

---

# **simple\_plotter Documentation**

***Release 0.0.0***

**Thies Hecker**

**Mar 30, 2020**



---

## Contents:

---

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Getting started</b>                               | <b>3</b>  |
| 1.1      | Desktop . . . . .                                    | 3         |
| 1.2      | Android . . . . .                                    | 3         |
| 1.3      | Configuration options . . . . .                      | 4         |
| 1.4      | Requirements . . . . .                               | 4         |
| 1.5      | Source code . . . . .                                | 4         |
| <b>2</b> | <b>User guide</b>                                    | <b>7</b>  |
| 2.1      | Overview . . . . .                                   | 7         |
| 2.2      | Defining an equation . . . . .                       | 7         |
| 2.3      | Creating a plot . . . . .                            | 8         |
| 2.4      | Adjust the plot appearance . . . . .                 | 8         |
| 2.5      | Constants . . . . .                                  | 9         |
| 2.6      | Plotting curve sets . . . . .                        | 10        |
| 2.7      | Plot labels . . . . .                                | 11        |
| 2.8      | Load, save and export . . . . .                      | 12        |
| 2.9      | Advanced usage . . . . .                             | 12        |
| <b>3</b> | <b>Licenses</b>                                      | <b>17</b> |
| 3.1      | simple-plotter . . . . .                             | 17        |
| 3.2      | simple-plotter-qt . . . . .                          | 18        |
| 3.3      | simple-plotter4a . . . . .                           | 18        |
| 3.4      | simple-plotter4a binary releases (Android) . . . . . | 18        |
| <b>4</b> | <b>Contributing</b>                                  | <b>43</b> |
| 4.1      | Concept . . . . .                                    | 43        |
| 4.2      | Package versions . . . . .                           | 43        |
| 4.3      | Building conda packages . . . . .                    | 43        |
| 4.4      | Testing . . . . .                                    | 44        |
| 4.5      | Project documentation . . . . .                      | 44        |
| <b>5</b> | <b>API reference</b>                                 | <b>45</b> |
| 5.1      | code_generator . . . . .                             | 45        |
| 5.2      | code_parser . . . . .                                | 49        |
| 5.3      | advanced_graph . . . . .                             | 50        |
| <b>6</b> | <b>Roadmap and Issue-tracker</b>                     | <b>53</b> |

|          |                           |           |
|----------|---------------------------|-----------|
| <b>7</b> | <b>Change log</b>         | <b>55</b> |
| 7.1      | Release 0.4.0 . . . . .   | 55        |
| 7.2      | Release 0.3.2 . . . . .   | 55        |
| 7.3      | Release 0.3.1 . . . . .   | 55        |
| 7.4      | Release 0.3.0 . . . . .   | 55        |
| 7.5      | Release 0.2.2 . . . . .   | 56        |
| 7.6      | Release 0.2.1 . . . . .   | 56        |
| 7.7      | Release 0.2.0 . . . . .   | 56        |
| 7.8      | Release 0.1.2 . . . . .   | 56        |
| 7.9      | Release 0.1.1 . . . . .   | 56        |
| 7.10     | Release 0.1.0 . . . . .   | 56        |
| <b>8</b> | <b>Indices and tables</b> | <b>57</b> |
|          | <b>Index</b>              | <b>59</b> |

*simple-plotter* is a code-generator to create python code for plotting functional 2D x,y-plots. The function equation has to be entered in python syntax (allowing the use of numpy statements).

*simple-plotter* provides a command line interface (CLI) tool, which takes simple JSON files as inputs to generate the code.

Graphical user interface (GUI) front-ends for *simple-plotter* are available as well.

The *simple-plotter-qt* package provides with a Qt-based GUI-frontend intended for desktop use.

*simple-plotter4a* provides an alternative kivy-based frontend, created primarily for the Android port (see screen shots below), which can be used on a desktop system as well.

For details on licenses see [Licenses](#) or the NOTICE and LICENSE files in the source code repositories of the individual components.

