Data Digitizer Documentation

Release 1.2.2

M. Skocic

Nov 08, 2023

CONTENTS

CHAPTER

ONE

GETTING STARTED

1.1 Description

DataDigitizer is a simple graphical tool for digitizing data from images.

In terminal enter the following command:

```
python -m datadigitizer
```

The cursor is used to point a specific position in the graph whereas all operations are done through keyboard combinations or through the main menu.

Legend:

- Red crosses are data points
- Blue crosses are Xmin and Xmax
- Green crosses are Ymin and Ymax

Commands:

- <Ctrl-o> for loading image.
- <Ctrl-a> add data point.
- <Hold a+Left Click> add data point.
- <Left Click> select a data point.
- <Hold Ctrl+Left Click> multiple data point selection.
- <Ctrl-g> set Xmin from last data point or from selected data point.
- <Ctrl-h> set Xmax from last data point or from selected data point.
- <Ctrl-j> set Ymin from last data point or from selected data point.
- <Ctrl-k> set Ymax from last data point or from selected data point
- <Ctrl-l> set all limits from last 4 data points or from last 4 selected data points.
- <Ctrl-n> remove all limits.
- <Ctrl-z> remove last data point.
- <Ctrl-d> remove selected data point.
- <Ctrl-D> remove all data points.
- <Ctrl-m> compute the data points.
- <Ctrl-t> view data table.
- <Ctrl-s> save data points.
- <Ctrl-w> clear all.

A pdf version of the documentation can be found here `DataDigitizer`. The source code can be viewed on GitHub.

1.2 How to install

```
$ python setup.py install

or

$ pip install datadigitizer
```

1.3 Dependencies

```
numpy>=1.20
matplotlib>=3.4.0
Pillow>=9.0.0
```

1.4 License

`GNU General Public License v3 (GPLv3)`

CHAPTER

TWO

USER GUIDE

2.1 Simple Extraction

2.1.1 Open the app

Start the application by entering in the terminal:

```
python -m datadigitizer
```

The main windows looks like in figure *Main Window*

A quick overview of the different commands is shown in `Help->How` to use.

2.1.2 Load the image

Load an image from which you want to extract data by pressing `<Ctrl-o>` or from `File->Load Image`. Browse to the image and click OK.

2.1.3 Position XY Limits

Position 4 crosses for the axis limits in the order Xmin, Xmax, Ymin, Ymax by pointing them with the mouse and adding them by pressing `<Ctrl-a>`. You can hold down `<a>`, point with the mouse and left click for adding a red cross.

2.1.4 Set XY Limits

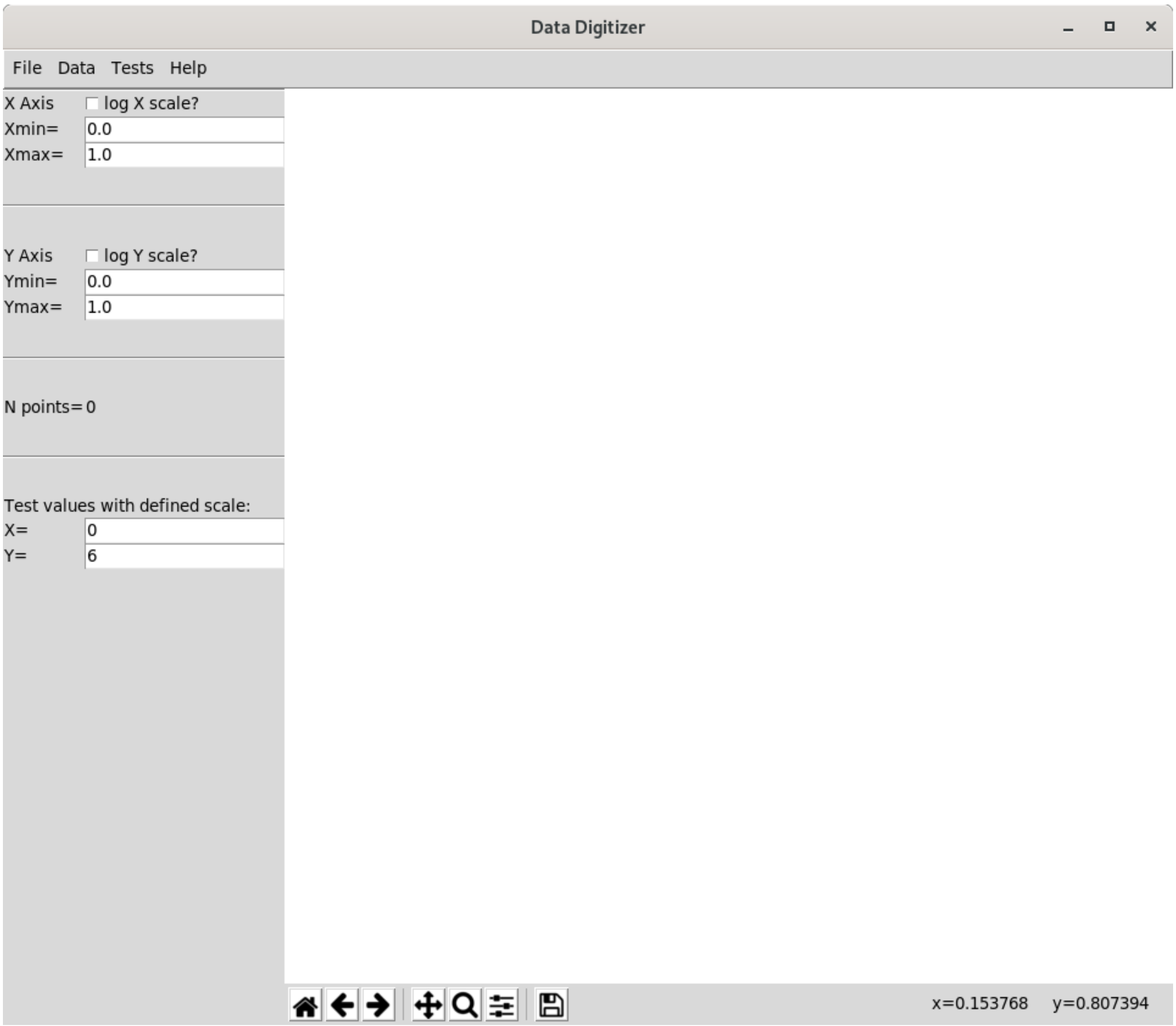
Press in the order `<Ctrl-k>`, `<Ctrl-j>`, `<Ctrl-h>`, `<Ctrl-g>`. It will set the Ymax, Ymin, Xmax and Xmin from the last (selected) data point, respectively.

