## Contents

1 Data Plotly 1

2 DataPlotly Basic Usage 5
   2.1 Add a Plot 6
   2.2 Multi Plots 7
   2.3 Save Plot 11
   2.4 Raw Plot Code 12
   2.5 Layer fields tips 13

3 Scatter Plot 15
   3.1 Plot Properties 16
   3.2 Plot Customizations 17

4 Box Plot 19
   4.1 Plot Properties 20
   4.2 Plot Customizations 21

5 Bar Plot 25
   5.1 Plot Properties 26
   5.2 Plot Customizations 27

6 Histogram 31
   6.1 Plot Properties 32
   6.2 Plot Customizations 33

7 Pie Plot 35
   7.1 Plot Properties 36
   7.2 Plot Customizations 37

8 2D Histogram 39
   8.1 Plot Properties 40
   8.2 Plot Customizations 40

9 Polar Plot 43
   9.1 Plot Properties 44
   9.2 Plot Customizations 45

10 Ternary Plot 47
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1 Plot Properties</td>
<td>48</td>
</tr>
<tr>
<td>10.2 Plot Customizations</td>
<td>49</td>
</tr>
<tr>
<td>11 Contour Plot</td>
<td>51</td>
</tr>
<tr>
<td>11.1 Plot Properties</td>
<td>52</td>
</tr>
<tr>
<td>11.2 Plot Customizations</td>
<td>55</td>
</tr>
<tr>
<td>12 Violin Plot</td>
<td>57</td>
</tr>
<tr>
<td>12.1 Plot Properties</td>
<td>57</td>
</tr>
<tr>
<td>12.2 Plot Customizations</td>
<td>58</td>
</tr>
<tr>
<td>13 DataPlotly as Processing provider</td>
<td>61</td>
</tr>
<tr>
<td>13.1 Activating DataPlotly for Processing</td>
<td>61</td>
</tr>
<tr>
<td>13.2 Simple DataPlotly usage</td>
<td>62</td>
</tr>
<tr>
<td>14 Call the plugin from the python console (BETA!)</td>
<td>65</td>
</tr>
<tr>
<td>14.1 Code example</td>
<td>65</td>
</tr>
<tr>
<td>15 Indices and tables</td>
<td>67</td>
</tr>
</tbody>
</table>
Data Plotly is a python plugin for QGIS that allows the creation of D3 like plots thanks to the Plotly library and the python API.

Plots are totally **dynamic** so you can interact with the plot, e.g. zooming, getting some information hovering the mouse and many other useful stuff.

Moreover, plot and map canvas are **connected**, this mean you can select an object on the plot and the same object will be selected in the map:

![Plot and Map interaction](image)

For each plot, tons of customizations are available.

With DataPlotly you can create different plots, even different plot types, overlay them in the same plot canvas or load them in different plot canvas:

Each plot type has its own configurations and customizations: it is very easy for you to choose the best solution that fits all your needs.

Besides the interactive plot canvas, plots can be exported as **png** static images or as pure **html** files. **Html** file keep the plot **interactivity**.

Finally, **plot raw html code** can be copied and pasted somewhere else, for example in html frame of the QGIS map composer or in an external website.

**Plotly library** javascript code is saved in the local plugin folder so the plugin is usable also without any Internet connection.
Fig. 2: simple plot example

Fig. 3: different plot types in the same map canvas
Fig. 4: different plot types in separated plot canvas
DataPlotly Basic Usage

DataPlotly interface has been designed in order to be simple but, at the same time, complete and with many options and customizations available.

Creating a plot is just easy as it sounds: choose the plot type, $x$ and $y$ columns, colors, labels, etc and press the Create Plot button. The plot is immediately shown in the plot canvas.

If you want to change some setting, e.g. the marker color or size, the $x$ column, go ahead and the hit the Update Plot button: the plot is then updated in the plot canvas.

DataPlotly is a docket widget meaning that you can move it within the QGIS interface. It is made up by 5 different widgets:

- fundamental plot settings
- additional plot customizations
- plot canvas
- user guide and help for each plot
- raw html code
2.1 Add a Plot

DataPlotly workflow is pretty straightforward:

1. choose the plot type (1)
2. set the layer and the field(s) you want to plot (2)
3. show the plot with the Create Plot button (3). The plot canvas is automatically visible.
You can change some plot settings and click on the Update button to update the plot with the changes.
If you want to start again, just click the Clean Plot Canvas button and the plot canvas will be empty.

