

# Architektura dużych projektów bioinformatycznych

Pakiety do obliczeń: naukowych,  
Inżynierskich i statystycznych  
Przegląd i porównanie

Bartek Wilczyński

23.11.2015

# Plan na dziś

- Pakiety do obliczeń: przegląd zastosowań
- różnice w zapotrzebowaniu: naukowcy, inżynierowie, statystycy/medycy
- Matlab/octave/scipy
- S-Plus/SPSS/projekt R
- Mathematica/Maxima/Sage
- Pakiety komercyjne vs. Open Source
- Excel?

# Typowi użytkownicy pakietów obliczeniowych

- Inżynierowie i projektanci (budownictwo, lotnictwo, motoryzacja, itp.)
- Naukowcy doświadczalni (fizycy, chemicy, materiałoznawcy, itp.)
- Statystycy (zastosowania w medycynie, ekonomii, biologii molekularnej, psychologii, socjologii, ubezpieczeniach, itp.)
- Matematycy (przede wszystkim matematyka stosowana )

# Obliczenia naukowe

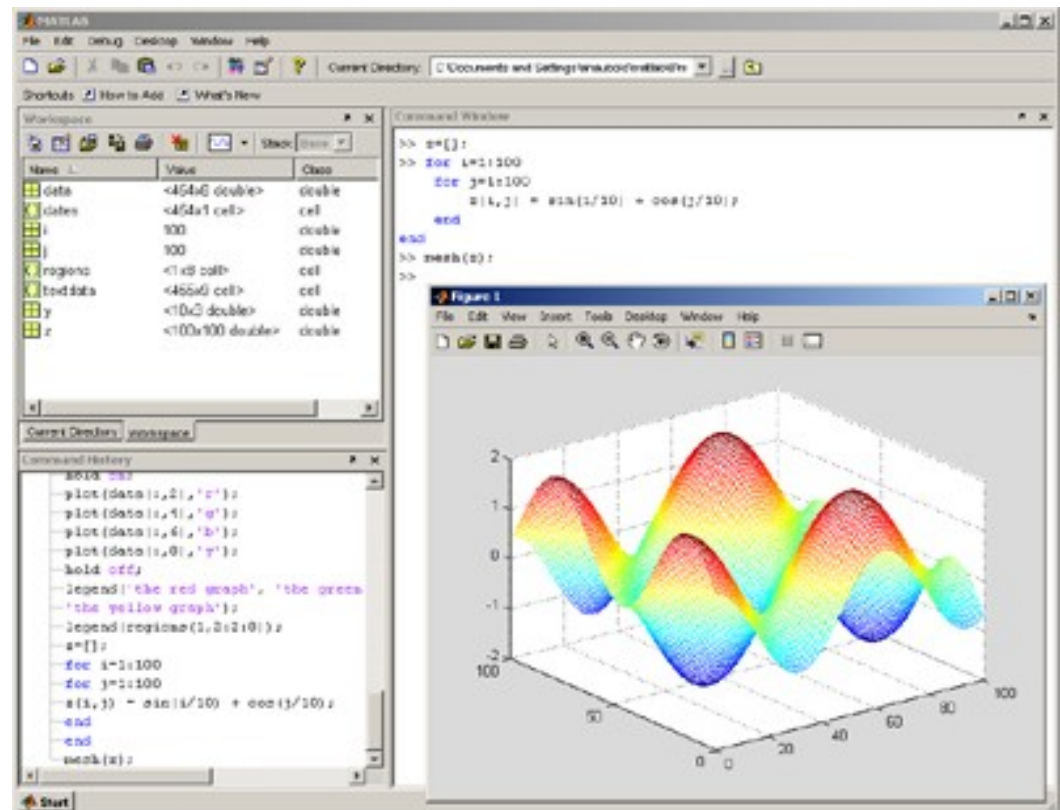
- Komputer jako “potężniejszy kalkulator”
- W zasadzie wszystko można zaprogramować samemu, ale każdemu mogą się przydać:
  - Interfejs użytkownika łatwiejszy niż typowego kompilatora
  - Możliwość zaawansowanej grafiki
  - Dobrze przetestowane standardowe procedury
  - Interfejsy do urządzeń
  - Wsparcie fachowców

