https://medium.datadriveninvestor.com/pandasgui-a-powerful-tool-for-exploring-visualizing-and-manipulating-pandas-dataframes-e0cdfdb90a11

PandasGUI: A Powerful Tool for Exploring, Visualizing and Manipulating Pandas DataFrames

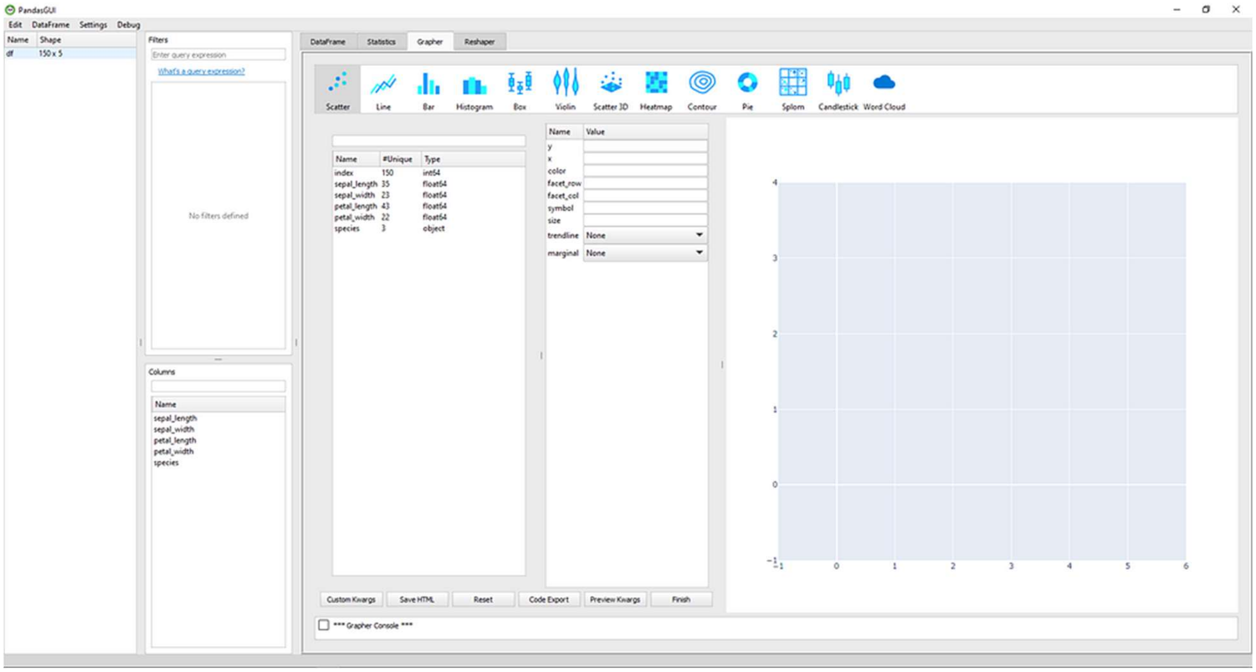
GUI for Viewing, Plotting, and Analyzing Pandas DataFrames.

·

Mar 3, 2024

159

1



Name	Shape
df	150 x 5

Name
sepal_length
sepal_width
petal_length
petal_width
species

Name	#Unique	Type
index	150	int64
sepal_length	35	float64
sepal_width	23	float64
petal_length	43	float64
petal_width	22	float64
species	3	object

Name	Value
y	
x	
color	
facet_row	
facet_col	
symbol	
size	
trendline	None
marginal	None

PandasGUI

Pandas is one of the most popular and powerful Python libraries for data analysis and manipulation. It provides a rich set of data structures and operations for working with tabular and time series data. However, sometimes it can be challenging to explore and understand the data using only the command line or a notebook interface. That's where PandasGUI comes in handy.

PandasGUI is a graphical user interface (GUI) for viewing, plotting, and analyzing Pandas DataFrames — Python module. It allows you to interact with your data in a visual and intuitive way, without writing any code. You can easily inspect, filter, sort, group, aggregate, pivot, and plot your data using a simple and user-friendly interface. You can also export your data or plots to various formats, such as CSV, Excel, HTML, PNG, or PDF.

