

# An introduction to Sage

Franco Saliola  
<saliola@gmail.com>

Fields Institute

3 May 2010

## Outline

- 1 What is Sage?
- 2 History
- 3 Community
- 4 Some useful features

# What is Sage?

Sage is . . .

a *distribution* of software

Sage is a *distribution* of software

When you install Sage, you get:

ATLAS	Automatically Tuned Linear Algebra Software
BLAS	Basic Fortran 77 linear algebra routines
Bzip2	High-quality data compressor
Cddlib	Double Description Method of Motzkin
Common Lisp	Multi-paradigm and general-purpose programming lang.
CVXOPT	Convex optimization, linear programming, least squares
Cython	C-Extensions for Python
F2c	Converts Fortran 77 to C code
Flint	Fast Library for Number Theory
FpLLL	Euclidian lattice reduction
FreeType	A Free, High-Quality, and Portable Font Engine

Sage is a *distribution* of software

When you install Sage, you get:

G95	Open source Fortran 95 compiler
GAP	Groups, Algorithms, Programming
GD	Dynamic graphics generation tool
Genus2reduction	Curve data computation
Gfan	Gröbner fans and tropical varieties
Givaro	C++ library for arithmetic and algebra
GMP	GNU Multiple Precision Arithmetic Library
GMP-ECM	Elliptic Curve Method for Integer Factorization
GNU TLS	Secure networking
GSL	Gnu Scientific Library
JsMath	JavaScript implementation of LaTeX

Sage is a *distribution* of software

When you install Sage, you get:

IML	Integer Matrix Library
IPython	Interactive Python shell
LAPACK	Fortan 77 linear algebra library
Lcalc	L-functions calculator
Libgcrypt	General purpose cryptographic library
Libgpg-error	Common error values for GnuPG components
Linbox	C++ linear algebra library
Matplotlib	Python plotting library
Maxima	computer algebra system
Mercurial	Revision control system
MoinMoin	Wiki

Sage is a *distribution* of software

When you install Sage, you get:

MPFI	Multiple Precision Floating-point Interval library
MPFR	C library for multiple-precision floating-point computations
ECLib	Cremona's Programs for Elliptic curves
NetworkX	Graph theory
NTL	Number theory C++ library
Numpy	Numerical linear algebra
OpenCDK	Open Crypto Development Kit
PALP	A Package for Analyzing Lattice Polytopes
PARI/GP	Number theory calculator
Pexpect	Pseudo-tty control for Python
PNG	Bitmap image support



Sage is a *distribution* of software

When you install Sage, you get:

PolyBoRi	Polynomials Over Boolean Rings
PyCrypto	Python Cryptography Toolkit
Python	Interpreted language
Qd	Quad-double/Double-double Computation Package
R	Statistical Computing
Readline	Line-editing
Rpy	Python interface to R
Scipy	Python library for scientific computation
Singular	fast commutative and noncommutative algebra
Scons	Software construction tool
SQLite	Relation database

Sage is a *distribution* of software

When you install Sage, you get:

Sympow	L-function calculator
Symmetrca	Representation theory
Sympy	Python library for symbolic computation
Tachyon	lightweight 3d ray tracer
Termcap	for writing portable text mode applications
Twisted	Python networking library
Weave	Tools for including C/C++ code within Python
Zlib	Data compression library
ZODB	Object-oriented database

Sage is a *distribution* of software

When you install Sage, you get:

Sympow	L-function calculator
Symmetrca	Representation theory
Sympy	Python library for symbolic computation
Tachyon	lightweight 3d ray tracer
Termcap	for writing portable text mode applications
Twisted	Python networking library
Weave	Tools for including C/C++ code within Python
Zlib	Data compression library
ZODB	Object-oriented database

