



mplhep

Histogram Visualization Needs in HEP

Andrzej Novak

PyHEP Workshop 2019
Abingdon, UK

Analysis Stack



----- Formerly self-contained in ROOT -----

uproot
awkward

Pratyush's and
Jim's talks

cofea
parsl spark

Lindsey's and
Ben's talks

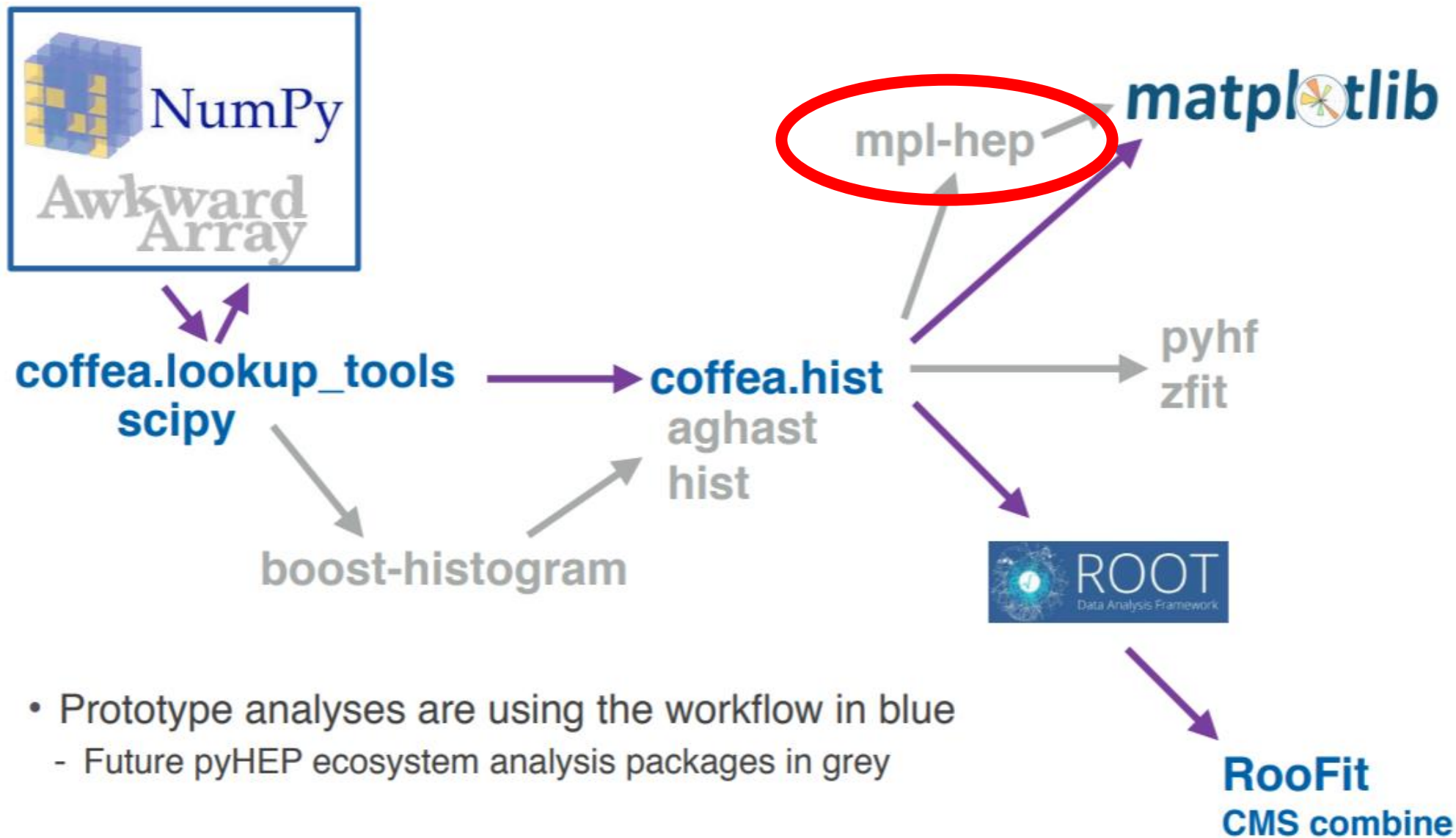
numpy
cofea
boost::histogram

Henry's talk

???

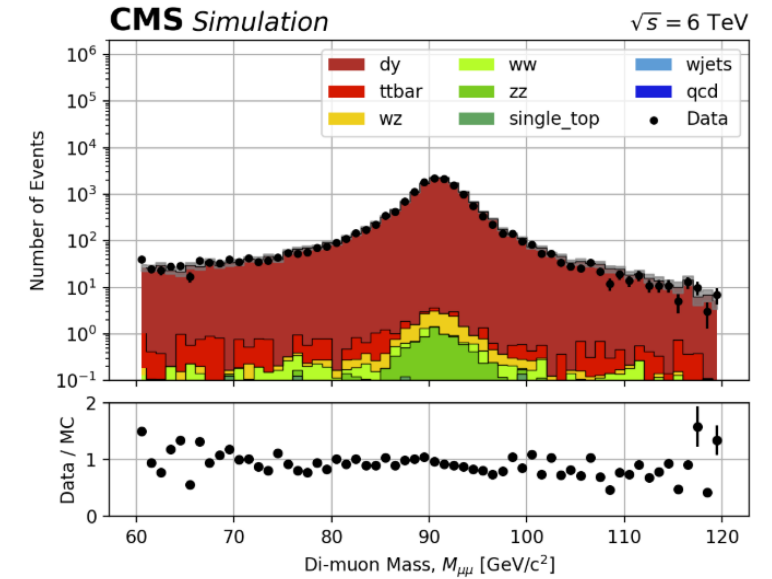
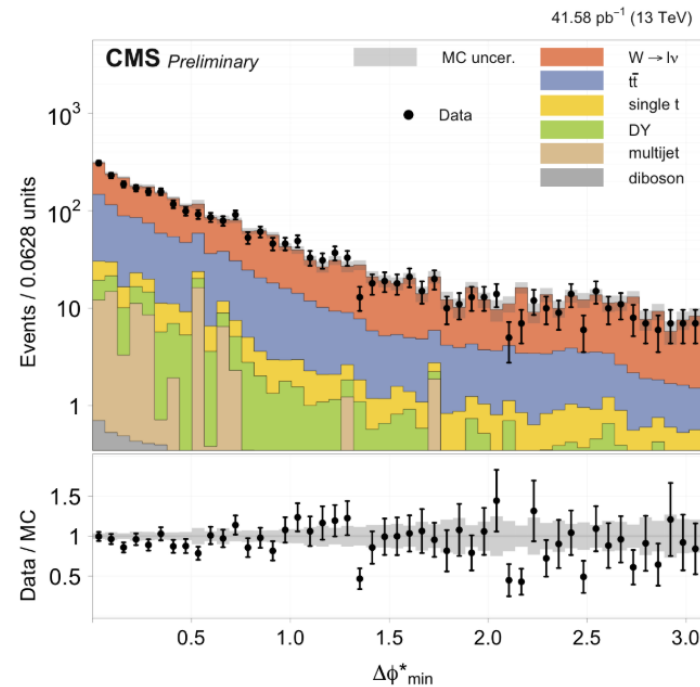
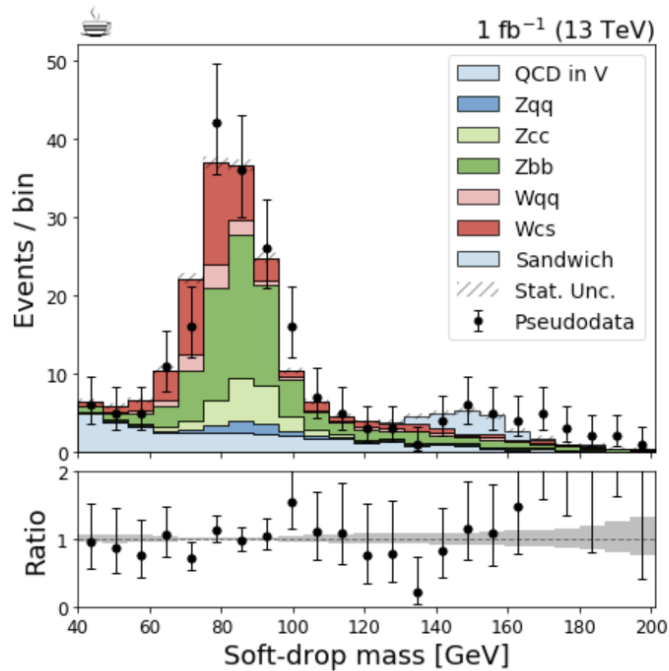
Why am I first again?

