

Teaching Mathematics with CAS Sage

Norbert Domes
KHS Fritzlar

Time 2014 - Krems, Austria - Wednesday, July 2

This is an edited version of my lecture.

Content

1. Sage and Sage Notebook
2. Teaching Examples (Worksheets)
3. Discussion and Conclusion

Preliminary note

Looking for an alternative to Derive, I started to work with Sage in the year 2008. In the following years I built up a Sage Notebook Server in the local network of our school and started to use Sage in teaching advanced courses.

My aim is to provide tutorials and examples focused on the maths topics we teach at German upper secondary schools (“gymnasiale Oberstufe”). The outcome of about four years teaching with Sage is documented here: www.mathe-mit-sage.de. The language of this site is German, but hopefully it may be of use even for non-German speakers.

1. Sage and Sage Notebook

1.1 Advantages

Sage is Open Source software and offers all benefits of these systems.

Sage uses the common and easy-to-learn programming language **Python**.

Therefore, a lot of Python modules are included and the Sage code written by the user is *mostly* pure Python code. There are a few (but important) adaptations of mathematical syntax.

Among other adaptations, here are some examples:

Sage	x^2	[1..4]	$f(x) = x+2$
Python	$x^{**}2$	range(1,5)	def f(x): return x+2 (But this python function is different to the symbolic function above.)

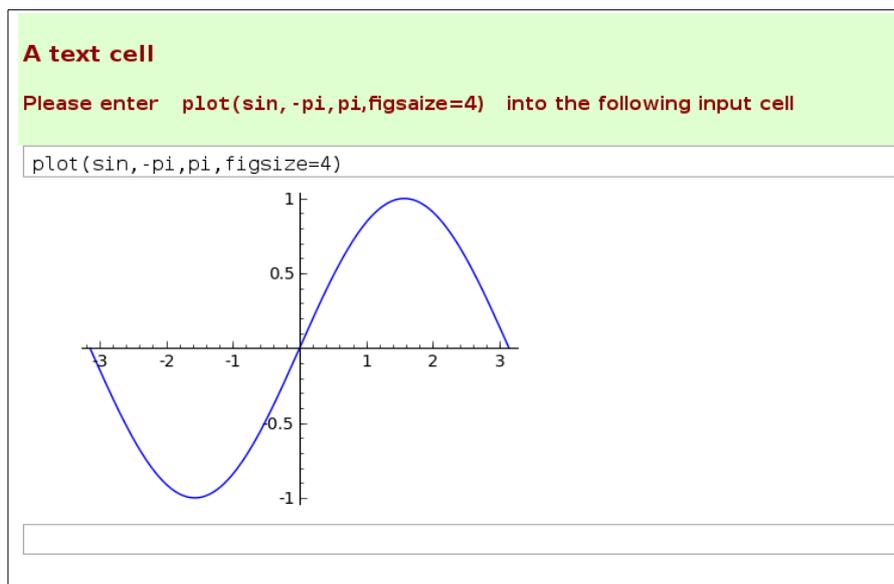
Sage includes numerous powerful maths software packages such as NumPy, SciPy, matplotlib, SymPy, Maxima, GAP, FLINT, R and many more. Sage provides a common user interface to these packages.

1.2 How to run Sage

There are many ways to install and use Sage. Sage installed on a local machine (a virtual box is needed on a windows machine) offers a command line interface. But it is much more comfortable to use the included Sage Notebook Server even with a local installation.

While working with Sage Notebook via a web browser, one can create, store and edit worksheets.

A worksheet is a HTML file containing input cells, output cells and text cells. Text cells may contain (typeset) explanations and instructions and even pictures. The combination of text cells, Sage code and respective output cells allows well-designed HTML files to document and present every form of mathematical work.



1.3 Integrated user management

One of the main advantages of the Sage Notebook is the integrated user management. With a Notebook Server in the local net of a school each student of a course can get a private account and access to the Sage Notebook at any computer of the local net without having to maintain a lot of local installations.

