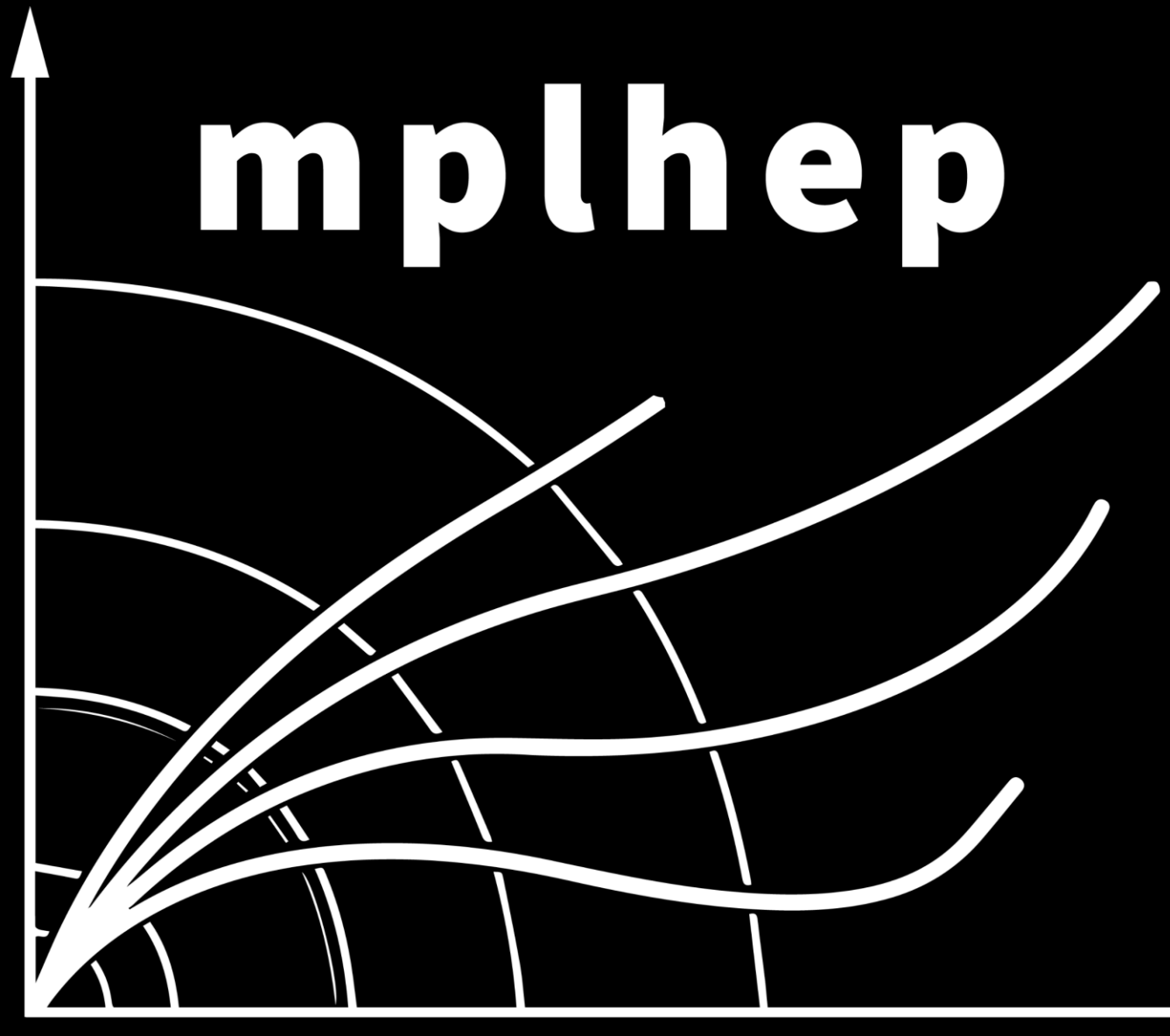


Histogram Visualization Needs in HEP

(in python)