- When `<Ctrl-k>` is pressed, the last of the 4 red crosses will become the Ymax value and will be colored in blue.
- When `<Ctrl-j>` is pressed, the last of the 3 red crosses will become the Ymin value and will be colored in blue.
- When `<Ctrl-h>` is pressed, the last of the 2 red crosses will become the Xmax value and will be colored in green.
- When `<Ctrl-k>` is pressed, the last red cross will become the Xmin value and will be colored in green.

You can set the XY limits in the different order if needed and you can also set all the limits at once from the last (selected) 4 data points (red crosses) by pressing `<Ctrl-l>`.

It is also possible to select a limit with a left click and adjust it by pressing left, right, up and down.

The limits can be reverted to data by pressing `<Ctrl-n>`.



X Axis □ log X scale?
Xmin= 0.0
Xmax= 1.0

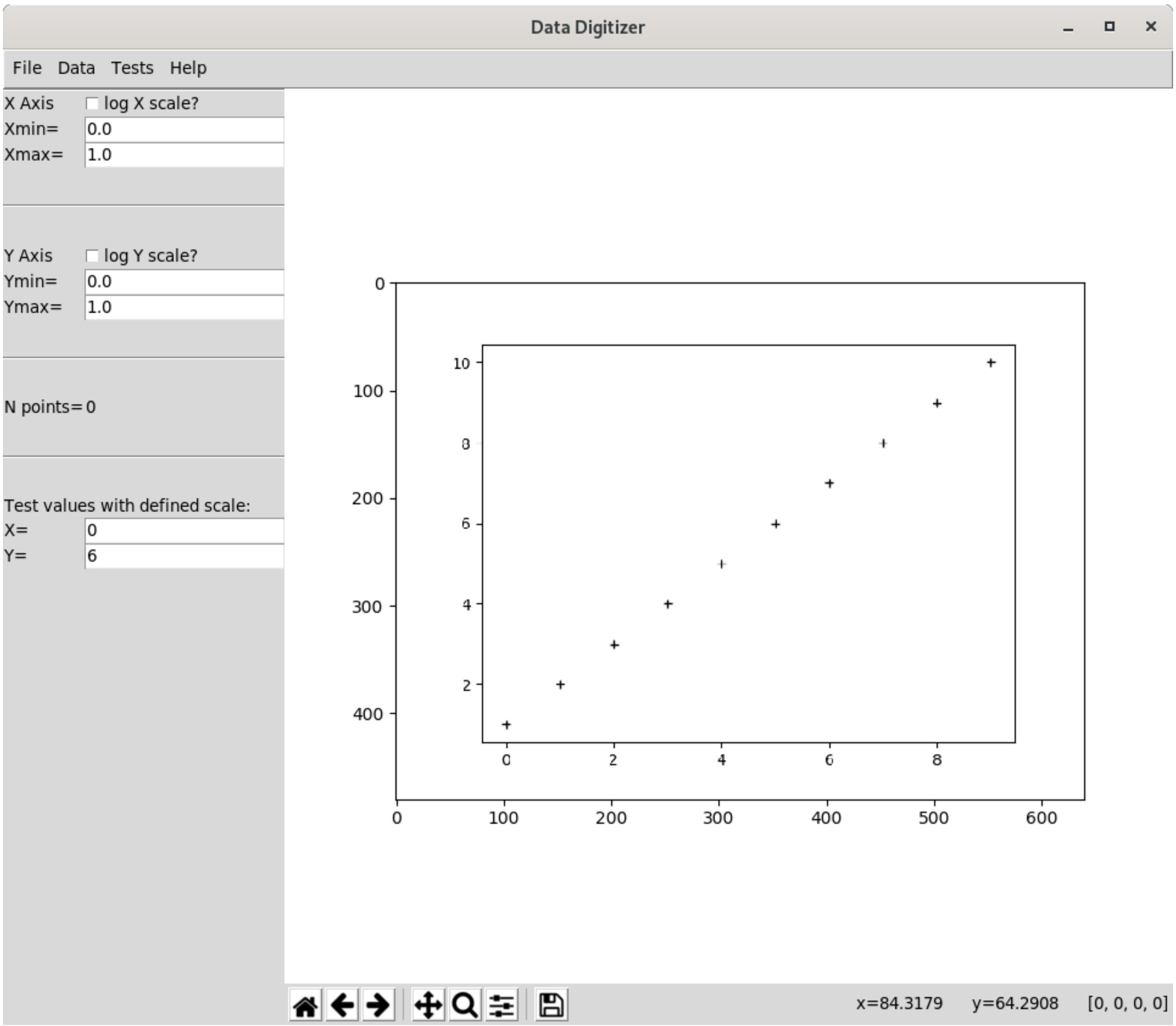
Y Axis □ log Y scale?
Ymin= 0.0
Ymax= 1.0