1.4 More Notebook Features

Publish worksheets

A user may publish worksheets (create a public readable copy of a worksheet). The copy can be automatically updated whenever the original worksheet is changed (option). Published worksheets can be transferred to other servers (download and upload). There is an option to enable the execution of “interacts” in published worksheets.

Sharing worksheets

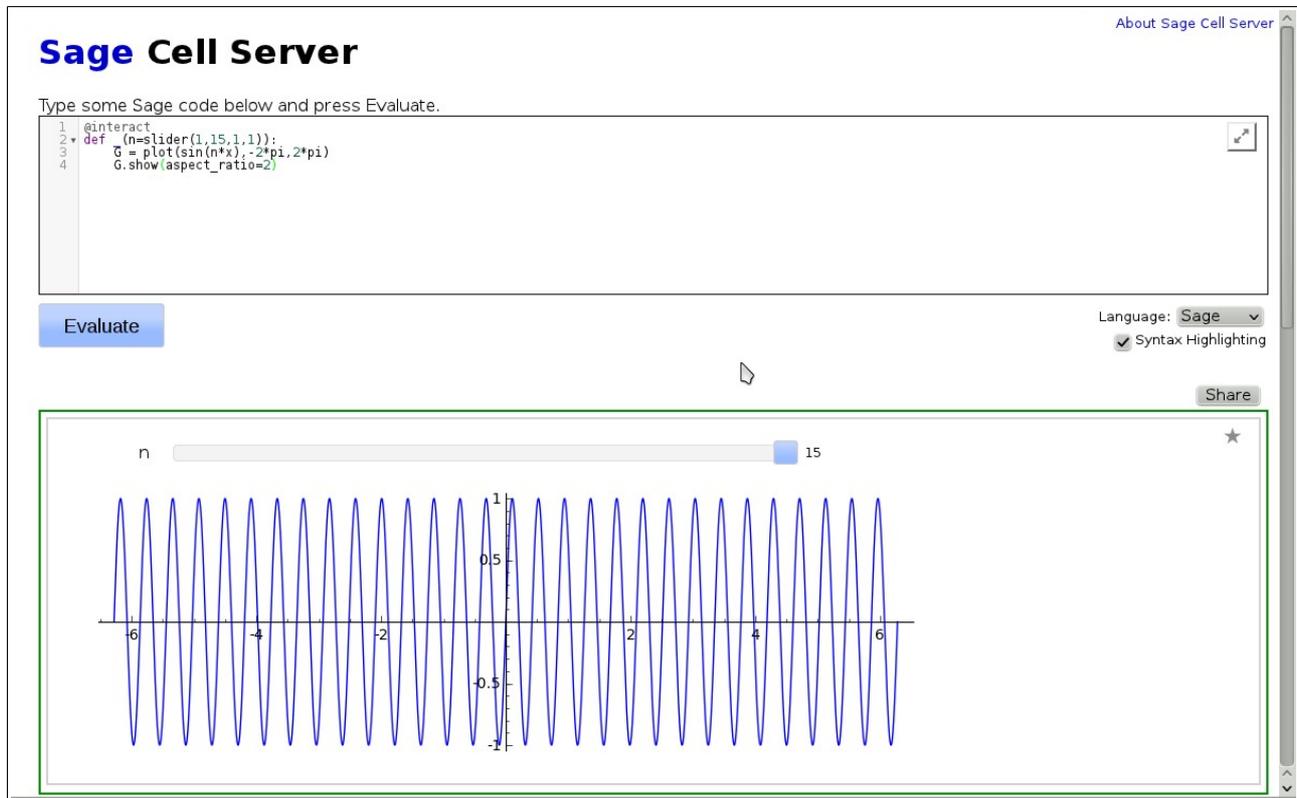
The owner of a worksheet can choose other users as collaborators. Thus, groups of students can share worksheets and work together.