We will see how to install and use PandasGUI, and demonstrate some of its features and functionalities. We will also see some code examples to illustrate how you can integrate PandasGUI with your existing Pandas workflow. By the end of this article, you will be able to use PandasGUI effectively to enhance your data analysis and visualization skills.

Installing PandasGUI

To install PandasGUI, you need to have Python 3.6 or higher and Pandas 1.0 or higher installed on your system. You can use pip or conda to install PandasGUI from PyPI or Anaconda Cloud, respectively. For example, to install PandasGUI using pip, you can run the following command in your terminal:

```
pip install pandasgui
```

Alternatively, you can install PandasGUI from the source code on GitHub, if you want to get the latest version or contribute to the development. To do that, you can clone the repository and run the setup script, as follows:

```
git clone https://github.com/adamerose/pandasgui.git
cd pandasgui
python setup.py install
```

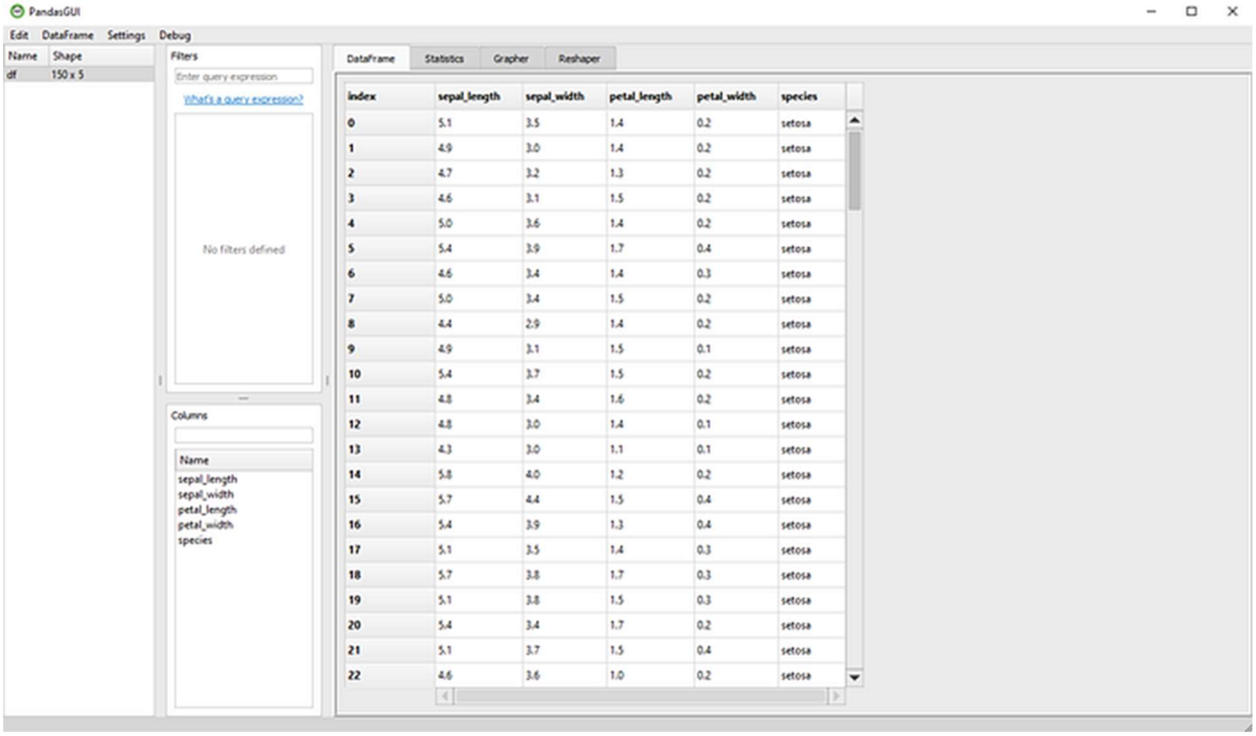
Using PandasGUI

To use PandasGUI, you need to import it in your Python script or notebook, along with Pandas. Then, you can use the `show` function to launch the GUI with one or more DataFrames as arguments. For example, to open the GUI with the built-in Iris dataset, you can run the following code:

```
import pandas as pd
import pandasgui

df = pd.read_csv('https://raw.githubusercontent.com/mwaskom/seaborn-
data/master/iris.csv')
pandasgui.show(df)
```

