

04.15-Further-Resources

November 10, 2024

1 Further Resources

A single part of a book can never hope to cover all the available features and plot types available in Matplotlib. As with other packages we've seen, liberal use of IPython's tab completion and help functions (see [Help and Documentation in IPython](#)) can be very helpful when exploring Matplotlib's API. In addition, Matplotlib's [online documentation](#) can be a helpful reference. See in particular the [Matplotlib gallery](#), which shows thumbnails of hundreds of different plot types, each one linked to a page with the Python code snippet used to generate it. This allows you to visually inspect and learn about a wide range of different plotting styles and visualization techniques.

For a book-length treatment of Matplotlib, I would recommend *Interactive Applications Using Matplotlib* (Packt), written by Matplotlib core developer Ben Root.

1.1 Other Python Visualization Libraries

Although Matplotlib is the most prominent Python visualization library, there are other more modern tools that are worth exploring as well. I'll mention a few of them briefly here:

- [Bokeh](#) is a JavaScript visualization library with a Python frontend that creates highly interactive visualizations capable of handling very large and/or streaming datasets.
- [Plotly](#) is the eponymous open source product of the Plotly company, and is similar in spirit to Bokeh. It is actively developed and provides a wide range of interactive chart types.
- [HoloViews](#) is a more declarative, unified API for generating charts in a variety of backends, including Bokeh and Matplotlib.
- [Vega](#) and [Vega-Lite](#) are declarative graphics representations, and are the product of years of research into how to think about data visualization and interaction. The reference rendering implementation is JavaScript, and the [Altair package](#) provides a Python API to generate these charts.

The visualization landscape in the Python world is constantly evolving, and I expect that this list may be out of date by the time this book is published. Additionally, because Python is used in so many domains, you'll find many other visualization tools built for more specific use cases. It can be hard to keep track of all of them, but a good resource for learning about this wide variety of visualization tools is <https://pyviz.org/>, an open, community-driven site containing tutorials and examples of many different visualization tools.