This feature enables the following teacher - student - work-flow :

- The teacher publishes a worksheet.
- The student works with a copy.
- The student shares his copy with the teacher.
- The teacher can write comments and make corrections.
- The teacher may publish exemplary worksheets.

1.5 Recent development

Sage Cell Server



The screenshot shows the Sage Cell Server interface. At the top, it says "Sage Cell Server" and "About Sage Cell Server". Below that, it prompts the user to "Type some Sage code below and press Evaluate." The code editor contains the following Sage code:

```
1 @interact
2 def (n=slider(1,15,1,1)):
3     G = plot(sin(n*x), -2*pi, 2*pi)
4     G.show(aspect_ratio=2)
```

Below the code editor is an "Evaluate" button. To the right, there is a "Language: Sage" dropdown menu and a "Syntax Highlighting" checkbox. A "Share" button is located at the bottom right of the code editor area. The main display area shows a plot of a sine wave with a slider for the parameter 'n' set to 15. The plot has a y-axis ranging from -1 to 1 and an x-axis ranging from -6 to 6. The sine wave is blue and oscillates between -1 and 1. A star icon is visible in the top right corner of the plot area.

<http://sagecell.sagemath.org/>

The Sage Cell Server offers a simple and quick access to Sage. There is no account and no log-in required, just go to the server and enter Sage commands. Sage Cell is a good choice for computations „on the fly“.

You can share your code by creating a permalink to your example.

Open <http://sagecell.sagemath.org/?q=eamgeg> to see the example above.

Therefore it is easy to link to Sage computations in your web files and even to embed executable Sage code in web files.

Sage Cloud Server

<http://cloud.sagemath.org>

Sage Cloud is an advanced alternative to Sage Notebook.

Some advanced features are:

- Collaboratively edit IPython notebooks and Sage worksheets.
- Real-time synchronization enables real-time collaboration.
- Creating projects and folders

Sage Cloud is still in a beta test phase. The response time may depend on the quality of your internet connection.

2. Teaching with Sage

In this chapter, I present a sample of far more than one hundred worksheets I have published on <http://nb.mathe-mit-sage.de>.

- First I would like to present some examples of Sage worksheets I use as teaching tools to visualize mathematical topics.
- Second I display some examples of worksheets to set challenging tasks and to extend the range of maths topics.

Besides this I use Sage to prepare lessons, to check solutions, to create graphics and tasks for exams and much more.

2.1 Visualization

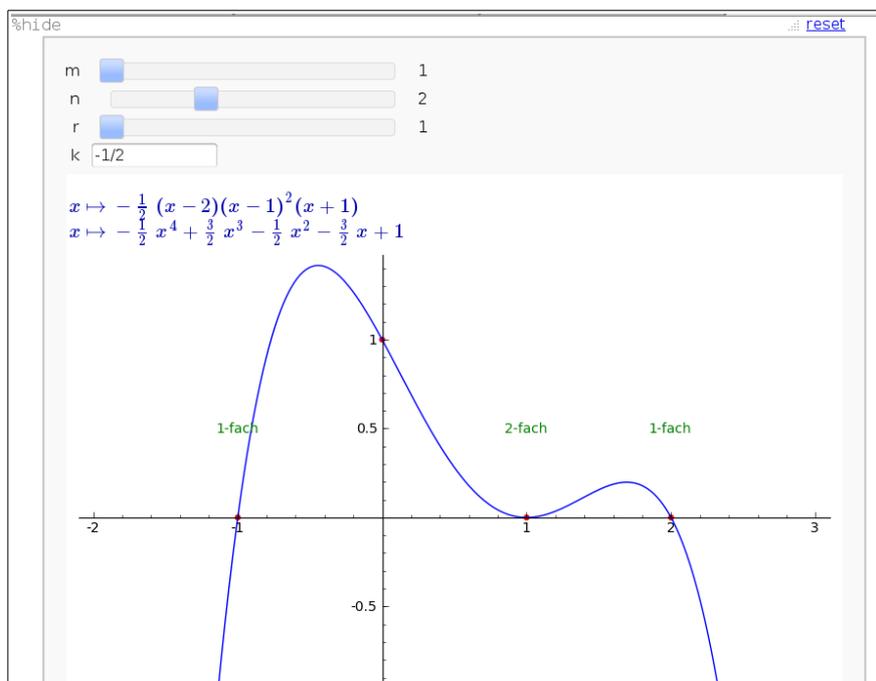
Sage offers an easy way to create interactive controls like sliders as well as animated graphics.