# Matlab i pakiety “inżynierskie”

- Rozwijany w latach 70'tych przez Cleve Moler'a jako narzędzie dla studentów informatyki, aby nie musieli używać zaawansowanych bibliotek fortranu
- Firma mathworks powstaje w 1984 i wydaje pierwszą wersję Matlab'a
- Najpopularniejszy wśród inżynierów, dobre całki numeryczne, rozwiązywanie równań i wykresy (również 3d)
- Bardzo popularny także do przetwarzania sygnałów i symulacji (simulink)
- Licencja komercyjna – niedrogi dla studentów, droższy dla uczelni, bardzo drogi dla przemysłu

# Toolbox'y Matlab'a

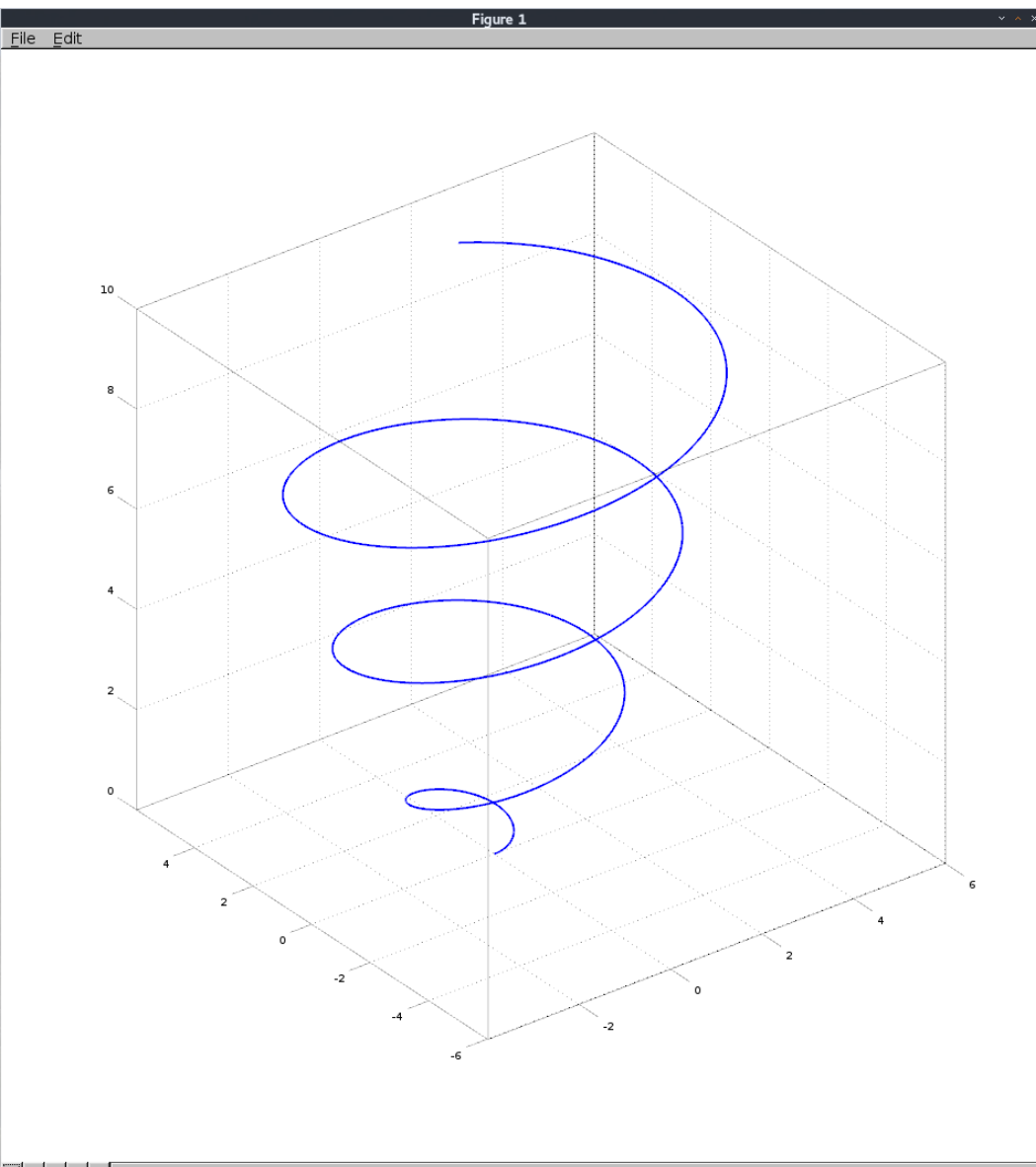
- Wiele dodatkowych (płatnych) bibliotek dla specjalistów
  - Symbolic math
  - Image processing
  - Financial toolbox
  - Bioinformatics
  - Optimization
  - SimBiology