N points= 0

Test values with defined scale:
X= 0
Y= 6

x=0.153768 y=0.807394

Fig. 1: Main Window



X Axis ☐ log X scale?
Xmin= 0.0
Xmax= 1.0

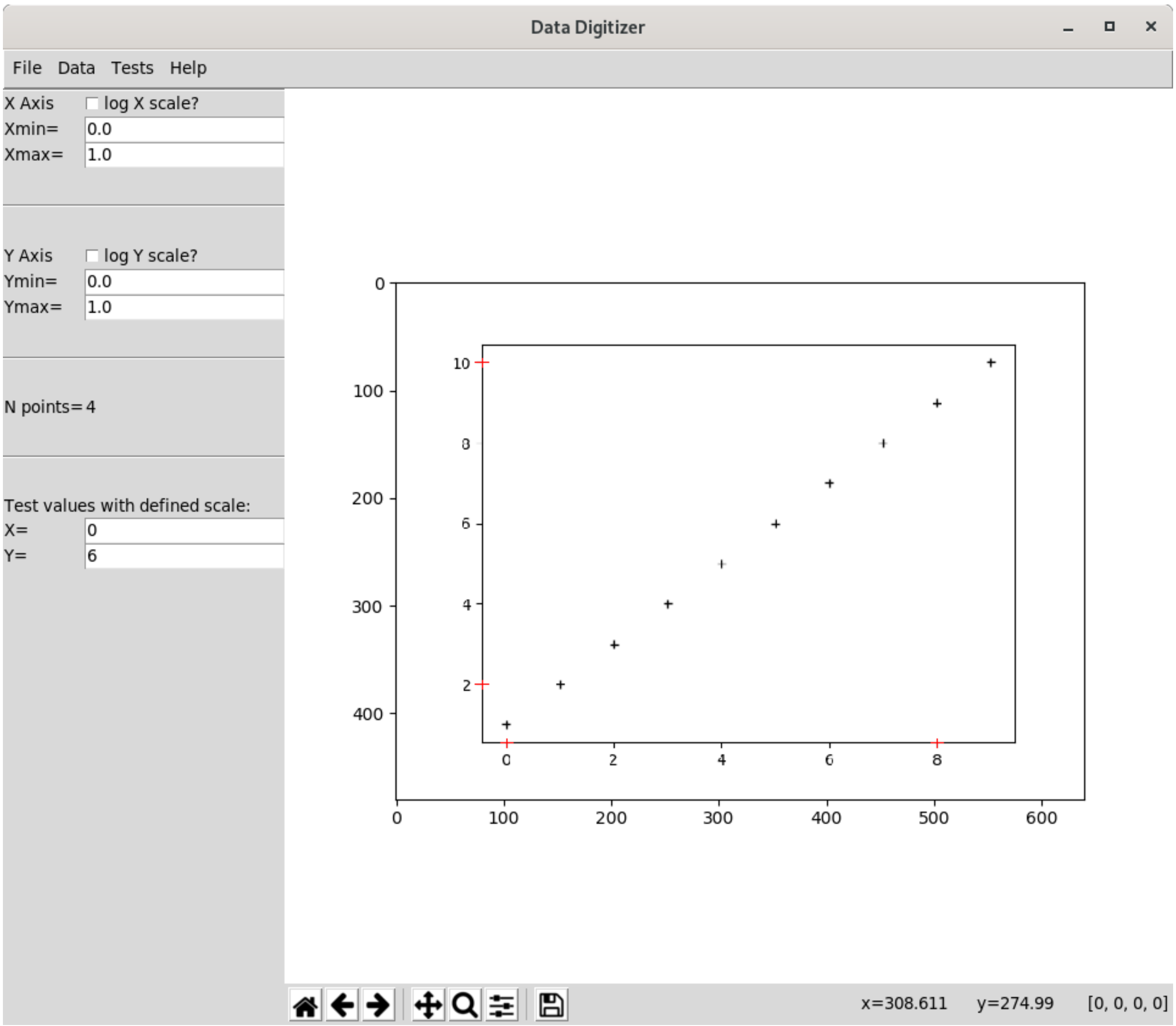
Y Axis ☐ log Y scale?
Ymin= 0.0
Ymax= 1.0

N points= 0

Test values with defined scale:
X= 0
Y= 6

x=84.3179 y=64.2908 [0, 0, 0, 0]

Fig. 2: Load the image from which to extract data



Data Digitizer

File Data Tests Help

X Axis □ log X scale?

Xmin= 0.0

Xmax= 1.0

Y Axis □ log Y scale?

Ymin= 0.0

Ymax= 1.0

N points= 4

Test values with defined scale:

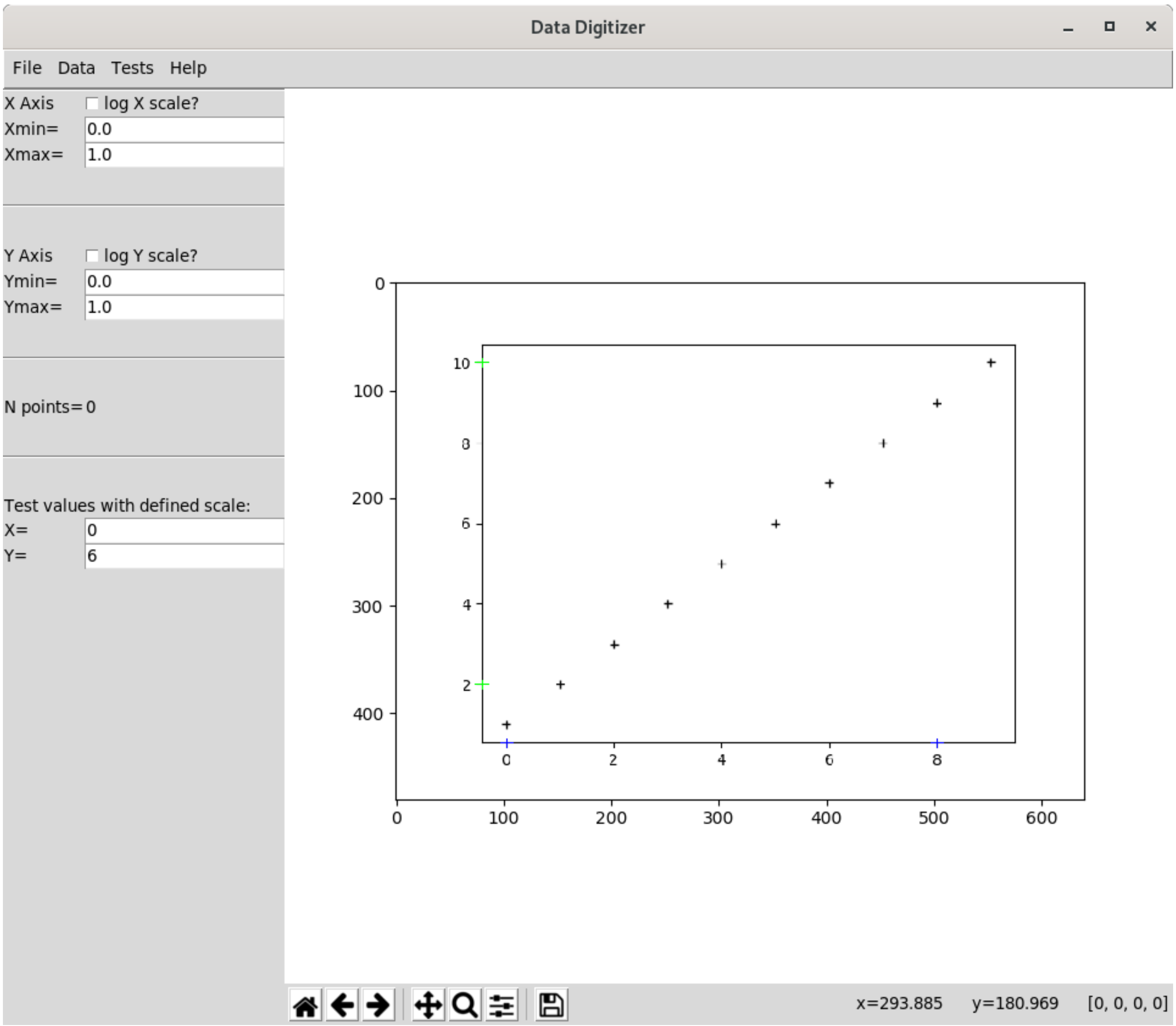
X= 0

Y= 6

x=308.611 y=274.99 [0, 0, 0, 0]

Fig. 3: Position limits for x and y axes

All shortcuts commands are also available through the menu Data.



Data Digitizer

File Data Tests Help

X Axis □ log X scale?
Xmin= 0.0
Xmax= 1.0

Y Axis □ log Y scale?
Ymin= 0.0
Ymax= 1.0

N points=0

Test values with defined scale:
X= 0
Y= 6

x=293.885 y=180.969 [0, 0, 0, 0]

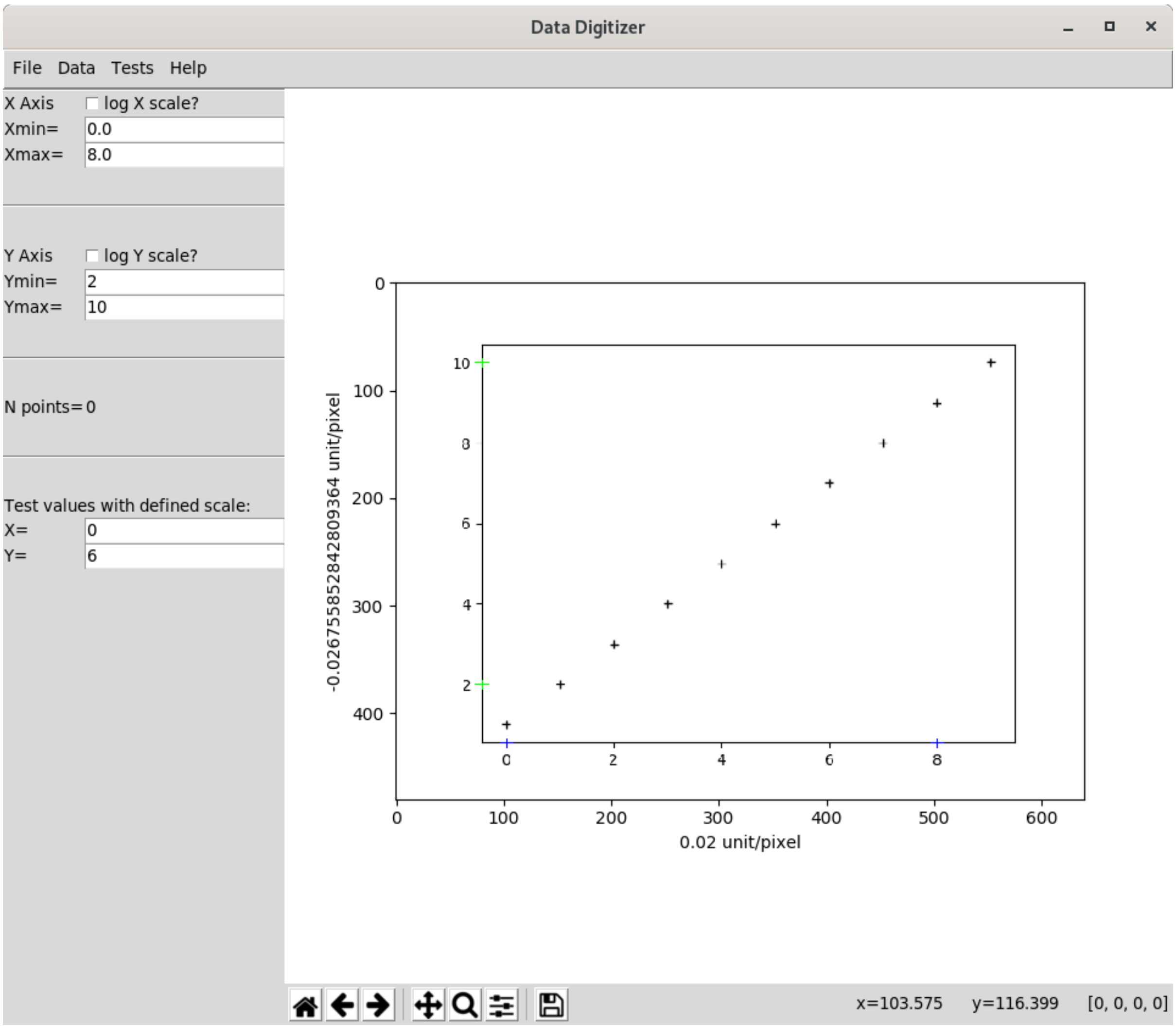
Fig. 4: Set XY limits

2.1.5 Enter XY Limits

Enter the corresponding value for Xmax, Xmin, Ymin and Ymax and press <`Enter`>. Switch to log scales if needed.