Naturally, there are plenty of animated or interactive graphics available in the net. Some of them offer much smarter interactive features than achievable with Sage (until now). So, where is the benefit to do this with Sage? In my experience, the main advantage is that I can change the worksheet to adapt it to my needs of teaching.

Further - depending on the level of the students - I intentionally refer to the Sage code to exemplify the maths operations behind the graphics.

Example: Multiplicity of roots

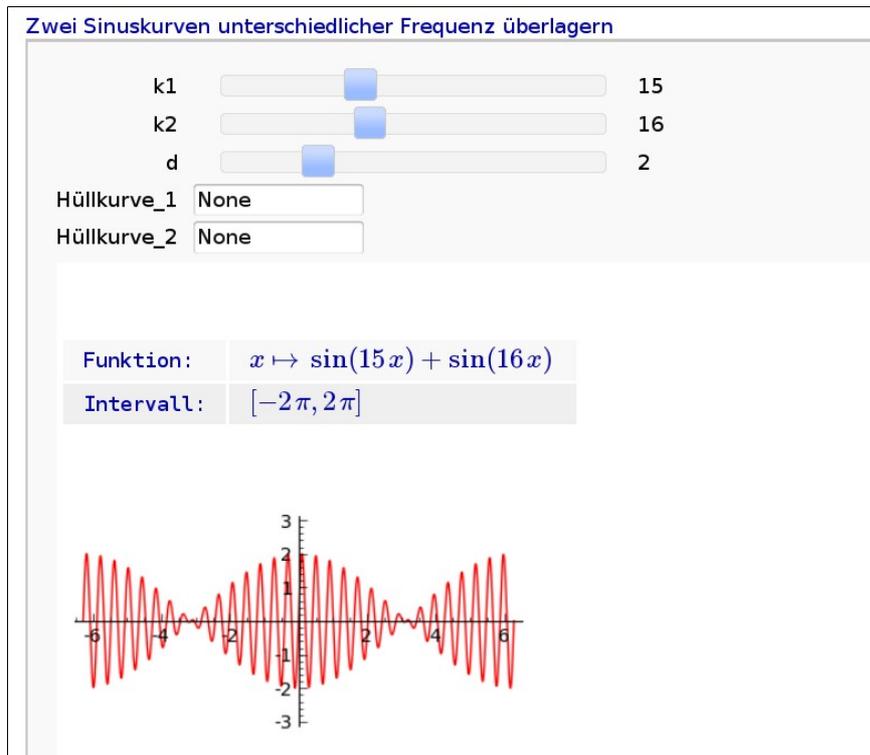
Discuss with students what will happen when changing the exponents, ask for their assumptions, show what happens and talk about the reasons.



