Parametr `bins=...` może podać liczbę albo zakresy “kubelków” w których zliczamy wartości
- Parametr `normed` określa, czy normujemy histogram, czy nie

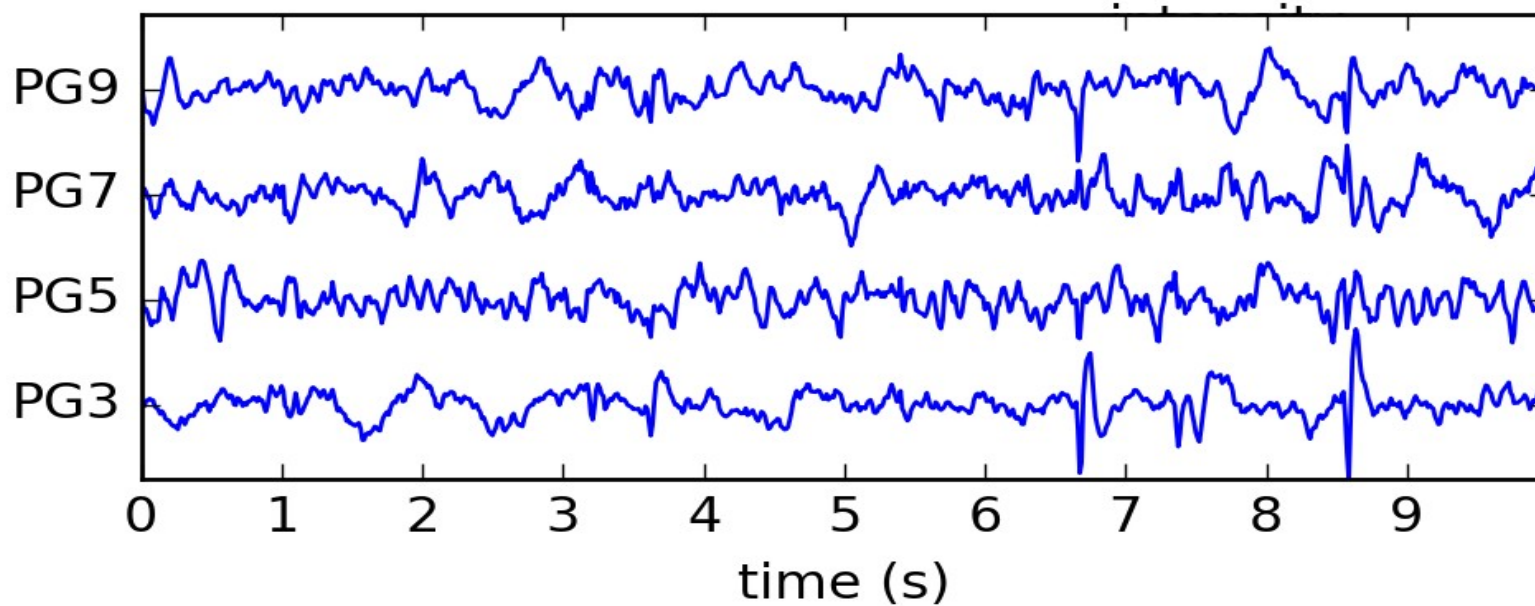
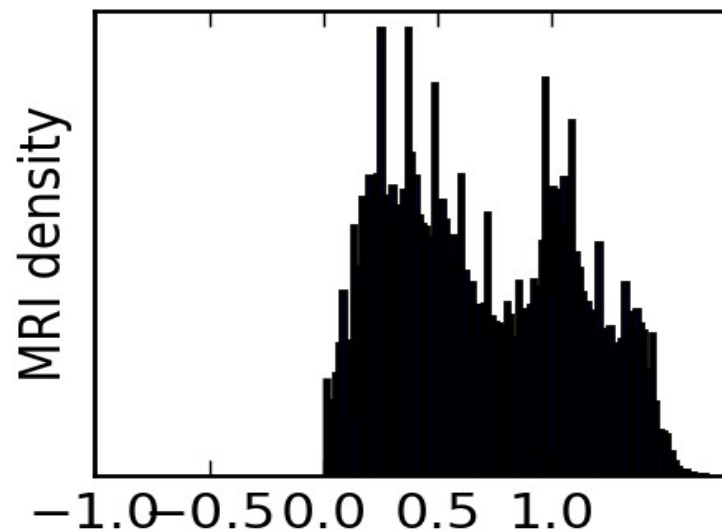
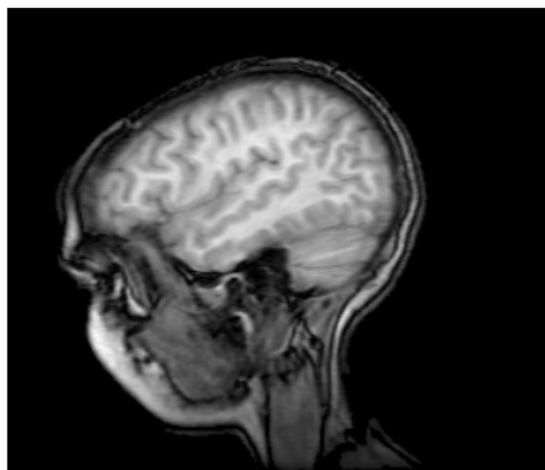