*... and more!*



```
> sage -singular
```

```
                SINGULAR                               /  Development
A Computer Algebra System for Polynomial Computations /  version 3-1-0
                by: G.-M. Greuel, G. Pfister, H. Schoenemann 0<
FB Mathematik der Universitaet, D-67653 Kaiserslautern  \  Mar 2009
>  \
```

```
> sage -maxima
```

```
Maxima 5.16.3 http://maxima.sourceforge.net
```

```
Using Lisp ECL 9.4.1
```

```
Distributed under the GNU Public License. See the file COPYING.
```

```
Dedicated to the memory of William Schelter.
```

```
The function bug_report() provides bug reporting information.
```

```
(%i1)
```

```
> sage -gp
```

```
GP/PARI CALCULATOR Version 2.3.3 (released)
amd64 running linux (x86-64/GMP-4.2.1 kernel) 64-bit version
compiled: Jul 10 2009, gcc-4.3.2 (Ubuntu 4.3.2-1ubuntu12)
(readline v5.2 enabled, extended help available)
```

```
Copyright (C) 2000-2006 The PARI Group
```

```
PARI/GP is free software, covered by the GNU General Public License, and
comes WITHOUT ANY WARRANTY WHATSOEVER.
```

```
Type ? for help, \q to quit.
```

```
Type ?12 for how to get moral (and possibly technical) support.
```

```
parisize = 8000000, primelimit = 500000
```

```
?
```

```
> sage -R
```

```
R version 2.6.1 (2007-11-26)
```

```
Copyright (C) 2007 The R Foundation for Statistical Computing  
ISBN 3-900051-07-0
```

```
R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.
```

```
    Natural language support but running in an English locale
```

```
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.
```

```
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.
```

```
>
```



Sage is . . .

a distribution of software  
for *mathematics research*

# Sage is software for mathematics research

Algebra	GAP, Maxima, Singular, ...
Exact linear algebra	Linbox, IML, ...
Numerical linear algebra	GSL, Scipy, Numpy, ...
Arbitrary precision arithmetic	GMP, MPFR, MPFI, NTL, ...
Calculus	Maxima, Sympy, ...
Combinatorics	Symmetrca, *-combinat, ...
Algebraic geometry	Singular, ...
Arithmetic geometry	PARI, NTL, mwrnk, ecm, ...
Graph theory	NetworkX, ...
Group theory	GAP, ...
⋮	⋮

Sage *combines* the power of many  
existing software.

# Sage combines software

[This example is from a talk by William Stein]

# Sage combines software

[This example is from a talk by William Stein]

Construct an elliptic curve using *John Cremona's table*:

```
sage: E = EllipticCurve('389a')
```

# Sage combines software

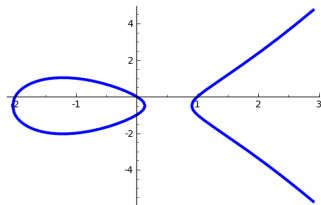
[This example is from a talk by William Stein]

Construct an elliptic curve using *John Cremona's table*:

```
sage: E = EllipticCurve('389a')
```

Use *matplotlib* to plot it:

```
sage: plot(E,thickness=3)
```



# Sage combines software

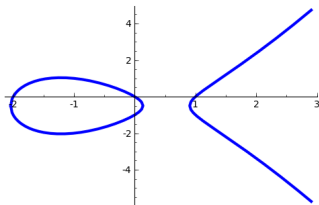
[This example is from a talk by William Stein]

Construct an elliptic curve using *John Cremona's table*:

```
sage: E = EllipticCurve('389a')
```

Use *matplotlib* to plot it:

```
sage: plot(E, thickness=3)
```



*mwrnk* to do a 2-descent:

```
sage: E.mwrnk()
```

```
Curve [0, 1, 1, -2, 0] : Rank = 2
```

# Sage combines software

*PARI* to compute Fourier coefficients  $a_n$ :

```
sage: E.anlist(15)
```

```
[0, 1, -2, -2, 2, -3, 4, -5, 0, 1, 6, -4, -4, -3, 10, 6
```



## Sage combines software

*PARI* to compute Fourier coefficients  $a_n$ :

```
sage: E.anlist(15)
[0, 1, -2, -2, 2, -3, 4, -5, 0, 1, 6, -4, -4, -3, 10, 6]
```

*lcalc* to compute zeros in the critical strip of the L-series:

```
sage: E.lseries().zeros(5)
[0.000000000, 0.000000000, 2.87609907, 4.41689608, 5.79
```

