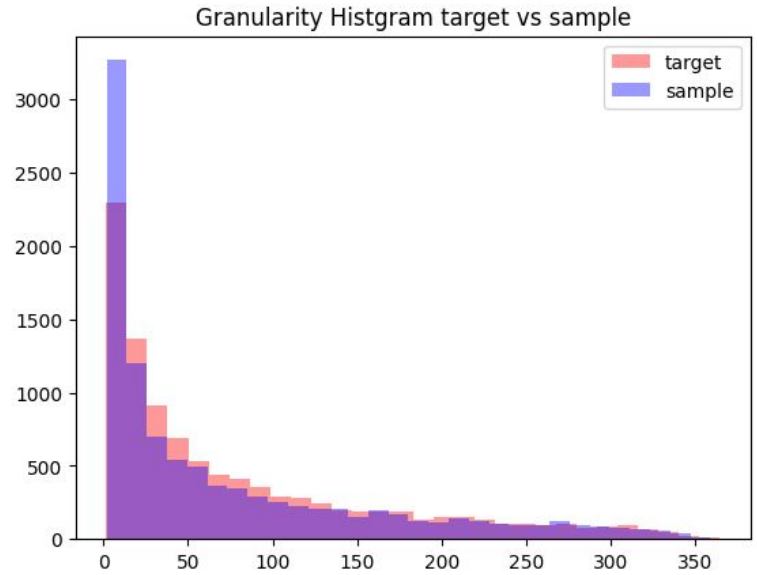
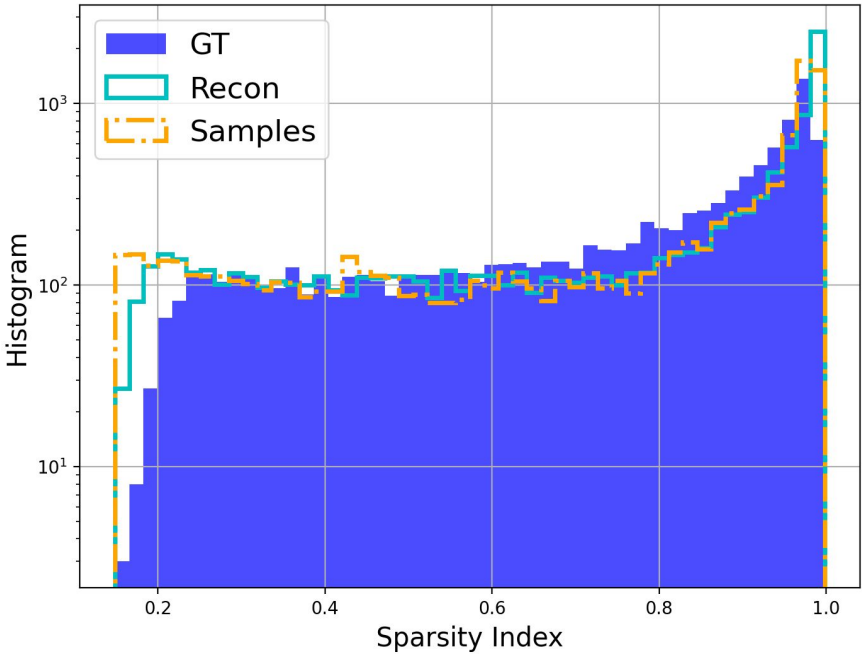
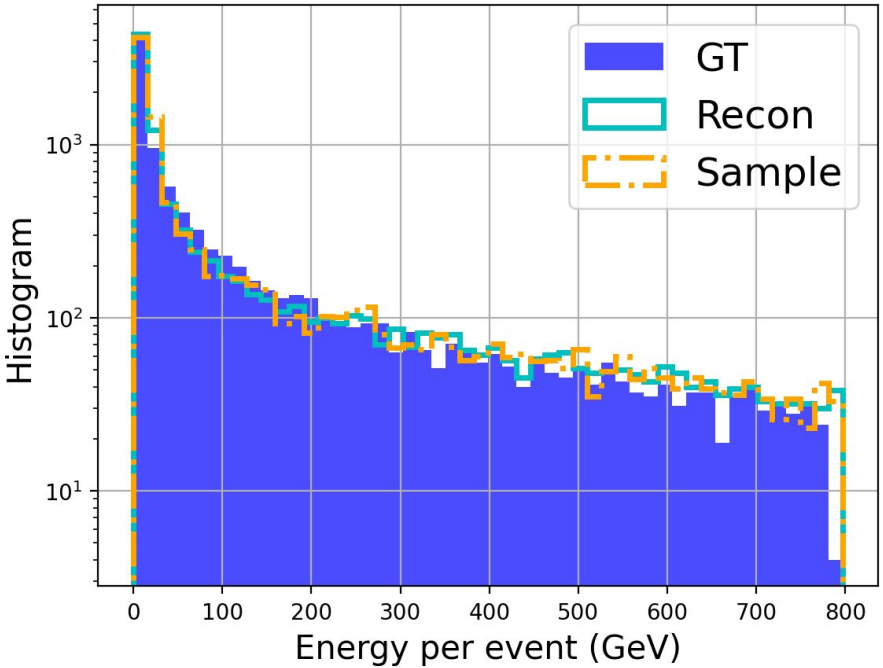
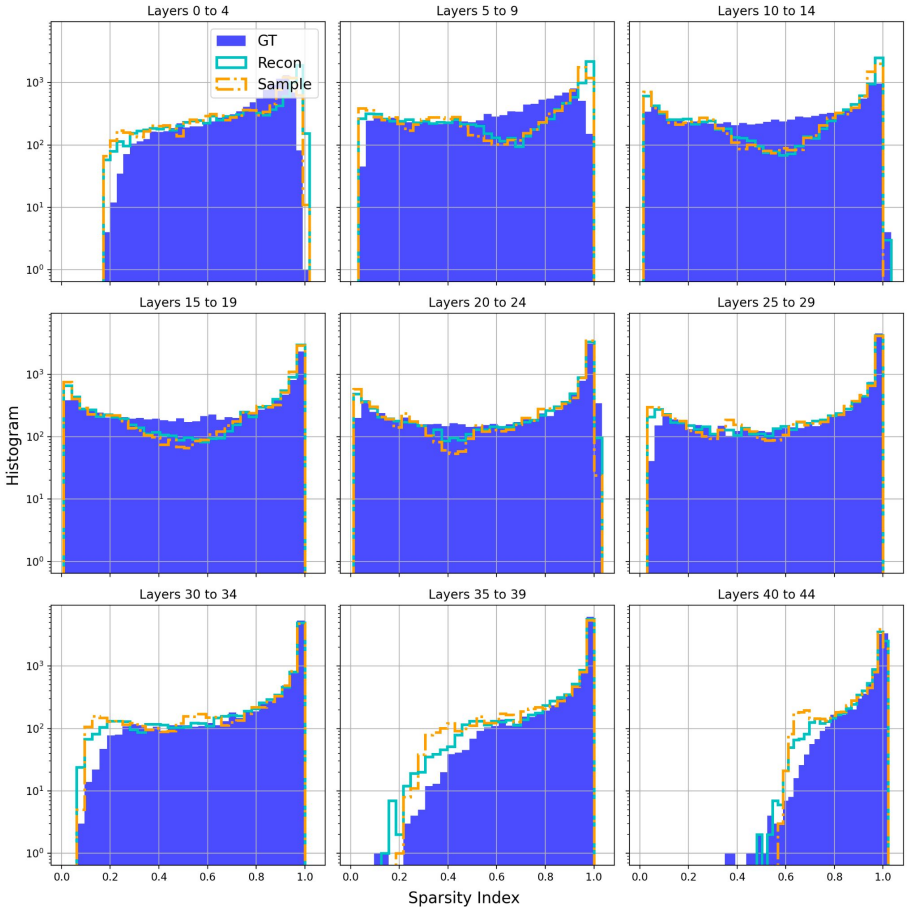
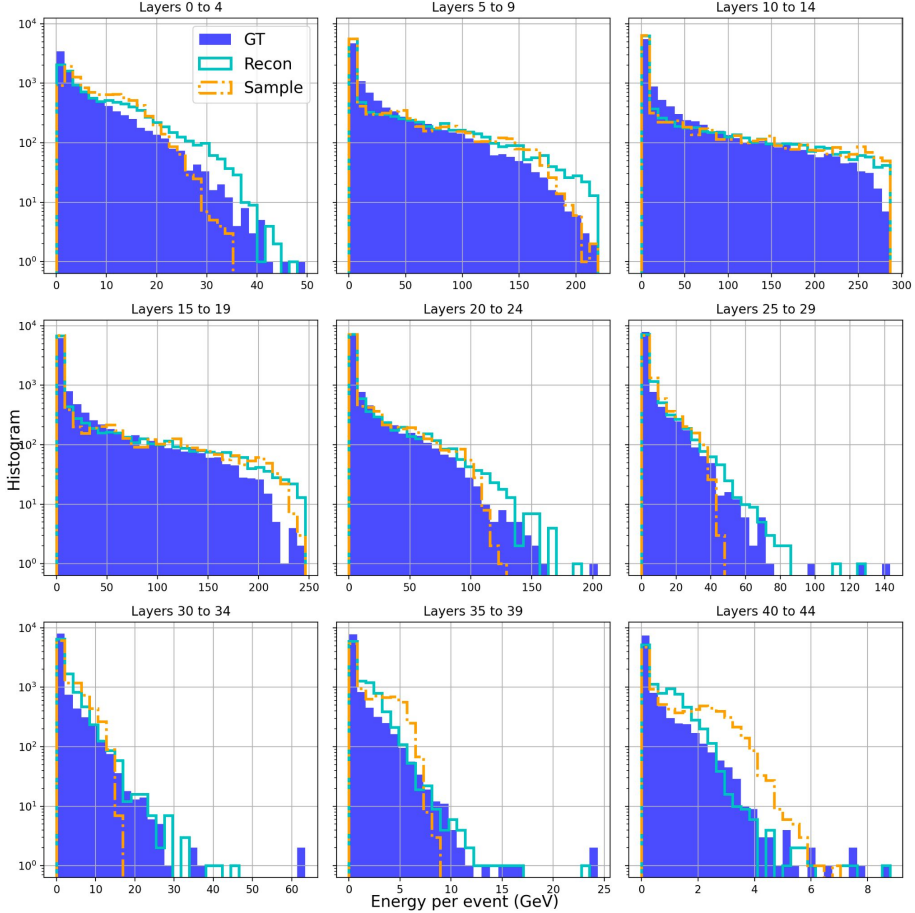
```python
def measure_single_granularity(data_tensor):
    # Calculate the differences between consecutive elements
    diffs = (data_tensor[9:,:] - data_tensor[:-9,:])
    std_g = torch.std(diffs, dim = 1)
    return std_g
```