2.2 Multi Plots

DataPlotly allows the creation of many different plots. Plots can be shown on the same plot canvas (overlapping) or each plot can be drawn in different rows or columns (subplots).

2.2.1 Overlapping Plots

You can add as many plots as you want within the same plot view. Plot types can be different and also the source layer can be different.

Note: results can be very strange depending on the plot type and on the fields you choose. Be careful!

To add other plot just repeat the steps of Add a Plot by choosing different plot types and/or just different layer fields, etc..
In the following picture, same plot type (scatterplot) and different fields of the same layer:
The following pictures show 2 different overlapping plot types (boxplot and scatterplot):
2.2.2 Subplots

You can choose to separate the plots in different plot canvas. It is particularly useful when the scales are very different or when overlapping too many data results in a messy plot canvas.

You just have to choose the plots and the fields as described in the section Add a Plot but you have to specify the SubPlots parameter from the combobox and choose if the plots have to be drawn in rows (default parameter) or in columns.

The following pictures show plots in rows and in columns:
Chapter 2. DataPlotly Basic Usage
2.3 Save Plot

Saving a plot, technically the plot canvas, is very simple. You can choose to save the plot as a png image or as html file.

**Note:** Saving the image as html file will keep the interactivity of the plot. The html file relies on the local DataPlotly installation, and cannot in general be used on other systems.

You just have to click on the correct button and choose the path where to save the image, both static or interactive.
2.4 Raw Plot Code

In addition to saving the plot as image or html file (see Save Plot) you can also copy the raw html code of the plot and embed it somewhere else.

A good place where to copy/paste the raw code is the html frame of the print composer of QGIS.

In order to copy the plot code, after the plot creation, just go in the Raw Plot tab: here you can see a long string. Right Click on the tab and choose Select All, then Right Click again and choose Copy (of course you can use keyboard shortcuts Ctrl + A for select all and Ctrl + C for copy):

Then you have your plot code copied in memory: you just have to choose where to paste it.

In the following example, the html code is pasted in the html frame of the print composer:

1. open the print composer and add an html frame
2. paste the code in the Source space

---

12 Chapter 2. DataPlotly Basic Usage
3. refresh the html code (3)

4. results will be shown in the frame (4)

2.5 Layer fields tips

2.5.1 Using Expressions

DataPlotly supports all the fields type (integers, floats, text, etc..) and will elaborate the data so that they can be correctly displayed.

Thanks to the QGIS API and custom widget, it is possible to add also Expressions instead of pure layer fields (e.g. field + 10, field1 * field2, etc).

You can use the Expression editor to add complex expressions (e.g. (field1 + 10) * (field2 * 10)) or you can type the expression directly in the combo box. Expressions are evaluated on the fly, so if the string is red, then the expression is not valid.
Another very handy options is to work only with the selected features on the layer.

Just check the **Use only selected features** check box and only the attributes of the selected features will be taken.
Here you will find the guide to every parameter of the scatter plot. If you need more generic information please see
\textit{DataPlotly Basic Usage}. 
3.1 Plot Properties

Layer: the combobox will display all the vector layers loaded in QGIS

X Field: the X field

Y Field: the Y field

Marker Color: marker color

Data Defined Override: you can add an Expression to define the size of the marker. If activated other options are available: Color Scale, Visible and Invert Color.

Marker Size: the size of the marker

Data Defined Override: you can add an Expression to define the size of the marker

Stroke Color: border color
Stroke Width: the width of the border
Marker Type: choose between Points, Lines or Points and Lines
Point Type or Line Type depending on the Marker Type chosen
Transparency: transparency level of the point/line

### 3.2 Plot Customizations

Show Legend: show the legend of the current plot
Horizontal Legend: check if you want to have an horizontal legend
Show Range Slider: add a slider below the plot

Plot Title: the plot title
Legend Title: the title of the legend

X Label: X label text

Y Label: Y Label text

Hover Infos: choose the information that will be shown when hovering on the Plot Canvas

Additional Hover Label: choose another field of the plot (or other values) that will be shown together with the other informations. This field supports expressions: e.g. 'The ID of this point is ' || ID:

Invert X Axis: check if you want to invert the ordering of the x axis

X Axis Mode: choose between Linear, Logarithmic or Categorized

Invert Y Axis: check if you want to invert the ordering of the y axis

Y Axis Mode: choose between Linear, Logarithmic or Categorized
HERE you will find the guide to every parameter of the box plot. If you need more generic information please the *DataPlotly Basic Usage*. 
4.1 Plot Properties

**Layer**: the combobox will display all the vector layers loaded in QGIS

**Grouping Field (Optional)**: the grouping field. If no field is added, a generic box plot for the Y value will be displayed

**Y Field**: the Y field

**Marker Color**: color of the box

**Stroke Color**: border color

**Stroke Width**: the width of the border

**Transparency**: transparency level of the box
4.2 Plot Customizations

Show Legend: show the legend of the current plot

Horizontal Legend: check if you want to have an horizontal legend

Plot Title: the plot title

Legend Title: the title of the legend

X Label: X label text

Y Label: Y Label text

Invert X Axis: check if you want to invert the ordering of the x axis

X Axis Mode: choose between Linear, Logarithmic or Categorized

Invert Y Axis: check if you want to invert the ordering of the y axis

Y Axis Mode: choose between Linear, Logarithmic or Categorized

Box Orientation: the orientation of the box, either Horizontal or Vertical

Show Statistics: choose to show also additional statistics on each box plot, like the mean or the standard deviation (shown in the picture below)
Outliers: show also the outliers (or all the points) next to the box. Some example in the following picture:
Here you will find the guide to every parameter of the bar plot. If you need more generic information please see *DataPlotly Basic Usage*.
5.1 Plot Properties

Layer: the combobox will display all the vector layers loaded in QGIS

X: the X field

Y Field: the Y field

Bar Color: color of the bar

Data Defined Override: you can add an Expression to define the size of the marker. If activated other options are available: Color Scale, Visible and Invert Color.

Stroke Color: border color

Stroke Width: the width of the border

Transparency: transparency level of the bar
5.2 Plot Customizations

Show Legend: show the legend of the current plot

Horizontal Legend: check if you want to have an horizontal legend

Plot Title: the plot title

Legend Title: the title of the legend

X Label: X label text

Y Label: Y label text

Invert X Axis: check if you want to invert the ordering of the x axis

Invert Y Axis: check if you want to invert the ordering of the y axis

Bar Orientation: the orientation of the bar, either Horizontal or Vertical

Bar Mode: this option is handy if you have more data to plot:

1. Grouped: the bars are placed one next the other
2. **Stacked**: bars are stacked one over the other
3. **Overlay**: bars can be overlayed. Adjust the *Transparency* setting
CHAPTER 6

Histogram

Here you will find the guide to every parameter of the histogram. If you need more generic information please see DataPlotly Basic Usage.
6.1 Plot Properties

Layer: the combobox will display all the vector layers loaded in QGIS

X: the values in this field will be binned and the count in each bin will be represented as the length of the bars

Marker Color: the color of the bars. To use different colors for individual bars, enter an expression that returns an array of color definition strings with as many elements as there are bars

For example

```python
array('red', 'green', 'blue', 'yellow', 'purple')
```

or, to use colors from a color ramp
array_foreach(
    generate_series(0, 7),
    ramp_color('Blues', @element/7)
)

Stroke Color: border color
Stroke Width: the width of the border
Transparency: transparency level of the bar

6.2 Plot Customizations

Show Legend: show the legend of the current plot
Horizontal Legend: check if you want to have an horizontal legend
Plot Title: the plot title
Legend Title: the title of the legend
X Label: X label text
Invert X Axis: check if you want to invert the ordering of the x axis
Invert Y Axis: check if you want to invert the ordering of the y axis
Bar Orientation: the orientation of the bar, either Horizontal or Vertical
Bar Mode: choose if you want to group, stack or overlay the data
Normalization: different type of bins calculations are available
Cumulative Histogram: check if you want to create a cumulative histogram
Invert Histogram Direction: check if you want to invert the histogram direction (only if cumulative option is chosen)
Choose manually the bins: check if want to choose manually the bins number and add a numeric value in the box
Bar gap: choose the space between the gaps, from 0 (no space) to 1 (maximum space)
Pie Plot

Here you will find the guide to every parameter of the pie plot. If you need more generic information please see *DataPlotly Basic Usage*. 

