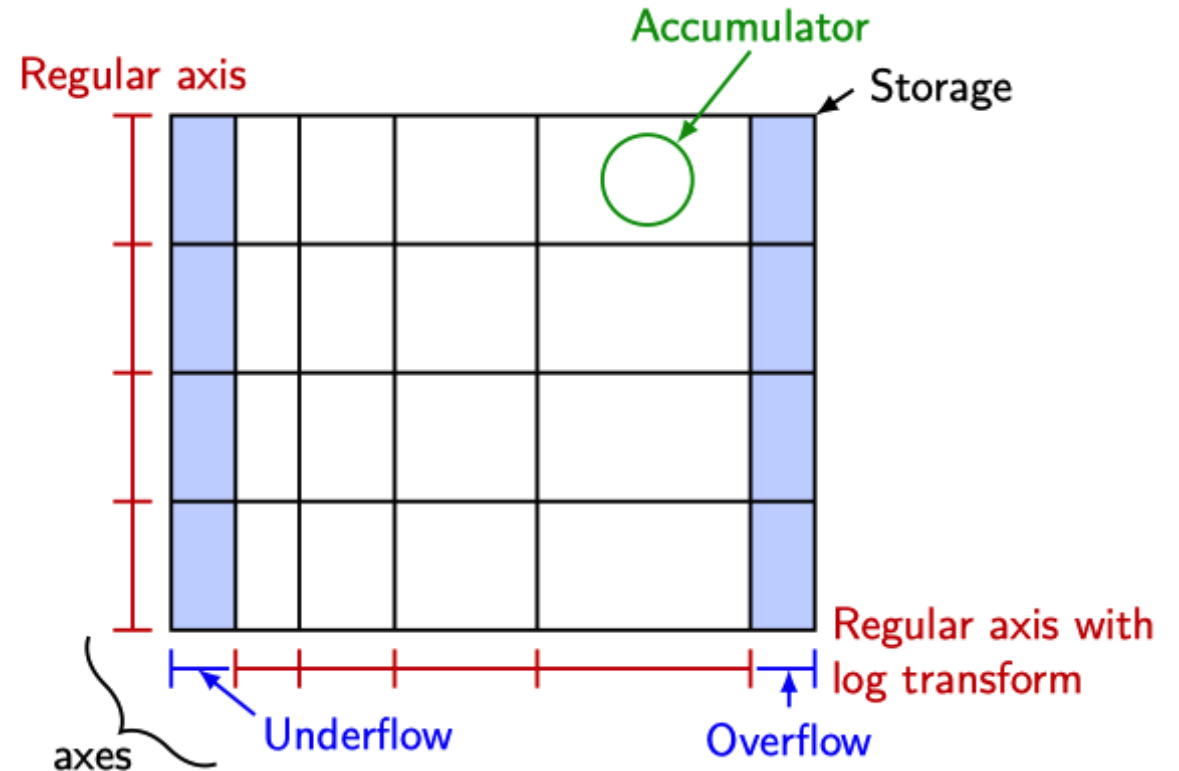


Hist: Histogramming for Analysis powered by Boost-histogram

2020-09-03 by Shuo Liu

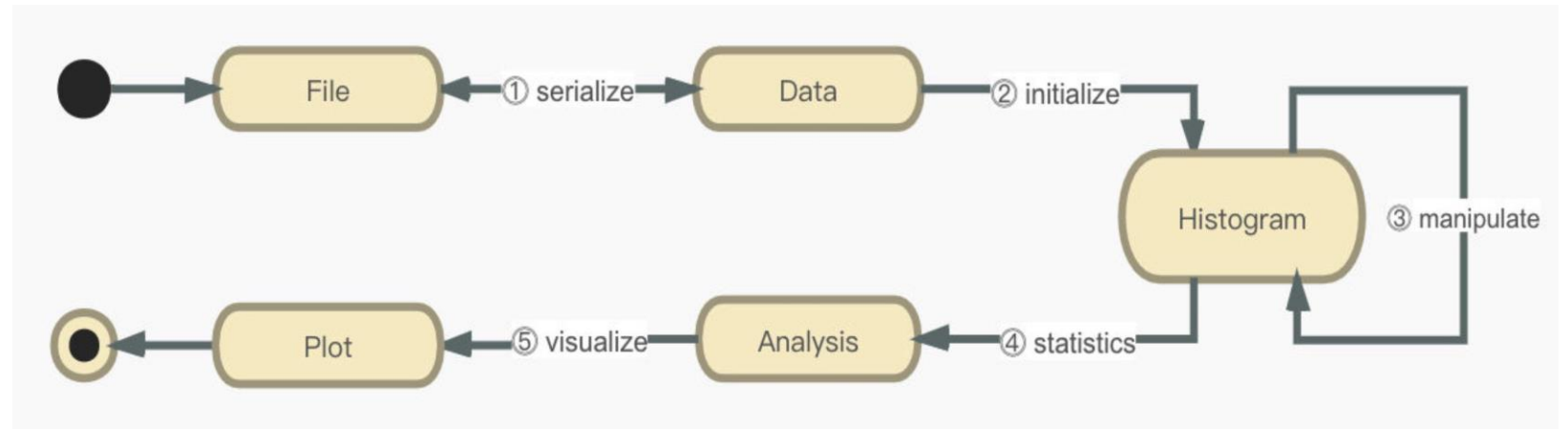
About Boost-histogram

- Creating
- Filling
- Slicing & Rebinning
- Accessing the contents
- Setting the contents
- Accessing Axes
- Saving Histograms
- Computing with Histograms



