model_gen_comp

December 29, 2022

[1]:
```
%cd ../../../
```

```
/home/jan/FMF/masters
```

[2]:
```
saved = "ml_hep_sim/notebooks/article_notebooks/saved/"
```

[3]:
```
from ml_hep_sim.notebooks.article_notebooks.test_runs import *
from ml_hep_sim.pipeline.pipes import *
from ml_hep_sim.pipeline.blocks import *

from ml_hep_sim.plotting.style import style_setup, set_size
from ml_hep_sim.stats.stat_plots import two_sample_plot

from ml_hep_sim.data_utils.higgs.process_higgs_dataset import LATEX_COLNAMES,␣
 ↪LOG_BIN_RANGES

import matplotlib
from scipy.optimize import curve_fit
import matplotlib.pyplot as plt
from tqdm import tqdm
import copy

set_size()
style_setup(seaborn_pallete=True)
```

[4]:
```
glow_train = run_glow_pipeline(
        False,
        False,
        False,
        sig=False,
        num_flows=[10],
        num_train=np.logspace(4, 6, 10).astype(int),
    )

maf_train = run_maf_pipeline(
        False,
        False,
        False,
        sig=False,
        use_mog=True,
        use_maf=True,
        num_mogs=[10],
        num_train=np.logspace(4, 6, 10).astype(int),
    )

spline_train = run_spline_pipeline(
        False,
        False,
        False,
        sig=False,
        num_splines=[32],
        name_str="",
        num_train=np.logspace(4, 6, 10).astype(int),
    )

pipelines = [glow_train[-2], maf_train[-1], spline_train[-1]]
```

```
100%|
                    | 10/10 [00:00<00:00, 16225.55it/s]
100%|
                    | 10/10 [00:00<00:00, 9910.93it/s]
100%|
                    | 10/10 [00:00<00:00, 9283.54it/s]
```

[5]:

```
# glow_pipeline = run_glow_pipeline(train=False, gen=False, test=False)[3][0]
# mafmademog_pipeline = run_maf_pipeline(train=False, gen=False, test=False,␣
 ↪use_mog=True, use_maf=True, num_mogs=[10], name_str="_rerun_10_2")[0]
# spline_pipeline = run_spline_pipeline(train=False, gen=False, test=False,␣
 ↪num_splines=[12])[0]

# pipelines = [glow_pipeline, mafmademog_pipeline, spline_pipeline]
```