<http://nb.mathe-mit-sage.de/home/pub/139/>

Sinus, frequency, amplitude

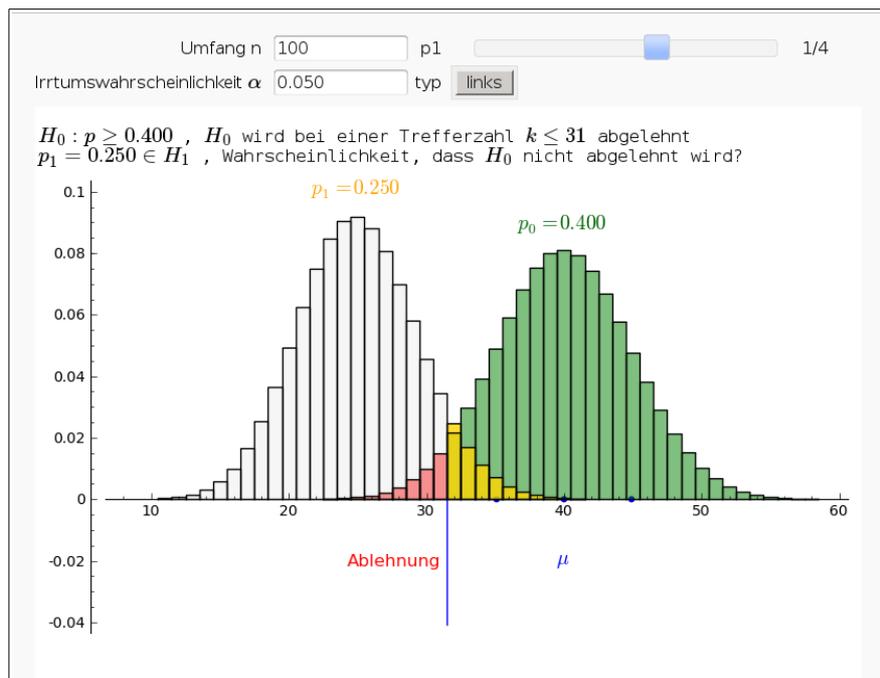
This worksheet provides different interacts to explore the frequency, amplitude or - as shown here - the interference of two sinus waves with a request to find the envelope.



<http://nb.mathe-mit-sage.de/home/pub/38/>

Error of the second kind

Understanding *the error of the second kind* poses a challenge for many students. Tasks of this worksheet are: Dynamically change an assumed probability of the alternative hypothesis or change the sample range to explore the impact of those parameters on the error of the second kind.



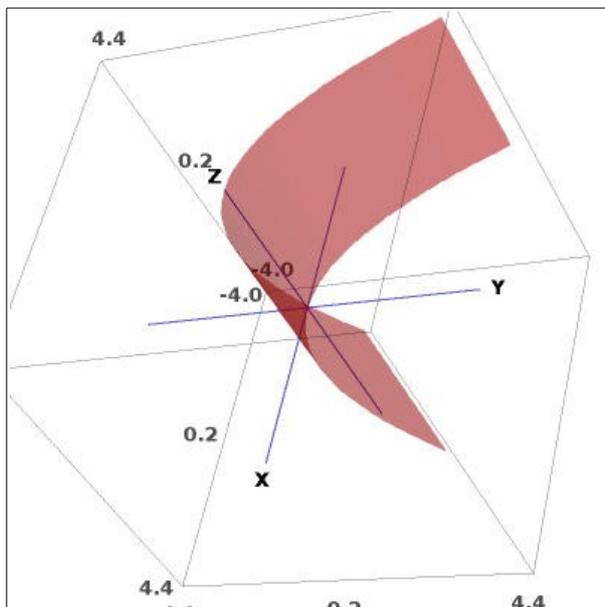
<http://nb.mathe-mit-sage.de/home/pub/102/>

2.2 More Examples - extending range of topics and challenging tasks

Parametric curves (and surfaces)

According to our curriculum we teach parametric representations of straight lines or planes.

The aim of this worksheet is to help students to acquire a deeper understanding of parametric representations.