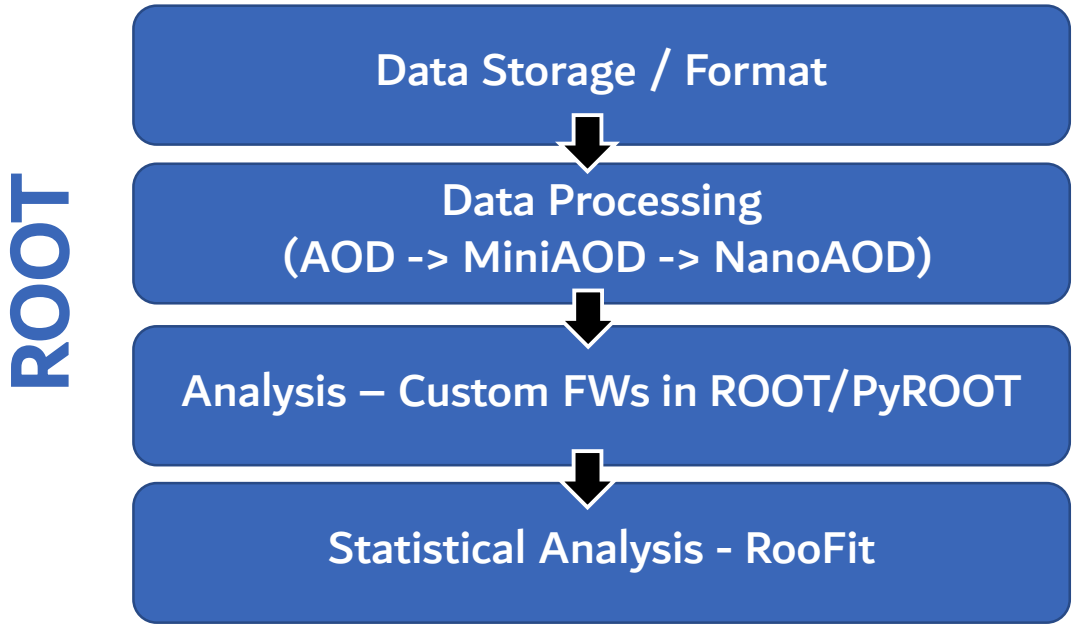
Andrzej Novak

Physics Institute III A

RWTH AACHEN UNIVERSITY

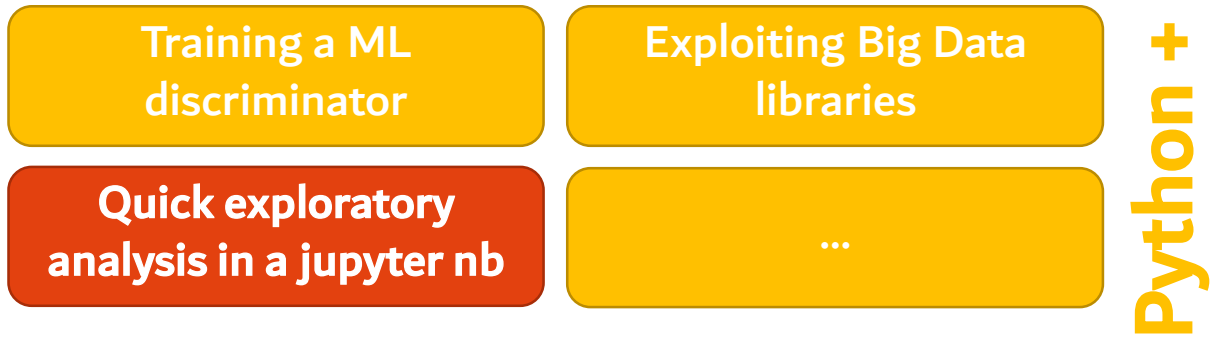
Situation in HEP (or at least in CMS)

- Workflow traditionally locked in ROOT



- Visualization largely standardized
  - ROOT, TDR macros

- What if it isn't?
  - Analyzers more comfortable in python
  - Want access to outside libraries/tech



- Visualization is a mess
  - matplotlib defaults very different from TDR
  - histograming not native

Quintessential HEP Plot

Pre-binned histogram
with uncertainties
?