[6]:

```
N = 5 * 10**6
device = "cuda"

res = []

for pipeline in pipelines:

    x_ConfigBuilderBlock, _, _, x_ModelTrainerBlock = pipeline.pipes
    x_ConfigBuilderBlock.config["datasets"]["data_params"]["subset_n"] =␣
 ↪[250000, 100000, N]
```

```
    x1 = ModelLoaderBlock(device=device)(x_ConfigBuilderBlock,␣
 ↪x_ModelTrainerBlock)

    x2 = DataGeneratorBlock(N, model_type="flow", chunks=100, device=device)(x1)
    x3 = GeneratedDataVerifierBlock(save_data=False, device=device,␣
 ↪rescale_data=True)(x1, x2)

    x4 = DatasetBuilderBlock()(x_ConfigBuilderBlock)
    x5 = ReferenceDataLoaderBlock(rescale_reference=None, device="cpu")(x4)

    pipe = Pipeline()
    pipe.compose(x1, x2, x3, x4, x5)
    pipe.fit()

    res.append(pipe.pipes[2].generated_data.cpu().numpy())

ref = x5.reference_data.cpu().numpy()

res.append(ref)
```

```
WARNING:root:fitting #0: <ml_hep_sim.pipeline.blocks.ModelLoaderBlock object at
0x7fba00cdb580>!
WARNING:root:fitting #1: <ml_hep_sim.pipeline.blocks.DataGeneratorBlock object
at 0x7fba00cdb5b0>!
100%|
                     | 100/100 [00:04<00:00, 24.48it/s]
WARNING:root:fitting #2: <ml_hep_sim.pipeline.blocks.GeneratedDataVerifierBlock
object at 0x7fba00cdb520>!
WARNING:root:Generated data check…
WARNING:root:nan OK
WARNING:root:pos-inf OK
WARNING:root:neg-inf OK
WARNING:root:pos-inf or neg-inf OK
WARNING:root:pos-inf or neg-inf or nan OK
WARNING:root:Scaled data check…
WARNING:root:nan OK
WARNING:root:pos-inf OK
WARNING:root:neg-inf OK
WARNING:root:pos-inf or neg-inf OK
WARNING:root:pos-inf or neg-inf or nan OK
WARNING:root:fitting #3: <ml_hep_sim.pipeline.blocks.DatasetBuilderBlock object
at 0x7fba00cdb4c0>!
WARNING:root:fitting #4: <ml_hep_sim.pipeline.blocks.ReferenceDataLoaderBlock
object at 0x7fba00cdb550>!
WARNING:root:fitting #0: <ml_hep_sim.pipeline.blocks.ModelLoaderBlock object at
0x7fba00cdb760>!
WARNING:root:fitting #1: <ml_hep_sim.pipeline.blocks.DataGeneratorBlock object
```

```
at 0x7fba008dbbe0>!
100%|
                     | 100/100 [00:57<00:00,  1.75it/s]
WARNING:root:fitting #2: <ml_hep_sim.pipeline.blocks.GeneratedDataVerifierBlock
object at 0x7fba008db280>!
WARNING:root:Generated data check…
WARNING:root:nan ERROR -> found 360 invalid values that will be removed
WARNING:root:pos-inf ERROR -> found 9 invalid values that will be removed
WARNING:root:neg-inf ERROR -> found 9 invalid values that will be removed
WARNING:root:pos-inf or neg-inf ERROR -> found 18 invalid values that will be
removed
WARNING:root:pos-inf or neg-inf or nan ERROR -> found 378 invalid values that
will be removed
/home/jan/FMF/masters/ml_hep_sim/data_utils/dataset_utils.py:59: RuntimeWarning:
overflow encountered in exp
  return 1 / (1 + np.exp(-x))
WARNING:root:Scaled data check…
WARNING:root:nan OK
WARNING:root:pos-inf OK
WARNING:root:neg-inf OK
WARNING:root:pos-inf or neg-inf OK
WARNING:root:pos-inf or neg-inf or nan OK
WARNING:root:fitting #3: <ml_hep_sim.pipeline.blocks.DatasetBuilderBlock object
at 0x7fba008db580>!
WARNING:root:fitting #4: <ml_hep_sim.pipeline.blocks.ReferenceDataLoaderBlock
object at 0x7fba008dbb50>!
WARNING:root:fitting #0: <ml_hep_sim.pipeline.blocks.ModelLoaderBlock object at
0x7fba008c3eb0>!
WARNING:root:fitting #1: <ml_hep_sim.pipeline.blocks.DataGeneratorBlock object
at 0x7fba0078ea90>!
100%|
                     | 100/100 [07:55<00:00,  4.76s/it]
WARNING:root:fitting #2: <ml_hep_sim.pipeline.blocks.GeneratedDataVerifierBlock
object at 0x7fba00796070>!
WARNING:root:Generated data check…
WARNING:root:nan OK
WARNING:root:pos-inf OK
WARNING:root:neg-inf OK
WARNING:root:pos-inf or neg-inf OK
WARNING:root:pos-inf or neg-inf or nan OK
WARNING:root:Scaled data check…
WARNING:root:nan OK
WARNING:root:pos-inf OK
WARNING:root:neg-inf OK
WARNING:root:pos-inf or neg-inf OK
WARNING:root:pos-inf or neg-inf or nan OK
WARNING:root:fitting #3: <ml_hep_sim.pipeline.blocks.DatasetBuilderBlock object
at 0x7fba007963a0>!
```

```
WARNING:root:fitting #4: <ml_hep_sim.pipeline.blocks.ReferenceDataLoaderBlock
object at 0x7fba00796280>!
```

[7]:

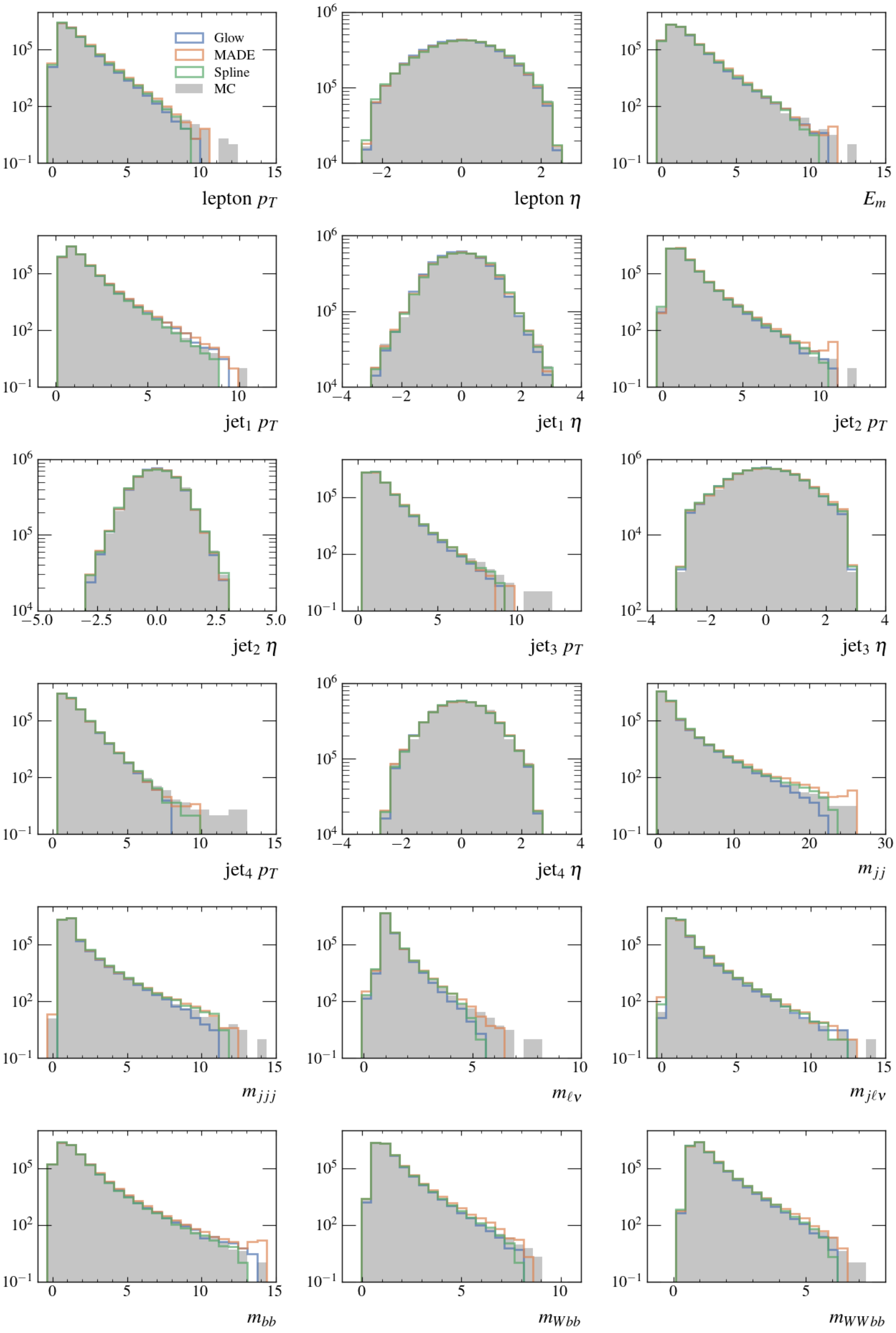
```python
def N_sample_plot(
    samples,
    axs,
    n_bins="auto",
    label=None,
    labels=None,
    log_scale=False,
    bin_range=None,
    xlim=None,
    titles=None,
    last_c="C7",
    **kwargs,
):
    n_features = samples[0].shape[1]

    if bin_range is not None:
        if not any(isinstance(el, list) for el in bin_range):
            bin_range = [bin_range] * n_features

    combined_sample = np.concatenate(samples)

    for feature in range(n_features):
        bin_edges = np.histogram_bin_edges(
            combined_sample[:, feature], bins=n_bins, range=bin_range[feature]␣
↪if bin_range else None
        )

        for i, sample in enumerate(samples):
            if i == len(samples) - 1:
                axs[feature].hist(sample[:, feature], bins=bin_edges,␣
↪histtype="stepfilled", color=last_c, lw=2, alpha=0.5)
            else:
                axs[feature].hist(sample[:, feature], bins=bin_edges,␣
↪histtype="step", **kwargs)

        if feature == 0 and label is not None:
            axs[feature].legend(label, fontsize=14)

        if labels is not None:
            axs[feature].set_xlabel(labels[feature], size=20)

        if log_scale:
            axs[feature].set_yscale("log")
        if xlim:
            axs[feature].set_xlim(xlim[feature])

        if titles is not None:
            axs[feature].set_title(titles[feature], size=15, loc="right")

    return axs
```

[8]:

```
LOG_BIN_RANGES = [
    [-1, 15],
    [-3, 3],
    [-1, 15],
    [-1, 12],
    [-4, 4],
    [-1, 14],
    [-5, 5],
    [-1, 14],
    [-4, 4],
    [-1, 15],
    [-4, 4],
    [-1.5, 30],
    [-1, 15],
    [-1, 10],
    [-1, 15],
    [-1, 15],
    [-1, 11],
    [-1, 8],
]

BIN_RANGES = [
    [0, 4],
    [-3, 3],
    [-0.1, 4],
    [0, 5],
    [-4, 4],
    [0, 4],
    [-5, 5],
    [0, 5],
    [-4, 4],
    [0, 5],
    [-3, 3],
    [0, 3],
    [0, 3],
    [0.75, 1.5],
    [0, 3],
    [0, 3],
    [0, 3],
    [0, 3],
]
```