This will open a new window with the PandasGUI interface, as shown below:



index	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa
10	5.4	3.7	1.5	0.2	setosa
11	4.8	3.4	1.6	0.2	setosa
12	4.8	3.0	1.4	0.1	setosa
13	4.3	3.0	1.1	0.1	setosa
14	5.8	4.0	1.2	0.2	setosa
15	5.7	4.4	1.5	0.4	setosa
16	5.4	3.9	1.3	0.4	setosa
17	5.1	3.5	1.4	0.3	setosa
18	5.7	3.8	1.7	0.3	setosa
19	5.1	3.8	1.5	0.3	setosa
20	5.4	3.4	1.7	0.2	setosa
21	5.1	3.7	1.5	0.4	setosa
22	4.6	3.6	1.0	0.2	setosa

PandasGUI Interface

As you can see, the interface consists of four main tabs: DataFrames, Statistics, Filters, and Plots. Let’s explore each of them in more detail.

DataFrames

The DataFrames tab shows the list of DataFrames that you have passed to the `show` function. You can select any DataFrame from the list to view its contents in a table. You can also perform various operations on the table, such as:

- Resize, reorder, or hide columns

- Sort rows by one or more columns
- Edit cell values
- Copy, paste, or delete rows or columns
- Search for text or values
- Export the table to CSV, Excel, HTML, or clipboard

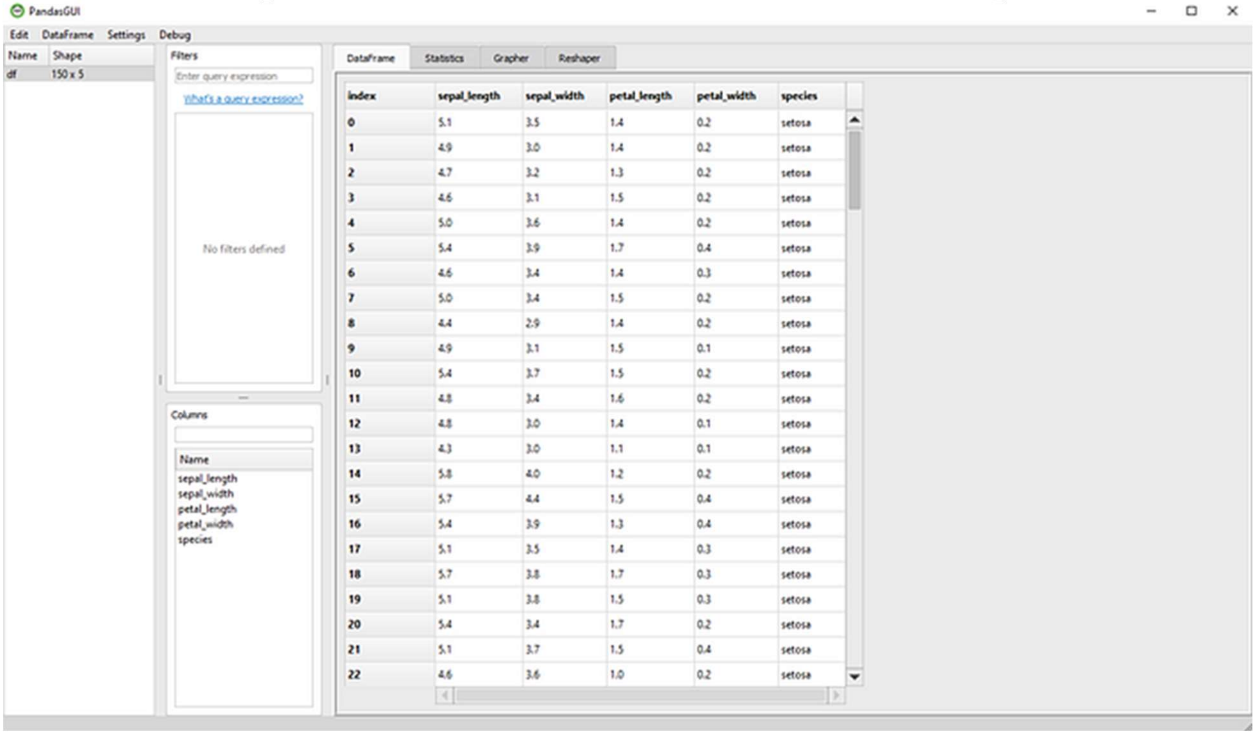
You can also access the DataFrame attributes and methods from the toolbar, such as:

- Shape: shows the number of rows and columns
- Info: shows the summary of the DataFrame, such as column names, data types, memory usage, etc.
- Describe: shows the descriptive statistics of the DataFrame, such as mean, median, standard deviation, etc.
- Head: shows the first n rows of the DataFrame
- Tail: shows the last n rows of the DataFrame
- Sample: shows a random sample of n rows from the DataFrame
- Dtypes: shows the data types of the columns
- Columns: shows the column names

- Index: shows the index values

You can also create new DataFrames from the toolbar, such as:

- Empty: creates an empty DataFrame
- Random: creates a DataFrame with random values
- Example: creates a DataFrame with some example data



Name	Shape
df	150 x 5

Name
sepal_length
sepal_width
petal_length
petal_width
species

index	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa
10	5.4	3.7	1.5	0.2	setosa
11	4.8	3.4	1.6	0.2	setosa
12	4.8	3.0	1.4	0.1	setosa
13	4.3	3.0	1.1	0.1	setosa
14	5.8	4.0	1.2	0.2	setosa
15	5.7	4.4	1.5	0.4	setosa
16	5.4	3.9	1.3	0.4	setosa
17	5.1	3.5	1.4	0.3	setosa
18	5.7	3.8	1.7	0.3	setosa
19	5.1	3.8	1.5	0.3	setosa
20	5.4	3.4	1.7	0.2	setosa
21	5.1	3.7	1.5	0.4	setosa
22	4.6	3.6	1.0	0.2	setosa

