A Appendix

A.1 Shower shape variables

We extend the list of shower shape variables described in Sec. A.1 :

Point level marginals. Marginals of each point feature by considering the set all the points from all the point clouds together.

Feature means $\langle\eta_i\rangle, \langle\phi_i\rangle, \langle r_i\rangle, \langle E_i\rangle$. Mean of each feature.

$$\langle\eta_i\rangle = \frac{\sum_j \eta_j^i}{\sum_j 1},\ \langle\phi_i\rangle = \frac{\sum_j \phi_j^i}{\sum_j 1},\ \langle r_i\rangle = \frac{\sum_j r_j^i}{\sum_j 1}\ \langle E_i\rangle = \frac{\sum_j E_j^i}{\sum_j 1}$$

where $r_j^i = \sqrt{(\eta_j^i)^2 + (\phi_j^i)^2}$ denotes the distance of the point in the lateral plane from the center.

Feature variances $\sigma_{\langle\eta_i\rangle}, \sigma_{\langle\phi_i\rangle}, \sigma_{\langle r_i\rangle}, \sigma_{\langle E_i\rangle}$. Variance of each feature. $\sigma_{\langle\eta_i\rangle} = \sqrt{\frac{\sum_j {\eta_j^i}^2}{\sum_j 1} - \langle\eta_i\rangle^2}$

Layer Energy $\bar{E}_i$. Denotes the total energy deposited in layer i of the shower. $\bar{E}_i = \sum_{j\in N_i} E_j^i$.

Total Energy E_{tot}. Total energy across all layers of the shower. $E_{\text{tot}} = \sum_{i\leq N} \bar{E}_i$.

Layer Centroids $\langle\eta_i\rangle_E$, $\langle\phi_i\rangle_E$, $\langle r_i\rangle_E$. Energy weighted mean of the features (η, ϕ, or r).

$$\langle\eta_i\rangle = \frac{\sum_j E_j^i \eta_j^i}{E_i},\ \langle\phi_i\rangle = \frac{\sum_j E_j^i \phi_j^i}{E_i},\ \langle r_i\rangle = \frac{\sum_j E_j^i r_j^i}{E_i}$$

The layer centroids can be interpreted as the center of energy in the lateral plane in respective dimensions.

Layer Lateral Width $\sigma_{\langle\eta_i\rangle_E}, \sigma_{\langle\phi_i\rangle_E}, \sigma_{\langle r_i\rangle_E}$. Denotes the standard deviation of the layer centroids.

$$\sigma_{\langle\eta_i\rangle_E} = \sqrt{\frac{\sum_j E_j^i (\eta_j^i)^2}{E_i} - \langle\eta_i\rangle_E^2}$$

The layer lateral widths can be interpreted as the spread around the center of energy in the lateral plane in respective dimensions. We drop the layer notation i from the above metrics when working with a single layer for brevity.

Layer Energy Fraction f_i. Fraction of the total energy deposited in layer i of the shower. $f_i = \bar{E}_i / E_{\text{tot}}$.

Energy Ratio $E_{\text{ratio},i}$. Ratio of the difference between highest and second highest energy intensity point or cell in layer i and their difference. $E_{\text{ratio},i} = \frac{E^i_{[1]} - E^i_{[2]}}{E^i_{[1]} + E^i_{[2]}}$.

Depth d. Deepest layer in the shower with non-zero energy deposit. $d = \max_i\{i : \max_j(E_j^i) > 0\}$.

Layer/Depth Weighted Total Energy l_d. Sum of the layer energies weighted by the layer number. $l_d = \sum_{i\leq N} i \cdot \bar{E}_i$.

Shower Depth s_d. Depth weighted total energy normalized by the total energy in the shower. $s_d = l_d / E_{\text{tot}}$.

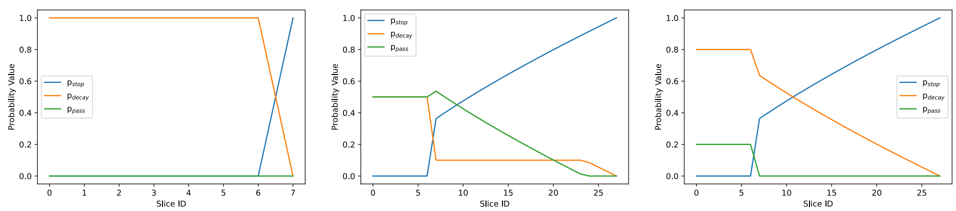
Shower Depth Width σ_{s_d}. Standard deviation of s_d in units of layer number.

$$\sigma_{s_d} = \sqrt{\frac{\sum_{i=0}^2 i^2 \cdot \bar{E}_i}{E_{\text{tot}}} - \left(\frac{\sum_{i=0}^2 i \cdot \bar{E}_i}{E_{\text{tot}}}\right)^2}$$