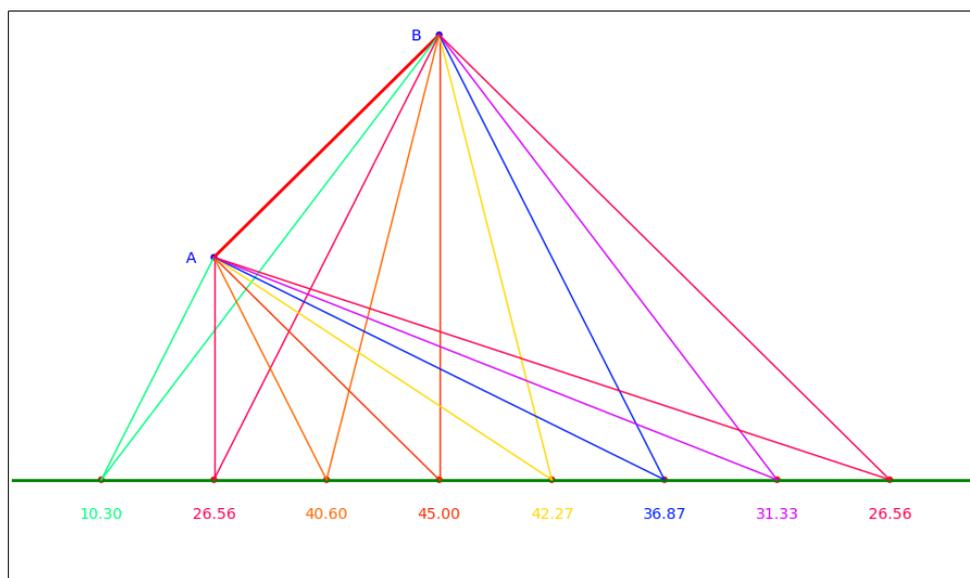


<http://nb.mathe-mit-sage.de/home/pub/169/>

Maximum angle

Find the point on the base line from where you see the points A and B in maximum angle.

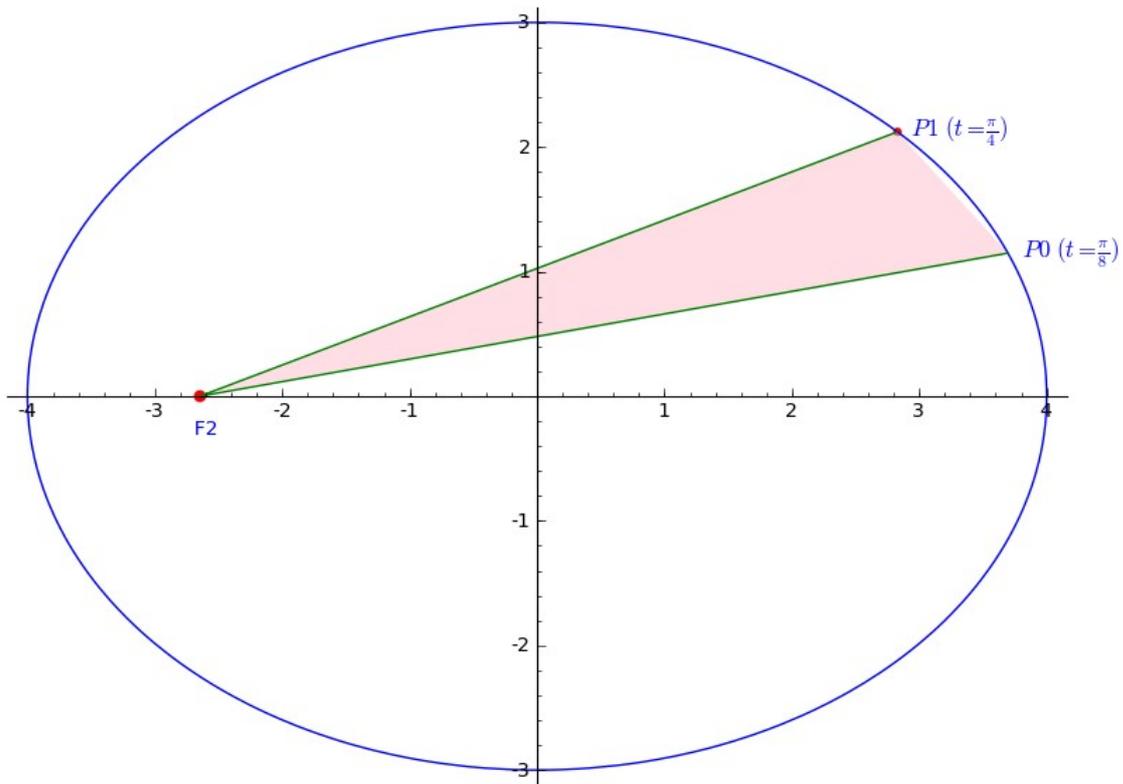
This turns out to be a rather demanding task for students. The worksheet integrates an interactive element with a slider to move the point along the base line. The aim of the worksheet is to motivate students to find a solution.