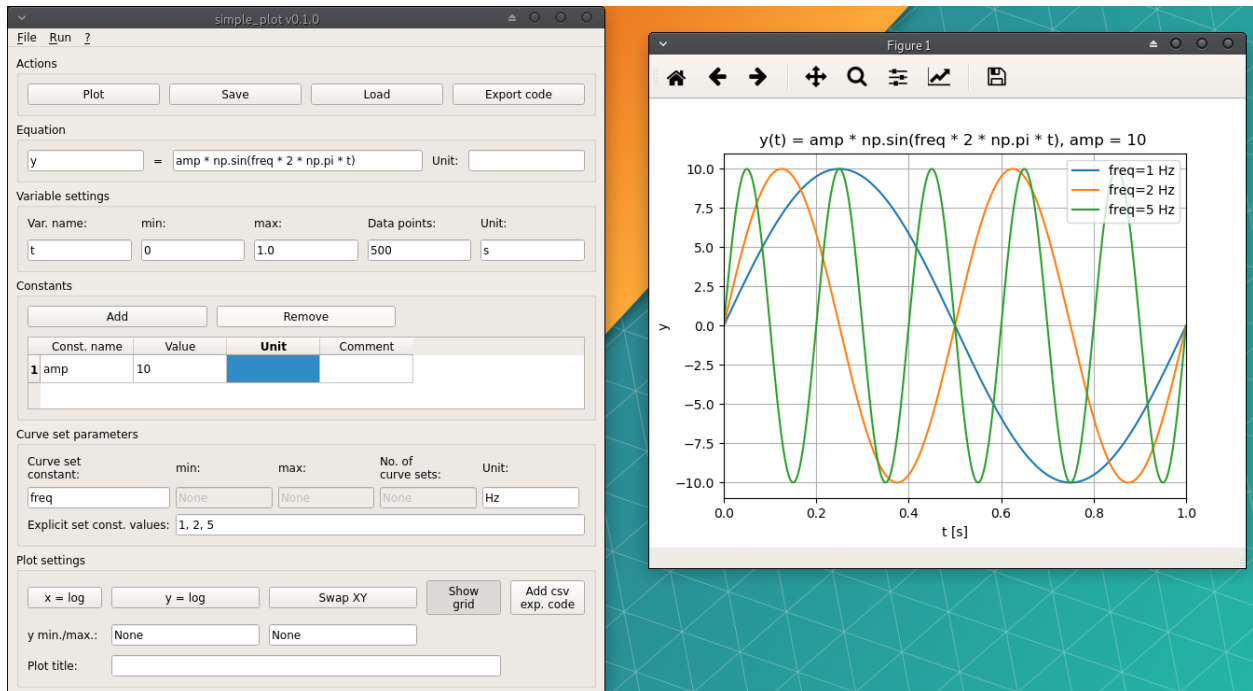


Fig. 1: Screen shot of the Qt-based GUI

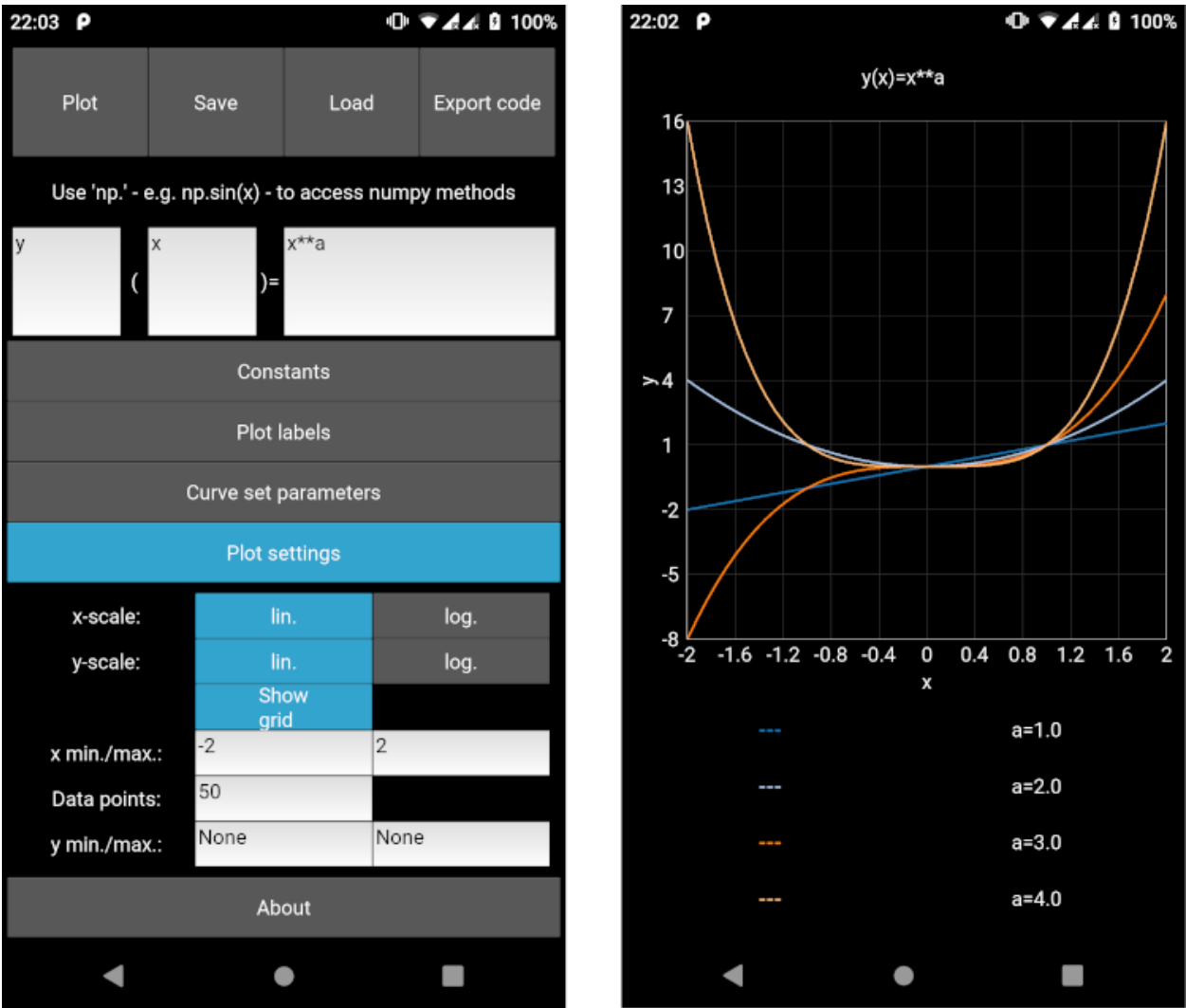


Fig. 2: Screen shot of the kivy-based GUI (running on Android)

This chapter will give a basic instructions for installing and running the program. For detailed usage instructions please see the [User guide](#).

### 1.1 Desktop

As *simple\_plotter* is entirely written in python it should run on any desktop platform with a python3 interpreter and support for the required packages (see [Requirements](#)).

The packages for *simple\_plotter* are available on PyPI. The easiest solution is to install the *simple-plotter-qt* package with the default desktop GUI front-end and plotting library (i.e. PyQt and matplotlib) using *pip*.

Open up a terminal and type:

```
pip install simple-plotter-qt
```

It will automatically install the requirements.

To launch simple-plotter just enter:

```
simple-plotter-qt
```

To install different GUI/plotting library options see [Configuration options](#).

### 1.2 Android

The Android app is available in the [F-Droid](#) app-store [work in progress...].

Alternatively you can build the Android APK from source. Follow the instructions on <https://gitlab.com/thecker/simple-plotter4a>

---

**Note:** Due to a broken matplotlib recipe in the *python-for-android* project the Android app is currently based on the *kivy-garden-graph* configuration - see [Configuration options](#).

---

## 1.3 Configuration options

The table below shows the currently available configuration options.

| Package           | Configuration     | plotting library |              | GUI frame work |      |
|-------------------|-------------------|------------------|--------------|----------------|------|
|                   |                   | matplotlib       | garden.graph | PyQt           | kivy |
| simple-plotter    |                   | x                | x            |                |      |
| simple-plotter-qt |                   | x                |              | x              |      |
| simple-plotter4a  | kivy-matplotlib   | x                |              |                | x    |
| simple-plotter4a  | kivy-garden-graph |                  | x            |                | x    |

The base package (*simple-plotter*) only provides a command line interface (CLI) for the plot-code generator. Use `pip install <Package>[<Configuration>]` to install one of the configuration options, where the [`<Configuration>`] is optional, if available - e.g.:

```
pip install simple-plotter4a[kivy-matplotlib]
```

Similarly you launch the programs with `simple-plotter-<Configuration>` - e.g.:

```
simple-plotter-kivy-matplotlib
```

## 1.4 Requirements

*simple\_plotter* is written in python3 and requires has following dependencies:

Mandatory:

- numpy
- setuptools\_scm
- jinja2

Optional:

- pyqt >= 5
- matplotlib>=2
- kivy>=1.11
- kivy-garden.graph>=0.4

The optional dependencies provide are related to the different [Configuration options](#).

## 1.5 Source code

The source code for the base package can be obtained from:



<https://gitlab.com/thecker/simple-plotter>

For Qt-based GUI see:

<https://gitlab.com/thecker/simple-plotter-qt>

and for the kivy-based GUI see:

<https://gitlab.com/thecker/simple-plotter4a>

If you would like to contribute see *Contributing*.



The following chapters describe the usage for the GUI. For details on the *simple-plotter* file format and the CLI tool see the *Advanced usage* chapter.

The screen shots in this user guide are taken from the Android port *simple-plotter4a*, but all features are available in the Qt-GUI as well.

## 2.1 Overview

*simple-plotter*'s main purpose is to serve as graphical calculator for creating 2D plots of x,y-functions and provides following features:

- define functions in python/NumPy syntax
- it gives access to all NumPy methods
- projects can be saved and restored
- projects can also be exported to python scripts
- constants and curve set parameters can be defined
- axes can be changed to logarithmic scale
- plot annotations (e.g. plot title and labels) are automatically generated

## 2.2 Defining an equation

To define an equation you must specify the values of the three input fields on the top:

|      |      |             |      |
|------|------|-------------|------|
| Save | Load | Export code | Plot |
|------|------|-------------|------|

Use 'np.' - e.g. np.sin(x) - to access numpy methods

|   |   |   |    |   |
|---|---|---|----|---|
| y | ( | x | )= | <code>np.tan(np.radians(alpha)) * x<br/>- g * (x / (np.cos(np.<br/>radians(alpha)) * v0))**2</code> |
|---|---|---|----|---|

**Function name (mandatory)** The function name (e.g. 'f' or 'y') will be displayed on the y-axis of the plot. It has no further effect on the calculation, so it is really just a label for the y-axis.

**Variable name (mandatory)** The name of the variable (e.g. 'x'), which will be displayed on the x-axis of the plot. In contrast to the function name this is not just used for the label. This value must match the variable in the equation field.

**Equation (mandatory)** Here you can define the equation (e.g.  $3x^2 + 2x + 5$ ). As mentioned above you will have to define a variable name in the variable field. Note, that your equation must be formatted in valid python syntax. If you are not familiar with python syntax have a look at the [Using Python as a Calculator](#) chapter in the official python tutorial. You can additionally use any function/constants of the NumPy package by prepending a 'np.' - e.g. your equation could be: `3 * np.sin(x) + 2 * np.cos(x / np.pi)`

## 2.3 Creating a plot

If you have entered the equation definition correctly just press the 'Plot' button and a graph will be rendered for your equation. You can alter the plot appearance in 'Plot settings' section.

## 2.4 Adjust the plot appearance

The plot appearance can be altered in the 'Plot settings' section.

| Plot settings |           |      |
|---------------|-----------|------|
| x-scale:      | lin.      | log. |
| y-scale:      | lin.      | log. |
|               | Show grid |      |
| x min./max.:  | 0         | 1    |
| Data points:  | 500       |      |
| y min./max.:  | None      | None |

You can change following parameters:

**x-scale** Sets the x-axis either to linear scale (default) or to logarithmic scale. Note, that for logarithmic scale only values greater than 0 can be displayed.

**y-scale** Sets the y-axis either to linear scale (default) or to logarithmic scale. Note, that for logarithmic scale only values greater than 0 can be displayed.

**show grid** Shows or hides the grid

**x-min./max. (mandatory)** Defines the bounds of x-axis. Both values are mandatory inputs. Note, that if you set the x-axis to logarithmic scale negative values will be ignored - i.e. the bounds will be re-calculated automatically.

**Data points (mandatory)** Specifies the number of data points used to calculate the x,y-values for each curve. Increase this value to create smoother curves.

**y-min./max. (optional)** Defines the bounds of x-axis. Both values are optional. If not defined they will be adjusted to fit all curve data points into the graph. Note, that if you set the x-axis to logarithmic scale negative values will be ignored - i.e. the bounds will be re-calculated automatically.

## 2.5 Constants

*simple-plotter* allows you to extend your equation with constants - e.g. ' $a*x**2 + b*x + c$ ', where a, b and c are constants.