New Features of Hist

- Serialization & Deserialization
- Initialization
- Manipulation
- Statistics
- Visualization



A Short Demo

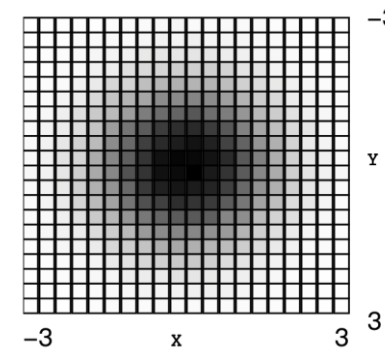
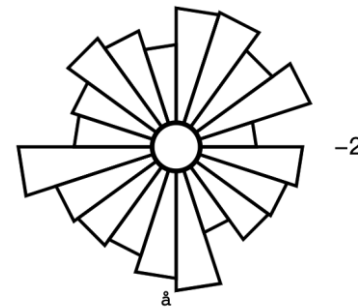
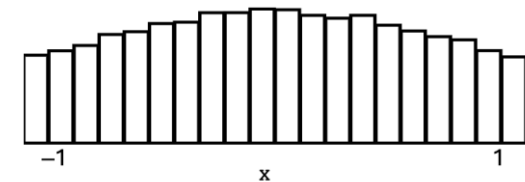
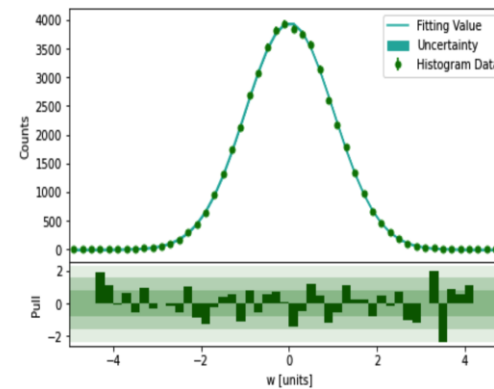
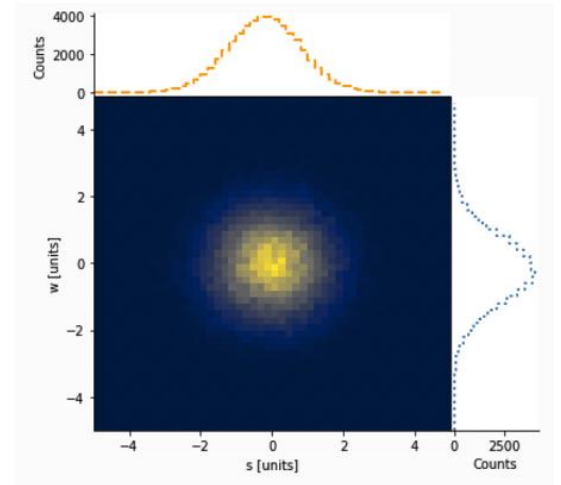
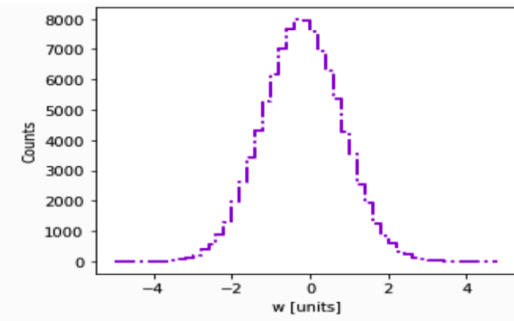


```
# You can create a histogram like this.
h = (
  hist.Hist()
    .Reg(10, 0, 1, name="x", label="x-axis")
    .Variable(range(10), name="y", label="y-axis")
    .Int64()
)

# Filling by names is allowed in hist.
hist.fill(y=[1, 4, 6], x=[3, 5, 2])

# New ways to manipulate the histogram.
h.project("x")
h[{"y": 1j + 3, "x": 5j}]
...

# Elegant plotting functions.
h.plot()
h.plot2d_full()
h.plot_pull(Callable)
...
```



Resources



Hist — [Code](#) | [Docs](#)

Boost-histogram — [Code](#) | [Docs](#) | [Chat](#) | [Stack](#)

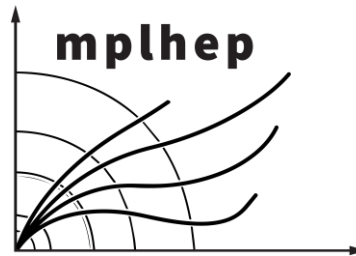
Scikit-HEP — [Org](#) | [Page](#) | [Chat](#)

Uproot-4.0 — [Code](#)

Aghast — [Code](#)

Matplotlib — [Code](#) | [Docs](#) | [Chat](#)

Mplhep — [Code](#)



Thanks !

Welcome to join us !