2.1.6 Add Data Points

Add data points by pointing them with the mouse and adding them by pressing <`Ctrl-a`>. You can hold down <a>, point with the mouse and left click for adding a red cross. Once a data point is added you can adjust its position by pressing left, right, up and down arrows. Press <`Ctrl-m`> or from the menu Data->Compute to compute the data with the definded XY scales. Press <`Ctrl-s`> or from the menu File->Save Data to save data.



X Axis □ log X scale?
Xmin= 0.0
Xmax= 8.0

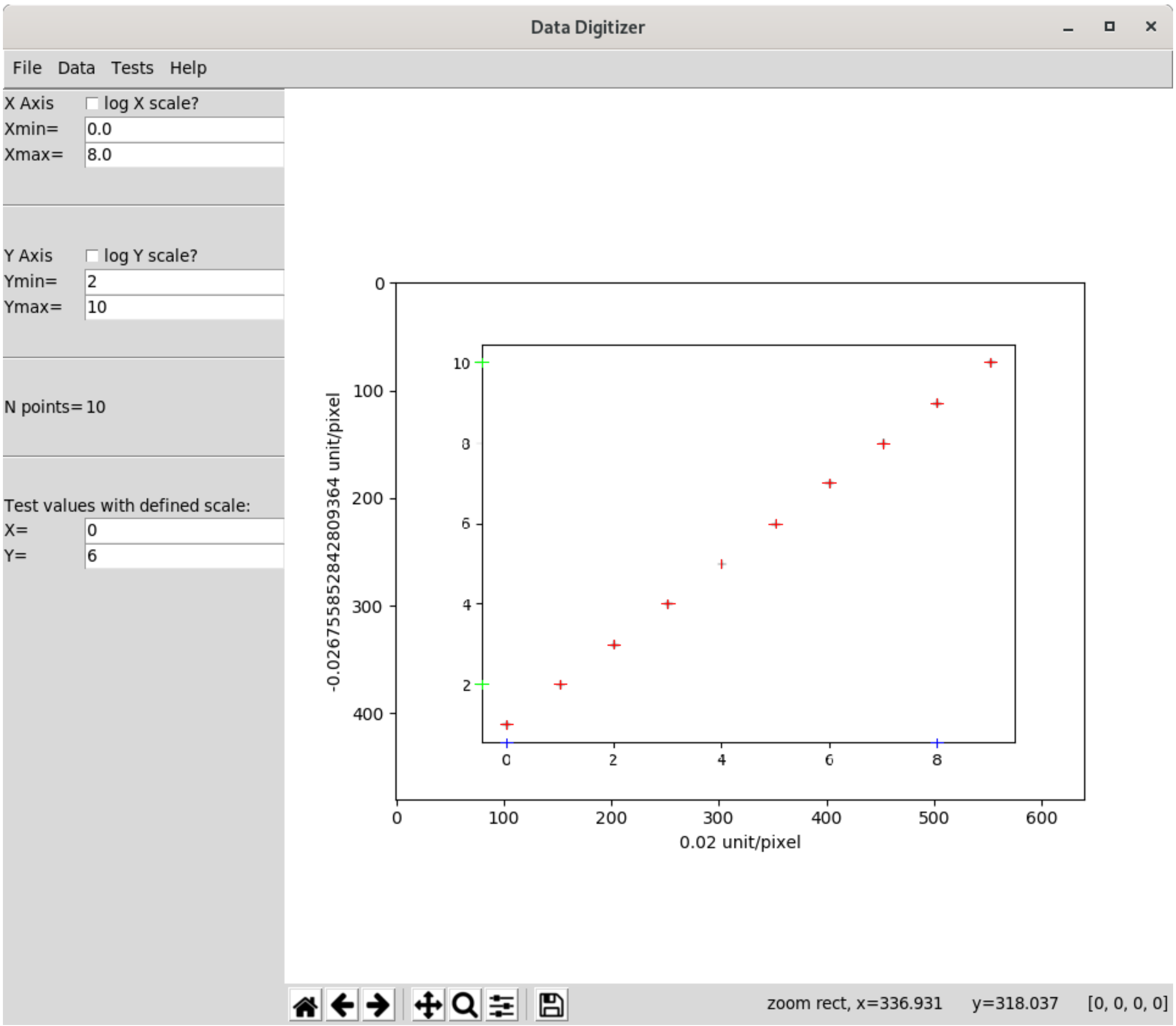
Y Axis □ log Y scale?
Ymin= 2
Ymax= 10

N points= 0

Test values with defined scale:
X= 0
Y= 6

x=103.575 y=116.399 [0, 0, 0, 0]

Fig. 5: Enter XY values



X Axis □ log X scale?

Xmin= 0.0

Xmax= 8.0

Y Axis □ log Y scale?

Ymin= 2

Ymax= 10

N points= 10

Test values with defined scale:

X= 0

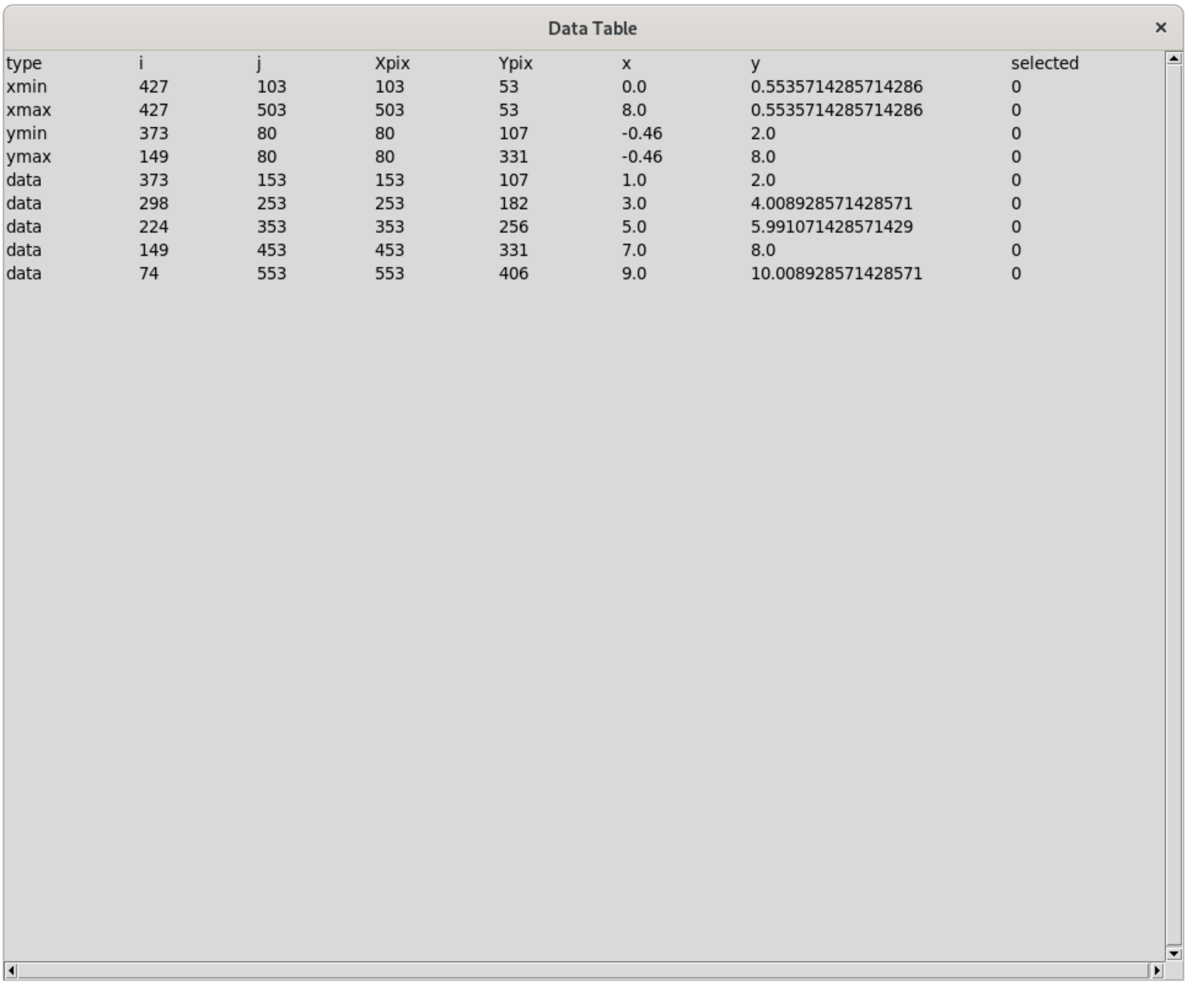
Y= 6

zoom rect, x=336.931 y=318.037 [0, 0, 0, 0]

Fig. 6: Set data values

2.1.7 View Data Points

The data values can be seen by pressing <`Ctrl-t`> or through the menu Data.