You can define two different types of constants:

- multiple single valued ('normal') constants - e.g.  $g = 9.81$
- one set constant to create curve sets - i.e. the set constant takes multiple values (see next chapter)

The ‘normal’ constants can be defined in the ‘Constants’ section. You can specify following properties per constant:

Constants

Add

| Const. name | Value | Unit             | Comment          |   |
|-------------|-------|------------------|------------------|---|
| g           | 9.81  | m/s <sup>2</sup> | gravity          | X |
| v0          | 30    | m/s              | initial velocity | X |

**Const. name (mandatory)** Name of the constant (as to be used in the equation).

**Value (mandatory)** Value of the constant. This can be either a simple value - e.g. ‘2.5’ - or python expression - e.g. ‘np.pi / 2’

**Unit (optional)** A unit for the constant - e.g. ‘kg’. This unit will only be used as a label in the plot title.

**Comment (optional)** A comment - e.g. to explain the meaning of the constant like ‘gravity’. This comment will be added as comment line in the python script file, if you export the project via the ‘Export’ button.

## 2.6 Plotting curve sets

If you want to create curve sets you can specify one constant as a curve set constant in the ‘Curve set parameters’ section. *simple-plotter* provides two ways of defining curve set values:

- in terms of a defined number of equidistant values between a min., max. value
- as a list of explicitly defined values

| Curve set parameters        |                            |   |
|-----------------------------|----------------------------|---|
| Set constant name:          | alpha                      |   |
| min./max. value:            |                            |   |
| No. of curve sets:          |                            |   |
| Unit:                       | °                          |   |
| Explicit set const. values: | 10, 20, 30, 35, 40, 45, 50 | This will override min./max. and number of sets |

You can define following properties:

**Set constant name (mandatory)** Name of the constant (as to be used in the equation).

**min./max. value (mandatory, if no explicit values are defined)** Limits of automatically generated values - requires no. of curve sets to be defined as well

**No of. curve sets (mandatory, if no explicit values are defined)** Number of sets values to generate - requires min. and max. value to be defined as well

**Unit (optional)** A unit for the set constant. Will be displayed in the plot legend with each set value

**Explicit set const. values (mandatory, if min./max. and/or number of curve sets not defined)** A comma separated list of explicit values. If this is defined the definition of min./max. and no. of curve sets will be ignored. If you want to switch back to min./max. definition just delete all text in this field.

## 2.7 Plot labels

You can define some plot annotations in the 'Plot labels' section.

| Plot labels |                  |
|-------------|------------------|
| Unit x:     | m                |
| Unit y:     | m                |
| Plot title: | Ballistic curves |

**Unit x (optional)** Appends a unit to the variable name on the x-axis

**Unit y (optional)** Appends a unit to the variable name on the y-axis

**Plot title (optional)** A user defined title to display above the plot. If this is empty or 'None' a plot title will automatically be generated from the function name, variable name, equation and defined constants - e.g. `'f(x)=a*x**2+b*x, a=3.0, b=5.0'`

## 2.8 Load, save and export

With the 'Save' button you can save your current project - i.e. the equation definition and additional parameters to a file and restore it via the 'Load' button.

The 'Export' button will export a python script, which can be run as a standalone script to create the plot.

---

**Note:** You cannot recreate the project from an exported python script (created via the 'Export' button). To restore your project in *simple-plotter* use the 'Save' button to save a project file.

---

## 2.9 Advanced usage

This chapter describes how the input file format is structured and how the CLI version of *simple-plotter* can be used to generate python code files.



## 2.9.1 File format

The input file for *simple-plotter* is a JSON file. See an example below:

```
{
  "file_format_version": "1.0",
  "export_csv": false,
  "formula": {
    "constants": [
      {
        "Comment": "",
        "Unit": "",
        "Value": "2",
        "Const. name": "amp"
      }
    ],
    "equation": "amp * np.sin(2* freq * np.pi * t)",
    "explicit_set_values": "None",
    "function_name": "y",
    "function_unit": "",
    "no_sets": "5",
    "set_max_val": "5",
    "set_min_val": "1",
    "set_var_name": "freq",
    "set_var_unit": "Hz",
    "var_name": "t",
    "var_unit": "s"
  },
  "plot_data": {
    "end_val": "1",
    "grid": true,
    "no_pts": "500",
    "plot_title": "",
    "start_val": "0",
    "swap_xy": false,
    "user_data": [],
    "x_log": false,
    "y_log": false,
    "y_max": "None",
    "y_min": "None"
  }
}
```

The file content can be divided into 3 parts:

**General information** The *file\_format\_version* identifies the version of the file format and is used to correctly parse the information TODO: add JSON schema for different versions *export\_csv* is intended for future use - currently this value has no effect.

**formula** This part contains all information, which defines the mathematical problem - e.g. the equation, constants, unit, etc. For details see the explanations on these properties above.

**plot\_data** This part contains all information, which define the appearance of the plot - e.g. start and values for x- and y-axis, lin. or log. scale, etc. For details see the explanations on these properties above.

## 2.9.2 CLI usage

The basic syntax to run the CLI tool is:

```
simple-plotter [options] -i <input_file_name>
```

Where following options are available:

**--help** Displays a help screen

**--input or -i** This is the only mandatory option, which must be followed by the JSON input file name

**--output or -o** If this option including an output file name (e.g. -o output.py) is set, the generated code will be saved to the corresponding file. If this option is omitted the code will just be printed to the terminal.

**--plotlib or -p** This option defines, which plotting library shall be used. Currently 'matplotlib' and 'kivy-garden/graph' are supported. If this option is omitted the code will be generated for matplotlib.

**--no-check or -n** This option disables the code checker. It is not recommended to use this option. If it is not checked the code checker will analyse the code for errors and illegal statements (e.g. un-allowed function calls) and thus provides some basic security.

Running the CLI tool on the example above (which can also be found as *sine\_waves.json* in the [examples folder](#) in source code repository) with following command will create a python script with code for a *kivy-garden/graph* plot and save it to 'test.py' file.

```
simple-plotter --plotlib kivy-garden/graph -i ./simple_plotter/examples/sine_waves.  
↪ json -o test.py
```

The content of the 'test.py' code file is shown below:

```
# this code has been auto-generated by simple_plotter  
import numpy as np  
  
import simple_plotter.core.advanced_graph as kv_graph  
  
# function definition  
def f(t, freq, amp):  
    return amp * np.sin(2* freq * np.pi * t)  
  
# definition of constants  
amp = 2  
  
# variable definition  
t = np.linspace(0, 1, num=500)  
  
# set constants definition  
freq = np.linspace(1, 5, 5)  
y = f(t, freq[0], amp)  
  
# plot setup  
graph = kv_graph.AnnotatedLinePlot(xlabel="t [s]",  
    ylabel="y",  
    y_grid_label=True, x_grid_label=True, padding=5,  
    x_grid=True, y_grid=True, xlog=False, ylog=False  
    , title="y(t)=amp * np.sin(2* freq * np.pi * t), amp = 2"  
)  
for set_const in freq:  
    results = f(t, set_const, amp)  
    graph.plot(t, results, label="freq="+str(set_const)+" [Hz]")
```

(continues on next page)

(continued from previous page)

```
# uncomment the code below, if you want run this as a standalone script
# from kivy.app import App
#
# class Plot(App):
#
#     def __init__(self):
#         super().__init__()
#
#     def build(self):
#         return graph
#
# Plot().run()
```

As can be seen the code to actually create the plot is out-commented. If you want to run this script to generate a standalone plot, you should un-comment the lines at the end of the script before you run it.



## CHAPTER 3

---

### Licenses

---

This chapter contains the license information of source code and binary releases of *simple-plotter* and related components.

---

**Note:** simple-plotter 0.4.0 or later is released under the MIT license, while earlier versions (up to 0.3.2) were mainly released under the GPLv3+ license including some code in the `advanced_graph` module taken from matplotlib under the matplotlib license. In simple-plotter version 0.4.0 the Qt-frontend (`simple_plotter/gui.py`) as well as the color map data in `simple_plotter/core/advanced_graph.py` taken from matplotlib have been removed to allow for a simpler and more permissive license of the core components. The Qt-frontend is still available as a separate project called ‘simple-plotter-qt’, which remains released under the GPLv3+.