Package ecosystem



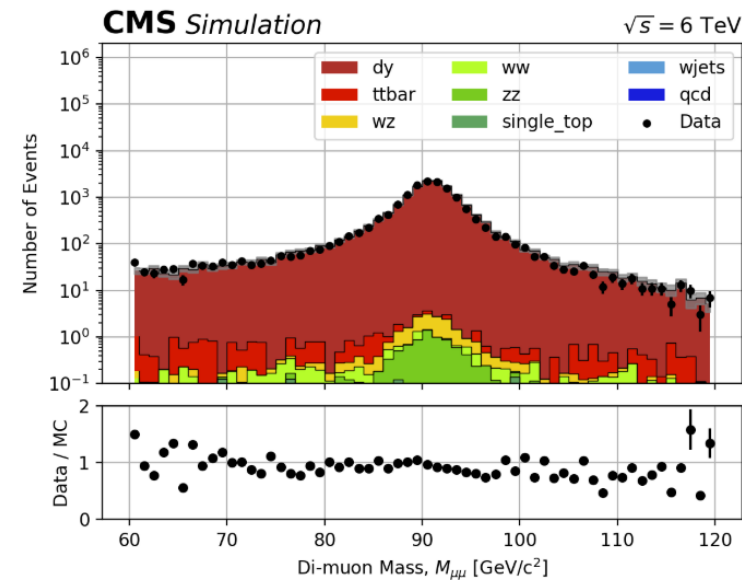
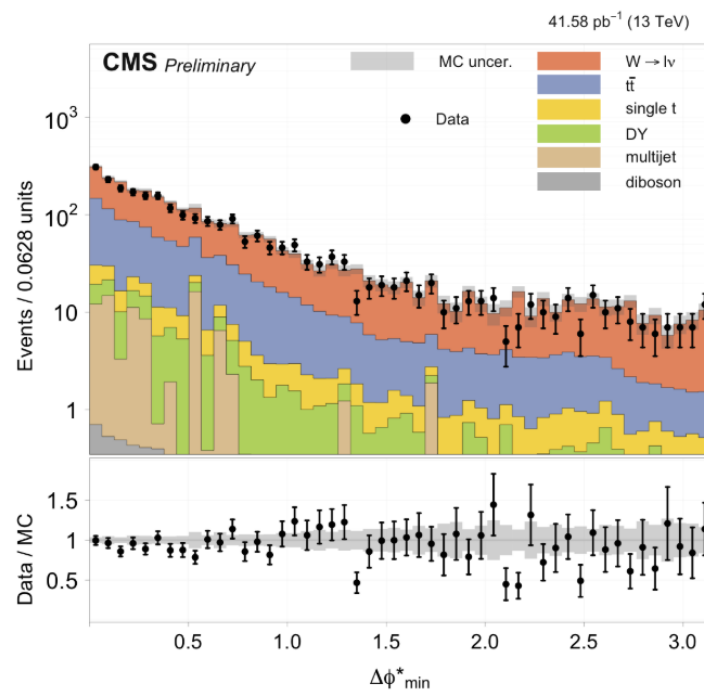
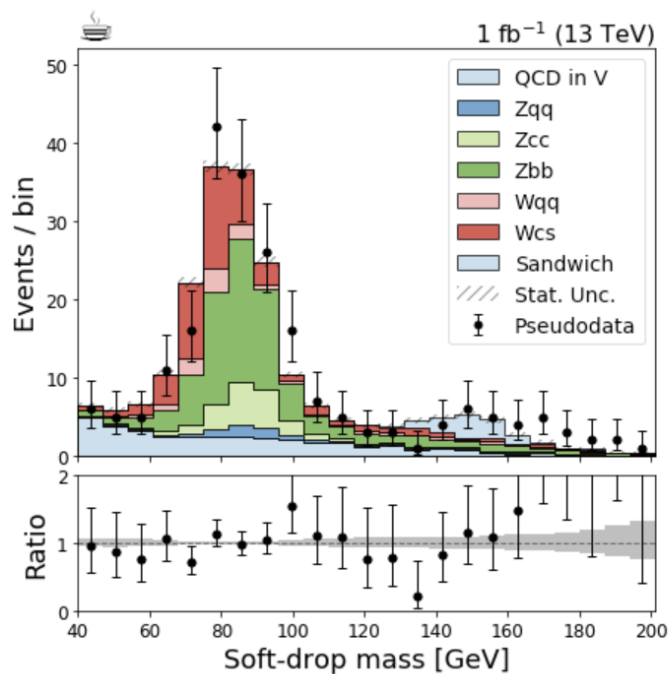
*stolen from Nick Smith

Situation Survey - [gitter](#), [Google Doc](#)

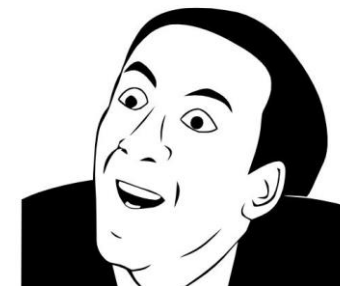


HEP Community is comfortable with matplotlib

Situation Survey - [gitter](#), [Google Doc](#)



HEP Community is comfortable with matplotlib

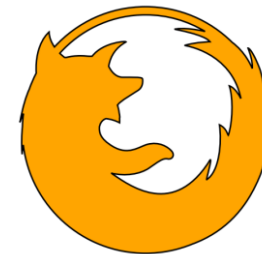


matplotlib



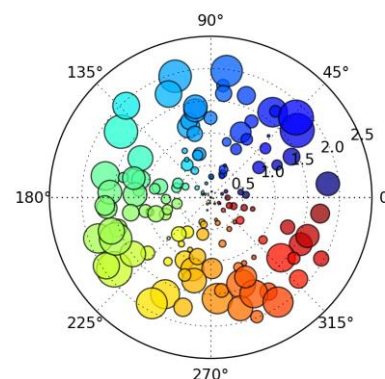
7.6k watchers

41.5k questions

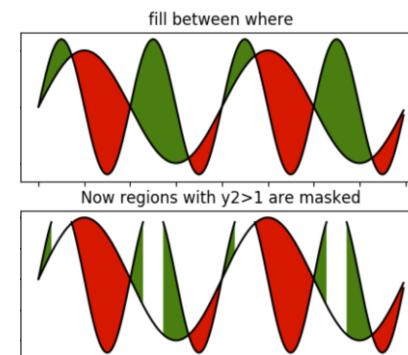
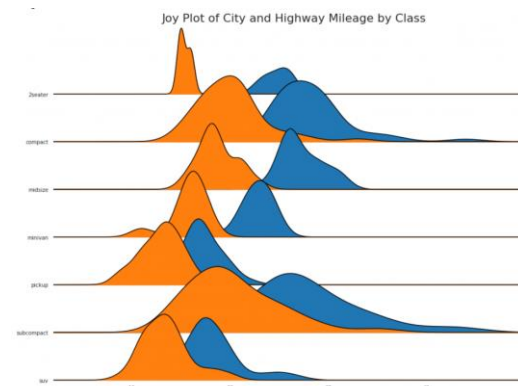
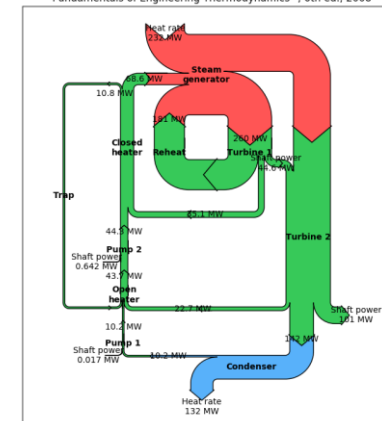


The Good

- Can **plot** just about **anything** + examples
- Many backends
- Almost **any problem** is **just a google search** away
 - **41 500 Q&A** on **StackOverflow**