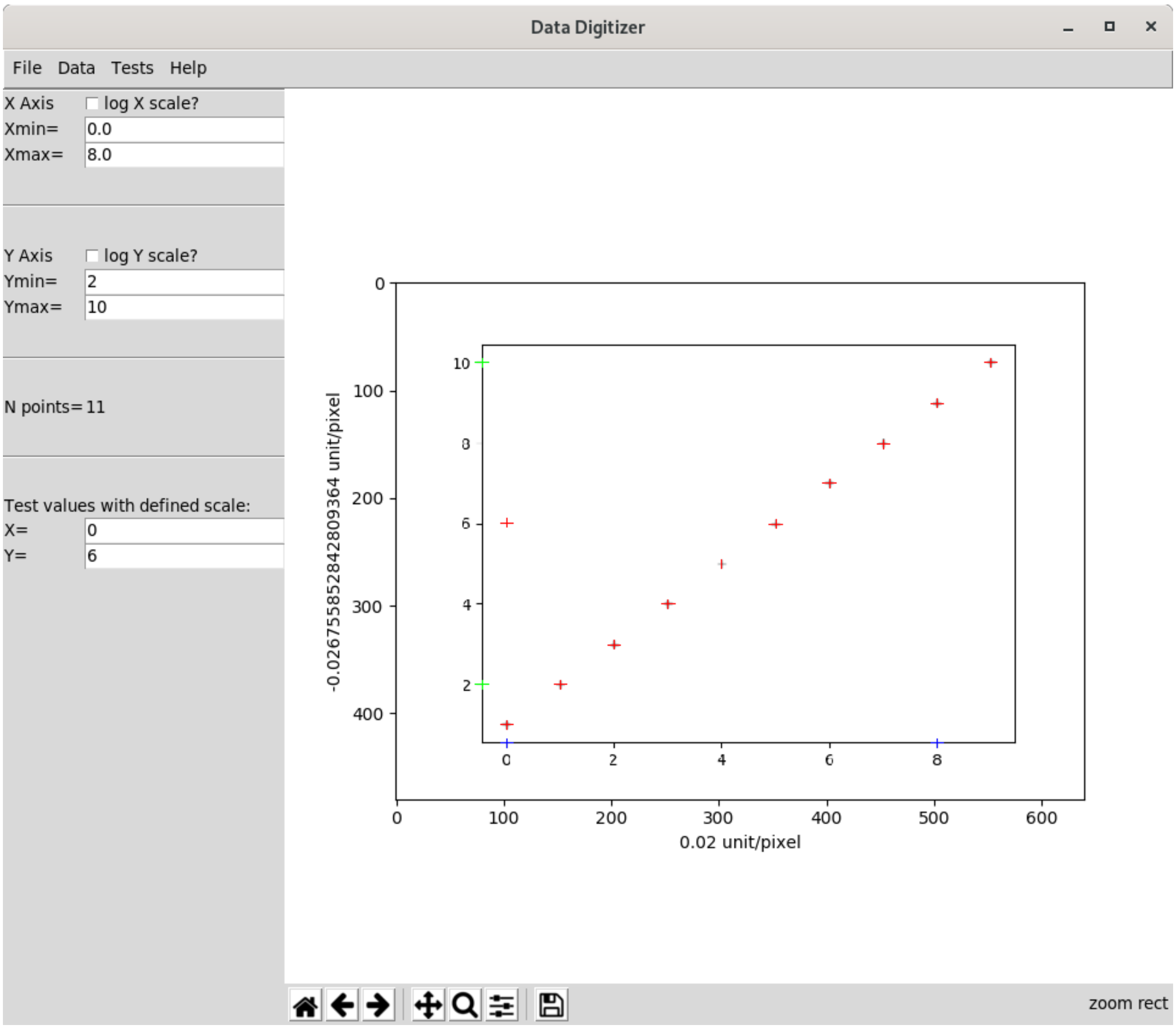


type	i	j	Xpix	Ypix	x	y	selected
xmin	427	103	103	53	0.0	0.5535714285714286	0
xmax	427	503	503	53	8.0	0.5535714285714286	0
ymin	373	80	80	107	-0.46	2.0	0
ymax	149	80	80	331	-0.46	8.0	0
data	373	153	153	107	1.0	2.0	0
data	298	253	253	182	3.0	4.008928571428571	0
data	224	353	353	256	5.0	5.991071428571429	0
data	149	453	453	331	7.0	8.0	0
data	74	553	553	406	9.0	10.008928571428571	0

Fig. 7: View data table

2.2 Test Scale Values

It is also possible de test the X/Y scales by entering values, e.g. X=0 and Y=6, press <`Enter`> to ckeck if the scales are properly set.



Data Digitizer

File Data Tests Help

X Axis □ log X scale?

Xmin= 0.0

Xmax= 8.0

Y Axis □ log Y scale?

Ymin= 2

Ymax= 10

N points= 11

Test values with defined scale:

X= 0

Y= 6

Fig. 8: Test scale

CHAPTER

THREE

RELEASE NOTES

3.1 Data Digitizer 1.2.2 Release Note

3.1.1 Summary

- Switch to pyproject.toml.

3.1.2 Download

PyPI

3.1.3 Contributors

Milan Skocic

3.1.4 What's changed

Full Changelog: https://github.com/MilanSkocic/PyDatadigitizer/compare/1.2.1…1.2.2

3.2 Data Digitizer 1.2.1 Release Note

3.2.1 Summary

- Refractoring and code cleaning.

3.2.2 Download

PyPI

3.2.3 Contributors

Milan Skocic

3.2.4 Commits

Full Changelog: https://github.com/MilanSkocic/PyDatadigitizer/compare/1.2.0. . . 1.2.1

3.3 Data Digitizer 1.2.0 Release Note

3.3.1 Changes

Change of numpy minimum version from 1.17 to 1.20 for taking advantage of the typing module.

Change of matplotlib minimum version from 3.0 to 3.4 for taking advantage of new features.

Simplify the settings in init file.

3.3.2 Download

PyPI

3.3.3 Contributors

Milan Skocic

3.3.4 Commits

Full Changelog: https://github.com/MilanSkocic/PyDatadigitizer/compare/1.1.2. . . 1.2.0

3.4 Data Digitizer 1.1.2 Release Note

3.4.1 Changes

Minor fixes in documentation.

3.4.2 Download

PyPI

3.4.3 Contributors

Milan Skocic

3.4.4 Commits

Full Changelog: https://github.com/MilanSkocic/PyDatadigitizer/compare/1.1.1…1.1.2

3.5 Data Digitizer 1.1.1 Release Note

3.5.1 Changes

Minor fixes in documentation.

3.5.2 Download

PyPI

3.5.3 Contributors

Milan Skocic

3.5.4 Commits

Full Changelog: https://github.com/MilanSkocic/PyDatadigitizer/compare/1.1.0…1.1.1

3.6 Data Digitizer 1.1.0 Release Note

3.6.1 Highlights

- Data table for visualizing the extracted data values as a toplevel window.
- Documentation bug fixes and improvements.

3.6.2 New Features

Data are visible as an array.

3.6.3 Download

PyPI

3.6.4 Contributors

Milan Skocic

3.6.5 Commits

Full Changelog: https://github.com/MilanSkocic/PyDatadigitizer/compare/1.0.2…1.1.0

3.7 Data Digitizer 1.0.2 Release Note

3.7.1 Highlights

- Documentation minor fixes.
- Explicit internal functions for converting (i,j) array indexes to (xpix, ypix) graph indexes
- Added `data folder` as an option for folder profile.

3.7.2 Future Changes

The data table will be visible in version 1.1 during the process of drawing data from the shortcut <Ctrl-t> or through the menu Data->View Data.

3.7.3 Compatibility notes

The full data array is now saved instead of (x, y) columns only. This changes does not bring any compatibility issue but it is worth mentionning that the format has changed.

3.7.4 Download

PyPI

3.7.5 Contributors

Milan Skocic

3.7.6 Commits

#c90ebea-MilanSkocic-ENH: Conversion as private functions

#be87095-MilanSkocic-ENH: Added conversion functions from i,j matrix indexes to xypix graph pixels

#94d1081-MilanSkocic-ENH: Separated array indexes i,j and pixel indexes in graph

#d39ca8f-MilanSkocic-ENH: Added folder options and full data array save

#5d0150d-MilanSkocic-ENH: Added options in folder profile

#61acb28-MilanSkocic-ENH: Refractoring according to pylint and mypy

3.8 Data Digitizer 1.0.1 Release Note

3.8.1 Highlights

Documentation minor fixes.

3.8.2 Contributors

Milan Skocic

3.8.3 Commits

#12ea7cb-MilanSkocic-DOC: Minor corrections in tutorial.rst.

#b171b83-MilanSkocic-DOC: Minor corrections in tutorial.

3.9 Data Digitizer 1.0.0 Release Note

3.9.1 Highlights

Initial release with basic features:

- Import image
- Set scale
- Multiple selection of data points
- Compute and save data

3.9.2 Contributors

Milan Skocic

3.9.3 Commits

PR#5-BUG: Fixed unwanted autoscale for each draw of the image threshold.

PR#4-ENH: New features with interactive points

PR#3-ENH: Cleanup.

PR#2-ENH: Added tests in the main app.

PR#1-ENH: Added tests.

#63fa293-Implementing FigureFrame in main app.

#6b25247-Initial commit with functional GUI and basic functions for adding points.

CHAPTER

FOUR

AUTOGENERATED DOCUMENTATION

4.1 Graphical FrontEnd

Main GUI.

`class datadigitizer.gui.AboutWindow(master)`

About window.