---

### 3.1 simple-plotter

```
simple-plotter - plot code generator for 2D functional line graphs
Copyright (C) 2019-2020 Thies Hecker
```

```
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
```

```
The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
```

```
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
```

(continues on next page)

(continued from previous page)

LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## 3.2 simple-plotter-qt

simple-plotter-qt - Qt based front-end for simple\_plotter  
Copyright (C) 2020 Thies Hecker

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

## 3.3 simple-plotter4a

simple-plotter4a - kivy based front-end for simple\_plotter  
Copyright (C) 2020 Thies Hecker

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

## 3.4 simple-plotter4a binary releases (Android)

### 3.4.1 simple-plotter4a rel. 0.1.0

This chapter shows the copyright and license notes of *simple-plotter4a* and the dependencies used to build *simple-plotter4a*.

```
simple-plotter4a - kivy based front-end for simple_plotter
Copyright (C) 2020 Thies Hecker
```

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

## Dependencies

### python3 3.7.1

Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018 Python Software Foundation; All Rights Reserved

<https://www.python.org/>

#### A. HISTORY OF THE SOFTWARE =====

Python was created in the early 1990s by Guido van Rossum at Stichting Mathematisch Centrum (CWI, see <http://www.cwi.nl>) in the Netherlands as a successor of a language called ABC. Guido remains Python's principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for National Research Initiatives (CNRI, see <http://www.cnri.reston.va.us>) in Reston, Virginia where he released several versions of the software.

In May 2000, Guido and the Python core development team moved to BeOpen.com to form the BeOpen PythonLabs team. In October of the same year, the PythonLabs team moved to Digital Creations, which became Zope Corporation. In 2001, the Python Software Foundation (PSF, see <https://www.python.org/psf/>) was formed, a non-profit organization created specifically to own Python-related Intellectual Property. Zope Corporation was a sponsoring member of the PSF.

All Python releases are Open Source (see <http://www.opensource.org> for the Open Source Definition). Historically, most, but not all, Python releases have also been GPL-compatible; the table below summarizes the various releases.

| Release        | Derived from | Year      | Owner | GPL-compatible? (1) |
|----------------|--------------|-----------|-------|---------------------|
| 0.9.0 thru 1.2 |              | 1991-1995 | CWI   | yes                 |
| 1.3 thru 1.5.2 | 1.2          | 1995-1999 | CNRI  | yes                 |

(continues on next page)

(continued from previous page)

|               |           |          |            |         |
|---------------|-----------|----------|------------|---------|
| 1.6           | 1.5.2     | 2000     | CNRI       | no      |
| 2.0           | 1.6       | 2000     | BeOpen.com | no      |
| 1.6.1         | 1.6       | 2001     | CNRI       | yes (2) |
| 2.1           | 2.0+1.6.1 | 2001     | PSF        | no      |
| 2.0.1         | 2.0+1.6.1 | 2001     | PSF        | yes     |
| 2.1.1         | 2.1+2.0.1 | 2001     | PSF        | yes     |
| 2.1.2         | 2.1.1     | 2002     | PSF        | yes     |
| 2.1.3         | 2.1.2     | 2002     | PSF        | yes     |
| 2.2 and above | 2.1.1     | 2001-now | PSF        | yes     |

## Footnotes:

- (1) GPL-compatible doesn't mean that we're distributing Python under the GPL. All Python licenses, unlike the GPL, let you distribute a modified version without making your changes open source. The GPL-compatible licenses make it possible to combine Python with other software that is released under the GPL; the others don't.
- (2) According to Richard Stallman, 1.6.1 is not GPL-compatible, because its license has a choice of law clause. According to CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1 is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's direction to make these releases possible.

## B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON

## PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2

1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.
2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018 Python Software Foundation; All Rights Reserved" are retained in Python alone or in any derivative version prepared by Licensee.
3. In the event Licensee prepares a derivative work that is based on or incorporates Python or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python.
4. PSF is making Python available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND

(continues on next page)



(continued from previous page)

DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee agrees to be bound by the terms and conditions of this License Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0

-----

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the Individual or Organization ("Licensee") accessing and otherwise using this software in source or binary form and its associated documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License Agreement, BeOpen hereby grants Licensee a non-exclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use the Software alone or in any derivative version, provided, however, that the BeOpen Python License is retained in the Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS" basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

(continues on next page)

(continued from previous page)

6. This License Agreement shall be governed by and interpreted in all respects by the law of the State of California, excluding conflict of law provisions. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between BeOpen and Licensee. This License Agreement does not grant permission to use BeOpen trademarks or trade names in a trademark sense to endorse or promote products or services of Licensee, or any third party. As an exception, the "BeOpen Python" logos available at <http://www.pythonlabs.com/logos.html> may be used according to the permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee agrees to be bound by the terms and conditions of this License Agreement.

#### CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1

1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 ("CNRI"), and the Individual or Organization ("Licensee") accessing and otherwise using Python 1.6.1 software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6.1 alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright (c) 1995-2001 Corporation for National Research Initiatives; All Rights Reserved" are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes): "Python 1.6.1 is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python 1.6.1 may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1013. This Agreement may also be obtained from a proxy server on the Internet using the following URL: <http://hdl.handle.net/1895.22/1013>".

3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS

(continues on next page)

(continued from previous page)

A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6.1, Licensee agrees to be bound by the terms and conditions of this License Agreement.

ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Stichting Mathematisch Centrum or CWI not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

## kivy 1.11.1

Copyright (c) 2010-2019 Kivy Team and other contributors

<https://kivy.org>

Copyright (c) 2010–2019 Kivy Team and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## numpy 1.16.4

Copyright © 2005-2019, NumPy Developers. All rights reserved.

<https://numpy.org/>

Copyright (c) 2005–2019, NumPy Developers.  
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- \* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- \* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- \* Neither the name of the NumPy Developers nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE

(continues on next page)

(continued from previous page)

OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The NumPy repository and source distributions bundle several libraries that are compatibly licensed. We list these here.

Name: Numpydoc

Files: doc/sphinxext/numpydoc/\*

License: 2-clause BSD

For details, see doc/sphinxext/LICENSE.txt

Name: scipy-sphinx-theme

Files: doc/scipy-sphinx-theme/\*

License: 3-clause BSD, PSF and Apache 2.0

For details, see doc/scipy-sphinx-theme/LICENSE.txt

Name: lapack-lite

Files: numpy/linalg/lapack\_lite/\*

License: 3-clause BSD

For details, see numpy/linalg/lapack\_lite/LICENSE.txt

Name: tempita

Files: tools/npv\_tempita/\*

License: BSD derived

For details, see tools/npv\_tempita/license.txt

Name: dragon4

Files: numpy/core/src/multiarray/dragon4.c

License: One of a kind

For license text, see numpy/core/src/multiarray/dragon4.c

## versioneer 0.18

<https://github.com/warner/python-versioneer>

taken from <https://github.com/warner/python-versioneer/blob/0.18/README.md>

License

To make Versioneer easier to embed, all its code is dedicated to the public domain. The `_version.py` that it creates is also in the public domain. Specifically, both are released under the Creative Commons "Public Domain Dedication" license (CC0-1.0), as described in <https://creativecommons.org/publicdomain/zero/1.0/> .

## jinja2 2.11.1

Copyright 2007 Pallets

<https://jinja.palletsprojects.com>

Copyright 2007 Pallets

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## **markupsafe 1.1.1**

Copyright 2010 Pallets

<https://palletsprojects.com/p/markupsafe/>

Copyright 2010 Pallets

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT

(continues on next page)

(continued from previous page)

HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## setuptools\_scm 3.5.0

Ronny Pfannschmidt

[https://github.com/pypa/setuptools\\_scm/](https://github.com/pypa/setuptools_scm/)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## kivy-garden.graph 0.4.0

Copyright (c) 2019- Kivy Team and other contributors

<https://github.com/kivy-garden/graph>

Copyright (c) 2019- Kivy Team and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

(continues on next page)

(continued from previous page)

LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## matplotlib 3.1.3

Copyright (c) 2012-2020 Matplotlib Development Team; All Rights Reserved

<https://matplotlib.org/>

License agreement for matplotlib versions 1.3.0 and later  
=====

1. This LICENSE AGREEMENT is between the Matplotlib Development Team ("MDT"), and the Individual or Organization ("Licensee") accessing and otherwise using matplotlib software in source or binary form and its associated documentation.
2. Subject to the terms and conditions of this License Agreement, MDT hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use matplotlib alone or in any derivative version, provided, however, that MDT's License Agreement and MDT's notice of copyright, i.e., "Copyright (c) 2012- Matplotlib Development Team; All Rights Reserved" are retained in matplotlib alone or in any derivative version prepared by Licensee.
3. In the event Licensee prepares a derivative work that is based on or incorporates matplotlib or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to matplotlib .
4. MDT is making matplotlib available to Licensee on an "AS IS" basis. MDT MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, MDT MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF MATPLOTLIB WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.
5. MDT SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF MATPLOTLIB FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING MATPLOTLIB , OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.
6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.
7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between MDT and Licensee. This License Agreement does not grant permission to use MDT trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

(continues on next page)



(continued from previous page)

8. By copying, installing or otherwise using matplotlib ,  
 Licensee agrees to be bound by the terms and conditions of this License  
 Agreement.