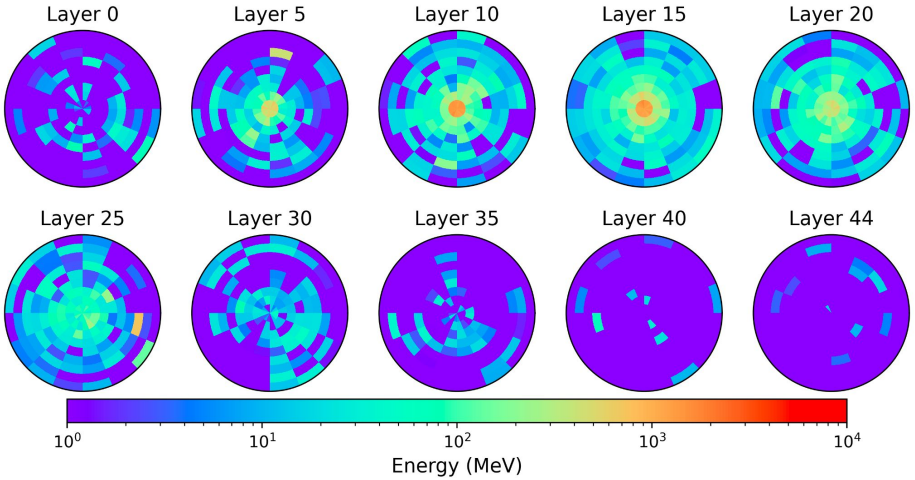



GT
Recon
Samples
Histogram
10^3
10^2
10^1
0.2
0.4
0.6
0.8
1.0
Sparsity Index


GT
Recon
Sample
Histogram
10^3
10^2
10^1
0
100
200
300
400
500
600
700
800
Energy per event (GeV)