## Sage combines software

*PARI* to compute Fourier coefficients  $a_n$ :

```
sage: E.anlist(15)
[0, 1, -2, -2, 2, -3, 4, -5, 0, 1, 6, -4, -4, -3, 10, 6]
```

*lcalc* to compute zeros in the critical strip of the L-series:

```
sage: E.lseries().zeros(5)
[0.000000000, 0.000000000, 2.87609907, 4.41689608, 5.79
```

*sympow* to compute the modular degree:

```
sage: E.modular_degree()
40
```

## Sage combines software

*PARI* to compute Fourier coefficients  $a_n$ :

```
sage: E.anlist(15)
[0, 1, -2, -2, 2, -3, 4, -5, 0, 1, 6, -4, -4, -3, 10, 6]
```

*lcalc* to compute zeros in the critical strip of the L-series:

```
sage: E.lseries().zeros(5)
[0.000000000, 0.000000000, 2.87609907, 4.41689608, 5.79
```

*sympow* to compute the modular degree:

```
sage: E.modular_degree()
40
```

*Magma* to compute the rank of the 3-selmer group:

```
sage: magma(E).ThreeSelmerGroup()
```

# Sage combines software

*“We implement all conversion routines, instead of expecting upstream to do it: we make them communicate with Sage, whether they want to or not. Resistance is futile.”*

*—William Stein*

Sage is a *new library* of software:  
it provides new or improved  
functionality not previously available.

Sage uses Python  
as its programming language.

## Sage uses Python

*Python* is an interpreted, modern and powerful programming language.

## Sage uses Python

*Python* is an interpreted, modern and powerful programming language.

*Interpreted* means Sage behaves like MuPAD, Maple, Mathematica, ...

```
sage: 2 + 3  
5
```



## Sage uses Python

- Sage  $\approx$  Python + a huge Python library
- Python is one of top 5 most used programming languages, with millions of users.
- Tens of thousands of third party Python packages are immediately available for use with Sage.
- Sage may be the first successful math software system to not invent its own new language just for mathematics.

## Sage uses Python

*Python is easy to read:*

*python:*

```
[8*A for A in range(10) if A%2 == 1]
```

*mathematics:*

$$\left\{ 8A \mid A \in \{0, 1, \dots, 9\} \text{ if } A \text{ is odd} \right\}$$

## Sage uses Python

*Python is easy to learn:*

- *Python Tutorial*

`docs.python.org`

- *Dive Into Python*

`www.diveintopython.org`

- *Sage Tutorial*

`sagemath.org/doc/tutorial`

Sage is . . .

a distribution of software  
for mathematics research

*licensed under the GPL*

# Sage is open-source software

You have the freedom:

- to run the program, for any purpose.
- to study how the program works, and adapt it to your needs.
- to redistribute copies so you can help your neighbour.
- to improve the program, and release your improvements to the public, so that the whole community benefits.

## Sage is open-source software

*“You can read Sylow’s Theorem and its proof in Huppert’s book in the library . . . then you can use Sylow’s Theorem for the rest of your life free of charge, but for many computer algebra systems license fees have to be paid regularly . . . .*

*With this situation two of the most basic rules of conduct in mathematics are violated: In mathematics information is passed on free of charge and everything is laid open for checking.”*

*—J. Neubüser (1993)  
(started GAP in 1986)*

## Sage is open-source software

*“I think, fundamentally, open source does tend to be more stable software. It’s the right way to do things. I compare it to **science versus witchcraft**.*

*In science, the whole system builds on people looking at other people’s results and building on top of them.*

*In witchcraft, somebody had a small secret and guarded it—but never allowed others to really understand it and build on it.”*

*—Linus Torvalds*

# Mission

The Sage Project aims to create a viable high-quality and open-source alternative to Magma, Maple, Mathematica, Matlab and MuPAD, and to foster a friendly community of users and developers.