### libffi 3.3-rc0

Copyright (c) 1996-2014 Anthony Green, Red Hat, Inc and others.

<https://sourceware.org/libffi/>

libffi - Copyright (c) 1996-2014 Anthony Green, Red Hat, Inc and others.  
 See source files for details.

Permission is hereby granted, free of charge, to any person obtaining  
 a copy of this software and associated documentation files (the  
 ``Software''), to deal in the Software without restriction, including  
 without limitation the rights to use, copy, modify, merge, publish,  
 distribute, sublicense, and/or sell copies of the Software, and to  
 permit persons to whom the Software is furnished to do so, subject to  
 the following conditions:

The above copyright notice and this permission notice shall be  
 included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ``AS IS'', WITHOUT WARRANTY OF ANY KIND,  
 EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF  
 MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.  
 IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY  
 CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,  
 TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE  
 SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### OpenSSL 1.1.1

Copyright (c) 1998-2018 The OpenSSL Project. All rights reserved. / Copyright (C) 1995-1998 Eric Young  
 (eay@cryptsoft.com), All rights reserved.

<http://www.openssl.org/>

LICENSE ISSUES  
 =====

The OpenSSL toolkit stays under a double license, i.e. both the conditions of  
 the OpenSSL License and the original SSLeay license apply to the toolkit.  
 See below for the actual license texts.

OpenSSL License  
 -----

```
/* =====
 * Copyright (c) 1998-2018 The OpenSSL Project. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
```

(continues on next page)

(continued from previous page)

```

* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
*
* 3. All advertising materials mentioning features or use of this
*    software must display the following acknowledgment:
*    "This product includes software developed by the OpenSSL Project
*    for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
*
* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
*    endorse or promote products derived from this software without
*    prior written permission. For written permission, please contact
*    openssl-core@openssl.org.
*
* 5. Products derived from this software may not be called "OpenSSL"
*    nor may "OpenSSL" appear in their names without prior written
*    permission of the OpenSSL Project.
*
* 6. Redistributions of any form whatsoever must retain the following
*    acknowledgment:
*    "This product includes software developed by the OpenSSL Project
*    for use in the OpenSSL Toolkit (http://www.openssl.org/)"
*
* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED.  IN NO EVENT SHALL THE OpenSSL PROJECT OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
* OF THE POSSIBILITY OF SUCH DAMAGE.
* =====
*
* This product includes cryptographic software written by Eric Young
* (eay@cryptsoft.com).  This product includes software written by Tim
* Hudson (tjh@cryptsoft.com).
*
*/

Original SSLeay License
-----

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
* All rights reserved.
*
* This package is an SSL implementation written

```

(continues on next page)

(continued from previous page)

```

* by Eric Young (eay@cryptsoft.com).
* The implementation was written so as to conform with Netscapes SSL.
*
* This library is free for commercial and non-commercial use as long as
* the following conditions are aheared to. The following conditions
* apply to all code found in this distribution, be it the RC4, RSA,
* lhash, DES, etc., code; not just the SSL code. The SSL documentation
* included with this distribution is covered by the same copyright terms
* except that the holder is Tim Hudson (tjh@cryptsoft.com).
*
* Copyright remains Eric Young's, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
*    must display the following acknowledgement:
*    "This product includes cryptographic software written by
*      Eric Young (eay@cryptsoft.com)"
*    The word 'cryptographic' can be left out if the rouines from the library
*    being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from
*    the apps directory (application code) you must include an acknowledgement:
*    "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
*
* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/

```

## SDL2\_image 2.0.4

Copyright (C) 1997-2018 Sam Lantinga <[slouken@libsdl.org](mailto:slouken@libsdl.org)>

[https://www.libsdl.org/projects/SDL\\_image/](https://www.libsdl.org/projects/SDL_image/)

```
/*
SDL_image:  An example image loading library for use with SDL
Copyright (C) 1997-2018 Sam Lantinga <slouken@libsdl.org>

This software is provided 'as-is', without any express or implied
warranty.  In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
   claim that you wrote the original software.  If you use this software
   in a product, an acknowledgment in the product documentation would be
   appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
   misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.
*/
```

## SDL2\_mixer 2.0.4

Copyright (C) 1997-2018 Sam Lantinga <slouken@libsdl.org>

[https://www.libsdl.org/projects/SDL\\_mixer/](https://www.libsdl.org/projects/SDL_mixer/)

```
/*
SDL_mixer:  An audio mixer library based on the SDL library
Copyright (C) 1997-2018 Sam Lantinga <slouken@libsdl.org>

This software is provided 'as-is', without any express or implied
warranty.  In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
   claim that you wrote the original software.  If you use this software
   in a product, an acknowledgment in the product documentation would be
   appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
   misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.
*/
```

## SDL2\_ttf 2.0.14

Copyright (C) 1997-2016 Sam Lantinga <slouken@libsdl.org>

[https://www.libsdl.org/projects/SDL\\_ttf/](https://www.libsdl.org/projects/SDL_ttf/)

```

/*
SDL_ttf:  A companion library to SDL for working with TrueType (tm) fonts
Copyright (C) 1997-2016 Sam Lantinga <slouken@libsdl.org>

This software is provided 'as-is', without any express or implied
warranty.  In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
   claim that you wrote the original software.  If you use this software
   in a product, an acknowledgment in the product documentation would be
   appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
   misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.
*/

```

## SDL2 2.0.9

Copyright (C) 1997-2018 Sam Lantinga <slouken@libsdl.org>

<https://www.libsdl.org/index.php>

```

Simple DirectMedia Layer
Copyright (C) 1997-2018 Sam Lantinga <slouken@libsdl.org>

This software is provided 'as-is', without any express or implied
warranty.  In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
   claim that you wrote the original software.  If you use this software
   in a product, an acknowledgment in the product documentation would be
   appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
   misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

```

## sqlite3-amalgamation 3150100

<https://www.sqlite.org/index.html>

copyright note taken from sqlite.h of sqlite-amalgamation 3150100:

2001 September 15

(continues on next page)

(continued from previous page)

The author disclaims copyright to this source code. In place of a legal notice, here is a blessing:

May you do good and not evil.  
May you find forgiveness for yourself and forgive others.  
May you share freely, never taking more than you give.

## setuptools 40.9.0

Copyright (C) 2016 Jason R Coombs <[jaraco@jaraco.com](mailto:jaraco@jaraco.com)>

<https://github.com/pypa/setuptools>

Copyright (C) 2016 Jason R Coombs <[jaraco@jaraco.com](mailto:jaraco@jaraco.com)>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## six 1.10.0

Copyright (c) 2010-2015 Benjamin Peterson

<http://pypi.python.org/pypi/six/>

Copyright (c) 2010-2015 Benjamin Peterson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR

(continues on next page)

(continued from previous page)

COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### pyjnius 1.2.1.dev0

Copyright (c) 2010-2017 Kivy Team and other contributors

<https://github.com/kivy/pyjnius>

Copyright (c) 2010-2017 Kivy Team and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### python-for-android 2019.10.06

Copyright (c) 2010-2017 Kivy Team and other contributors

<https://python-for-android.readthedocs.io/en/latest/>

Copyright (c) 2010-2017 Kivy Team and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

(continues on next page)

(continued from previous page)

OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## hidapi

Copyright (c) 2010, Alan Ott, Signal 11 Software, All rights reserved.

part of SDL2 2.0.9 source code

HIDAPI can be used under one of three licenses.

1. The GNU General Public License, version 3.0, in LICENSE-gpl3.txt
2. A BSD-Style License, in LICENSE-bsd.txt.
3. The more liberal original HIDAPI license. LICENSE-orig.txt

The license chosen is at the discretion of the user of HIDAPI. For example:

1. An author of GPL software would likely use HIDAPI under the terms of the GPL.

2. An author of commercial closed-source software would likely use HIDAPI under the terms of the BSD-style license or the original HIDAPI license.

Below the content of the LICENSE-bsd.txt file are given:

Copyright (c) 2010, Alan Ott, Signal 11 Software  
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- \* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- \* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- \* Neither the name of Signal 11 Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.