Rankine Power Cycle: Example 8.6 from Moran and Shapiro
"Fundamentals of Engineering Thermodynamics", 6th ed., 2008



matplotlib



7.6k watchers

41.5k questions

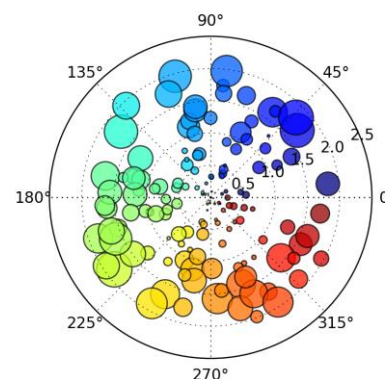


The Good

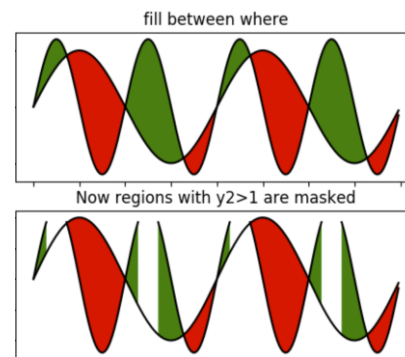
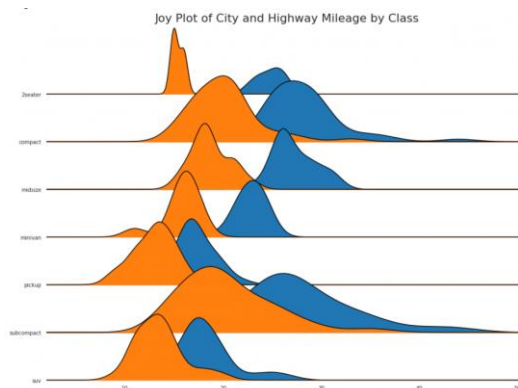
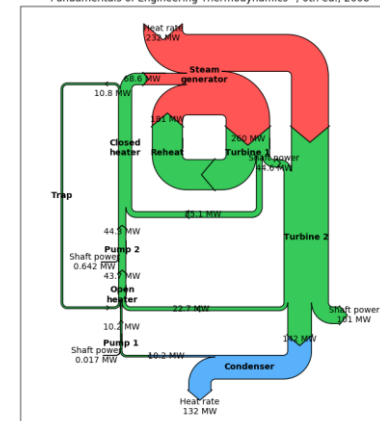
- Can **plot** just about **anything** + examples
- Many backends
- Almost **any problem** is just a google search away
 - 41 500 Q&A on StackOverflow

The Not So Good for HEP

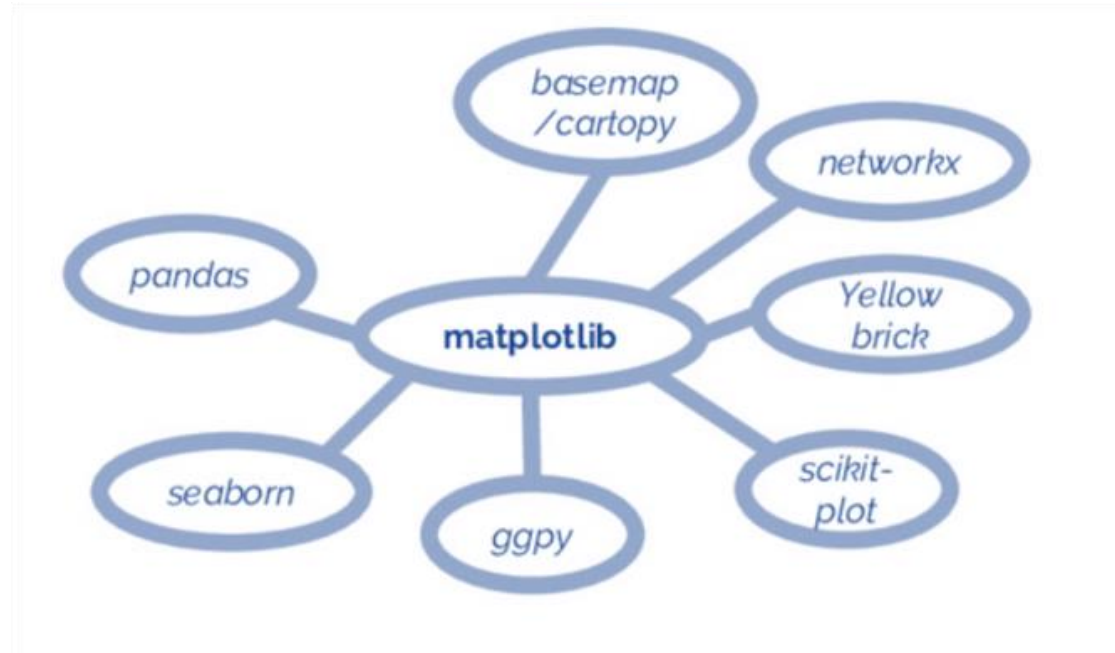
- Styling takes a **lot of tinkering**
 - Defaults very different from ATLAS/CMS style
- Common HEP plots not easily accessible
 - **1D & 2D pre-binned histograms**



Rankine Power Cycle: Example 8.6 from Moran and Shapiro
"Fundamentals of Engineering Thermodynamics", 6th ed., 2008



Building on matplotlib

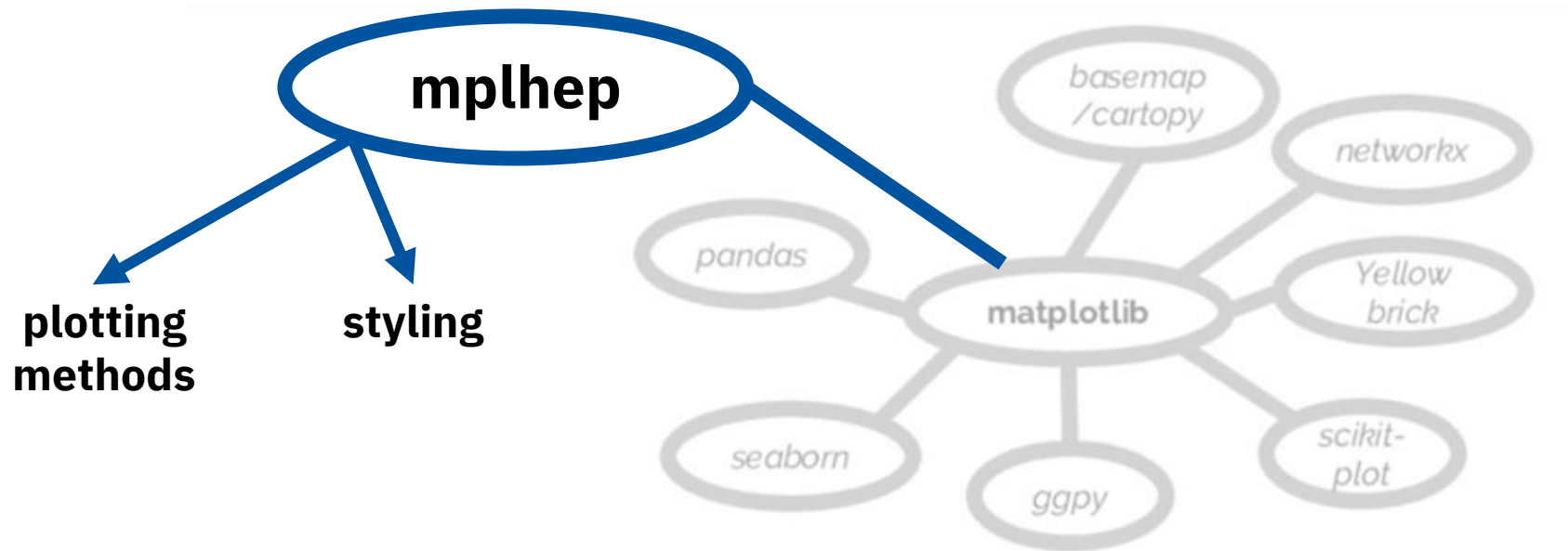


Common Idea:

Keep matplotlib as a **versatile, well-tested** backend, and provide a new domain-specific API

*adapted from Jake VanderPlas

Building on matplotlib



Common Idea:

Keep matplotlib as a **versatile, well-tested** backend, and provide a new domain-specific API