[9]:

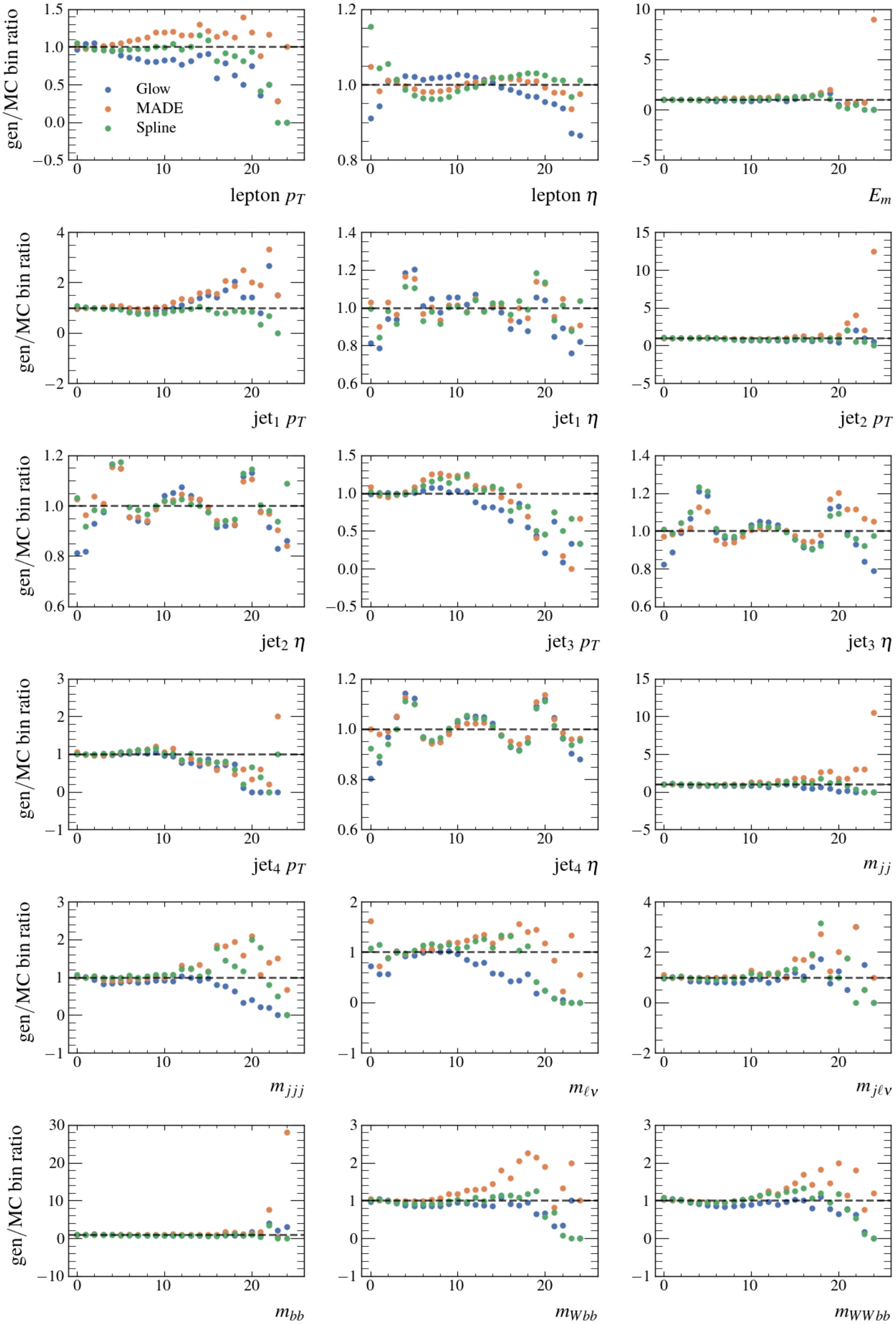
```
fig, axs = plt.subplots(6, 3, figsize=(13, 19))
axs = axs.flatten()

N_sample_plot(res, axs, n_bins=25, log_scale=True,
              labels=LATEX_COLNAMES, lw=2, alpha=0.7,
              label=["Glow", "MADE", "Spline", "MC"],
              xlim=LOG_BIN_RANGES, bin_range=LOG_BIN_RANGES)
plt.tight_layout()
plt.savefig(saved + "mc_gen_dists_1000000.pdf")
```