<http://nb.mathe-mit-sage.de/home/pub/13/>

Kepler's second law

The task is to create an animated graphic to illustrate Kepler's second law. The students suggested an approximate solution. The worksheet takes this suggestion and provides a framework which the students have to finalize.



<http://nb.mathe-mit-sage.de/home/pub/184/>

3. Discussion

3.1 Disadvantages

Sage is not designed as an educational software, it is a huge and powerful tool to do mathematics. Therefore, Sage is not focused on secondary school level, the documentation comes mostly with examples on university level. Students (and maths teachers) may likely be discouraged by the complexity or the sometimes distracting error messages.

Sage is very strict concerning the underlying algebraic structures. When doing symbolic calculations, Sage returns symbolic expressions. Numeric solutions may be complex numbers.

3.2 Unexpected answers

In some cases, Sage returns no solution or the answer differs from the answer of a pocket calculator. Such “unexpected answers” issues are not specific to Sage. But when teaching with Sage it is important to understand some basic ideas of Sage concerning algebraic structures. The teacher should be able to explain what happens and students need some hints how to outwit Sage in these cases.

“Unexpected answers” may be obstacles but on the other hand they provide opportunities to explain mathematics.

Cube root

Until now there is no built-in command in Sage to compute the n-th real root. Expressions such as $(-8)^{\frac{1}{3}}$ will give surprising answers for students who are only used to real numbers.

This worksheet discusses how to deal with the case when Sage returns complex numbers.

Cube Root

$$\sqrt[3]{8} = ? \quad (8)^{\frac{1}{3}}$$

There is no `nth_root` command in sage, so you have to enter a power.

```
8^(1/3)
2
```

That's the answer your pocket calculator will give too.

But what happens now?

$$\sqrt[3]{-8} = ? \quad (-8)^{\frac{1}{3}}$$

```
c = ((-8)^(1/3)); c
2(-1)^(1/3)
```

Your pocket calculator returns -2

Let's ask for a numerical answer.

```
c.n()
1.0000000000000000 + 1.73205080756888i
```

Students often consider maths rules as general rules. They are not used to look at the required conditions. From their point of view the next example displays an unexpected answer.

$a^n \cdot b^n = (ab)^n$
<pre>eqn2 = a^n * b^n == (a*b)^n eqn2 ; bool(eqn2)</pre>
$a^n b^n = (ab)^n$ False
<pre>eqn2a = eqn2.subs(a=-1,b=-2,n=1/2); eqn2a; bool(eqn2a)</pre>
$i\sqrt{-2} = \sqrt{2}$ False
<pre>eqn2a.simplify_full()</pre>
$-\sqrt{2} = \sqrt{2}$

<http://nb.mathe-mit-sage.de/home/pub/187/>

solve(sin(x) == x/2)

Sage tries to find a symbolic (and therefore exact) solution, fails and returns a slightly changed equation. Students have to learn how to cope with such situations, for example by using plots to explore the issue or by switching to numerical solutions.