*adapted from Jake VanderPlas

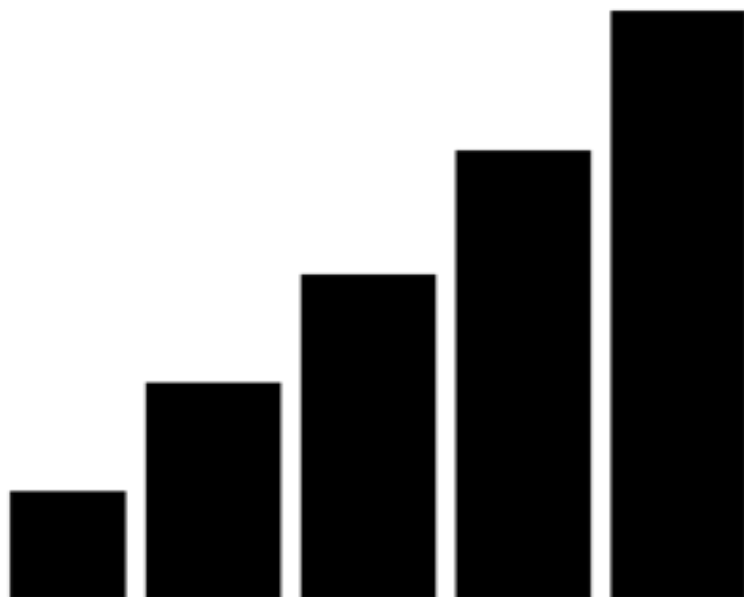
mplhep

Plotting Methods

- API handling
 - Ideally PR changes to matplotlib
- Future
 - Handle plotting for histogramming libraries

Styling

- Easy publication grade styling
 - Distribute styles/fonts
- Save real analyzer time



Plotting 1D Histogram

API for histogram inputs

Unified interface for different styles

Hide data handling

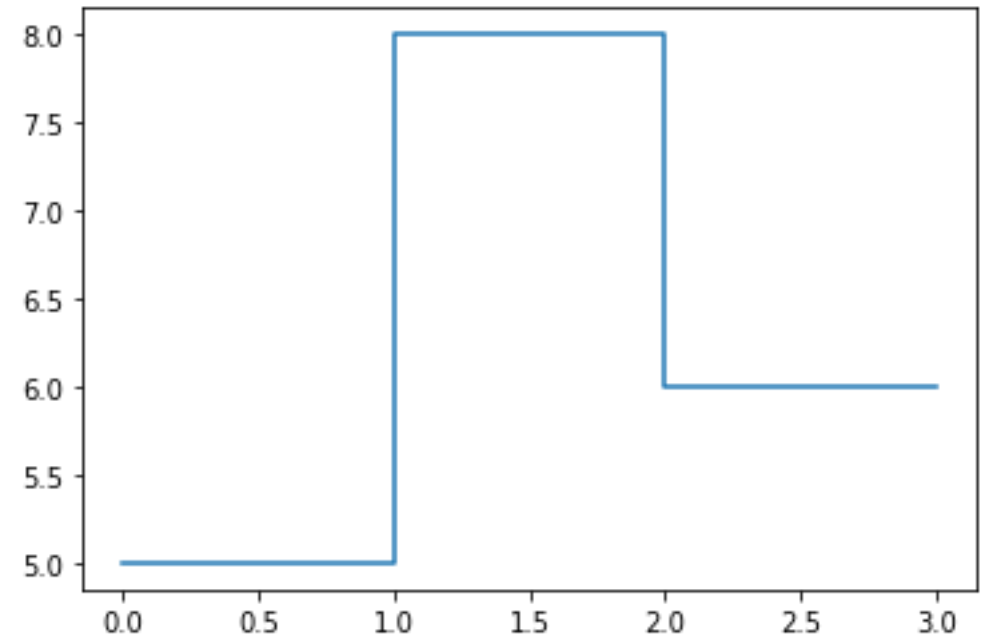
1D Histogram

```
import numpy as np
import matplotlib.pyplot as plt
```

```
bins = [0,1,2,3]
```

```
h = [5,8,6]
```

```
f, ax = plt.subplots()
ax.step(bins, np.r_[h, h[-1]],
       where='post')
```



1D Histogram

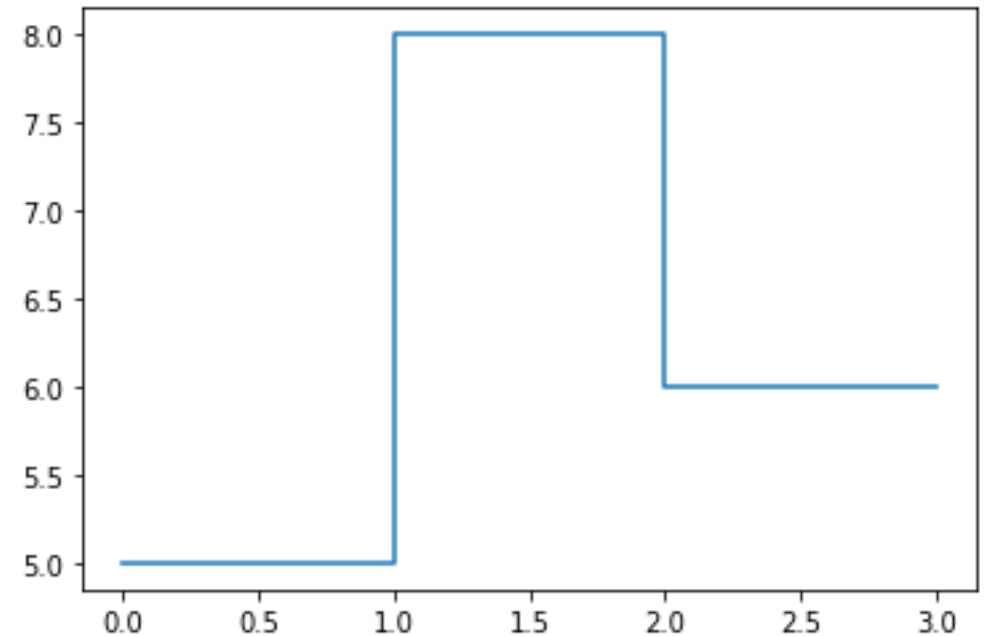
```
import numpy as np
import matplotlib.pyplot as plt
```

```
bins = [0,1,2,3]
```

```
h = [5,8,6]
```

```
f, ax = plt.subplots()
```

```
ax.step(bins, np.r_[h, h[-1]],  
       where='post')
```

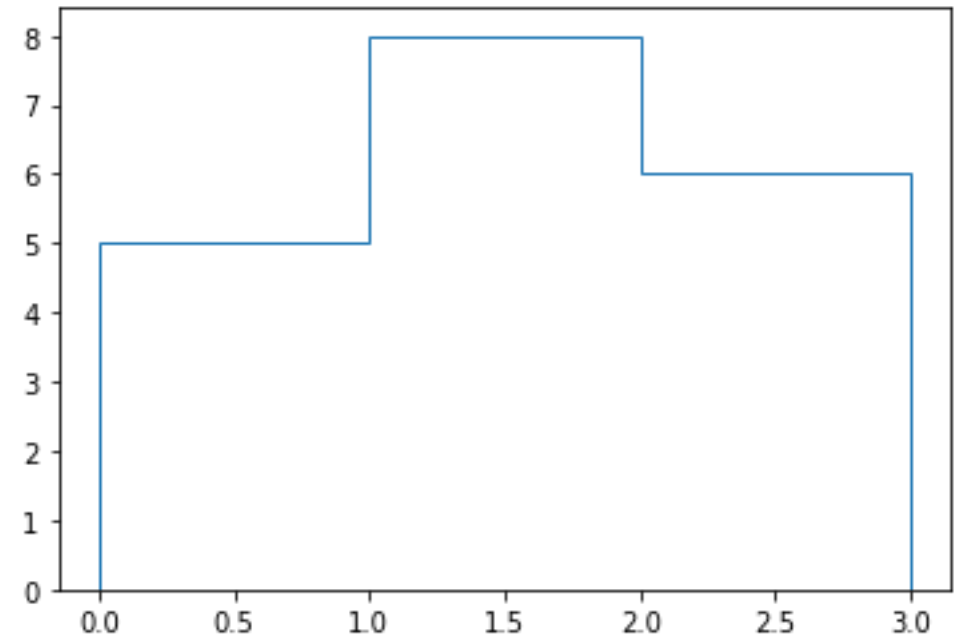


1D Histogram

```
import numpy as np
import matplotlib.pyplot as plt

bins = [0,1,2,3]
h = [5,8,6]

f, ax = plt.subplots()
centers = bins[:-1] + np.diff(bins)/2
ax.hist(centers, bins=bins,
        weights=h, histtype='step')
```