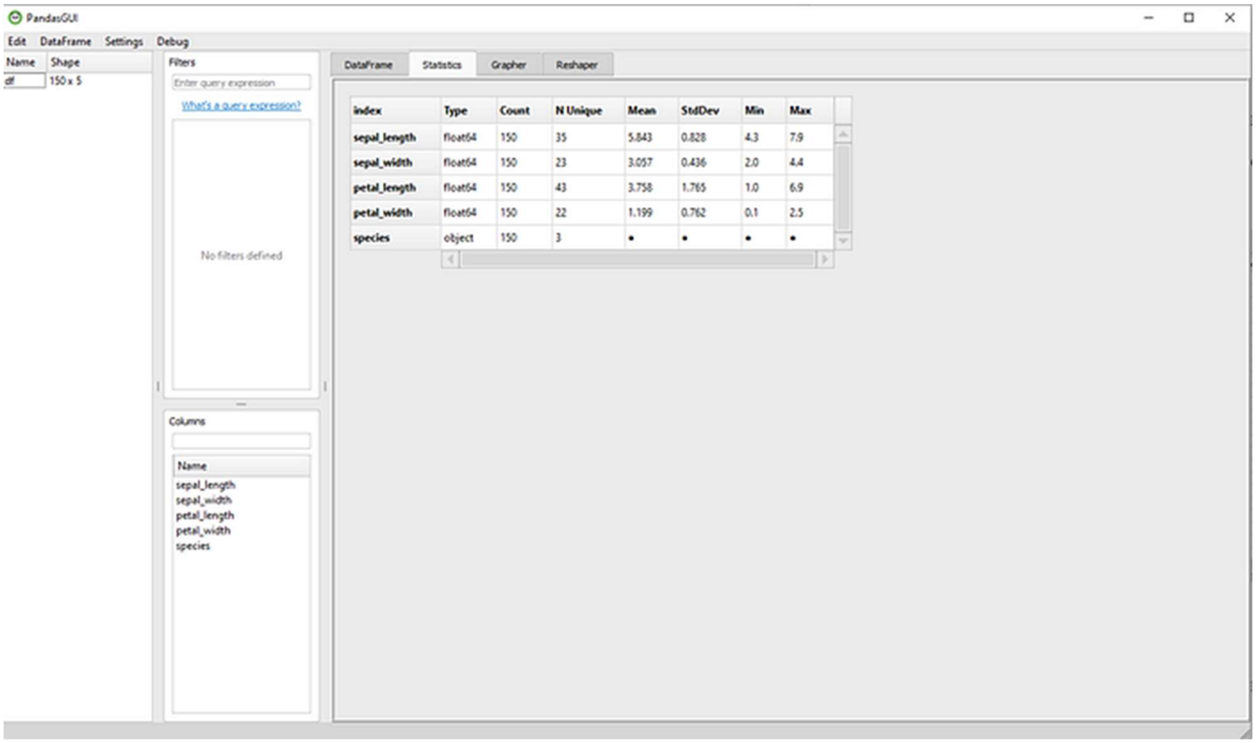
Dataframes

Statistics

The Statistics tab shows the summary statistics of the selected DataFrame, such as:

- Count: the number of non-missing values
- Unique: the number of unique values
- Top: the most frequent value
- Freq: the frequency of the most frequent value
- Mean: the arithmetic mean
- Std: the standard deviation
- Min: the minimum value
- 25%: the 25th percentile
- 50%: the 50th percentile (median)
- 75%: the 75th percentile
- Max: the maximum value

You can also view the distribution of the values in each column using histograms, box plots, or density plots. You can customize the plots by changing the bin size, outliers, bandwidth, etc. You can also export the plots to PNG, PDF, or clipboard.



Name	Shape
df	150 x 5

index	Type	Count	N Unique	Mean	StdDev	Min	Max
sepal_length	float64	150	35	5.843	0.828	4.3	7.9
sepal_width	float64	150	23	3.057	0.436	2.0	4.4
petal_length	float64	150	43	3.758	1.765	1.0	6.9
petal_width	float64	150	22	1.199	0.762	0.1	2.5
species	object	150	3	•	•	•	•

