

# Architektura dużych projektów bioinformatycznych

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

Bartek Wilczyński

10.4.2019

# Plan na przyszły tydzień: quiz

- Kto używał debuggerów: pdb, pycharm, idle
- Kto używał profilerów: profile, cProfile, kernprofiler, runsnakerun
- Kto używał narzędzi do benchmarkowania: timeit, pystone, etc.
- Kto używał modułu inspect
- Kto używał systemów do logowania/raportowania błędów (logging, warnings)?

# 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

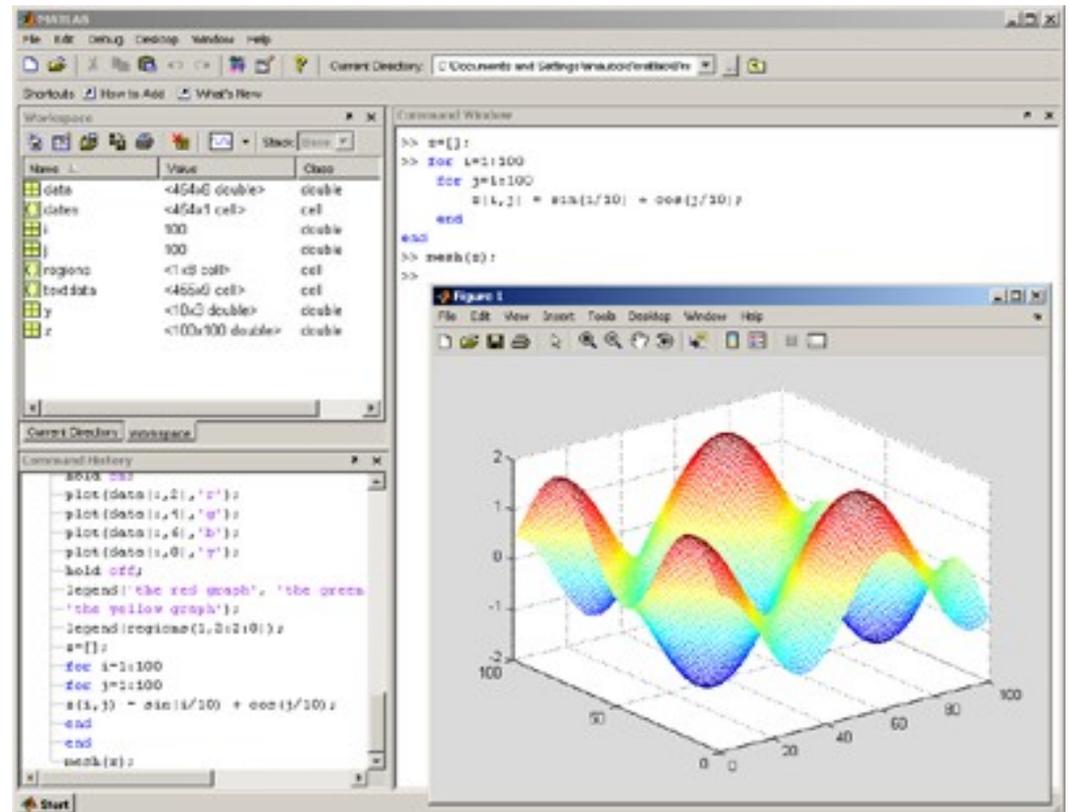
- Komputer jako “potężniejszy kalkulator”
- W zasadzie wszystko można zaprogramować samemu, ale każdemu mogą się przydać:
  - Interfejs użytkownika łatwiejszy niż typowego kompilatora
  - Możliwość zaawansowanej grafiki
  - Dobrze przetestowane standardowe procedury
  - Interfejsy do urządzeń
  - Wsparcie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

# Matlab i pakiety “inżynierskie”

- Rozwijany w latach 70'tych przez Clive 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