## High-quality code and documentation

All new code is:

- rigorously tested
- well documented
- peer-reviewed

New releases every 2–3 weeks

## Some history of the Sage project

- *1999-2005.* William Stein wrote over 25,000 lines of Magma code for his research. Decided that Magma was a bad long term investment since he couldn't see or modify the internals.
- *Jan. 2005.* William Stein started Sage.
- *Feb. 2005.* Sage version 0.1: a Python library gluing together PARI, Maxima, Python, Singular e GAP.
- *Feb. 2006.* Sage version 1.0 released; and the "first annual" Sage Days workshop.

## Some history of the Sage project

- *Nov. 2007.* Sage won first place in Les Trophées du Libre competition (honours the best existing free software)
- *Dec. 2007.* Sage gets slashdotted:

+ - IT: Open Source 'Sage' Takes Aim at High End Math Software

Posted by [CmdrTaco](#) on Saturday December 08 2007, @12:19PM  
from the [that'll-take-awhile](#) dept.

[coondoggie](#) writes

"A [new open source mathematics program](#) is looking to push aside commercial software commonly used in mathematics education, in large government laboratories and in math-intensive research. The program's backers say the software, called Sage, can do anything from mapping a 12-dimensional object to calculating rainfall patterns under global warming."



▶ software it sage octave math story

## Some history of the Sage project

- *Feb. 2008.* Sage Days 7 on Combinatorics
- *Summer 2008.* MuPAD-Combinat switches to Sage
- \$\$\$: NSF, DoD, Google, Microsoft, Sun, UW, other universities, private donations, . . . , Fields!
- *Current version:* Sage-4.4

## Sage Days!

- Intensive workshop to develop and implement new features and to attract new users and developers.
- Any developed software is made freely available as part of Sage.
- Sufficiently novel algorithms are submitted for publication.

There have been over 20 Sage Days!

## Sage Days in 2010

- Sage Days 19: Seattle, WA (January 2010)
- Sage Days 20: Marseille (February 2010)
- Sage Days 20.25: Montreal (March 2010)
- Sage Days 20.5: Fields Institute (May 2010)
- Sage Days 21: Seattle, WA (June 2010)
- Sage-Combinat/Chevie: France (June 2010)
- Sage Days 22: Berkeley, CA (July 2010)
- Sage Days 23: Leiden, Netherlands (July 2010)
- Sage Days 24: Linz, Austria (July 2010)
- Sage Days 25: Mumbai, India (August 2010)
- Sage Days 26: Kaiserslautern, Germany (August 2010)

# Sage Community

## mailing lists

sage-devel for development

sage-combinat-devel for combinatorics

sage-windows for Microsoft Windows Port

sage-nt for number theory

sage-finance for finance

sage-flame for flame wars

sage-release for releases

sage-edu for education

sage-grid for scientific grid computing

## irc-channel

#sage-devel on freenode.net

## Map of contributors to the Sage project



There are currently 184 contributors in 118 different places.



# Useful features

$\text{\LaTeX}$ 

In this  $\text{\LaTeX}$  file, I typed:

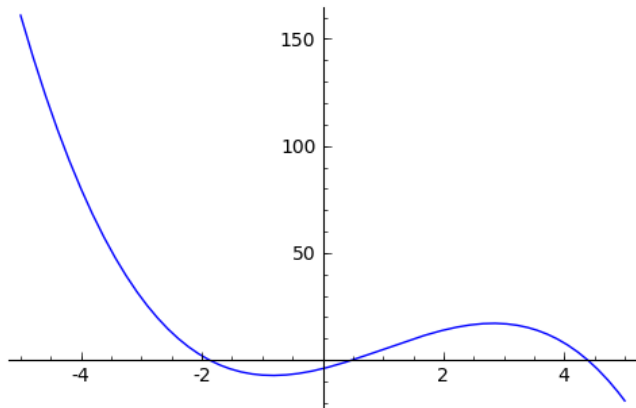
```
\sageplot{plot(-x^3+3*x^2+7*x-4,-5,5)}
```

$\text{\LaTeX}$ 

In this  $\text{\LaTeX}$  file, I typed:

```
\sageplot{plot(-x^3+3*x^2+7*x-4,-5,5)}
```

and it got replaced by:



# L<sup>A</sup>T<sub>E</sub>X

In this L<sup>A</sup>T<sub>E</sub>X file:

```
\begin{sagesilent}
  sigma = Permutation([7,3,1,5,2,6,8,4])
  P, Q = sigma.robinson_schensted()
\end{sagesilent}
```

Let  $\sigma = \text{sage}\{\sigma\}$ . The  
RSK algorithm produces  
the tableaux:

```
\[\text{sage}\{P\} \quad \text{sage}\{Q\}\]
```

L<sup>A</sup>T<sub>E</sub>X

It got replaced with:

Let  $\sigma = [7, 3, 1, 5, 2, 6, 8, 4]$ . The RSK algorithm produces the tableaux:

1	2	4	8
3	5	6	
7			

1	4	6	7
2	5	8	
3			

L<sup>A</sup>T<sub>E</sub>X

It got replaced with:

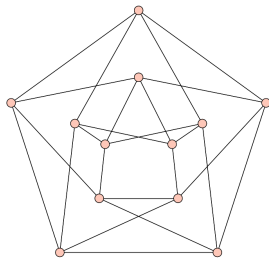
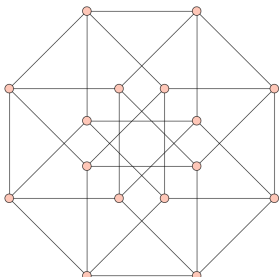
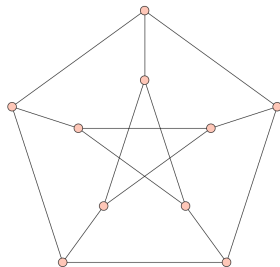
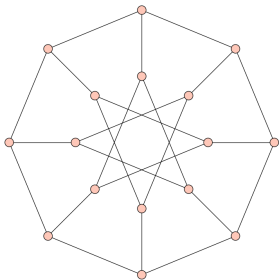
Let  $\sigma = [7, 3, 1, 5, 2, 6, 8, 4]$ . The RSK algorithm produces the tableaux:

1	2	4	8
3	5	6	
7			

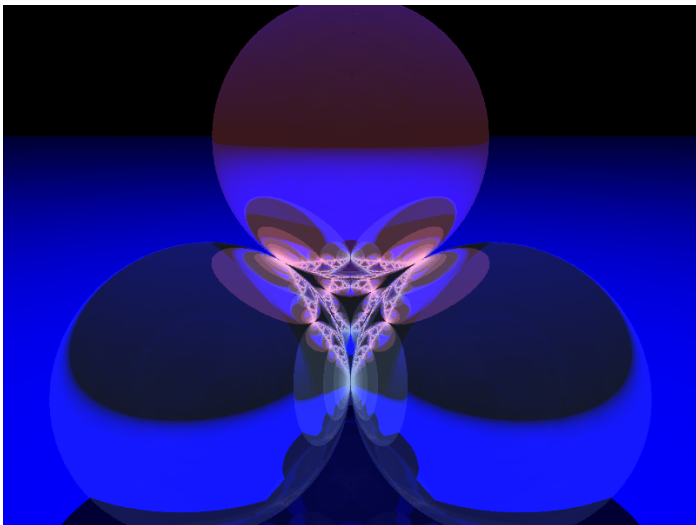
1	4	6	7
2	5	8	
3			

This is done with the *sagetex* package, written by Dan Drake. Of course, the package is included with Sage.