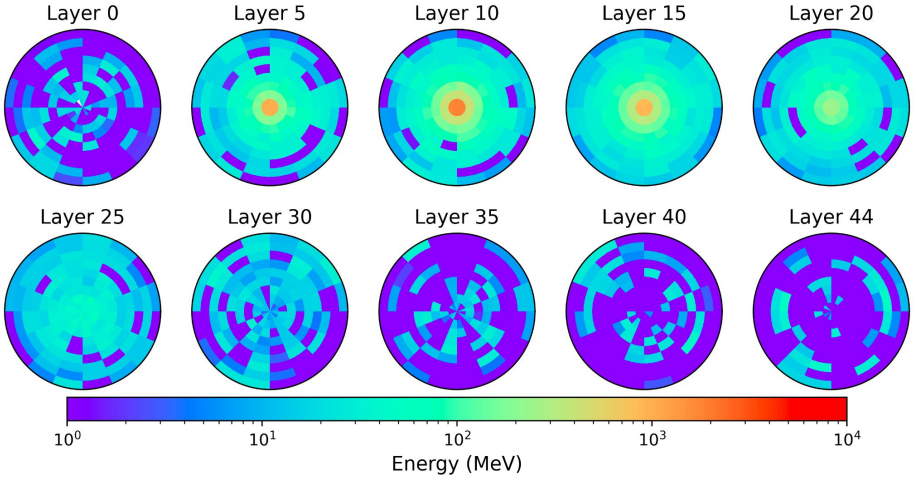

Layers 0 to 4
Layers 5 to 9
Layers 10 to 14
Layers 15 to 19
Layers 20 to 24
Layers 25 to 29
Layers 30 to 34
Layers 35 to 39
Layers 40 to 44
GT
Recon
Sample
Histogram
Sparsity Index

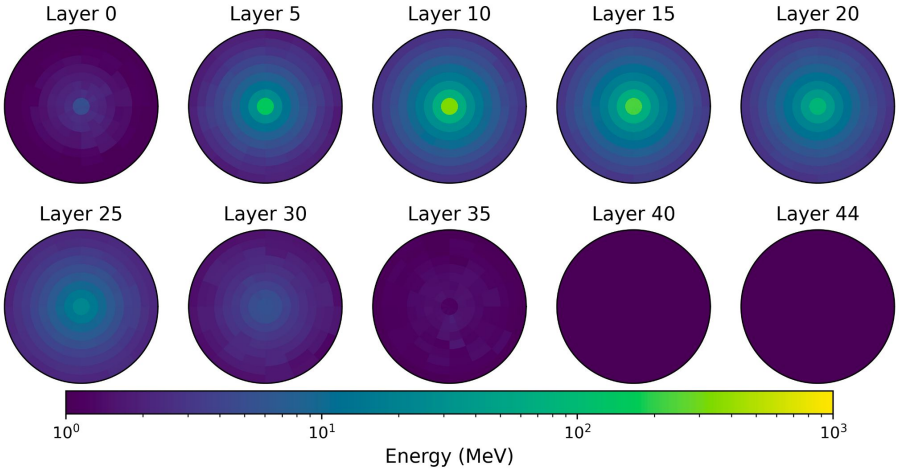

Layers 0 to 4
Layers 5 to 9
Layers 10 to 14
Layers 15 to 19
Layers 20 to 24
Layers 25 to 29
Layers 30 to 34
Layers 35 to 39
Layers 40 to 44
GT
Recon
Sample
Histogram
Energy per event (GeV)