Name
sepal_length
sepal_width
petal_length
petal_width
species

Statistics

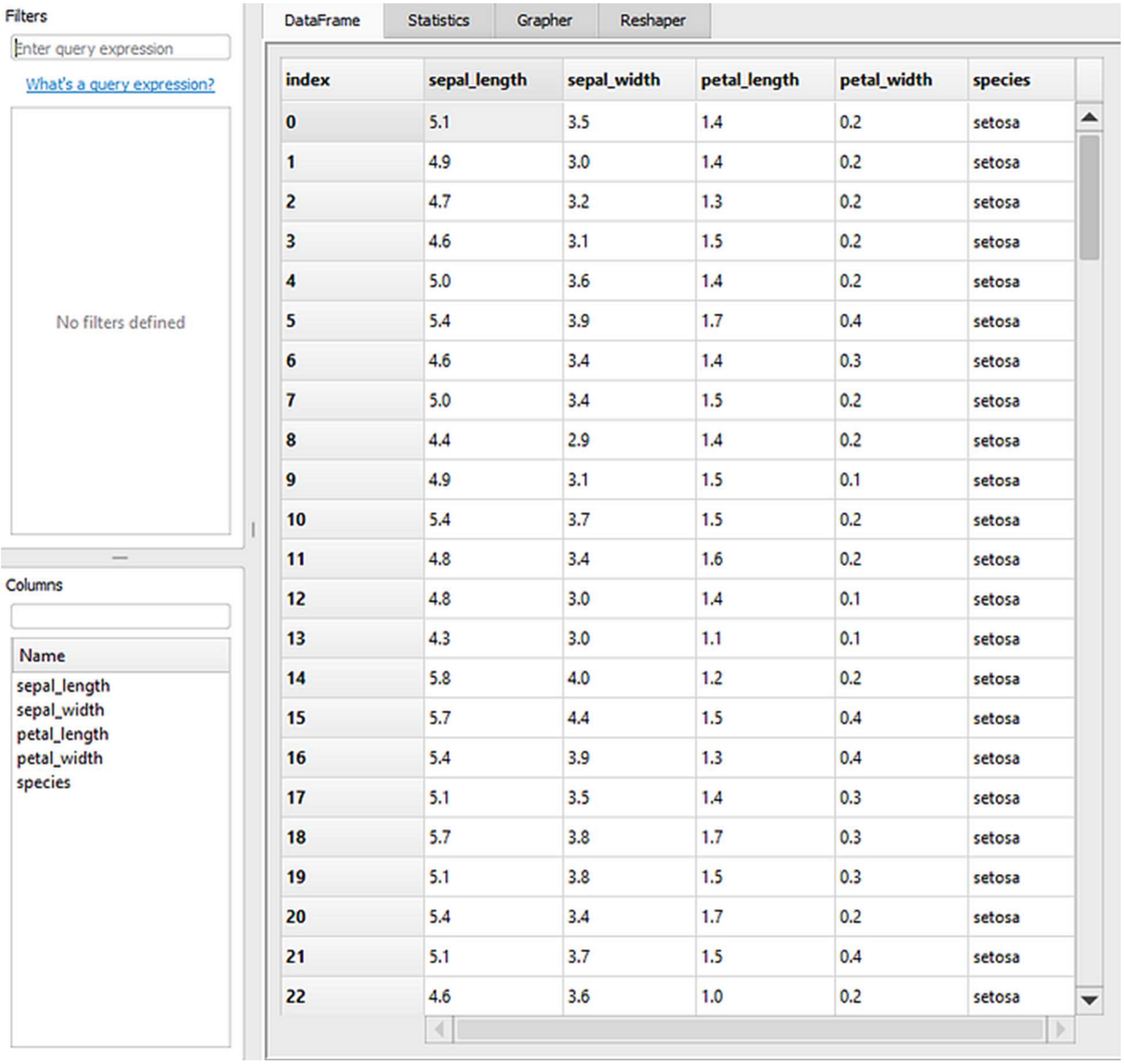
Filters

The Filters tab allows you to filter the rows of the selected DataFrame based on one or more conditions. You can create filters using the following operators:

- ==: equal to
- !=: not equal to
- <: less than
- <=: less than or equal to
- >: greater than

- >=: greater than or equal to
- in: in a list of values
- not in: not in a list of values
- contains: contains a substring
- not contains: does not contain a substring
- startswith: starts with a prefix
- endswith: ends with a suffix
- isna: is missing
- notna: is not missing

You can also combine multiple filters using the logical operators and, or, and not. You can also save, load, or delete filters from the toolbar.



index	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa
10	5.4	3.7	1.5	0.2	setosa
11	4.8	3.4	1.6	0.2	setosa
12	4.8	3.0	1.4	0.1	setosa
13	4.3	3.0	1.1	0.1	setosa
14	5.8	4.0	1.2	0.2	setosa
15	5.7	4.4	1.5	0.4	setosa
16	5.4	3.9	1.3	0.4	setosa
17	5.1	3.5	1.4	0.3	setosa
18	5.7	3.8	1.7	0.3	setosa
19	5.1	3.8	1.5	0.3	setosa
20	5.4	3.4	1.7	0.2	setosa
21	5.1	3.7	1.5	0.4	setosa
22	4.6	3.6	1.0	0.2	setosa