35
7.1 Plot Properties

Layer: the combobox will display all the vector layers loaded in QGIS

Grouping: features with the same value in the Grouping field are assigned to the same pie segment

Y Field: the Y field determines the size of the pie segment. If features are assigned to the same pie segment because they have the same value in the Grouping field the corresponding values in the Y field are summed up to determine the size of the pie segment. To create a pie chart displaying the number of features belonging to each pie segment enter a \( j \) instead of a field name

Marker color: the color used for filling the first pie segment. Subsequent pie segments are filled using the default Plotly color scale.

To control the color of the individual pie segments use an expression yielding color strings depending on the value in the Grouping field.
For example, to create a pie chart showing the population in each continent using the `Natural Earth ne_110m_admin_0_countries` layer, choose `CONTINENT` as the Grouping field and `POP_EST` as the Y field. Then, to set the colors for the pie segments use the following expression in the Marker color field:

```python
map_get(
    map(
        'Asia', '#8dd3c7',
        'Europe', '#ffffb3',
        'North America', '#bebada',
        'South America', '#fb8072',
        'Africa', '#80b1d3',
        'Oceania', '#fdb462',
        'Antarctica', '#b3de69',
        'Seven seas (open ocean)', '#fccde5'
    ), "CONTINENT"
)
```

### 7.2 Plot Customizations

**Show Legend**: show the legend of the current plot

**Legend Title**: the title of the legend

**Horizontal Legend**: check if you want to have an horizontal legend
2D Histogram

Here you will find the guide to every parameter of the 2D Histogram plot. If you need more generic information please see *DataPlotly Basic Usage.*
8.1 Plot Properties

Layer: the combobox will display all the vector layers loaded in QGIS
X Field: the X field
Y Field: the Y field
Color Scale: choose between many different color scales

8.2 Plot Customizations

Show Legend: show the legend of the current plot
Plot Title: the plot title
Legend Title: the title of the legend
X Label: X label text
Y Label: Y Label text
Invert X Axis: check if you want to invert the ordering of the X axis
Invert Y Axis: check if you want to invert the ordering of the Y axis
Here you will find the guide to every parameter of the polar plot. If you need more generic information please see *DataPlotly Basic Usage*. 
9.1 Plot Properties

Layer: the combobox will display all the vector layers loaded in QGIS
X Field: the X field
Y Field: the Y field
Marker Color: marker color
Marker Size: the size of the marker
Stroke Color: border color
Stroke Width: the width of the border
Transparency: transparency level of the marker/line
9.2 Plot Customizations

Show Legend: show the legend of the current plot
Plot Title: the plot title
Legend Title: the title of the legend
Here you will find the guide to every parameter of the ternary plot. If you need more generic information please see *DataPlotly Basic Usage*. 
10.1 Plot Properties

Layer: the combobox will display all the vector layers loaded in QGIS

X Field: the X field

Y Field: the Y field

Z Field: the Z field

Marker Color: marker color

Data Defined Override: you can add an Expression to define the size of the marker. If activated other options are available: Color Scale, Visible and Invert Color.

Marker Size: the size of the marker

Data Defined Override: you can add an Expression to define the size of the marker
Stroke Color: border color
Stroke Width: the width of the border
Point Type: marker type
Transparency: transparency level of the marker/line

10.2 Plot Customizations

Show Legend: show the legend of the current plot
Horizontal Legend: check if you want to have an horizontal legend
Plot Title: the plot title
Legend Title: the title of the legend
X Label: X label text
Y Label: Y Label text
Z Label: Z Label text
Additional Hover Label: choose another field of the plot (or other values) that will be shown together with the other informations. This field supports expressions: e.g. 'The ID of this point is ' || ID:
Here you will find the guide to every parameter of the contour plot. If you need more generic information please see *DataPlotly Basic Usage*. 
11.1 Plot Properties

Layer: the combobox will display all the vector layers loaded in QGIS

X Field: the X field

Y Field: the Y field

Contour Type: choose the type of contour:

1. Fill: the plot is filled everywhere:
2. **Heatmap**: a heatmap like plot will be created:
3. **Only Lines**: only the contour lines will be displayed:
Show Lines: uncheck if no contour lines should be added to the plot
Color Scale: choose between many different color scales

11.2 Plot Customizations

Show Legend: show the legend of the current plot
Plot Title: the plot title
Legend Title: the title of the legend
Here you will find the guide to every parameter of the violin plot. If you need more generic information please the DataPlotly Basic Usage.