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

bins = range(0,11)
rng = np.random.normal(6,3,200)
h, bins = np.histogram(rng, bins)
weights = np.random.uniform(.1,.4, len(h))

hw = h * weights
yerr = 1 / np.sqrt(hw)
bin_centers = bins[:-1] + np.diff(bins)/2

fig, ax = plt.subplots()
ax.step(bins, np.r_[hw, hw[-1]], where='post')
ax.errorbar(bin_centers, hw, yerr=yerr,
            ls='none')
ax.set_ylim(0, ax.get_ylim()[-1])
fig.show()
```

Quintessential HEP Plot

Pre-binned histogram

with uncertainties

?

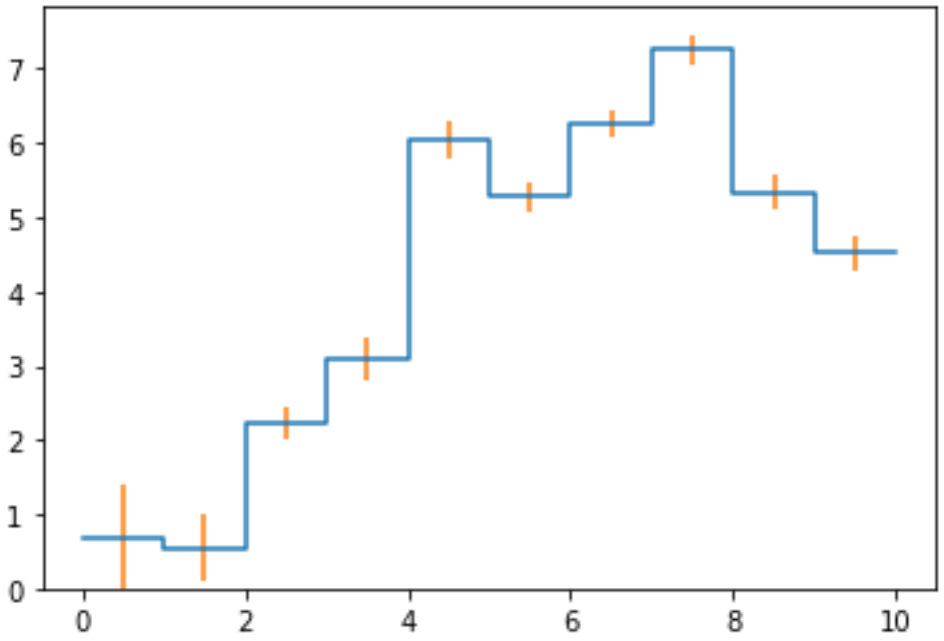
```
import numpy as np
import matplotlib.pyplot as plt

bins = range(0,11)
rng = np.random.normal(6,3,200)
h, bins = np.histogram(rng, bins)
weights = np.random.uniform(.1,.4, len(h))

hw = h * weights
yerr = 1 / np.sqrt(hw)
bin_centers = bins[:-1] + np.diff(bins)/2

