Sage-mathematics

From ArchWiki

Sage (http://www.sagemath.org) is a program for numerical and symbolic mathematical computation that uses Python as its main language. It is meant to provide an alternative for commercial programs such as Maple, Matlab, and Mathematica.

Sage provides support for the following:

- **Calculus**: using Maxima and SymPy.
- **Linear Algebra**: using the GSL, SciPy and NumPy.
- **Statistics**: using R (through RPy) and SciPy.
- **Graphs**: using matplotlib.
- An **interactive shell** using IPython.
- Access to **Python modules** such as PIL, SQLAlchemy, etc.

Related articles

Matlab

Octave

Mathematica

Contents

Installation

Sage can be installed from the official repositories:

- `sage-mathematics` (https://www.archlinux.org/packages/?name=sage-mathematics) contains the command-line version;
- `sage-notebook` (https://www.archlinux.org/packages/?name=sage-notebook) includes the web-based notebook interface.

Usage

Sage mainly uses Python as a scripting language with a few modifications (http://www.sagemath.org/doc/tutorial/afterword.html#section-mathannoy) to make it better suited for mathematical computations.

Sage command-line

Sage can be started from the command-line:

```
$ sage
```

For information on the Sage command-line see this page (http://www.sagemath.org/doc/reference/cmd/index.html).

The command-line is based on the IPython shell so you can use all its tricks (http://www.sagemath.org/doc/tutorial/interactive_shell.html) with Sage. For an extensive tutorial on IPython see the community maintained IPython Cookbook (http://wiki.ipython.org/Cookbook).

Note, however, that it is not very comfortable for some uses such as plotting. When you try to plot something, for example:

```
sage: plot(sin,(x,0,10))
```

Sage opens a browser window with the Sage Notebook.

Sage Notebook

A better suited interface for advanced usage in Sage is the Notebook. To start the Notebook server from the command-line, execute:

```
$ sage -n
```

The notebook will be accessible in the browser from http://localhost:8080 and will require you to login.

However, if you only run the server for personal use, and not across the internet, the login will be an annoyance. You can instead start the Notebook without requiring login, and have it automatically pop up in a browser, with the following command:

```
$ sage -c "notebook(automatic_login=True)"
```

For a more comprehensive tutorial on the Sage Notebook see the Sage documentation (http://www.sagemath.org/doc/reference/notebook/index.html). For more information on the `notebook()` command see this page (http://www.sagemath.org/doc/reference/notebook/sagenb/notebook/notebook.html).

Cantor

Cantor (http://edu.kde.org/applications/mathematics/cantor/) is an application included in the KDE Edu Project. It acts as a front-end for various mathematical applications such as Maxima, Sage, Octave, Scilab, etc. See the Cantor page (http://wiki.sagemath.org/Cantor) on the Sage wiki for more information on how to use it with Sage.

Cantor can be installed with the `kdeedu-cantor` (https://www.archlinux.org/packages/?name=kdeedu-cantor) package or as part of the `kde` (https://www.archlinux.org/groups/x86_64/kde/) or `kdeedu` (https://www.archlinux.org/groups/x86_64/kdeedu/) groups, available in the official repositories.

Documentation

For local documentation, one can compile it into multiple formats such as HTML or PDF. To build the whole Sage reference, execute the following command (as root):

```
# sage --docbuild reference html
```

This builds the HTML documentation for the whole *reference* tree (may take longer than an hour). An option is to build a smaller part of the documentation tree, but you would need to know what it is you want. Until then, you might consider just browsing the online reference (http://www.sagemath.org/doc/).

For a list of documents see `sage --docbuild --documents` and for a list of supported formats see `sage --docbuild --formats` .

Optional additions

SageTeX

If you have installed TeX Live on your system, you may be interested in using SageTeX (http://www.sagemath.org/doc/tutorial/sagetex.html), a package that makes the inclusion of Sage code in LaTeX files possible. TeX Live is made aware of SageTeX automatically so you can start using it straight away.

As a simple example, here is how you include a Sage 2D plot in your TEX document (assuming you use `pdflatex`):

- include the `sagetex` package in the preamble of your document with the usual

```
\usepackage{sagetex}
```

- create a `sagesilent` environment in which you insert your code:

```
\begin{sagesilent}
dob(x) = sqrt(x^2 - 1) / (x * arctan(sqrt(x^2 - 1)))
dpr(x) = sqrt(x^2 - 1) / (x * log( x + sqrt(x^2 - 1)))
p1 = plot(dob,(x, 1, 10), color='blue')
p2 = plot(dpr,(x, 1, 10), color='red')
ptot = p1 + p2
ptot.axes_labels(['$\\xi$','$\\frac{R_h}{\\max(a,b)}$'])
\end{sagesilent}
```

- create the plot, e.g. inside a `float` environment:

```
\begin{figure}
\begin{center}
\sageplot[width=\linewidth]{ptot}
\end{center}
\end{figure}
```

- compile your document with the following procedure:

```
$ pdflatex <doc.tex>
$ sage <doc.sage>
$ pdflatex <doc.tex>
```

- you can have a look at your output document.

The full documentation of SageTeX is available on CTAN (http://www.ctan.org/pkg/sagetex).

Troubleshooting

TeX Live does not recognize SageTex

If your TeX Live installation does not find the SageTex package, you can try the following procedure (as root or use a local folder):

- Copy the files to the texmf directory:

```
# cp /opt/sage/local/share/texmf/tex/* /usr/share/texmf/tex/
```

- Refresh TeX Live:

```
# texhash /usr/share/texmf/
texhash: Updating /usr/share/texmf/.//ls-R...
texhash: Done.
```

See also

- Official Website (http://www.sagemath.org/)
- Sage Documentation (http://www.sagemath.org/doc/)
- Planet Sage (http://planet.sagemath.org/)
- Sage Wiki (http://wiki.sagemath.org/)
- Software Used by Sage (http://www.sagemath.org/links-components.html)