Introduction to the SageMath software

(https://www.sagemath.org)

slides available at
http://www.labri.fr/perso/vdelecro/teaching.html

June 1st 2017, Mini-course HSE Moscow
by Vincent Delecroix, CNRS researcher at LaBRI Bordeaux (France)
Why do you want (good) math softwares?

verify your small computations,
make nice illustrations,
make huge computations, test conjectures,
develop new algorithms,
experimental physics or mathematics, make conjectures.
Why do you want (good) math softwares?

- verify your small computations,
Why do you want (good) math softwares?

- verify your small computations,
- make nice illustrations,
Why do you want (good) math softwares?

- verify your small computations,
- make nice illustrations,
- make huge computations, test conjectures,
- develop new algorithms,
- experimental physics or mathematics, make conjectures.
Why do you want (good) math softwares?

- verify your small computations,
- make nice illustrations,
- make huge computations, test conjectures,
- develop new algorithms,
Why do you want (good) math softwares?

- verify your small computations,
- make nice illustrations,
- make huge computations, test conjectures,
- develop new algorithms,
- experimental physics or mathematics, make conjectures.

```
In [57]: from scipy.integrate import ode
   ...: y0, t0 = [1.0j, 2.0j], 0j
   ...: def f(t, y, arg1): return [1j*arg1*y[0] + y[1], -arg1*y[1]**2]
   ...: def jac(t, y, arg1): return [[1j*arg1, 1j], [0j, -arg1*2j*y[1]]]
   ...: r = ode(f, jac).set_integrator('zvode', method='bdf', with_jacobian=True)
   ...: r.set_initial_value(y0, t0).set_f_params(2.0j).set_jac_params(2.0j)
   ...: t1 = 10j
   ...: dt = 1j
   ...: while r.successful() and r.t < t1:
   ...:     r.integrate(r.t+dt)
   ...:     print("%6g %6g", (r.t, r.y[0]))
```
Why SageMath

Four commercial softwares are dominating the field of mathematical computations: Mathematica, Maple, Matlab, Magma...
Why SageMath

Four commercial softwares are dominating the field of mathematical computations: Mathematica, Maple, Matlab, Magma...

- These softwares are expensive.
Why SageMath

Four commercial softwares are dominating the field of mathematical computations: Mathematica, Maple, Matlab, Magma. . .

- These softwares are expensive.
- No way to verify how they work.
Why SageMath

Four commercial softwares are dominating the field of mathematical computations: Mathematica, Maple, Matlab, Magma...

- These softwares are expensive.
- No way to verify how they work.
- Sometimes impossible to get bugs corrected.
Why SageMath

Four commercial softwares are dominating the field of mathematical computations: Mathematica, Maple, Matlab, Magma. . .

- These softwares are expensive.
- No way to verify how they work.
- Sometimes impossible to get bugs corrected.
- They can disappear.
Why SageMath

Four commercial softwares are dominating the field of mathematical computations: Mathematica, Maple, Matlab, Magma...

- These softwares are expensive.
- No way to verify how they work.
- Sometimes impossible to get bugs corrected.
- They can disappear.

Free mathematical softwares exists since the 80’s like GAP, R, Maxima, PARI/GP. Each of them is specialized. SageMath is an international project started in 2005 and aims to cover a large range of mathematics.
The developer map

One contributor from Russia!? (Alexey U. Gudchenko, contributed 6 years ago)
The developer map

One contributor from Russia!? (Alexey U. Gudchenko, contributed 6 years ago)
Four important ingredients

- open source license
Four important ingredients

open source license

popular programming language
Four important ingredients

- open source license
- popular programming language
- generic notebook interface

V. Delecroix (LaBRI)
Four important ingredients

- open source license
- popular programming language
- generic notebook interface
- on the shoulders of hundreds of free mathematical libraries

»Every free computer algebra system I’ve tried has reinvented many times the wheel without being able to build the car.«
- SageMath is free.
- Can be downloaded from internet https://www.sagemath.org/
- Source code at https://git.sagemath.org/sage.git/
- Contributions open to anyone https://trac.sagemath.org
SageMath is free.

Can be downloaded from internet https://www.sagemath.org/

Source code at https://git.sagemath.org/sage.git/

Contributions open to anyone https://trac.sagemath.org

Freeness will remain forever and is guaranteed by the GNU GPL license.
Python is a very popular programming language that is easy to learn and close to mathematical notation.

The set \( \{ x \in \{1, \ldots, 20\} : 2x^2 - 1 \text{ is prime} \} \) can be constructed in Sage as

\[ [x \text{ for } x \text{ in } [1..20] \text{ if is_prime(2*x^2 - 1)]} \]
Python is a very popular programming language that is easy to learn and close to mathematical notation. The set

\[ \{ x \in \{1, \ldots, 20\} : 2x^2 - 1 \text{ is prime} \} \]

can be constructed in Sage as

\[ [x \text{ for } x \text{ in } [1..20] \text{ if is_prime}(2*x^2 - 1)] \]
Python is a very popular programming language that is easy to learn and close to mathematical notation. The set

$$\{ x \in \{1, \ldots, 20\} : 2x^2 - 1 \text{ is prime} \}$$

can be constructed in Sage as

$$[x \text{ for } x \text{ in } [1..20] \text{ if } \text{is_prime}(2x^2 - 1)]$$

Python is used for many other purposes: web programming, script language, biology, data analysis, etc.
Jupyter is a generic web interface for programming language. It can be used with many different languages and softwares: Sage, PARI/GP, C, C++, etc. The list of kernels can be found at

https://github.com/jupyter/jupyter/wiki/Jupyter-kernels
Sage is built on top of hundreds of scientific libraries and softwares.

- **GMP or MPIR**: C library for arbitrary precision integers
- **flint**: a C library for algebra
- **linbox**: C++ library for exact linear algebra
- **PARI/GP**: a CAS for number theory
- **GAP**: a CAS for group computations
- complete list at
  