# Alternatywy openSource

- GNU Octave (rozpoczęty w 1988, wydania od 1992, rozwijany przez John'a W. Eatona, chemika z University of Wisconsin-Madison)
  - W zasadzie kompatybilny z Matlab'em
  - John W. Eaton Inc. - consulting
- Scipy stack – zestaw bibliotek python'a do obliczeń naukowych
  - Wiele bibliotek, rozwijanych przez niezależne grupy
  - System pakietów, edytor i dystrybucja organizowana przez firmę Enthought, również komercyjne dystrybucje i consulting
  - Wiele konferencji tematycznych dla naukowców i pracowników przemysłu - także źródło dochodu

# Interfejs Octave



```
octave
-----
~ » octave
GNU Octave, version 3.8.1
Copyright (C) 2014 John W. Eaton and others.
This is free software; see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. For details, type 'warranty'.

Octave was configured for "x86_64-unknown-linux-gnu".

Additional information about Octave is available at http://www.octave.org.

Please contribute if you find this software useful.
For more information, visit http://www.octave.org/get-involved.html

Read http://www.octave.org/bugs.html to learn how to submit bug reports.
For information about changes from previous versions, type 'news'.

octave:1> t=[0:0.01:20];
octave:2> x=sqrt(t).*cos(t);
octave:3> y=sqrt(t).*sin(t);
octave:4> z=0.5*t;
octave:5> graph=plot3(x,y,z)
graph = -17.921
octave:6> set(graph(1), "linewidth", 2)
octave:7> □
```



# Interfejs Enthought Canopy

The image displays the Canopy IDE interface, which is a Python development environment. The main window is titled "Editor - Canopy" and shows a code editor with Python code. The code includes comments and function calls, such as `num_vars = int` and `draw_circle_patch`. The interface is divided into several panels:

- File Browser:** Located at the top left, it shows a list of files and folders.
- Welcome to Canopy:** A central panel with the Canopy logo and a message: "Hi, welcome to Canopy! Login to your Enthought account or create one." Below this are icons for the Editor, Package Manager, and Doc Browser.
- Recent files:** A list of recently opened files, including `application.py`, `application_launcher.py`, `util.py`, `prisma.py`, and `mac_metadata.py`.
- Documentation Browser:** A window on the right side showing the Canopy documentation. It includes a search bar, a navigation menu, and a list of links to various resources, such as "Python Tutorial", "Python Documentation", "IPython", "NumPy", "SciPy", "Traits", "TraitsUI", "Enaml", "Envisage", "Chaco", "Hovast", and "Matplotlib Gallery".
- 3D Visualization:** A window titled "Mayavi Scene 1" showing a 3D visualization of a complex, multi-colored, swirling structure, likely a visualization of a mathematical or scientific model.

The bottom of the interface shows a command prompt with the following text:

```
Type "?" for some information.  
In [1]: !run "/var/folders/tr/pfk3l1cgl4ndrbppzjwq_tq4000gn/T/tmpff1uoh.py"  
In [2]: from mayavi import mlab  
In [3]: mlab.test_plot3d()  
Out[3]: <mayavi.modules.surface.Surface at 0x122be4400>  
In [4]:
```

# S-Plus dla statystyków

- Język S zaprojektowany w laboratoriach Bell Labs przez Johna Chambers'a
- Implementacja przez R. Douglas'a Martina, profesora statystyki w Seattle
- Wydany komercyjnie w 1988 jako S-Plus, potem kolejno “przejmowany” przez różne korporacje aż do 2008, kiedy przejęła go firma TIBCO
- Adresowany do statystyków akademickich i przemysłowych
- Ogólny, bez specjalnych specjalizacji

# Projekt R – Implementacja OpenSource języka S

- Rozpoczęty ok. 1995 projekt stworzenia darmowej implementacji języka S
- Ross Ithaka (obecnie Genentech) and Robert Gentleman (obecnie Univ. Auckland)
- W tej chwili zarządzany przez “R foundation”
- Wiele firm “wspierających” R
- Zachęcam do obejrzenia slajdów Chambers'a:  
[www.r-project.org/useR-2006/Slides/Chambers.pdf](http://www.r-project.org/useR-2006/Slides/Chambers.pdf)

# Rstudio – interfejs do R'a

The screenshot displays the RStudio environment with the following components:

- Source Editor:** Contains R code for loading data, summarizing it, and creating a faceted scatter plot.
- Console:** Shows the execution output of the code, including summary statistics for 'x', 'y', and 'z' variables, and the execution of the plotting functions.
- Workspace:** Lists the loaded data object 'diamonds' (53940 observations) and the 'p' object (ggplot object).
- Plots:** Displays a faceted scatter plot titled 'Diamond Pricing' showing Price vs. Carat, faceted by Clarity.

```
1 library(ggplot2)
2 source("plots/formatPlot.R")
3
4 view(diamonds)
5 summary(diamonds)
6
7 summary(diamonds$price)
8 aveSize <- round(mean(diamonds$carat), 4)
9 clarity <- levels(diamonds$clarity)
10
11 p <- qplot(carat, price,
12            data=diamonds, color=clarity,
13            xlab="Carat", ylab="Price",
14            main="Diamond Pricing")
15
```

**Console Output:**

```
Min.   : 0.000   Min.   : 0.000   Min.   : 0.000
1st Qu.: 4.710   1st Qu.: 4.720   1st Qu.: 2.910
Median : 5.700   Median : 5.710   Median : 3.530
Mean   : 5.731   Mean   : 5.735   Mean   : 3.539
3rd Qu.: 6.540   3rd Qu.: 6.540   3rd Qu.: 4.040
Max.   :10.740   Max.   :58.900   Max.   :31.800
> summary(diamonds$price)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  326   950   2401   3933   5324   18820
> aveSize <- round(mean(diamonds$carat), 4)
> clarity <- levels(diamonds$clarity)
> p <- qplot(carat, price,
+           data=diamonds, color=clarity,
+           xlab="Carat", ylab="Price",
+           main="Diamond Pricing")
>
> format.plot(p, size=24)
>
```

**Workspace Data:**

Variable	Value
diamonds	53940 obs. of 10 variables
aveSize	0.7979
clarity	character [8]
p	ggplot [8]

**Plots:**

The plot is a faceted scatter plot titled "Diamond Pricing". The x-axis is labeled "Carat" (ranging from 0.0 to 3.5) and the y-axis is labeled "Price" (ranging from 0 to 15000). The data points are colored by "Clarity", with a legend on the right showing categories: I1, SI2, SI1, VS2, VS1, VVS2, VVS1, and IF.

# System pakietów w R

- Istotna jest możliwość rozwijania własnych “pakietów” w R (coś na kształt “toolbox’ów” Matlab'a)
- Jest to proces całkowicie demokratyczny, każdy może wysłać pakiet i umieścić go w repozytorium CRAN (Comprehensive R Archive Network)
- Możliwość automatycznej instalacji pakietów z CRAN
- Pewne standardy dokumentacji (winiety) zgodne z “literate programming”

# Bioconductor

- Dystrybucja wybranych pakietów R do analizy danych biomedycznych
- Nacisk na łatwiejszą instalację i lepszą dokumentację pakietów
- Stabilny cykl wydań (2 razy do roku)
- Szkolenia adresowane do biologów i medyków
- Kompletnie not-for-profit
- Finansowany z grantów (ok 7-8 osób core team)

# Mathematica - interfejs

The image shows a Mathematica notebook interface with several windows and panels.

**Main Notebook Window:**

- Code:**

```

aa = Table[A[[i, j]], {i, 1, 4}, {j, 1, 4}];
bb = Table[B[[i, j]], {i, 1, 4}, {j, 1, 4}];
ll = Table[L[[i]], {i, 1, 4}];
mm = Table[M[[i]], {i, 1, 4}];
ss = LinearSolve[aa, l];
(ss + mm)[[3]] / (aa + bb)[[3]]

```
- Output:**

```

{0.}
{0.}
{1.}
{0.}

```

0.4

■ p=2: Neumann-Randintegrale

Auf jedem Face kommen die grade. Daher: nur x = 1 wird bei

```

nE[1][y_, z_] = y * (1 - z);
nE[2][y_, z_] = z * (1 - y);
nE[3][y_, z_] = z * (1 - y);
nE[4][y_, z_] = y * (1 - z);
For[i = 1, i <= 4, i++,
  Print[
    Integrate[Integrate[nE[i][y, z] * f[1, y, z], {y, 0, 1}], {z, 0, 1}]]

```

0.0138889

0.0138889

0.0277778

0.0277778

```

nF[y_, z_] = y * (1 - y) * z * (1 - z);
Integrate[Integrate[nF[y, z] * f[1, y, z], {y, 0, 1}], {z, 0, 1}] // N

```

0.00694444
- Plot:** A 2D plot showing four Bessel functions:  $J_0(x)$  (red),  $J_1(x)$  (green),  $J_2(x)$  (blue), and  $J_9(x)$  (magenta). The x-axis ranges from 0 to 20, and the y-axis ranges from -0.4 to 1.0.