## docutils

<http://docutils.sourceforge.net/>

```

=====
Copying Docutils
=====

:Author: David Goodger
:Contact: goodger@python.org
:Date: $Date: 2020-02-06 12:47:25 +0000 (Thu, 06 Feb 2020) $
:Web site: http://docutils.sourceforge.net/
:Copyright: This document has been placed in the public domain.

Most of the files included in this project have been placed in the
public domain, and therefore have no license requirements and no
restrictions on copying or usage; see the `Public Domain Dedication`_
below. There are a few exceptions_, listed below.
Files in the Sandbox_ are not distributed with Docutils releases and
may have different license terms.

Public Domain Dedication
=====

The persons who have associated their work with this project (the
"Dedicator": David Goodger and the many contributors to the Docutils
project) hereby dedicate the entire copyright, less the exceptions_
listed below, in the work of authorship known as "Docutils" identified
below (the "Work") to the public domain.

The primary repository for the Work is the Internet World Wide Web
site <http://docutils.sourceforge.net/>. The Work consists of the
files within the "docutils" module of the Docutils project Subversion
repository (Internet host docutils.svn.sourceforge.net, filesystem path
/svnroot/docutils), whose Internet web interface is located at
<http://docutils.svn.sourceforge.net/viewvc/docutils/>. Files dedicated to the
public domain may be identified by the inclusion, near the beginning
of each file, of a declaration of the form::

    Copyright: This document/module/DTD/stylessheet/file/etc. has been
    placed in the public domain.

Dedicator makes this dedication for the benefit of the public at large
and to the detriment of Dedicator's heirs and successors. Dedicator
intends this dedication to be an overt act of relinquishment in
perpetuity of all present and future rights under copyright law,
whether vested or contingent, in the Work. Dedicator understands that
such relinquishment of all rights includes the relinquishment of all
rights to enforce (by lawsuit or otherwise) those copyrights in the
Work.

Dedicator recognizes that, once placed in the public domain, the Work
may be freely reproduced, distributed, transmitted, used, modified,
built upon, or otherwise exploited by anyone for any purpose,
commercial or non-commercial, and in any way, including by methods
that have not yet been invented or conceived.

```

(continues on next page)

(continued from previous page)

```
(This dedication is derived from the text of the `Creative Commons
Public Domain Dedication`. [#]_)

.. [#] Creative Commons has `retired this legal tool`__ and does not
   recommend that it be applied to works: This tool is based on United
   States law and may not be applicable outside the US. For dedicating new
   works to the public domain, Creative Commons recommend the replacement
   Public Domain Dedication CC0_ (CC zero, "No Rights Reserved"). So does
   the Free Software Foundation in its license-list_.

   __ http://creativecommons.org/retiredlicenses
   .. _CC0: http://creativecommons.org/about/cc0

Exceptions
=====

The exceptions to the `Public Domain Dedication`_ above are:

* docutils/writers/s5_html/themes/default/iepngfix.htc:

    IE5.5+ PNG Alpha Fix v1.0 by Angus Turnbull
    <http://www.twinhelix.com>. Free usage permitted as long as
    this notice remains intact.

* docutils/utils/math/__init__.py,
  docutils/utils/math/latex2mathml.py,
  docutils/writers/xetex/__init__.py,
  docutils/utils/error_reporting.py,
  docutils/test/transforms/test_smartquotes.py:

  Copyright © Günter Milde.
  Released under the terms of the `2-Clause BSD license`_
  (`local copy <licenses/BSD-2-Clause.txt>`__).

* docutils/utils/smartquotes.py

  Copyright © 2011 Günter Milde,
  based on `SmartyPants`_ © 2003 John Gruber
  (released under a 3-Clause BSD license included in the file)
  and smartypants.py © 2004, 2007 Chad Miller.
  Released under the terms of the `2-Clause BSD license`_
  (`local copy <licenses/BSD-2-Clause.txt>`__).

  .. _SmartyPants: http://daringfireball.net/projects/smartypants/

* docutils/utils/math/math2html.py,
  docutils/writers/html4css1/math.css

  Copyright © Alex Fernández
  These files are part of eLyXer_, released under the `GNU
  General Public License`_ version 3 or later. The author relicensed
  them for Docutils under the terms of the `2-Clause BSD license`_
  (`local copy <licenses/BSD-2-Clause.txt>`__).

  .. _eLyXer: http://www.nongnu.org/elyxer/
```

(continues on next page)

(continued from previous page)

```

* docutils/utils/roman.py, copyright by Mark Pilgrim, released under the
  `Python 2.1.1 license`_ (`local copy`_).

__ licenses/python-2-1-1.txt

* tools/editors/emacs/rst.el, copyright by Free Software Foundation,
  Inc., released under the `GNU General Public License`_ version 3 or
  later (`local copy`_).

__ licenses/gpl-3-0.txt

The `2-Clause BSD license`_ and the Python licenses are OSI-approved_
and GPL-compatible_.

Plaintext versions of all the linked-to licenses are provided in the
licenses_ directory.

.. _sandbox: http://docutils.sourceforge.net/sandbox/README.html
.. _licenses: licenses/
.. _Python 2.1.1 license: http://www.python.org/2.1.1/license.html
.. _GNU General Public License: http://www.gnu.org/copyleft/gpl.html
.. _2-Clause BSD license: http://www.spdx.org/licenses/BSD-2-Clause
.. _OSI-approved: http://opensource.org/licenses/
.. _license-list:
.. _GPL-compatible: http://www.gnu.org/licenses/license-list.html

```

## pygments

Copyright (c) 2006-2017 by the respective authors (see AUTHORS file). All rights reserved.

<https://pygments.org/>

```

Copyright (c) 2006-2017 by the respective authors (see AUTHORS file).
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
  documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

```

(continues on next page)

(continued from previous page)

(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## glew

Copyright (C) 2008-2016, Nigel Stewart <nigels@users.sourceforge.net> Copyright (C) 2002-2008, Milan Ikits <milan.ikits@ieee.org> Copyright (C) 2002-2008, Marcelo E. Magallon <mmagallo@debian.org> Copyright (C) 2002, Lev Povalahev All rights reserved.

<http://glew.sourceforge.net/>

The OpenGL Extension Wrangler Library  
Copyright (C) 2008-2016, Nigel Stewart <nigels@users.sourceforge.net>  
Copyright (C) 2002-2008, Milan Ikits <milan.ikits@ieee.org>  
Copyright (C) 2002-2008, Marcelo E. Magallon <mmagallo@debian.org>  
Copyright (C) 2002, Lev Povalahev  
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- \* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- \* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- \* The name of the author may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## simple-plotter 0.3.2

Copyright (C) 2019-2020 Thies Hecker

<https://simple-plotter.readthedocs.io/en/latest/>

simple-plotter - simple plot code generator and GUI front-end  
Copyright (C) 2019-2020 Thies Hecker

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or

(continues on next page)

(continued from previous page)

(at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

License for color map used in advance\_graph module  
=====

The color table in standard\_colors method is taken from matplotlib v3.1.3:  
[https://github.com/matplotlib/matplotlib/blob/v3.1.3/lib/matplotlib/\\_cm.py](https://github.com/matplotlib/matplotlib/blob/v3.1.3/lib/matplotlib/_cm.py)

Copyright (c) 2012-2020 Matplotlib Development Team; All Rights Reserved

See the matplotlib license agreement below.

License agreement for matplotlib versions 1.3.0 and later  
=====

1. This LICENSE AGREEMENT is between the Matplotlib Development Team ("MDT"), and the Individual or Organization ("Licensee") accessing and otherwise using matplotlib software in source or binary form and its associated documentation.
2. Subject to the terms and conditions of this License Agreement, MDT hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use matplotlib alone or in any derivative version, provided, however, that MDT's License Agreement and MDT's notice of copyright, i.e., "Copyright (c) 2012- Matplotlib Development Team; All Rights Reserved" are retained in matplotlib alone or in any derivative version prepared by Licensee.
3. In the event Licensee prepares a derivative work that is based on or incorporates matplotlib or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to matplotlib .
4. MDT is making matplotlib available to Licensee on an "AS IS" basis. MDT MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, MDT MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF MATPLOTLIB WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.
5. MDT SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF MATPLOTLIB FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING MATPLOTLIB , OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

(continues on next page)

(continued from previous page)

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between MDT and Licensee. This License Agreement does not grant permission to use MDT trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using matplotlib , Licensee agrees to be bound by the terms and conditions of this License Agreement.

If you would like to contribute to the project feel free to fork the project and raise a merge request on gitlab.  
Find some development related remarks below:

### 4.1 Concept

*simple\_plotter* is primarily a very specific python-code generator - i.e. the parameters provided by the GUI front-end will be used to generate a python script, which is then executed to generate the plot.

The scripts will be created based on jinja2 templates - see `simple_plotter/core/templates`.

Some code checking is done using the `ast` module to provide basic security (e.g. prevent unwanted commands or imports). See the `CodeChecker` class in the `code_parser` module.

### 4.2 Package versions

`setuptools_scm` is used to fetch the version strings from git tags/commits automatically.