Input

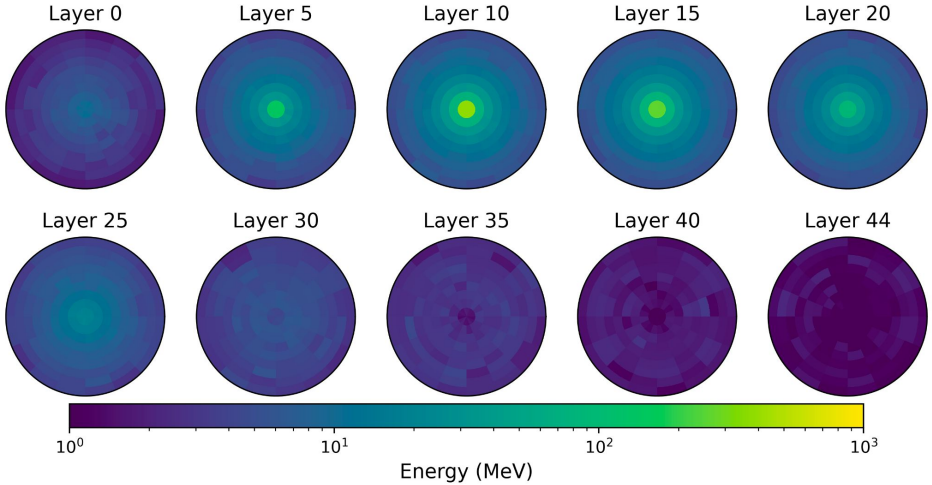
sample



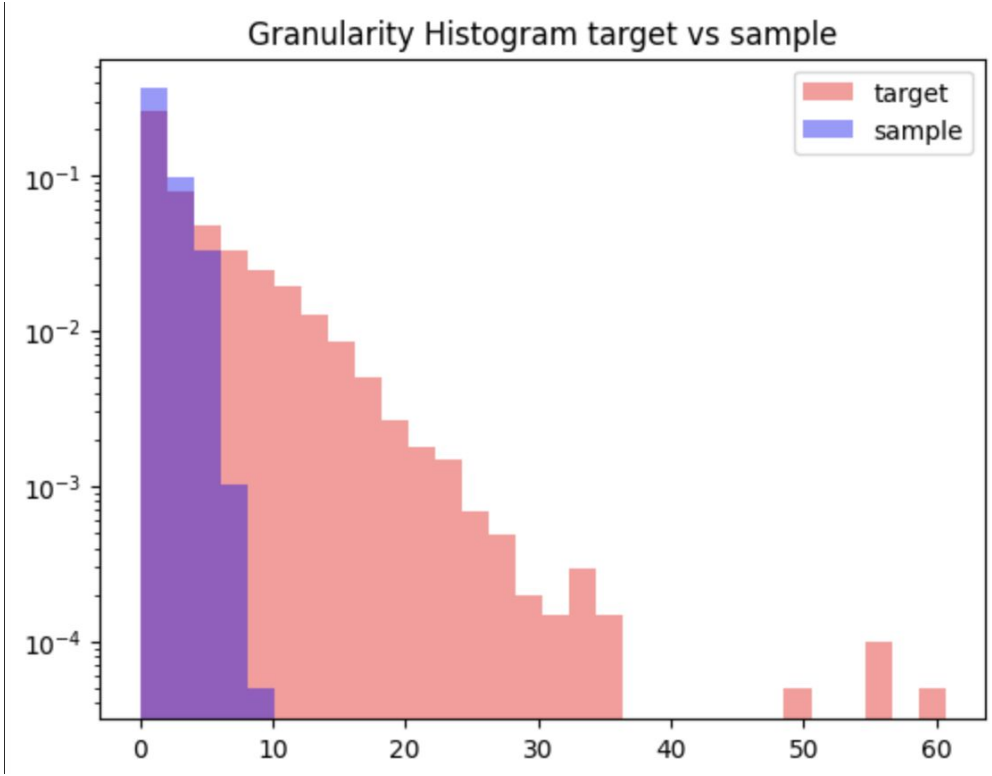


Input

sample



Another check

For each event:
GR = std [E(x) / batch_mean(x)]
where x is the voxel position

Right shows the histogram of GR on 10000 events for target and sample