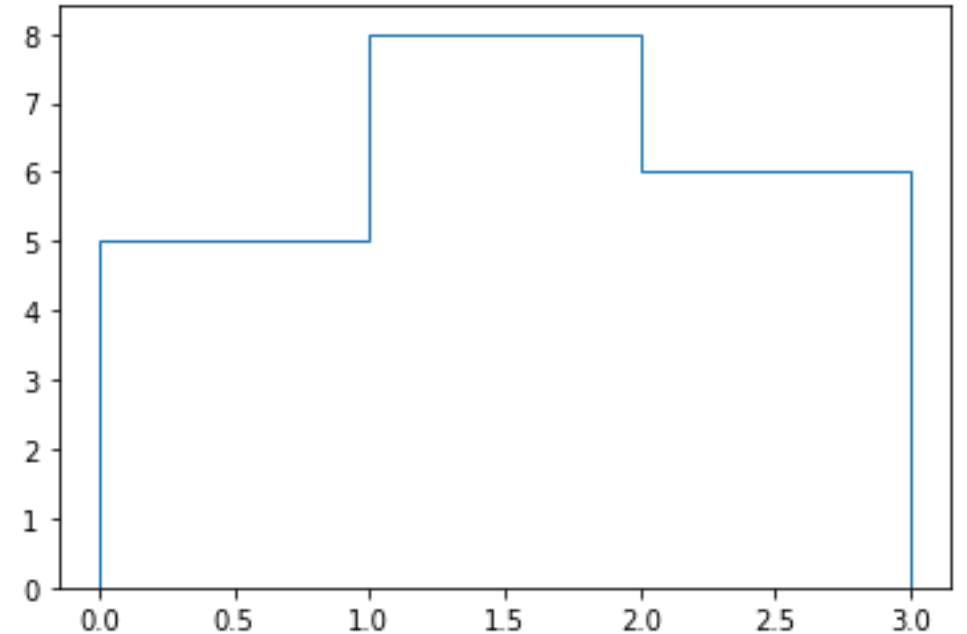


1D Histogram

```
import numpy as np
import matplotlib.pyplot as plt

bins = [0,1,2,3]
h = [5,8,6]

f, ax = plt.subplots()
centers = bins[:-1] + np.diff(bins)/2
ax.hist(centers, bins=bins,
        weights=h, histtype='step')
```



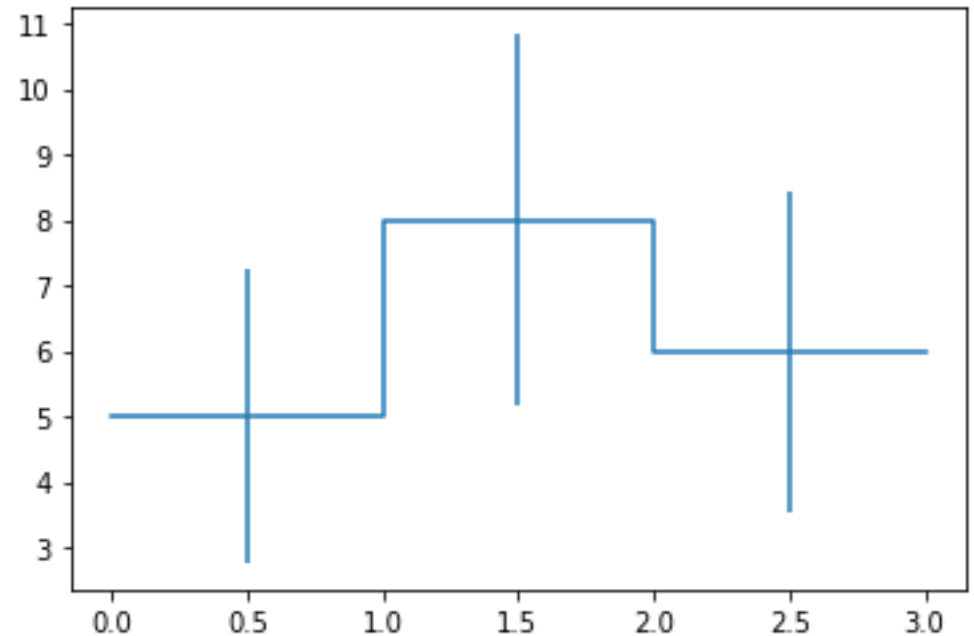
1D Histogram – Errorbars

```
import numpy as np
import matplotlib.pyplot as plt

bins = [0,1,2,3]
h = [5,8,6]

f, ax = plt.subplots()
s, = ax.step(bins, np.r_[h, h[-1]],
             where='post')

errors = np.sqrt(h)
centers = bins[:-1] + np.diff(bins)/2
ax.errorbar(centers, h, yerr=errors,
           fmt='none', c=s.get_color())
```



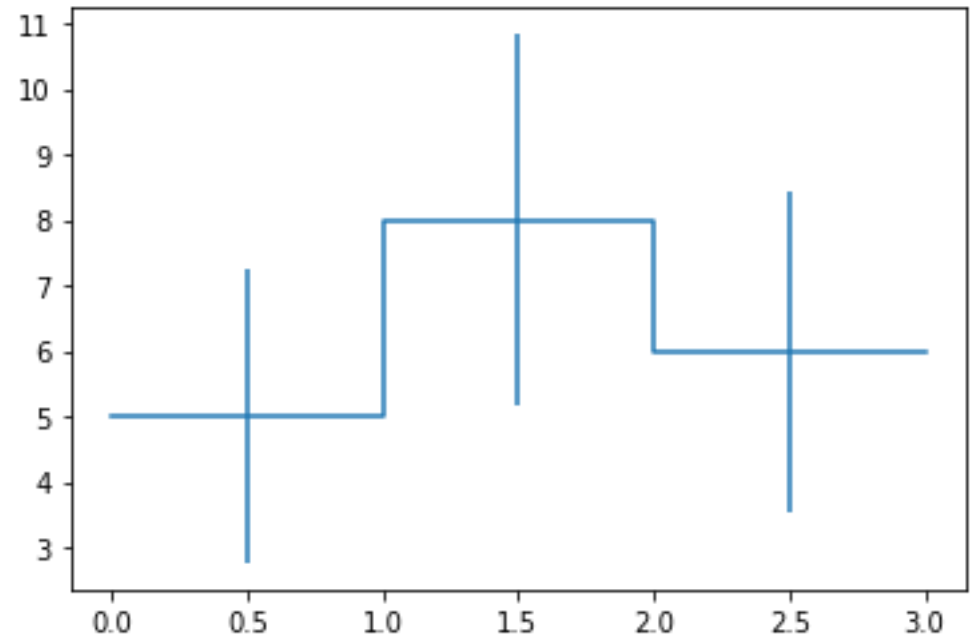
1D Histogram – Errorbars

```
import numpy as np
import matplotlib.pyplot as plt

bins = [0,1,2,3]
h = [5,8,6]

f, ax = plt.subplots()
s, = ax.step(bins, np.r_[h, h[-1]],
             where='post')

errors = np.sqrt(h)
centers = bins[:-1] + np.diff(bins)/2
ax.errorbar(centers, h, yerr=errors,
           fmt='none', c=s.get_color())
```