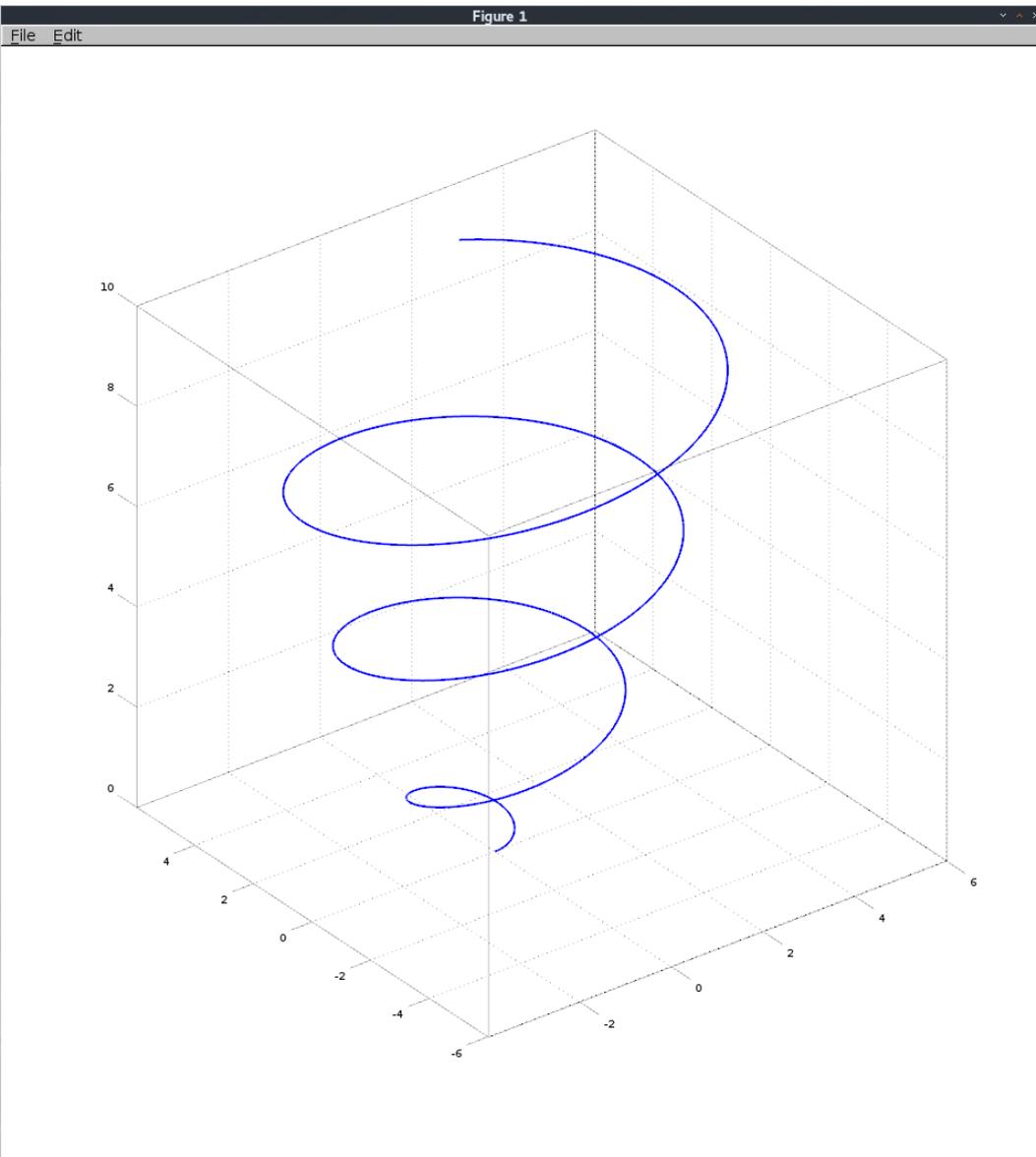
- Wiele dodatkowych (płatnych) bibliotek dla specjalistów
  - Symbolic math
  - Image processing
  - Financial toolbox
  - Bioinformatics
  - Optimization
  - SimBiology