**Help Browser Window:**

- Content:**

BesselJ[n, z] gives the Bessel function of the first kind  $J_n(z)$ .

$J_n(z)$  satisfies the differential equation  $z^2 y'' + z y' + (z^2 - n^2) y = 0$ .

BesselJ[n, z] has a branch cut discontinuity in the complex z plane running from  $-\infty$  to 0.

```

Plot[{BesselJ[0, x], BesselJ[1, x], BesselJ[2, x], BesselJ[9, x]},
  {x, 0, 20},
  PlotStyle -> {RGBColor[1, 0, 0], RGBColor[0, 1, 0], RGBColor[0, 0, 1],
    RGBColor[1, 0, 1]}]

```
- Navigation:** Go Back, Hide Categories
- Categories:** Plot3D, ListPlot3D (selected), ParametricPlot3D
- Text:**

of a surface representing an array of height values.

element of the surface shaded according to the specification in

cs.

e.

representing z values. There will be holes in the surface correspond-

ensions  $(m-1) \times (n-1)$ .

or RGBColor, or SurfaceColor objects.

[Math`ComputationalGeometry`](#).

**Bottom Panel:**

- Code:**

```

In[1]:= ListPlot3D[Table[Sin[x y] + Random[Real, {-0.15, 0.15}],
  {x, 0, 3 Pi / 2, Pi / 15}, {y, 0, 3 Pi / 2, Pi / 15}];

```
- Plot:** A 3D surface plot showing a wavy surface with a grid. The x and y axes range from 0 to approximately 4.7 (3π/2), and the z-axis ranges from -1 to 1.

# Obliczenia symboliczne - Mathematica

- Opracowana w latach 1980'tych przez Stephen'a Wolframa
- Jeden z pierwszych w historii pakietów umożliwiających obliczenia symboliczne
- Bardzo popularna wśród studentów amerykańskich, którzy muszą “zaliczyć” rachunek różniczkowy
- Obecnie także w wersji online: Wolfram Alpha
- Konkurencyjne pakiety: Maple, Mathcad



# Obliczenia symboliczne Open Source

- Maxima (1992-), a wcześniej Macsymba (1968-1982)
- Wydana w 1998 na licencji GPL
- Napisana w języku lisp
- Wiele konkurencyjnych interfejsów (WXMaxima, Gmaxima itp)
- Skupiona na obliczeniach symbolicznych

# SAGE notebook

- Stosunkowo nowy projekt
- Połączenie wielu środowisk obliczeniowych
  - Python (Numpy, Scipy, Sympy, matplotlib, Networkx)
  - Maxima
  - R
  - GAP, FLINT, GD, JMOL, PALP, Singular
- Środowisko w przeglądarce, sesja na serwerze lub “w chmurze”

# Interfejs SAGE

## Use Sage to Solve Equations

last edited on April 11, 2011 05:45 PM by admin

[Save](#) [Save & quit](#) [Discard & quit](#)

File... Action... Data...   Typeset

 [Print](#) [Worksheet](#) [Edit](#) [Text](#) [Undo](#) [Share](#) [Publish](#)

```
var('a b c d e f x y')
```

```
(a, b, c, d, e, f, x, y)
```

```
show(solve(a*x^2 + b*x + c == 0, x)[0])
```

$$x = -\frac{b + \sqrt{-4ac + b^2}}{2a}$$

```
show(solve(x^3 + a*x + b == 0, x)[0])
```

$$x = \frac{(-i\sqrt{3} + 1)a}{6 \left(\frac{1}{18}\sqrt{4a^3 + 27b^2}\sqrt{3} - \frac{1}{2}b\right)^{\frac{1}{3}}} - \frac{1}{2}(i\sqrt{3} + 1) \left(\frac{1}{18}\sqrt{4a^3 + 27b^2}\sqrt{3} - \frac{1}{2}b\right)^{\frac{1}{3}}$$

```
solve([a*x + b*y == c, d*x + e*y == f], x, y)
```

```
[[x == -(b*f - c*e)/(a*e - b*d), y == (a*f - c*d)/(a*e - b*d)]]
```

[evaluate](#)

# Excel?

- Najpopularniejszy pakiet do obliczeń
- Bardzo prosty interfejs
- Często stosowany również w bio-informatyce
- Ma spore ograniczenia (np. Maksymalna liczba linii w arkuszu), które utrudniają rozwój projektów prowadzonych w arkuszu
- Brak możliwości efektywnego testowania,
- brak debuggerów