1D Histogram - Add Legend

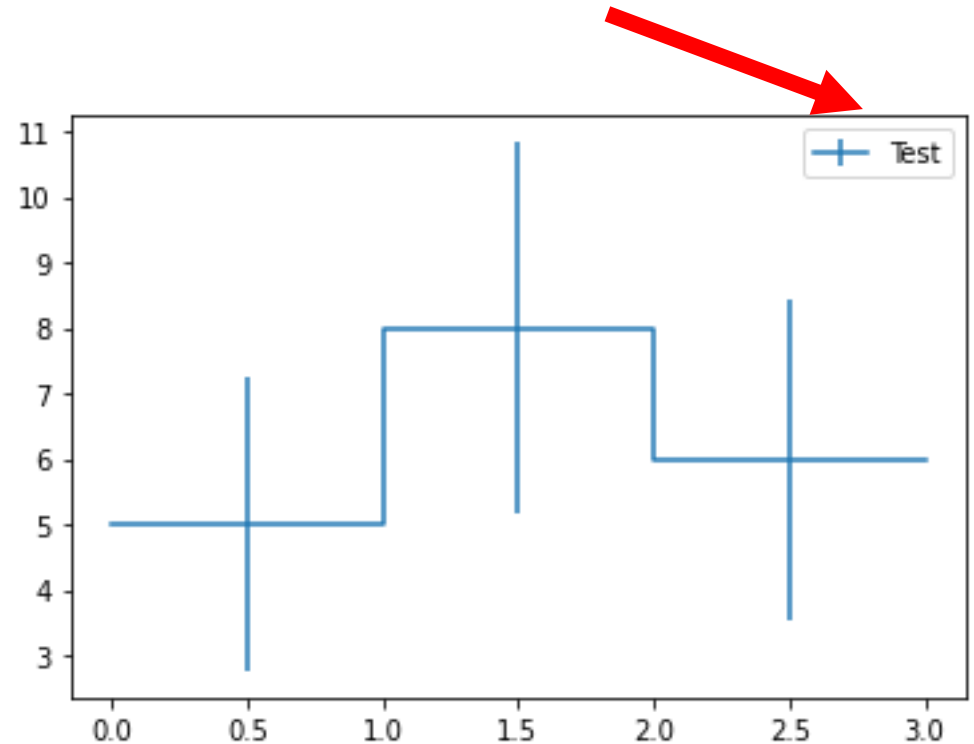
```
import numpy as np
import matplotlib.pyplot as plt

bins = [0,1,2,3]
h = [5,8,6]

f, ax = plt.subplots()
s, = ax.step(bins, np.r_[h, h[-1]],
             where='post')

errors = np.sqrt(h)
centers = bins[:-1] + np.diff(bins)/2
e = ax.errorbar(centers, h, yerr=errors,
               fmt='none', c=s.get_color())

ax.legend([(s, e)], ['Test'])
```



1D Histogram - Solution

Just data manipulation problems

- Create a **seaborn** style wrapper
- Externalize and **hide all superfluous steps**
- Keep **API** as close as possible to `plt.hist()`

Other functionality easy to bake in

- Stacked histograms
- Filled/step type, closed/open steps
- Labels at bin edges

1D Histogram - Solution

```
import numpy as np
import matplotlib.pyplot as plt
import mplhep as hep
```

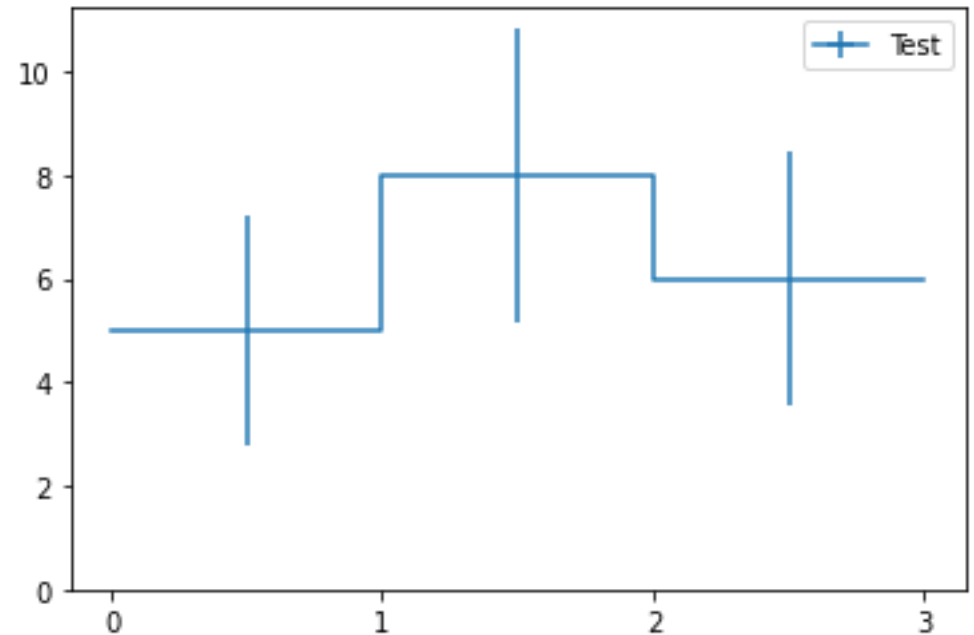
```
bins = [0,1,2,3]
```

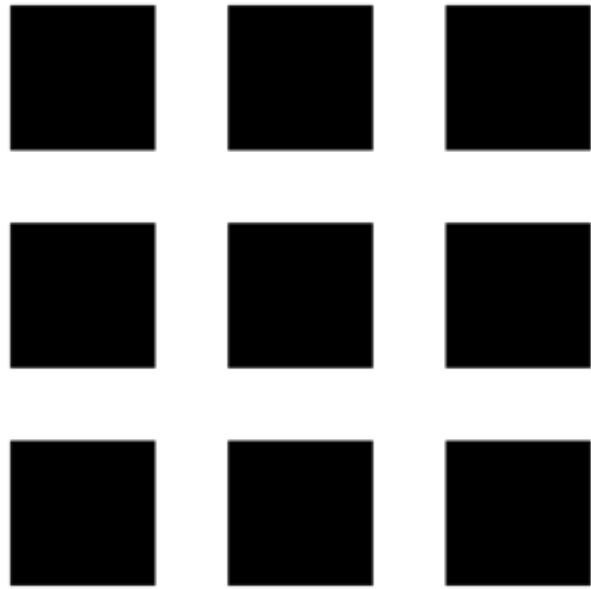
```
h = [5,8,6]
```

```
f, ax = plt.subplots()
```

```
hep.histplot(h, bins, yerr=True,
              label="Test")
```

```
ax.legend()
```





Plotting 2D Histogram

Cleaner API

2D Histogram

```
import numpy as np
import matplotlib.pyplot as plt
import mplhep as hep
```

```
xbins = [0,1,2,3]
```

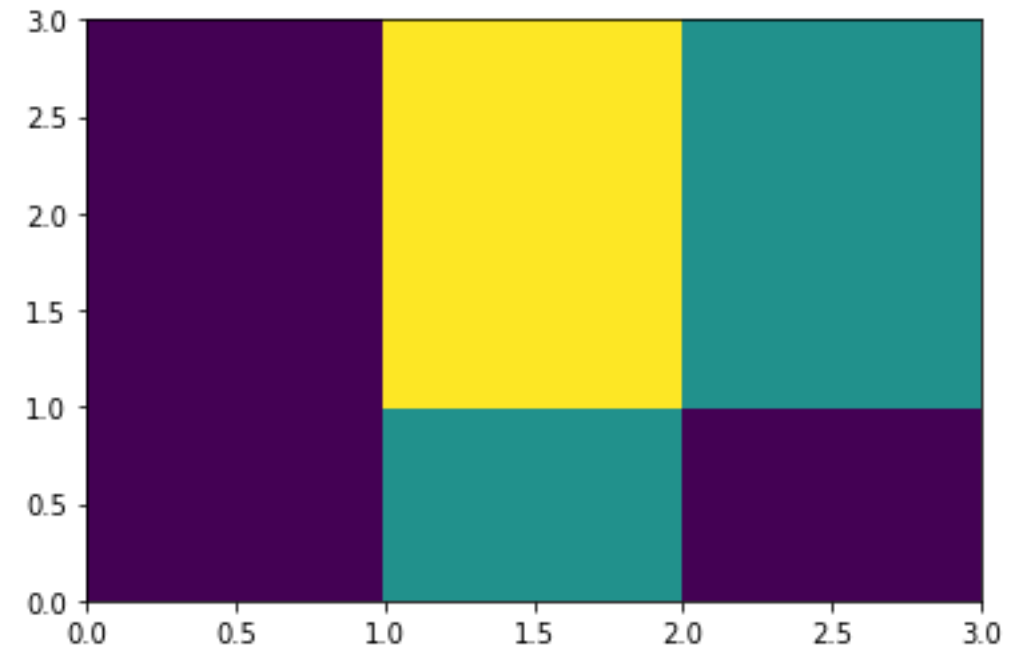
```
ybins = [0,1,3]
```

```
H = [[1,2,1],[1,3,2]]
```

```
fig, ax = plt.subplots()
```

```
X, Y = np.meshgrid(xbins, ybins)
```

```
pc = ax.pcolormesh(X, Y, H)
```