fig, ax = plt.subplots()
ax.step(bins, np.r_[hw, hw[-1]], where='post')
ax.errorbar(bin_centers, hw, yerr=yerr,
            ls='none')
ax.set_ylim(0, ax.get_ylim()[-1])
fig.show()
```

Quintessential HEP Plot



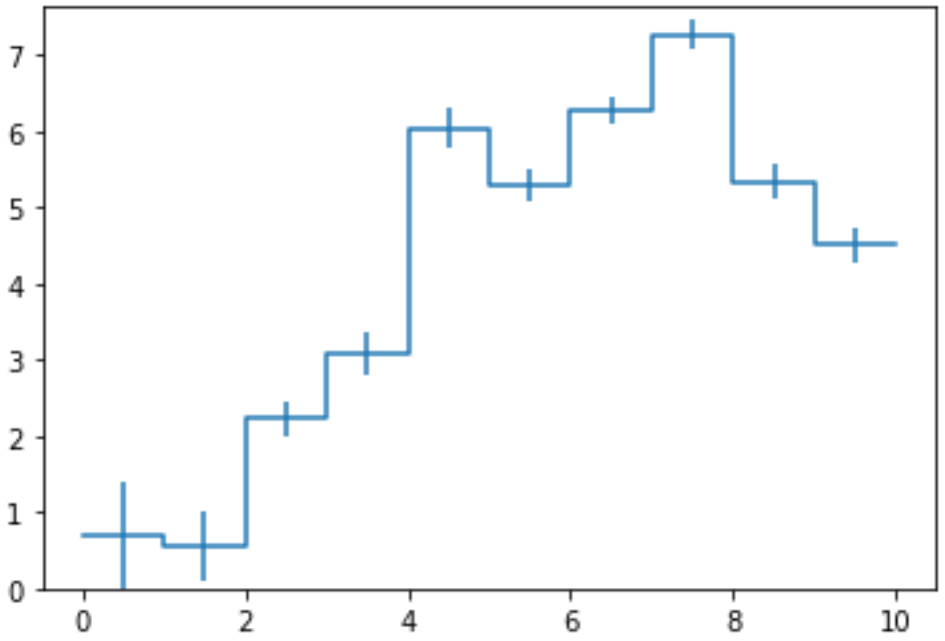

```
import numpy as np
import matplotlib.pyplot as plt

bins = range(0,11)
rng = np.random.normal(6,3,200)
h, bins = np.histogram(rng, bins)
weights = np.random.uniform(.1,.4, len(h))

hw = h * weights
yerr = 1 / np.sqrt(hw)
bin_centers = bins[:-1] + np.diff(bins)/2

fig, ax = plt.subplots()
ax.step(bins, np.r_[hw, hw[-1]], where='post')
ax.errorbar(bin_centers, hw, yerr=yerr,
            ls='none')
ax.set_ylim(0, ax.get_ylim()[-1])
fig.show()
```

Quintessential HEP Plot

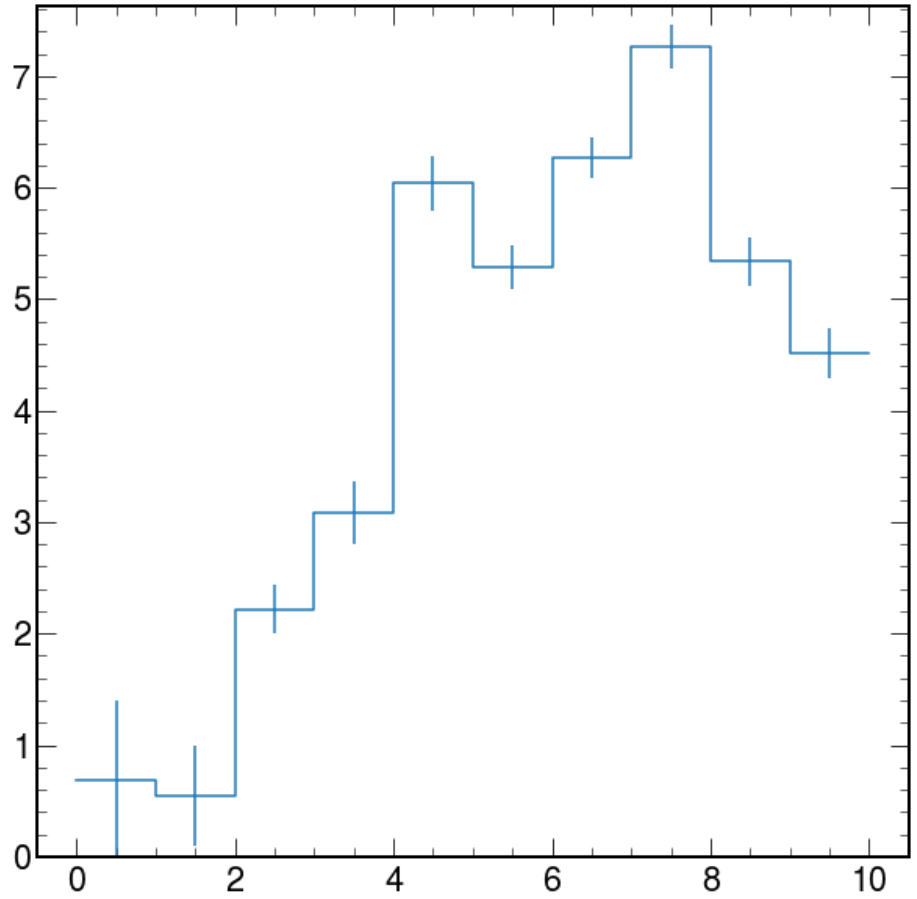