### 4.3 Building conda packages

There will probably be no need to update the `meta.yaml` manually. `conda-build`'s jinja templating functionality is used in `meta.yaml` to fetch the package information from `setup.py` (except for the package version).

Since `setuptools_scm` is used, there is no version argument, that can be imported with the `load_setup_py_data` function in `meta.yaml`.

Therefore **conda packages should be built using the `build_conda_package.sh` shell script.**

This script first reads the (auto-generated) version from `setup.py` and passes it to an environment variable, which is used in the `meta.yaml`.

---

**Note:** Unfortunately there is currently no kivy-garden.graph package for conda. So conda packages will miss this dependency! You will have to install kivy-garden.graph via pip.

---

## 4.4 Testing

gitlab CI/CD will run the test functions (pytest) - see tests/ on each commit and update the [code coverage report](#). Tests are currently only defined for the simple\_plotter.core modules (not for the GUI).

## 4.5 Project documentation

Documentation sources for sphinx are stored in the doc/ folder. To build the documentation you need *sphinx* and the *sphinx-rtd-theme* package installed.

When commits to master are pushed to the gitlab repository the documentation is automatically generated on readthedocs:

<https://simple-plotter.readthedocs.io/latest>

If tags are pushed documentation will be generated for

<https://simple-plotter.readthedocs.io/stable>



This section describes the API of the backend - i.e. the **code\_generator** and **code\_parser** module.

## 5.1 code\_generator

The module consists of a class *DataHandler* which includes the code generator methods and some data container classes (*Formula* and *PlotData*).

`simple_plotter.core.code_generator.check_valid_input(input_value)`

Checks if an input value corresponds to a valid value - i.e. not None, "", "None", ...

**Parameters** `input_value` – Any type of input value

**Returns** True if valid

**Return type** bool

### 5.1.1 Formula

```
class simple_plotter.core.code_generator.Formula(function_name='y',  var_name='x',
                                                equation='x**2',      con-
                                                stants=None,  set_var_name=None,
                                                set_min_val=None,
                                                set_max_val=None,  no_sets=None,
                                                var_unit=None, function_unit=None,
                                                set_var_unit=None,      ex-
                                                plicit_set_values=None)
```

Data container for definition of the equation

**Parameters**

- **function\_name** (*str*) – Name of the “return value” - e.g. `y = ...`
- **var\_name** (*str*) – Name of the function variable - e.g. `x`

- **equation** (*str*) – String representation of an equation (python, numpy code)
- **constants** (*list*) – List with a dictionary for each constant - see notes
- **set\_var\_name** (*str*) – Name of the set parameter
- **set\_min\_val** (*float*) – Min. value of the set parameter
- **set\_max\_val** (*float*) – Max. value of the set parameter
- **no\_sets** (*int*) – Number of set parameter values to create between min. and max. value
- **var\_unit** (*str*) – Unit for the variable (only used for display)
- **function\_unit** (*str*) – Unit of the function return value (only used for display)
- **set\_var\_unit** (*str*) – Unit of set parameter (only used for display)
- **explicit\_set\_values** (*str*) – String of explicit values for set parameters (like a list separated with comma)

### Notes

- the constants dictionary consists of following keys (all values are strings): “Const. name”, “Value”, “Unit” and “Comment”

## 5.1.2 PlotData

```
class simple_plotter.core.code_generator.PlotData (start_val=-10.0, end_val=10.0,  
                                                no_pts=50, x_log=False,  
                                                y_log=False, swap_xy=False,  
                                                grid=False, user_data=None,  
                                                y_min=None, y_max=None,  
                                                plot_title=None)
```

Data container for plot definition

### Parameters

- **start\_val** (*float*) – start of x value range
- **end\_val** (*float*) – end of x value range
- **no\_pts** (*int*) – Number of data points for x value
- **x\_log** (*bool*) – Sets x scale to logarithmic if True
- **y\_log** (*bool*) – Sets y scale to logarithmic if True
- **swap\_xy** (*bool*) – Swaps x and y axis if True
- **grid** (*bool*) – Enables grid if True
- **user\_data** (*list*) – Not implemented yet
- **y\_min** (*float*) – Min y value for plot display
- **y\_max** (*float*) – Max y value for plot display
- **plot\_title** (*str*) – Manually defined plot title (if None, plot title will be created from formula automatically)

## Notes

- All float values will be converted to str and may also be passed as str directly.

### 5.1.3 DataHandler

```
class simple_plotter.core.code_generator.DataHandler (formula, plot_data,  
                                                    export_csv=False,  
                                                    filename=None,  
                                                    code_checker=None,  
                                                    plot_lib='matplotlib')
```

Main class for parser - includes all code generator methods

#### Parameters

- **formula** (*Formula*) – Formula container object
- **plot\_data** (*PlotData*) – plot data container object
- **export\_csv** (*bool*) – If true code to export curve values to csv will be embedded
- **filename** (*str*) – Name of the project file (JSON)
- **code\_checker** (*CodeChecker*) – Code checker object
- **plot\_lib** (*str*) – Library used for plotting - either 'matplotlib' or 'kivy-garden/graph'

**check\_code()**

Checks the code for each segment

#### Returns

Consisting of:

- list: List of error codes
- list: List of error logs

**Return type** tuple

## Notes

- **following error codes exist:**
  - 0: Error in imports
  - 1: Error in function definition
  - 2: Error in constants definition
  - 3: Error variable definition
  - 4: Error in set constants definition
  - 5: Error in plot call
  - 6: Error in plot setup

**check\_const\_validity()**

Checks the validity of defined constants

**Returns** True if all constants are valid, false otherwise

**Return type** bool

---

**Note:** Constants need to have at least a name and a value

---

**static check\_valid\_input** (*input\_value*)

Checks if an input value corresponds to a valid value - i.e. not None, "", "None",...

**Returns** True if valid input value

**Return type** bool

**combine\_code** ()

Assembles complete code

**Returns** combined python code

**Return type** str

**load\_project** (*filename*)

loads a project from JSON file

**plot** ()

plots the function, if code is valid

**Returns**

Consisting of:

- list: Error codes
- list: Error logs

**Return type** tuple

**save\_project** (*filename=None*)

saves project object to JSON file

**write\_constdefs** ()

Write constants definition

**write\_csv\_export** ()

writes the code to export to filename.csv

**Returns** python code for export to csv file

**Return type** str

**write\_funcdef** ()

Writes the function definition

**write\_imports** ()

Writes the imports

**write\_plot\_call** ()

Writes the plot call code

**write\_plot\_setup** ()

writes the plot setup code

**Returns** python code for plotting section

**Return type** str

**write\_py\_file** (*filename*)

writes a python file with code generated by combine\_code

**write\_setvardef()**  
writes definition for set variable

**Returns** python code for definitions

**Return type** str

**write\_vardef()**  
Write variable definition

## 5.2 code\_parser

This module consist of a data container class *Code* and the *CodeChecker* class, which uses *ast* to analyze code elements and identify unwanted code.

### 5.2.1 Code

**class** simple\_plotter.core.code\_parser.**Code**(code\_str)  
Container for code

**Parameters** code\_str (str) – Code in string representation

**code\_lines**  
Returns code as list (each line is a list element)

**code\_tree**  
Returns ast root node

**print\_code\_lines()**  
Prints code with line numbers

### 5.2.2 CodeChecker

**class** simple\_plotter.core.code\_parser.**CodeChecker**(allowed\_imports=None,  
allowed\_calls=None, al-  
lowed\_names=None, al-  
lowed\_aliases=None, al-  
lowed\_FunctionDefs=None)  
Checker for illegal statements in code

**Parameters**

- **allowed\_imports** (list) – List of allowed imports
- **allowed\_calls** (list) – List of allowed calls
- **allowed\_names** (list) – List of allowed names (e.g. function names not covered in methods of allowed aliases)
- **allowed\_aliases** (list) – List of allowed module/class/function aliases
- **allowed\_FunctionDefs** (list) – List of allowed names for Function definition

**allowed\_imports**  
List of allowed imports

**Type** list

**allowed\_calls**

List of allowed calls

**Type** list

**allowed\_names**

List of allowed names (e.g. function names not covered in methods of allowed aliases)

**Type** list

**allowed\_aliases**

List of allowed module/class/function aliases

**Type** list

**allowed\_node\_types**

List of node ast node types (Classes) - defaults to Import, FunctionDef, Assign,

**Type** list

**Call and Expr****allowed\_FunctionDefs**

List of allowed names for Function definition

**Type** list

**check\_code** (*code*)

Checks if the code is valid

**Parameters** **code** (*Code*) – Code object to analyze

**Returns**

consisting of:

- bool: True if code is valid
- list: Error log

**Return type** tuple

**first\_pass** (*code*)

Check if nodes are in allowed node types

**Parameters** **code** (*Code*) –

**generic\_visit** (*node*)

Called if no explicit visitor function exists for a node.