Glow
MADE
Spline
MC
10^{-1}
10^{2}
10^{5}
10^{4}
10^{6}
10^{2}
lepton p_T
lepton η
E_m
jet$_1$ p_T
jet$_1$ η
jet$_2$ p_T
jet$_2$ η
jet$_3$ p_T
jet$_3$ η
jet$_4$ p_T
jet$_4$ η
m_{jj}
m_{jjj}
$m_{\ell\nu}$
$m_{j\ell\nu}$
m_{bb}
m_{Wbb}
m_{WWbb}
0
5
10
15
−2
2
−4
4
−5.0
−2.5
0.0
2.5
5.0
20
30

[10]:

```
fig, axs = plt.subplots(6, 3, figsize=(13, 19))
axs = axs.flatten()

n_bins = 25

a, b, c, mc = res

combined_sample = np.concatenate([a, b, c])

for i in tqdm(range(18)):
    bin_edges = np.histogram_bin_edges(combined_sample[:, i], bins=n_bins)

    hist_a, _ = np.histogram(a[:, i], bins=bin_edges)
    hist_b, _ = np.histogram(b[:, i], bins=bin_edges)
    hist_c, _ = np.histogram(c[:, i], bins=bin_edges)
    hist_mc, _ = np.histogram(mc[:, i], bins=bin_edges)

    axs[i].scatter(range(len(hist_a)), hist_a / hist_mc, s=25)
    axs[i].scatter(range(len(hist_b)), hist_b / hist_mc, s=25)
    axs[i].scatter(range(len(hist_c)), hist_c / hist_mc, s=25)

    axs[i].set_xlim(-1, 26)
    axs[i].set_xlabel(LATEX_COLNAMES[i])

    if i in [0, 3, 6, 9, 12, 15]:
        axs[i].set_ylabel("gen$/$MC bin ratio")

    if i == 0:
        axs[i].legend(["Glow", "MADE", "Spline"])

    axs[i].axhline(1, c="k", ls='--', alpha=0.7)

plt.tight_layout()
plt.savefig(saved + "ratios.pdf")
```

```
 17%|
| 3/18 [00:03<00:15,  1.06s/it]/tmp/ipykernel_618221/4122431777.py:18:
RuntimeWarning: divide by zero encountered in true_divide
  axs[i].scatter(range(len(hist_a)), hist_a / hist_mc, s=25)
/tmp/ipykernel_618221/4122431777.py:19: RuntimeWarning: divide by zero
encountered in true_divide
  axs[i].scatter(range(len(hist_b)), hist_b / hist_mc, s=25)
/tmp/ipykernel_618221/4122431777.py:20: RuntimeWarning: invalid value
encountered in true_divide
  axs[i].scatter(range(len(hist_c)), hist_c / hist_mc, s=25)
 50%|
| 9/18 [00:09<00:09,  1.05s/it]/tmp/ipykernel_618221/4122431777.py:18:
RuntimeWarning: invalid value encountered in true_divide
  axs[i].scatter(range(len(hist_a)), hist_a / hist_mc, s=25)
/tmp/ipykernel_618221/4122431777.py:20: RuntimeWarning: divide by zero
encountered in true_divide
  axs[i].scatter(range(len(hist_c)), hist_c / hist_mc, s=25)
100%|
                      | 18/18 [00:18<00:00,  1.05s/it]
```


gen/MC bin ratio
Glow
MADE
Spline
1.5
1.0
0.5
0.0
−0.5
0
10
20
lepton p_T
1.2
1.0
0.8
lepton η
10
5
0
−5
E_m
4
2
0
−2
jet$_1$ p_T
1.4
1.2
1.0
0.8
0.6
jet$_1$ η
15
10
5
0
−5
jet$_2$ p_T
1.2
1.0
0.8
0.6
jet$_2$ η
1.5
1.0
0.5
0.0
−0.5
jet$_3$ p_T
1.4
1.2
1.0
0.8
0.6
jet$_3$ η
3
2
1
0
−1
jet$_4$ p_T
1.2
1.0
0.8
0.6
jet$_4$ η
15
10
5
0
−5
m_{jj}
3
2
1
0
−1
m_{jjj}
2
1
0
−1
$m_{\ell\nu}$
4
2
0
−2
$m_{j\ell\nu}$
30
20
10
0
−10
m_{bb}
3
2
1
0
−1
m_{Wbb}
3
2
1
0
−1
m_{WWbb}

[ ]:

```
```