# Histogramy sześciokątne 2d

- Funkcja `hexbin()`, pozwala na zliczenia w 2d
- Zliczenia mogą być logarytmiczne, lub liniowe
- Wartości są reprezentowane w sześciokątnych polach



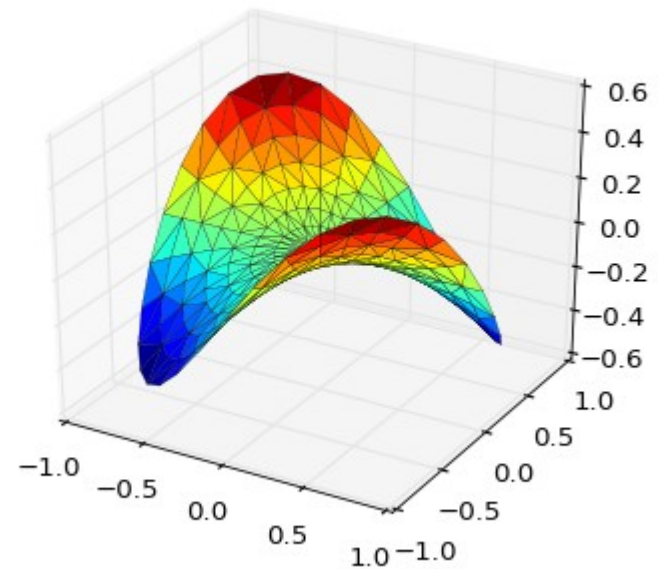
# Ciekawszy przykład





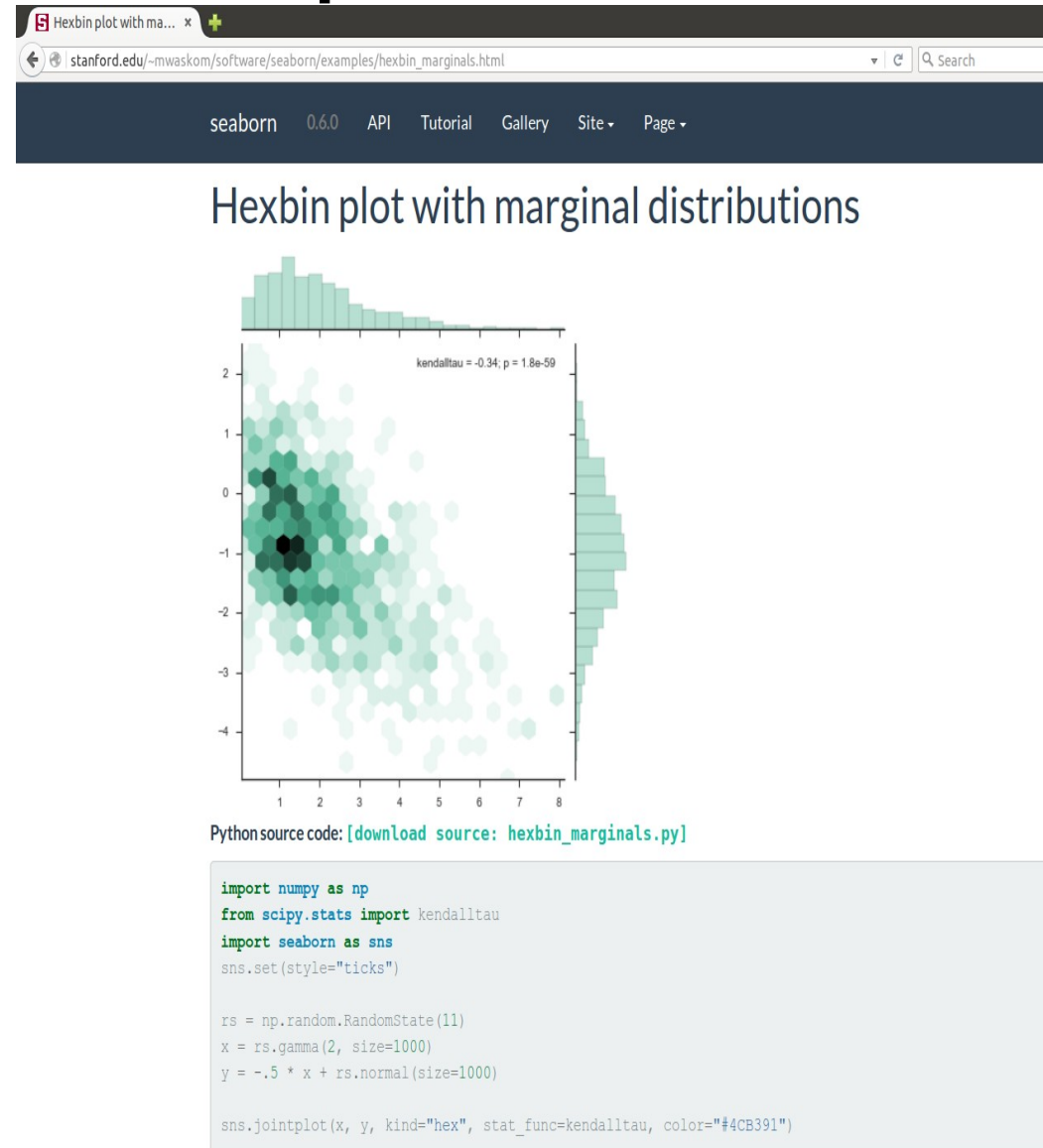
# Wykresy 3d

- Można wyświetlać też wykresy w 3d
- Trzeba stworzyć specjalny obiekt `axes3d`
- Można też kolorować z użyciem map kolorów



# Nowe pakiety rozszerzające matplotlib

- **Seaborn** – kilka nowych rodzajów wykresów, nieco ładniejszy standardowy “styl” wykresów
- **Bokeh** – biblioteka umożliwiająca renderowanie wykresów a'la matplotlib jako obiektów html



# Różne biblioteki generujące obrazy

- Matplotlib jest biblioteką dającą interfejs programistyczny (API) do kreślenia wykresów, ale sama korzysta z różnych bibliotek generujących obrazy (tzw. backend'ów)
- W szczególności, najczęściej wykorzystywana jest biblioteka Agg, do grafiki rastrowej, ale można też korzystać z cairo do generowania grafiki wektorowej
- Możemy wybierać backend w skrypcie (przed wywołaniem funkcji plot lub figure):

```
import matplotlib
matplotlib.use('PS')    # generate postscript output by default
```