# 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, firma, która czerpie dochody z konsultingu Octave'a
- SciLab – dawniej psilab, rozwijany od lat 90tych w INRIA we Francji, przekształcony w spółkę Scilab enterprises, przejęty w 2017 przez ESI Group, dostępny na licencji GNU
- 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 konsulting
  - 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 a Python script. The script includes comments and code for plotting a radar chart. The code is as follows:

```
27 -----  
28 num_vars = int  
29         Number of variables for radar chart.  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000
```

The interface includes a "File Browser" on the left, a "Documentation Browser" on the right, and a "Welcome to Canopy" dialog box in the foreground. The dialog box contains the Canopy logo, a welcome message, and buttons for "Editor", "Package Manager", and "Doc Browser". Below the dialog box, there is a "Recent files" section with a list of files and buttons for "Restore previous session" and "Open an existing file". The bottom of the interface shows a command prompt with the following commands and output:

```
Type "?" for some information.  
In [1]: !run "/var/folders/tr/pfk3lqg4ndrbppjwep_cp4000@gn/T/bepfftu0ll.py"  
In [2]: from mayavi import mlab  
In [3]: mlab.test_plot3d()  
Out[3]: <mayavi.modules.surface.Surface at 0x122be44d0>  
In [4]:
```

The "Documentation Browser" window shows the "Canopy User Guide" and "Online Help" sections. The "Online Help" section lists various Python libraries and frameworks, including Python Tutorial, Python Documentation, IPython, NumPy, SciPy, Traits, TraitsUI, Enaml, Envisage, Chaco, Mayavi, and Matplotlib Gallery. The "Mayavi Scene 1" window in the bottom right corner displays a 3D visualization of a complex, multi-colored, swirling structure.

# 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 specjalizacji w jakiejś dziedzinie zastosowań

# 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 interface with the following components:

- Source Editor:** Contains R code for loading data, summarizing it, and creating a ggplot2 scatter plot titled "Diamond Pricing".
- Console:** Shows the execution output, including summary statistics for the 'diamonds' dataset and the execution of the plotting commands.
- Workspace:** Lists the loaded data object 'diamonds' (53940 observations) and the 'p' object (a ggplot object).
- Plots Panel:** Displays the resulting scatter plot of Price vs. Carat, colored 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 Panel:** The plot is titled "Diamond Pricing" and shows Price on the y-axis (0 to 15000) and Carat on the x-axis (0.0 to 3.5). 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)

# 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

# Mathematica - interfejs

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

**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[nF[y, z] * f[1, y, z], {y, 0, 1}], {z, 0, 1}]]

```

0.0138889

0.0138889

0.0277778

0.0277778

0.00694444

**AnalysisII-Sperb-SS01/U9.nb Window:**

- Text:**

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 plane running from  $-\infty$  to 0.
- Code:**

```

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]}]

```
- Figure:** A 2D plot showing four Bessel functions:  $J_0(x)$  (red),  $J_1(x)$  (green),  $J_2(x)$  (blue), and  $J_9(x)$  (magenta) over the interval  $x \in [0, 20]$ . The y-axis ranges from -0.4 to 1.0.

**Help Browser:**

- 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 Window:**

- Code:**

```

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

```
- Figure:** A 3D surface plot showing a wavy surface with a grid. The x-axis ranges from 0 to 20, the y-axis from 0 to 15, and the z-axis from -1 to 1.

# Obliczenia symboliczne Open Source

- Maxima (1992-), a wcześniej Macsyma (1968-1982)
- Wydana w 1998 na licencji GPL
- Napisana w języku lisp
- Wiele konkurencyjnych interfejsów (WXMaxima, Gmaxima itp)
- Maxima skupiona na obliczeniach symbolicznych, bez większej funkcjonalności w numeryce

# SAGE math cloud

(dawniej sage notebook)

- Stosunkowo nowy projekt (inny niż sage synapse)
- 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... sage  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)]]
```

# 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

**WHEN IT COMES TO DATA  
ANALYSIS**

**WE EXCEL**

COMMENT

Open Access



# Gene name errors are widespread in the scientific literature

Mark Ziemann<sup>1</sup>, Yotam Eren<sup>1,2</sup> and Assam El-Osta<sup>1,3\*</sup>

## Abstract

The spreadsheet software Microsoft Excel, when used with default settings, is known to convert gene names to dates and floating-point numbers. A programmatic scan of leading genomics journals reveals that approximately one-fifth of papers with supplementary Excel gene lists contain erroneous gene name conversions.