```
import numpy as np
import matplotlib.pyplot as plt
import mplhep as hep

bins = range(0,11)
rng = np.random.normal(6,3,200)
h, bins = np.histogram(rng, bins)
weights = np.random.uniform(.1,.4, len(h))

fig, ax = plt.subplots()
hep.plot.hplot(h, bins, weights, yerr='Poisson')
fig.show()
```

Quintessential HEP Plot

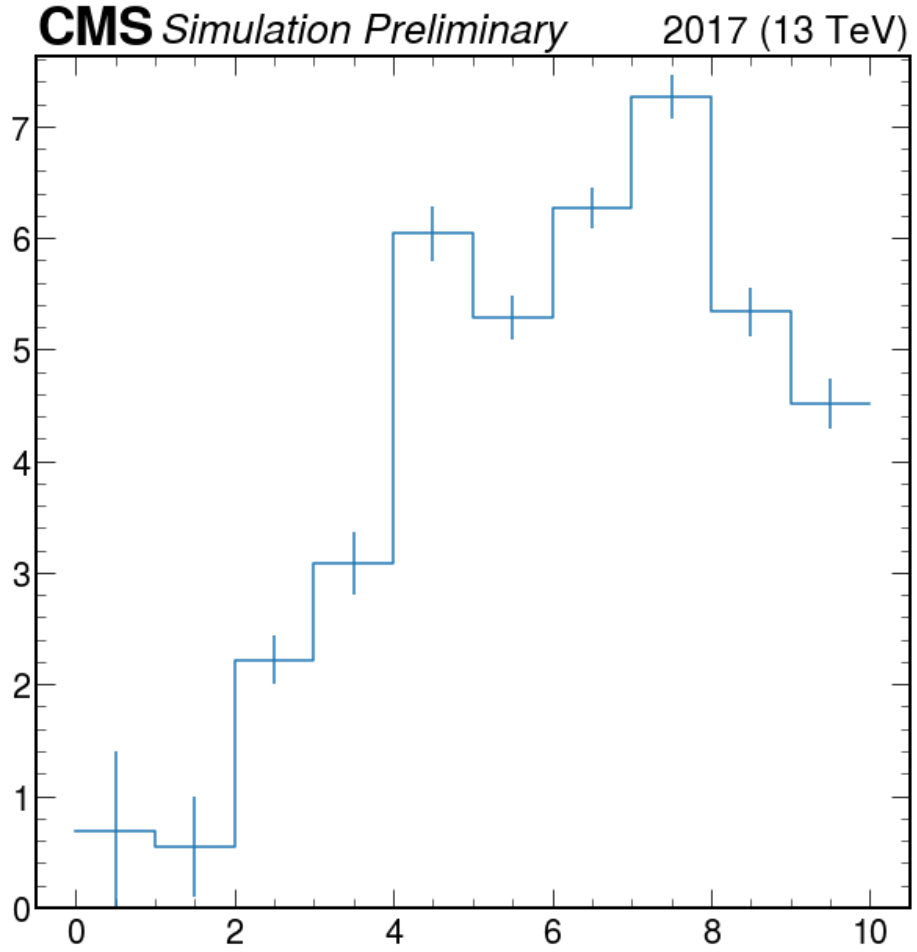