A.2 Details on different variations of SUPA datasets

A.2.1 Parameters

Fig. 5 shows the remaining parameters used for generating SUPA variations (see Table 2 for details on other parameters). SUPAv1 is most deterministic as particles always split in the first six sub-layers with no deposits ($p_{split} = 1$ and $p_{stop} = 0$ for all sub-layers < 7), further since $p_{stop} = 1$ at sub-layer 7, all the particles get deposited. Thus each event/example in SUPAv1 has exactly $128 (= 2^7)$ points. Further, since α is fixed to 0, all splits are symmetric and energy is always halved at each split, thus all deposits have the same energy value. SUPAv5 has higher p_{split} in the initial sub-layers (< 7) than SUPAv2-4, while p_{stop} is the same for all of them, thus SUPAv5 has more number of hits/points than SUPAv2-4 in the respective sub-layers or layers.

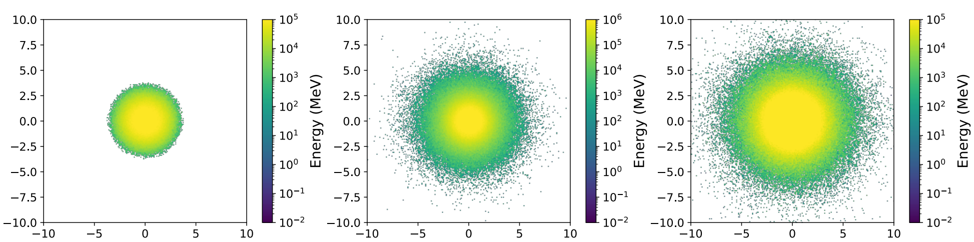


(a) SUPAv1 (b) SUPAv2, SUPAv3, SUPAv4 (c) SUPAv5

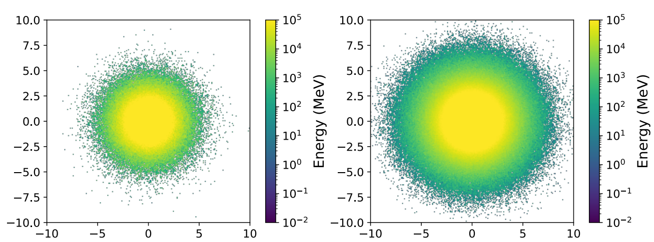
Figure 5: Parameters p_{split}, p_{stop}, p_{pass} for SUPA variations

A.2.2 Shower Shape Variables

Fig. 6 shows the average events for different variations of SUPA datasets and Figs. Fig. 7 - 12 shows the histograms of the various shower shape variables for all SUPA datasets.



(a) SUPAv1 (b) SUPAv2 (c) SUPAv3



(d) SUPAv4 (e) SUPAv5

Figure 6: Average event representation for different variations of SUPA datasets

A.3 Point Cloud Generative Models

PointFlow PointFlow [Yang et al., 2019] is a flow based model with a PointNet-like encoder and a continuous normalizing flow (CNF) decoder. Additionally, the latents (encoder outputs) are modeled