**Keywords:** Microsoft Excel, Gene symbol, Supplementary data

**Abbreviations:** GEO, Gene Expression Omnibus; JIF, journal impact factor

The problem of Excel software (Microsoft Corp., Redmond, WA, USA) inadvertently converting gene symbols to dates and floating-point numbers was originally described in 2004 [1]. For example, gene symbols such as *SEPT2* (Septin 2) and *MARCH1* [Membrane-Associated Ring Finger (C3HC4) 1, E3 Ubiquitin Protein Ligase] are converted by default to ‘2-Sep’ and ‘1-Mar’, respectively. Furthermore, RIKEN identifiers were described to be automatically converted to floating point numbers (i.e. from accession ‘2310009E13’ to ‘2.31E+13’). Since that report, we have uncovered further instances where

frequently reused. Our aim here is to raise awareness of the problem.

We downloaded and screened supplementary files from 18 journals published between 2005 and 2015 using a suite of shell scripts. Excel files (.xls and .xlsx suffixes) were converted to tabular separated files (tsv) with *ssconvert* (v1.12.9). Each sheet within the Excel file was converted to a separate tsv file. Each column of data in the tsv file was screened for the presence of gene symbols. If the first 20 rows of a column contained five or more gene symbols, then it was suspected to be a list of gene symbols, and then a regular expression (regex) search of the entire column was applied to identify gene symbol errors. Official gene symbols from Ensembl version 82, accessed November 2015, were obtained for *Arabidopsis thaliana*, *Caenorhabditis elegans*, *Drosophila melanogaster*, *Danio rerio*, *Escherichia coli*, *Gallus gallus*, *Homo sapiens*, *Mus musculus*, *Oryza sativa* and *Saccharomyces cerevisiae* [2]. The regex search used was similar to that described previously by Zeeberg and colleagues [1], with the added screen for dates in other formats (e.g. DD/MM/YY and MM-DD-YY). To expedite analysis of supplementary files from multi-disciplinary journals, we limited the articles screened to those that have the keyword ‘genome’ in the title or abstract (*Science*, *Nature* and *PLoS One*). Excel files (.xls and .xlsx) deposited in NCBI Gene Expression Omnibus (GEO) [3] were also

**Table 1** Results of the systematic screen of supplementary Excel files for gene name conversion errors

Journal <sup>a</sup>	Number of Excel files screened	Number of gene lists found	Number of papers with gene lists	Number of supplementary files affected	Number of papers affected	Number of gene names converted
<i>PLoS One</i>	7783	2202	994	220	170	4240
<i>BMC Genomics</i>	11464	1650	801	218	158	4932
<i>Genome Res</i>	2607	580	251	114	68	3180
<i>Nucleic Acids Res</i>	2117	540	315	88	67	1661
<i>Genome Biol</i>	2678	664	257	97	63	1878
<i>Genes Dev</i>	932	395	190	75	55	1593
<i>Hum Mol Genet</i>	980	372	168	48	27	1724
<i>Nature</i>	482	150	74	27	23	1375
<i>BMC Bioinformatics</i>	1790	235	152	26	21	534
<i>RNA</i>	569	127	77	20	15	1341
<i>Nat Genet</i>	264	70	37	12	9	178
<i>Bioinformatics</i>	731	112	67	11	6	339
<i>PLoS Comput Biol</i>	177	79	32	6	6	46
<i>PLoS Biol</i>	143	54	29	7	5	206
<i>Mol Biol Evol</i>	995	112	79	7	4	56
<i>Science</i>	172	36	19	7	3	451
<i>Genome Biol Evol</i>	490	32	25	2	2	121
<i>DNA Res</i>	801	57	30	2	2	6
<i>Total</i>	35175	7467	3597	987	704	23861

<sup>a</sup>The 18 journals investigated are ordered by the number of papers affected by gene name conversion errors