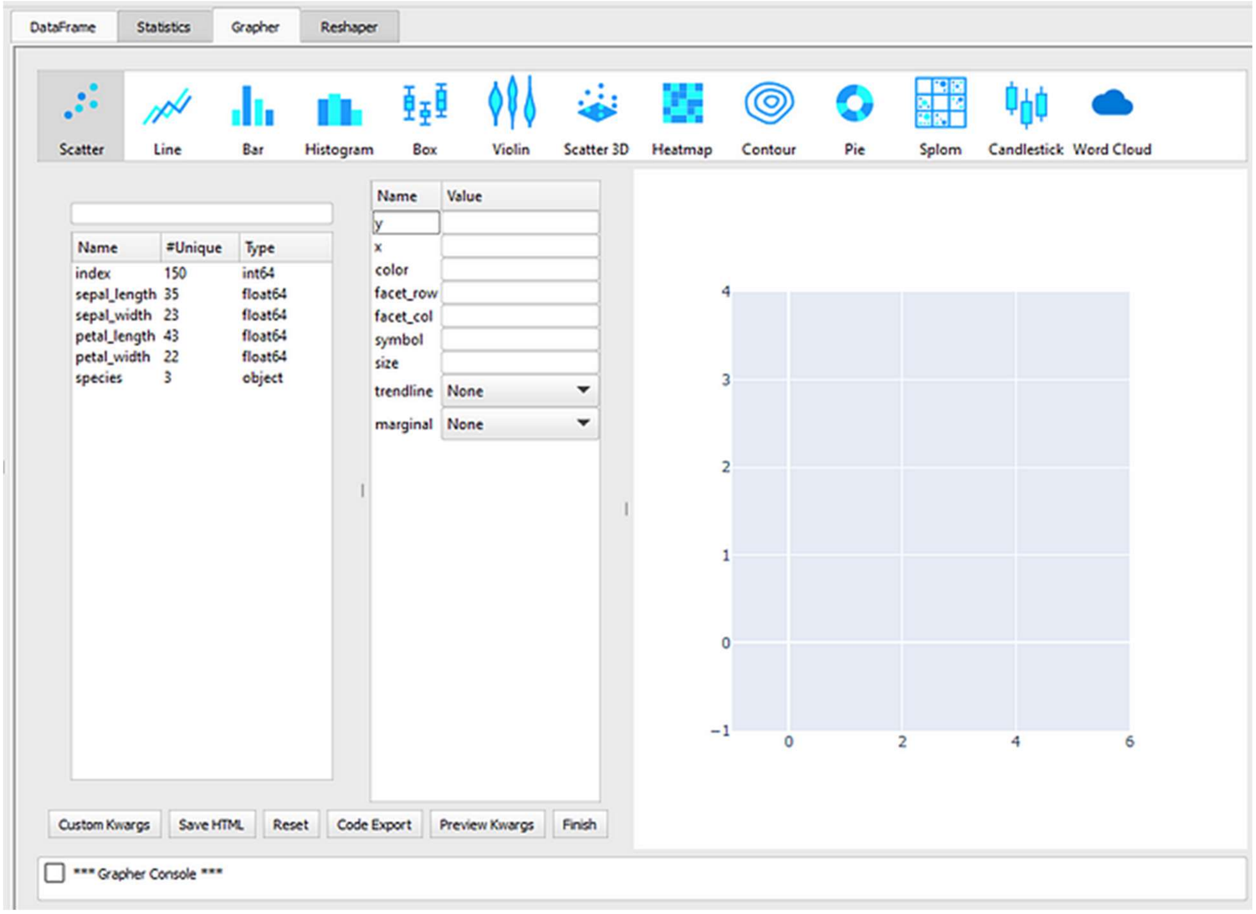
Filters

Plots

The Plots tab allows you to create various types of plots using the selected DataFrame. You can choose from the following plot types:

- Scatter: shows the relationship between two numerical variables
- Line: shows the trend of a numerical variable over time or another numerical variable
- Bar: shows the frequency or amount of a categorical variable
- Pie: shows the proportion of a categorical variable
- Histogram: shows the distribution of a numerical variable
- Box: shows the summary statistics of a numerical variable
- Density: shows the probability density of a numerical variable
- Hexbin: shows the density of points in a two-dimensional space
- Heatmap: shows the correlation or intensity of two numerical variables
- Surface: shows the relationship between three numerical variables

You can customize the plots by changing the x-axis, y-axis, color, size, shape, legend, title, labels, etc. You can also export the plots to PNG, PDF, or clipboard.



Name	#Unique	Type
index	150	int64
sepal_length	35	float64
sepal_width	23	float64
petal_length	43	float64
petal_width	22	float64
species	3	object

Name	Value
y	
x	
color	
facet_row	
facet_col	
symbol	
size	
trendline	None
marginal	None

Plots

Integrating PandasGUI with Pandas

One of the great features of PandasGUI is that it can be integrated with your existing Pandas workflow. You can use the `gui` attribute of any DataFrame to access the PandasGUI interface. For example, if you have a DataFrame called `df`, you can simply type `df.gui` in your Python script or notebook to launch the GUI. You can also use the `gui` attribute to access the DataFrames, Statistics, Filters, and Plots objects of PandasGUI, and use their methods and properties. For example, you can use `df.gui.filters` to get the list of filters applied to the DataFrame, or use `df.gui.plots.scatter` to create a scatter plot.