```
import numpy as np
import matplotlib.pyplot as plt
import mplhep as hep

bins = range(0,11)
rng = np.random.normal(6,3,200)
h, bins = np.histogram(rng, bins)
weights = np.random.uniform(.1,.4, len(h))

plt.style.use(hep.style.ROOT)
fig, ax = plt.subplots()
hep.plot.hplot(h, bins, weights, yerr='Poisson')
fig.show()
```

Quintessential HEP Plot

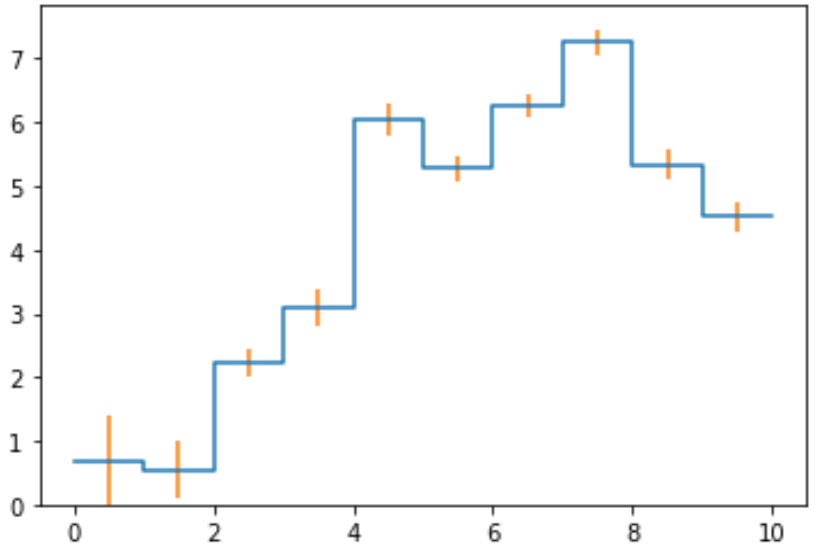



```
import numpy as np
import matplotlib.pyplot as plt
import mplhep as hep

bins = range(0,11)
rng = np.random.normal(6,3,200)
h, bins = np.histogram(rng, bins)
weights = np.random.uniform(.1,.4, len(h))

plt.style.use(hep.style.ROOT)
fig, ax = plt.subplots()
hep.plot.hplot(h, bins, weights, yerr='Poisson')
hep.cms.cmslabel(ax, year='2017')
fig.show()
```

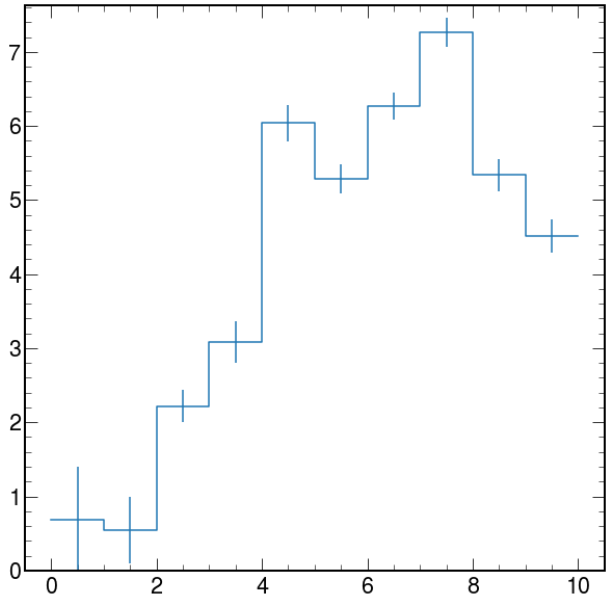
matplotlib



matplotlib

mplhep

```
pip install mplhep
```



More

- Previous discussion
  - https://gitter.im/HSF/mpl-hep
  - Google Doc
- Spec (TBD)
  - Experiment specific plot styles easily ✓
    - Serve Free Helvetica look-alikes ✓
  - Plotting pre-binned 1D histograms easily ✓
  - Plotting 2D histograms easily
  - More Common methods...