2D Histogram

```
import numpy as np
import matplotlib.pyplot as plt
import mplhep as hep
```

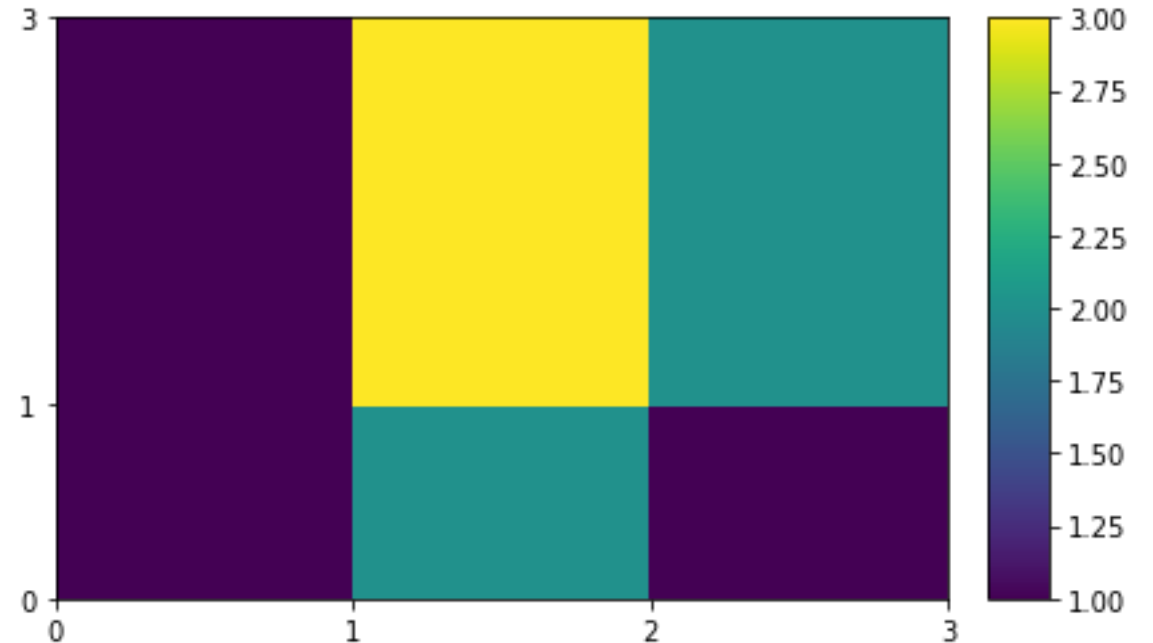
```
xbins = [0,1,2,3]
```

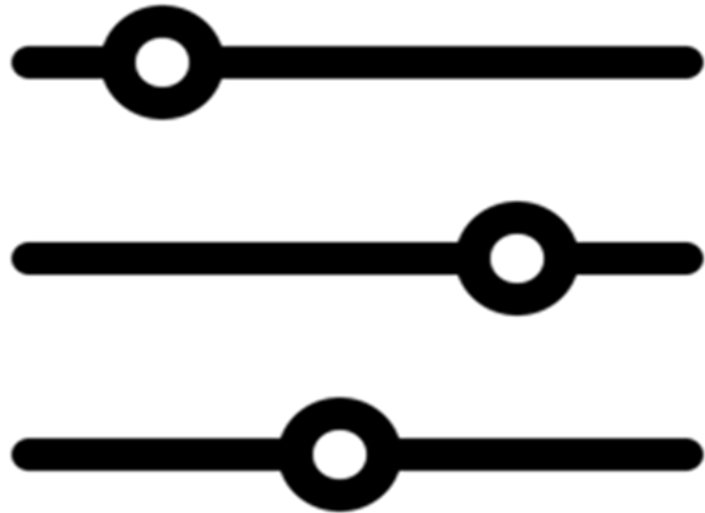
```
ybins = [0,1,3]
```

```
H = [[1,2,1],[1,3,2]]
```

```
fig, ax = plt.subplots()
```

```
hep.hist2dplot(H, xbins, ybins)
```





Styling

Style sheets

Helper function

Fonts

Styling

matplotlib has style sheet functionality

- style sheets can be dictionaries

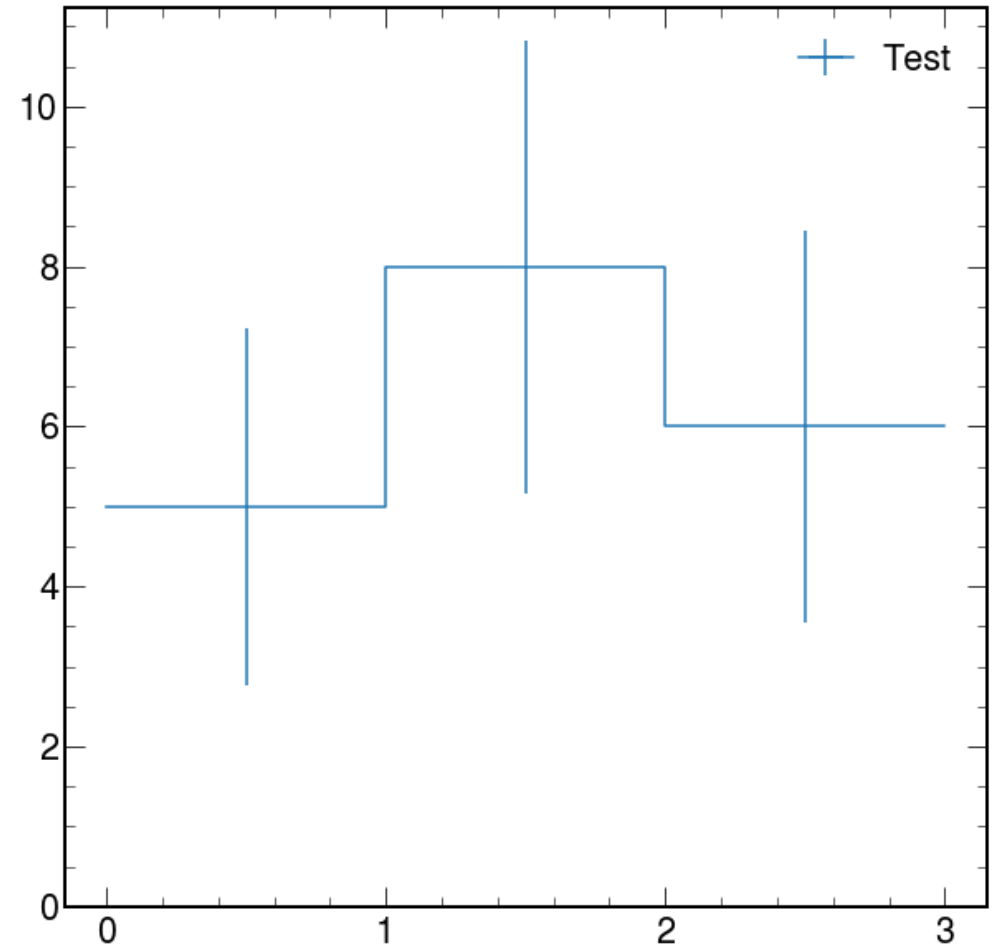
```
import matplotlib.pyplot as plt
import mplhep as hep
plt.style.use([hep.style.ROOT])
```

- styles can be chained as desired
(default/fonts/ticks/grids/LaTeX)