Parameters

master: tkinter widget
Container.

`class datadigitizer.gui.App(master=None)`

The cursor is used to point a specific position in the graph whereas all operations are done through keyboard combinations or through the main menu.

Legend:

- Red crosses are data points
- Blue crosses are Xmin and Xmax
- Green crosses are Ymin and Ymax

Commands:

- <Ctrl-o> for loading image.
- <Ctrl-a> add data point.
- <Hold a+Left Click> add data point.
- <Left Click> select a data point.
- <Hold Ctrl+Left Click> multiple data point selection.
- <Ctrl-g> set Xmin from last data point or from selected data point.
- <Ctrl-h> set Xmax from last data point or from selected data point.
- <Ctrl-j> set Ymin from last data point or from selected data point.
- <Ctrl-k> set Ymax from last data point or from selected data point
- <Ctrl-l> set all limits from last 4 data points or from last 4 selected data points.
- <Ctrl-n> remove all limits.
- <Ctrl-z> remove last data point.
- <Ctrl-d> remove selected data point.
- <Ctrl-D> remove all data points.
- <Ctrl-m> compute the data points.

- <Ctrl-t> view data table.
- <Ctrl-s> save data points.
- <Ctrl-w> clear all.

Parameters

master: tkinter.Tk instance
Root instanciation of tkinter.

Methods

run()	Start the application.
stop()	Stop the main tk loop.

run()

Start the application.

stop()

Stop the main tk loop.

class datadigitizer.gui.**DataTable**(*master*, ***kwargs*)

Scrolled data table widget.

Parameters

master: tkinter widget
Master container.

kwargs: dict, optional
Keyword arguments for the scrolled frame.

Methods

set_new_data(data)	Set new data in the displayed data table.

set_new_data(*data*)

Set new data in the displayed data table.

Parameters

data
[structured array, shape=(n,)] Numpy structured array used for registering the extracted data.

class datadigitizer.gui.**DataWindow**(*master*)

How to use window.

Parameters

master: tkinter widget
Container.

class datadigitizer.gui.**FigureFrame**(*master*, ***kwargs*)

Tk frame encapsulating a matplotlib figure and a toolbar.

Parameters

kwargs: dict, optional
Keyword arguments for the tk frame.

Methods

refresh()	Refresh the plot.

refresh()

Refresh the plot.

class datadigitizer.gui.**HowToUse**(*master*)

How to use window.

Parameters

master: tkinter widget
Container.

class datadigitizer.gui.**ScrolledFrame**(*master*, ***kwargs*)

Scrolled Frame widget which may contain other widgets and can have a 3D border.

Parameters

master: tkinter widget
Master container.

kwargs: dict, optional
Keyword arguments for the scrolled frame.

Attributes

canvas
Return the canvas that contains the scrollbars.

frame
Return the frame that contains the widgets.

property **canvas**

Return the canvas that contains the scrollbars.

property **frame**

Return the frame that contains the widgets.

class datadigitizer.gui.**Transform**(*values_min: float*, *values_max: float*, *pix_min: int | float*, *pix_max: int | float*, *which: str = 'linear'*)

Transform class converting values coordinates into pixel coordinates.

Parameters

values_min: int, float
Minimum value.

values_max: int, float
Maximum value.

pix_min: int, float
Minimum pixel.

pix_max: int, float
Maximum pixel.

which: str, optional
Which kind of transform i.e. linear or log.

Attributes

backward_scale
Return the scale for transforming pixels into values.

forward_scale
Return the scale for transforming values into pixels.

Methods

backward(x)	Transform pixels to values.
forward(x)	Transform values to pixels.

backward(*x: int* | *float* | *ndarray*)

Transform pixels to values.

Parameters

x: int or floats or array-like, shape(n,)
Pixels to be transformed.

Returns

values: int or floats or array-like, shape(n,)
Values corresponding to the pixels.

Notes

$$x = (x_{pix} - x_{pix,min})\frac{x_{max} - x_{min}}{x_{pix,max} - x_{pix,min}} + x_{min}$$

property backward_scale

Return the scale for transforming pixels into values.

$$\frac{x_{max} - x_{min}}{x_{pix,max} - x_{pix,min}}$$

forward(*x: int* | *float* | *ndarray*)

Transform values to pixels.

Parameters

x: int or floats or array-like, shape(n,)
Values to be transformed.

Returns

pixels: int or floats or array-like, shape(n,)
Values corresponding to the pixels.

Notes

$$x_{pix} = (x - x_{min})\frac{x_{pix,max} - x_{pix,min}}{x_{max} - x_{min}} + x_{pix,min}$$

property forward_scale

Return the scale for transforming values into pixels.

$$\frac{x_{pix,max} - x_{pix,min}}{x_{max} - x_{min}}$$

4.2 Icon

Create icon for the app.

4.3 Settings

Settings module.

`datadigitizer.settings.`**`save_cfg`**`()`

Save the configuration file.

4.4 Tests

Tests module.

`class` `datadigitizer.tests.`**`TestPlotData`**`(`*methodName='runTest'*`)`

Create an instance of the class that will use the named test method when executed. Raises a ValueError if the instance does not have a method with the specified name.

Methods

test_linear()	Test linear plot.
test_loglog()	Test log-log plot.
test_xlog()	Test x semi-log plot.
test_ylog()	Test y semi-log plot.

`test_linear``()`

Test linear plot.

`test_loglog``()`

Test log-log plot.

`test_xlog``()`

Test x semi-log plot.

`test_ylog``()`

Test y semi-log plot.

`datadigitizer.tests.`**`test_linear`**`()` → Path

Generate the linear plot and data.

Returns

fpath: Path object
Path to the linear plot.

`datadigitizer.tests.`**`test_loglog`**`()` → Path

Generate the log-log plot and data.

Returns

fpath: Path object
Path to the log-log plot.

`datadigitizer.tests.test_xlog()` → Path

Generate the semi-log plot and data.

Returns

fpath: Path object
Path to the semi-log plot.

`datadigitizer.tests.test_ylog()` → Path

Generate the semi-log plot and data.

Returns

fpath: Path object
Path to the semi-log plot.

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

INDEX