## 5.3 advanced\_graph

This module includes a class *AnnotatedLinePlot* which is a specialized extension of the kivy-garden.graph package. The *AnnotatedLinePlot* adds some matplotlib-like behavior (e.g. automatic axis-scaling, a legend, plot title) to the garden.graph *LinePlot* class.

### 5.3.1 AnnotatedLinePlot

```
class simple_plotter.core.advanced_graph.AnnotatedLinePlot (title=None,  
                                                         show_legend=True,  
                                                         **kwargs)
```

Extends the kivy-graden/graph by some additional features, like a title, legend, automatic colors

**Parameters**

- **title** (*str*) – Title to place above the plot
- **show\_legend** (*bool*) – If true legend will be shown below the graph
- **\*\*kwargs** – Keyword arguments for `kivy_garden.graph.Graph` class

**get\_data\_limits** ()

Calculate x- and y-data limits

**Returns**

tuple consisting of:

- float: min. x-value
- float: max. x-value
- float: min. y-value
- float: max. y-value

**Return type** tuple**plot** (*x\_values*, *y\_values*, *color=None*, *label=None*)

Adds a plot to the graph

**Parameters**

- **x\_values** (*list*) – List of x-values
- **y\_values** (*list*) – List of y-values
- **color** (*tuple*) – RGBA color value (0..1.) for the curves color- e.g. (0.5, 1., 1., 1.)
- **label** (*str*) – Legend label for the curve

---

**Note:** If color is None the color will automatically be assigned based on the predefined standard colors.

---

**recalculate\_axis\_limits** ()

Recalculates the axis limits and updates the graph

**recalculate\_x\_axis** ()

Recalculates the x-axis limits and ticks

**Returns**

tuple consisting of:

- float: y-axis min. value
- float: y-axis max. value
- float: tick major spacing
- int: number minor ticks

**Return type** tuple**recalculate\_y\_axis** ()

Recalculates y-axis limits

**Returns**

tuple consisting of:

- float: y-axis min. value
- float: y-axis max. value
- float: tick major spacing
- int: number minor ticks

**Return type** tuple

**standard\_colors**

Returns a list of 16 standard color RGBA values

**Type** list

**x\_data\_limits**

Returns data limits (xmin and xmax) in terms of x-coordinates

**Type** tuple

**y\_data\_limits**

Returns data limits (ymin and ymax) in terms of y-coordinates

**Type** tuple



## CHAPTER 6

---

### Roadmap and Issue-tracker

---

For bug-reports and feature request please use the Issue tracker on the project's gitlab-page:

<https://gitlab.com/thecker/simple-plotter/-/issues>

The development roadmap can be found on gitlab as well:

see <https://gitlab.com/thecker/simple-plotter/-/milestones>



### 7.1 Release 0.4.0

**Date** 2020-03-30

- project restructured: Qt-frontend code moved to new project simple-plotter-qt, matplotlib code removed → **license changed to MIT** (for simple-plotter base)
- CLI added to simple-plotter
- bug-fix: setting ymax value with kivy-garden/graph caused error

### 7.2 Release 0.3.2

**Date** 2020-03-21

- removed jsonpickle for saving projects (now explicit JSON conversion, resolves issue #1)
- bug fix (Qt-GUI): loading project with fewer constants than currently defined caused crash

### 7.3 Release 0.3.1

**Date** 2020-03-09

- bug fix (in jinja2 template): Plot failed, when y-min./max. and title label were defined

### 7.4 Release 0.3.0

**Date** 2020-03-08

- complete restructuring - base modules are now in simple\_plotter.core

- bug fixes: crashes with logarithmic axes using kivy-frontend fixed
- documentation: User guide added

## 7.5 Release 0.2.2

**Date** 2020-02-27

- support for kivy-garden/graph based plotting library added (enables support for Android version of simple-plotter)

## 7.6 Release 0.2.1

**Date** 2019-11-02

- bug fix: loading of example project from older version failed

## 7.7 Release 0.2.0

**Date** 2019-11-02

- code parser added (limits allowed imports, calls, etc.)
- code generator now uses jinja templates
- unit tests added
- export csv removed

## 7.8 Release 0.1.2

**Date** 2019-10-18

- automatic version strings added (using setuptools\_scm)
- package definitions moved from meta.yaml to setup.py

## 7.9 Release 0.1.1

**Date** 2019-10-14

- fixed documentation builds

## 7.10 Release 0.1.0

**Date** 2019-10-14

- Initial release

## CHAPTER 8

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



## A

allowed\_aliases (simple\_plotter.core.code\_parser.CodeChecker attribute), 50

allowed\_calls (simple\_plotter.core.code\_parser.CodeChecker attribute), 49

allowed\_FunctionDefs (simple\_plotter.core.code\_parser.CodeChecker attribute), 50

allowed\_imports (simple\_plotter.core.code\_parser.CodeChecker attribute), 49

allowed\_names (simple\_plotter.core.code\_parser.CodeChecker attribute), 50

allowed\_node\_types (simple\_plotter.core.code\_parser.CodeChecker attribute), 50

AnnotatedLinePlot (class in simple\_plotter.core.advanced\_graph), 50

## C

check\_code() (simple\_plotter.core.code\_generator.DataHandler method), 47

check\_code() (simple\_plotter.core.code\_parser.CodeChecker method), 50

check\_const\_validity() (simple\_plotter.core.code\_generator.DataHandler method), 47

check\_valid\_input() (in module simple\_plotter.core.code\_generator), 45

check\_valid\_input() (simple\_plotter.core.code\_generator.DataHandler static method), 48

Code (class in simple\_plotter.core.code\_parser), 49

code\_lines (simple\_plotter.core.code\_parser.Code at-

tribute), 49

code\_tree (simple\_plotter.core.code\_parser.Code attribute), 49

CodeChecker (class in simple\_plotter.core.code\_parser), 49

combine\_code() (simple\_plotter.core.code\_generator.DataHandler method), 48

## D

DataHandler (class in simple\_plotter.core.code\_generator), 47

## F

first\_pass() (simple\_plotter.core.code\_parser.CodeChecker method), 50

Formula (class in simple\_plotter.core.code\_generator), 45

## G

generic\_visit() (simple\_plotter.core.code\_parser.CodeChecker method), 50

get\_data\_limits() (simple\_plotter.core.advanced\_graph.AnnotatedLinePlot method), 51

## L

load\_project() (simple\_plotter.core.code\_generator.DataHandler method), 48

## P

plot() (simple\_plotter.core.advanced\_graph.AnnotatedLinePlot method), 51

plot() (simple\_plotter.core.code\_generator.DataHandler method), 48

PlotData (class in simple\_plotter.core.code\_generator), 46

`print_code_lines()` (simple\_plotter.core.code\_parser.Code method), 49  
*ple\_plotter.core.advanced\_graph.AnnotatedLinePlot attribute*), 52

## R

`recalculate_axis_limits()` (simple\_plotter.core.advanced\_graph.AnnotatedLinePlot method), 51  
`recalculate_x_axis()` (simple\_plotter.core.advanced\_graph.AnnotatedLinePlot method), 51  
`recalculate_y_axis()` (simple\_plotter.core.advanced\_graph.AnnotatedLinePlot method), 51

## S

`save_project()` (simple\_plotter.core.code\_generator.DataHandler method), 48  
`standard_colors` (simple\_plotter.core.advanced\_graph.AnnotatedLinePlot attribute), 52

## W

`write_constdefs()` (simple\_plotter.core.code\_generator.DataHandler method), 48  
`write_csv_export()` (simple\_plotter.core.code\_generator.DataHandler method), 48  
`write_funcdef()` (simple\_plotter.core.code\_generator.DataHandler method), 48  
`write_imports()` (simple\_plotter.core.code\_generator.DataHandler method), 48  
`write_plot_call()` (simple\_plotter.core.code\_generator.DataHandler method), 48  
`write_plot_setup()` (simple\_plotter.core.code\_generator.DataHandler method), 48  
`write_py_file()` (simple\_plotter.core.code\_generator.DataHandler method), 48  
`write_setvardef()` (simple\_plotter.core.code\_generator.DataHandler method), 48  
`write_vardef()` (simple\_plotter.core.code\_generator.DataHandler method), 49

## X

`x_data_limits` (sim-

## Y

`y_data_limits` (simple\_plotter.core.advanced\_graph.AnnotatedLinePlot attribute), 52