Key Takeaways and Insights

Here are some key takeaways and insights that you can get from this article:

- PandasGUI is a GUI for viewing, plotting and analyzing Pandas DataFrames — Python module.
- PandasGUI allows you to interact with your data in a visual and intuitive way, without writing any code.
- PandasGUI has four main tabs: DataFrames, Statistics, Filters, and Plots, each with its own features and functionalities.
- PandasGUI can be integrated with your existing Pandas workflow using the `gui` attribute of any DataFrame.
- PandasGUI can be used for various purposes and applications, such as data analysis, visualization, machine learning, data science, data journalism, data education, data exploration, and data fun.

Comparing PandasGUI with Other GUI Tools for Pandas

PandasGUI is not the only GUI tool available for Pandas. There are other alternatives that you can use to interact with your data in a graphical way. Some of the most popular ones are:

- QGrid: A Jupyter notebook widget for interactive grid-based editing of DataFrames.
- D-Tale: A web-based application for viewing, manipulating, and analyzing DataFrames.
- PivotTable.js: A Jupyter notebook extension for creating pivot tables and charts from DataFrames.
- Lux: A Python library that automatically generates visualizations from DataFrames based on user intent.
- HiPlot: A lightweight interactive visualization tool for exploring high-dimensional data.

How does PandasGUI compare with these other tools? What are the advantages and disadvantages of using PandasGUI over them? Here are some points to consider:

- PandasGUI is a standalone application that can be launched from any Python script or notebook, while QGrid, PivotTable.js, and Lux are Jupyter notebook widgets that require a running notebook server. This means that PandasGUI can be used independently of Jupyter, and can handle larger and more complex DataFrames without affecting the notebook's performance. However, this also means that PandasGUI is less integrated with the notebook environment, and does not support features such as cell magic, inline plots, or interactive widgets.

- PandasGUI and D-Tale are both web-based applications that use QtWebEngine as the backend, while QGrid, PivotTable.js, and Lux use HTML, CSS, and JavaScript as the frontend. This means that PandasGUI and D-Tale can leverage the power and flexibility of Qt, and offer more features and functionalities than the other tools. However, this also means that PandasGUI and D-Tale are heavier and slower than the other tools, and may have compatibility issues with some browsers or platforms.

- PandasGUI and PivotTable.js are both focused on creating and customizing plots from DataFrames, while QGrid, D-Tale, and Lux are more focused on viewing and manipulating DataFrames. This means that PandasGUI and PivotTable.js can offer more plot types and options than the other tools, and can help you visualize and analyze your data more effectively. However, this also means that PandasGUI and PivotTable.js are less versatile and comprehensive than the other tools, and may not cover all the aspects and operations of data analysis and manipulation.

- PandasGUI and Lux are both designed to be intuitive and user-friendly, while QGrid, D-Tale, and PivotTable.js are more technical and advanced. This means that PandasGUI and Lux can be easier to use and learn than the other tools, and they can help you explore and understand your data without writing any code. However, this also means that PandasGUI and Lux are less flexible and customizable than the other tools, and may not suit all the needs and preferences of different users.

Conclusion

PandasGUI, a GUI for viewing, plotting, and analyzing Pandas DataFrames — Python module. We have seen how to install and use PandasGUI, and demonstrated some of its features and functionalities. We have also seen some code examples to illustrate how you can integrate PandasGUI with your existing Pandas workflow. PandasGUI is a very handy and powerful tool that can help you enhance your data analysis and visualization skills and experience. Whether you are a beginner or an expert, PandasGUI can offer you a visual and interactive way to explore and understand your data, without writing any code. It can also complement your existing Pandas workflow by integrating with the `gui` attribute of any DataFrame. It can also be used for various purposes and applications, such as data analysis, visualization, machine learning, data science, data journalism, data education, data exploration, and data fun.