12.1 Plot Properties

Layer: the combobox will display all the vector layers loaded in QGIS
**Grouping Field (Optional):** the grouping field. If no field is added, a generic violin plot for the Y value will be displayed

**Y Field:** the Y field

**Marker Color:** color of the violin

**Stroke Color:** border color

**Stroke Width:** the width of the border

**Transparency:** transparency level of the violin

### 12.2 Plot Customizations

**Show Legend:** show the legend of the current plot

**Horizontal Legend:** check if you want to have an horizontal legend

**Plot Title:** the plot title

**Legend Title:** the title of the legend

**X Label:** X label text

**Y Label:** Y Label text

**Box Orientation:** the orientation of the violin, either *Horizontal* or *Vertical*

**Outliers:** show also the outliers (or all the points) next to the box. Some example in the following picture:
Violin side: if the violin plot should be both sides, only right or only left
Show Mean Line: uncheck to not show a line corresponding to the mean
DataPlotly as Processing provider

From version 1.4 DataPlotly has been added as Processing provider thank to the work of Michaël Douchin of 3Liz. This opens the doors to infinite possibilities:

- using all the Processing methods with DataPlotly
- creating batch plots with one click
- adding DataPlotly in the Processing Graphical Modeler

13.1 Activating DataPlotly for Processing

First thing to do is activating Processing and add DataPlotly as provider. If not added by default you can go in Settings -> Options and click on the last tab named Processing.

Expand the Provider menu and activate the DataPlotly provider:
And then you will see the DataPlotly provider in the Processing Toolbox:

### 13.2 Simple DataPlotly usage

The **Build a generic plot** algorithm is a simplified version of DataPlotly plugin: this means you don’t have all the customizations available, but still you can create awesome plots.
The interface is very simple and the plot creation process is very straightforward:

- choose the layer
- choose the plot type
- adjust the additional options (plot title, X axis, Y axis and color)

DataPlotly will create 2 results:

- html file with the final plot
- json file with all the plot specifications

The plot can be opened in the Processing -> Result Viewer:
And by simple double clicking on the menu entry, the plot will be opened in your default browser:

If you want to save also the json file, you have to specify the path in the DataPlotly Processing Window.
Call the plugin from the python console (BETA!)

DataPlotly comes with some simple API that can be used from the python console. Actually the method accept a structured dictionary as input, calls and open the dialog of the plugin populating each field with the dictionary values and creating the final plot.

**Note:** not all the customization are (yet) available in this method, but you can easily update the plot with the *Update Plot* button. See DataPlotly Basic Usage

### 14.1 Code example

The following example is very simple and straightforward. Supposing you have already a layer loaded in the QGIS legend we will access to is, take 2 fields and building a simple scatter plot.

Open QGIS and the python console. The example considers the $PH$ and $T$ fields of the layer (just look at the code and change the fields according to your layer):

```python
# create the VectorLayer object from with iface
vl = iface.activeLayer()

# import the plugins
from qgis.utils import plugins

# create the instance of the DataPlotly plugin
my = plugins['DataPlotly']

# initialize and empty dictionary
dq = {}

# create nested dictionaries for plot and layour properties
dq['plot_prop'] = {}
dq['layout_prop'] = {}
```

(continues on next page)
# start to fill the dictionary with values you want

# plot type
dq['plot_type'] = 'scatter'
# QgsVectorLayer object
dq['layer'] = vl

# choose the plot properties
dq['plot_prop']['x'] = [i['O2'] for i in vl.getFeatures()]
dq['plot_prop']['y'] = [i['EC'] for i in vl.getFeatures()]
dq['plot_prop']['marker'] = 'markers'
dq['plot_prop']['x_name'] = 'O2'
dq['plot_prop']['y_name'] = 'EC'

# choose the layout properties
dq['layout_prop']['legend'] = True
dq['layout_prop']['range_slider'] = {}
dq['layout_prop']['range_slider']['visible'] = False

# call the method that opens the dialog
my.loadPlot(dq)

Using the example code shown above should open the DataPlotly dialog, create the plot and also populate the dialog fields (comboboxes, checkboxes, etc.) with the values chosen (where possible).

**Warning:** this method is still in **BETA** and there are several known issues.
CHAPTER 15

Indices and tables

• genindex
• modindex
• search