# SageTeX and Graphs

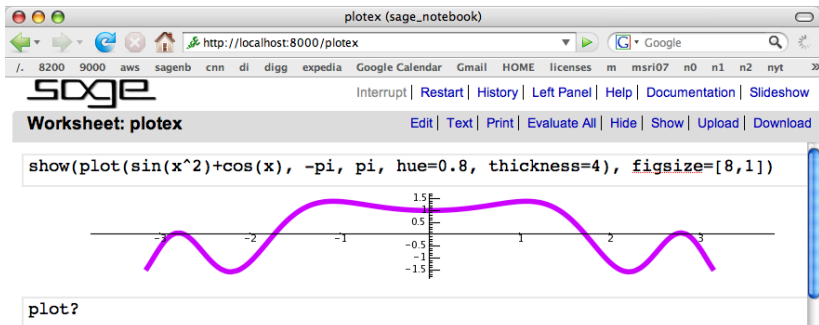


# SageTeX and Tachyon (3D rendering)





# Notebook interface



The screenshot shows a web browser window titled "plotex (sage\_notebook)" with the URL "http://localhost:8000/plotex". The browser's address bar and search bar are visible. Below the browser, the Sage logo is displayed. A navigation bar contains links: "Interrupt", "Restart", "History", "Left Panel", "Help", "Documentation", and "Slideshow". Below this, the text "Worksheet: plotex" is shown, followed by a secondary navigation bar with links: "Edit", "Text", "Print", "Evaluate All", "Hide", "Show", "Upload", and "Download".

The main content area contains a code cell with the following Python code:

```
show(plot(sin(x^2)+cos(x), -pi, pi, hue=0.8, thickness=4), figsize=[8,1])
```

Below the code, a plot is displayed. The plot shows a magenta curve on a coordinate system. The x-axis ranges from approximately -3.5 to 3.5, with major ticks at -3, -2, -1, 1, 2, and 3. The y-axis ranges from -1.5 to 1.5, with major ticks at -1.5, -1, -0.5, 0.5, 1, and 1.5. The curve starts at approximately (-3.5, -1.5), crosses the x-axis at x ≈ -3.14, reaches a local maximum at x ≈ -1.5, a local minimum at x ≈ 0.5, and crosses the x-axis again at x ≈ 3.14, ending at approximately (3.5, -1.5).

Below the plot, there is a text cell containing the text "plot?".

```
Type: <type 'instance'>
Definition: plot( [noargspec] )
Docstring:
```

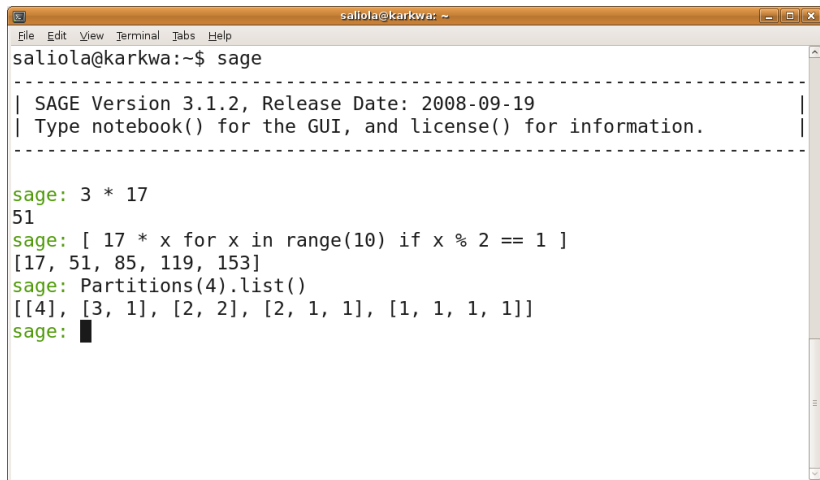
Use plot by writing

```
plot(X, ...)
```

where X is a SAGE object that either is callable and returns numbers that can be coerced to floats, or has a plot method that returns a GraphicPrimitive object.

Type plot.options for a dictionary of the default options for plots. You can change this to change the defaults for all future plots. Use plot.reset() to reset to the default options.

# Command line interface



```
saliola@karkwa: ~  
File Edit View Terminal Tabs Help  
saliola@karkwa:~$ sage  
-----  
| SAGE Version 3.1.2, Release Date: 2008-09-19  
| Type notebook() for the GUI, and license() for information.  
-----  
  
sage: 3 * 17  
51  
sage: [ 17 * x for x in range(10) if x % 2 == 1 ]  
[17, 51, 85, 119, 153]  
sage: Partitions(4).list()  
[[4], [3, 1], [2, 2], [2, 1, 1], [1, 1, 1, 1]]  
sage: █
```

Includes extensive and beautiful documentation

# Yoda!

# @interact!

# Cython