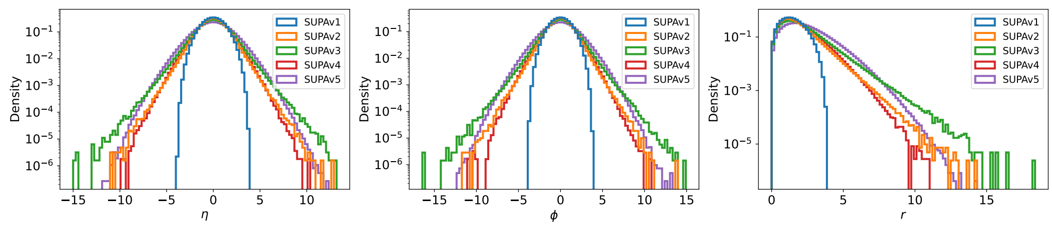


Figure 7: Histograms of point level distributions

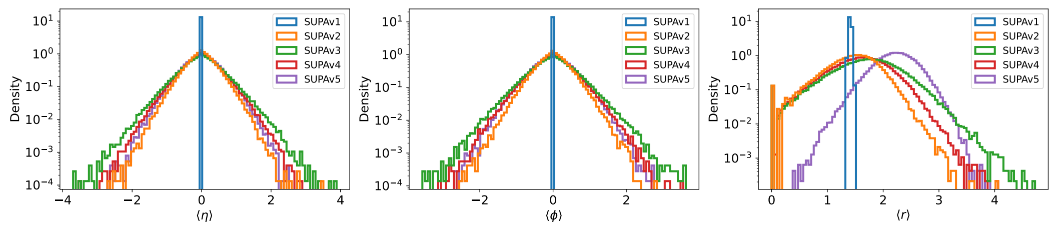


Figure 8: Histograms of feature means

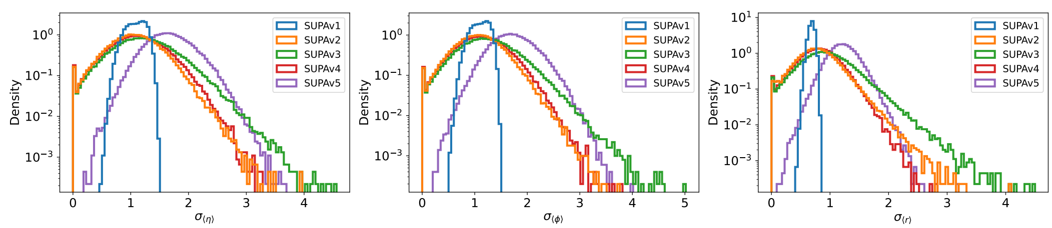
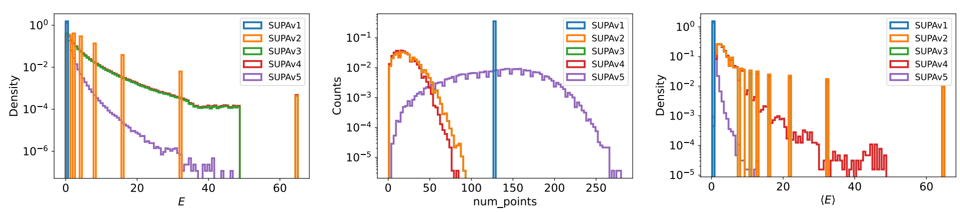
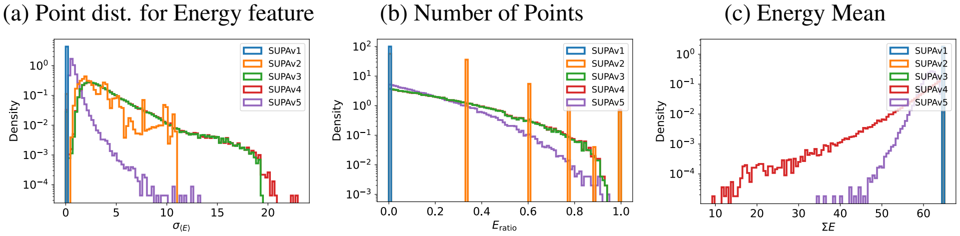


Figure 9: Histograms of feature variances



(a) Point dist. for Energy feature (b) Number of Points (c) Energy Mean



(d) Energy Variance (e) Cell Energy Ratio (f) Layer Energy

Figure 10: Histograms of various shower shape variables

with another CNF to enable sampling. We adapted the PointFlow code to handle variable number of points with masking and masked batch norm. The encoder consists of 1D convolutions with filter sizes 128, 128, 256 and 512, followed by a three-layer MLP with 256 and 128 hidden dimensions to convert the point cloud into its latent representation of size 128. The CNF decoder has four conditional `concatsquash` layers with a hidden dimension of 128 and the latent CNF has three

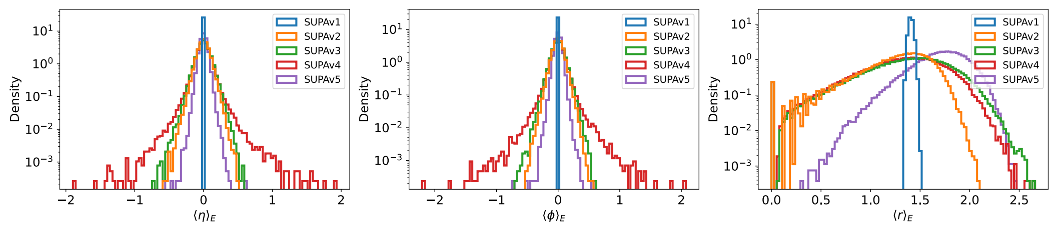


Figure 11: Histograms of layer centroids

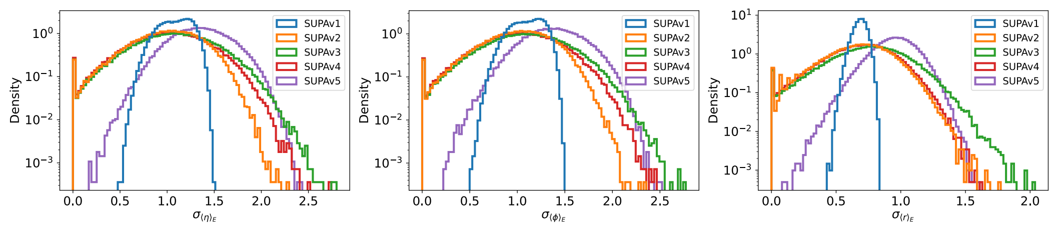


Figure 12: Histograms of layer widths

`concatsquash` layers with a hidden dimension of 64. The overall architecture has $0.7M$ trainable parameters.