This worksheet also shows an example of a bug. Of course there are bugs in Sage, and this also can be an occasion to deepen the understanding of mathematics.

9 days ago by ndomes
<pre>eqn = (sin(x) == x/2) solve(eqn, x)</pre>
$[x = 2 \sin(x)]$
<pre>g = plot(x/2, -pi, pi) + plot(sin, -pi, pi) g.show(figsize=(4,3))</pre>
<pre>find_root(eqn, -pi, pi)</pre>
0.0

<http://nb.mathe-mit-sage.de/home/pub/186/>

3.3 How to deal with disadvantages?

Lower the threshold!

- Tutorials and well commented examples for beginners!
There is an increasing number of Sage tutorials dedicated to secondary school level or undergraduate level. Some examples (in German, the last one in English):
 - <http://nb.mathe-mit-sage.de/pub/?section=Anleitungen>
 - http://www.sagemath.org/de/html/thematische_anleitungen/sage_gymnasium.html
 - <http://www-rohan.sdsu.edu/~mosulliv/sagetutorial/sagecalc.html>
- Internationalizing Sage Notebook and Sage documentation.
A lot of work is already done: there are different language versions of the Notebook and the main documentation and reference. The native language helps students (and teachers) to familiarize themselves with Sage. However, it is far too much work to translate all internal doc-strings and error-messages.

Do we need an *educational version of Sage* to hide the underlying algebraic structures and avoid those difficulties as presented above? It would be too much work to create and maintain such a Sage version. As long as Sage is an extensive software and under active development it seems impossible to protect students from all surprises. On the other hand there should be improvements and patches related to teaching at school like a cube root command to increase the benefits of teaching maths with Sage.

3.4 Students Responses

Students approve Sage when it is used by the teacher to give short presentations visualizing mathematics, such as exploring sets of curves, mapping matrices, probability distributions and much more. Only a part of the students (in my experience half of them) like to work with Sage as a tool. One reason is the complexity of Sage, a further reason: they are not used to algorithmic problem solving and to do programming, which requires additional training and practice to reach a level of success. Students do not like to do additional work on top, if it is (from their point of view) not needed for exams. So if they don't have to use Sage in their exams they consider using Sage of minor importance during the lessons.

There is a reciprocal effect due to the fact that many maths teachers have little experience in programming and therefore stay away from complex and open systems such as Sage.

Likely Sage will overburden those students who require much assistance in learning maths.

A statement that reveals a complete refusal:

“ I don't like sage because no one knew how to use the program. It seems to be pretty advanced and it's hard to learn & understand the code on top of the classwork. So on the end of semester projects over more complex topics the sage work was the biggest obstacle because I didn't know how to use to program well.”

Linear Algebra with Sage and “A First Course in Linear Algebra”
Jane Long, Stephen F. Austin State University, Sage Edu Days 4, June 14, 2012

On the other hand I received approving feedback from my students:

“ ... always exciting lessons with Sage in the computer lab...”

„ Your elaborated Sage worksheets and visualizations were particularly helpful...”

“...the visualizations in the computer lab gave us an understanding of mathematics that would not have been provided by other teachers.”

3.5 Conclusion

Sage is a powerful tool for teaching maths. Sage is a good choice for specialised maths workshops, additional study groups, preparatory courses in cooperation of school and university – all of them addressing highly motivated students.

Guidance for beginners is important - especially on secondary school level.

Sage is even a good choice as CAS for advanced courses, provided all students have access to Sage during the lessons and at home and working with Sage is part of tests and exams.

Sage Notebook and alternatively Sage Cloud provide useful collaboration frameworks.

Well-prepared Sage worksheets for presentations and visualisations are in any case recommendable for both advanced and basis courses.

References

<http://www.sagemath.org>

<http://wiki.sagemath.org>

<http://ask.sagemath.org>

<http://www.mathe-mit-sage.de>