Experiment specific styles

hep.style.ROOT

hep.style.ATLAS



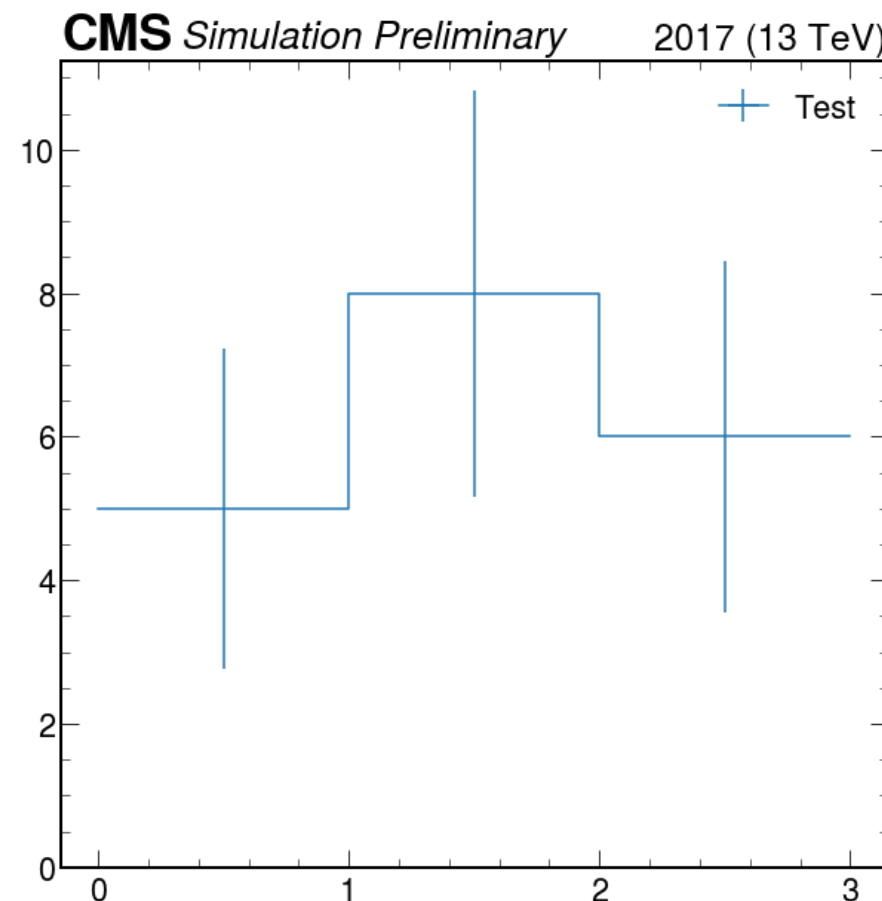
Styling – Helper Functions

Common methods can be aggregated

- Generic - square plot, legend order
- Experiment specific
- CMS Example
 - Experiment and luminosity labels

```
hep.cms.cmslabel(ax, data=False)
```

Input welcome for other experiments



Styling - Fonts

Use **mplhep** for consistent font

- Updates font manager on import
- Helvetica is only available on OSX
- **Helvetica** & **Arial** are proprietary
 - Cannot be distributed
- **TeX Gyre Heros** is a free alternative

CMS *Simulation*

CMS *Simulation*

CMS *Simulation*

Styling - Fonts

Use **mplhep** for consistent font

- Updates font manager on import
- Helvetica is only available on OSX
- **Helvetica** & **Arial** are proprietary
 - Cannot be distributed
- **TeX Gyre Heros** is a free alternative

TeX Gyre Heros

Helvetica

Arial

Summary - mplhep

Adds **plotting methods** for HEP data

- **1D Histograms**

```
hep.histplot(h, bins)
```

- **2D Histograms**

```
hep.hist2dplot(H, xbins, ybins)
```

Facilitates **easy** styling to **publication quality plots**

```
plt.style.use(hep.style.ROOT)
```

mplhep plots in CMS-PAS-HIG-18-045

Already saves time and code!

Summary - mplhep

Next Steps

Write documentation & Compile HEP gallery

Improve `hep.hist2dplot()` with `sns.heatmap()` like features

Identify and include other plotting methods

- **Input Welcome!**

Thank You

Github

<https://github.com/andrzejnovak/mplhep>

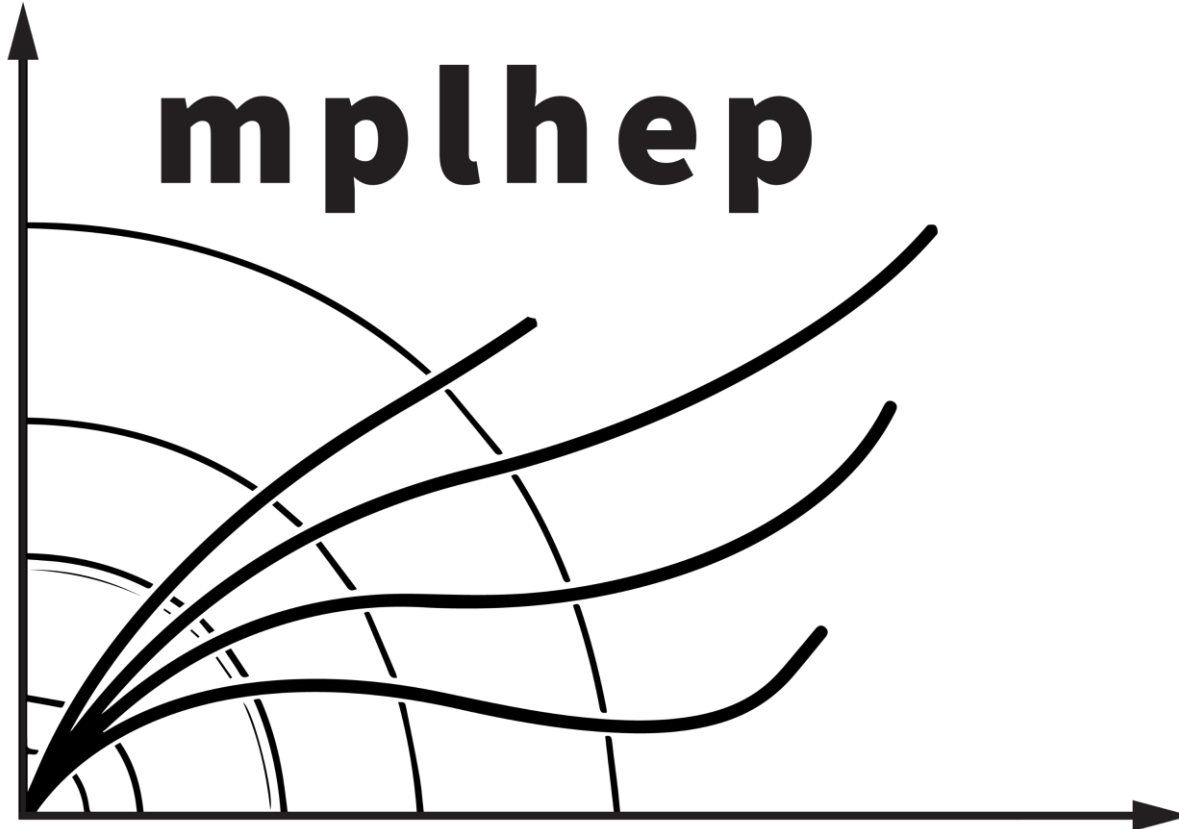
Gitter

<https://gitter.im/HSF/mpl-hep>

Andrzej Novak

Student @ RWTH Aachen

Higgs to charm decays @ CMS



BackUp

Why are we so bad at fonts in HEP?

... so bad I can tell the OS of whoever made the plots in the last two ATLAS papers

Measurement of J/ψ production in association with a W^\pm boson with pp data at 8 TeV / ATLAS Collaboration
A measurement of the production of a prompt J/ψ meson in
LHC. [...]
arXiv:1909.13626 ; CERN-EP-2018-352. - 2019. - 35 p.
Fulltext - Previous draft version - Fulltext

[GeV]

Search for the Higgs boson decays $H \rightarrow ee$ and $H \rightarrow e\mu$ in pp collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector,
Searches for the Higgs boson decays $H \rightarrow ee$ and $H \rightarrow e\mu$ in pp collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector,
arXiv:1909.10235 ; CERN-EP-2019-184. - 2019. - 31 p.
Fulltext - Previous draft version - Fulltext

[GeV]

Both experiments plot with ROOT default font 42

```
4 : helvetica-medium-r-normal "Arial"
```

Different OS ↔ Different Result

GGG

Helvetica and Arial* are proprietary and cannot be packaged

Open alternatives – TeX Gyre Heros, Fira Sans (includes math font)