SetVAE SetVAE Kim et al. [2021] is a transformer-based hierarchical VAE for set-structured data which learns latent variables at multiple scales, capturing coarse-to-fine dependency of the set elements while achieving permutation invariance. We set the number of heads to 4, the dimension of the initial set to 64, the hidden dimension to 64, the number of mixtures for the initial set to 4, and the number of inducing points in the hierarchical setup to $[2, 4, 8, 16, 32]$. The overall architecture has $0.5M$ trainable parameters.

Transflowmer The *Transflowmer* is flow-architecture using Real NVP layers [Dinh et al., 2016]. As the events are point clouds of varying cardinality, the coupling layers of the flow are required to be permutation equivariant and able to process a varying number of inputs. To satisfy these constraints, we use transformers [Vaswani et al., 2017] without positional encoding in the coupling layers. The overall architecture consists of 16 coupling layers, each of them is parametrised by a 3 transformer layers with $d_{model} = 32$. The overall architecture has $2.1M$ parameters.

We train all the models with $100K$ training examples.

A.4 Experiments on SUPA datasets

We train point cloud generative models, PointFlow [Yang et al., 2019], SetVAE [Kim et al., 2021], and Transflowmer on SUPA datasets. In this section, we show histogram plots to compare the generative performance across different shower shape variables. For all these plots, the axes limits are chosen according to the ground truth data and generated samples can have probability mass outside the shown range.

A.4.1 SUPAv1

Figs. 13 - 18 show the histograms of various shower shape variables for SUPAv1 and samples generated with PointFlow, SetVAE, and Transflowmer.

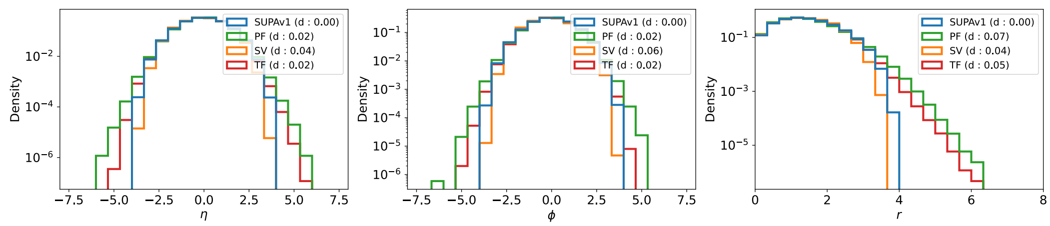


Figure 13: Histograms of point distributions for η, ϕ, and r

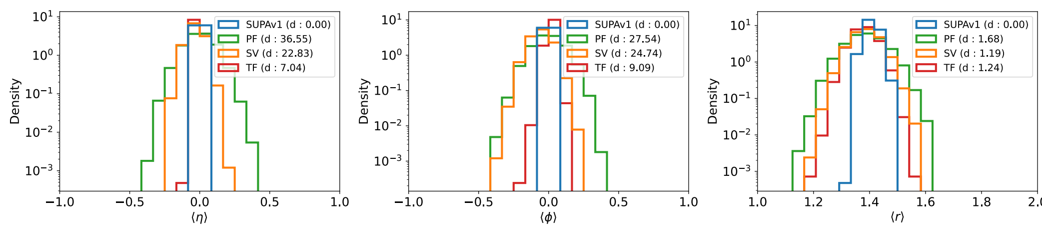


Figure 14: Histograms of sample means for different features

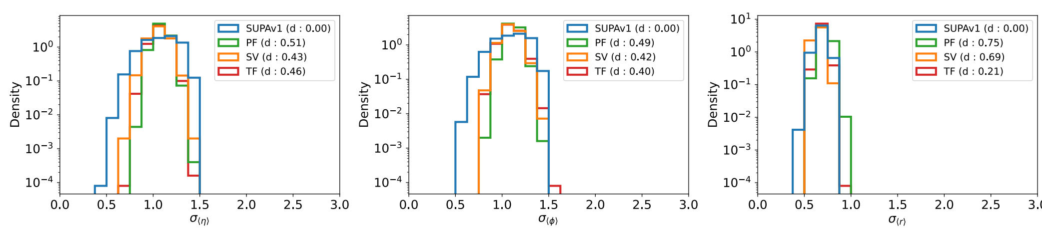


Figure 15: Histograms of sample variance for different features

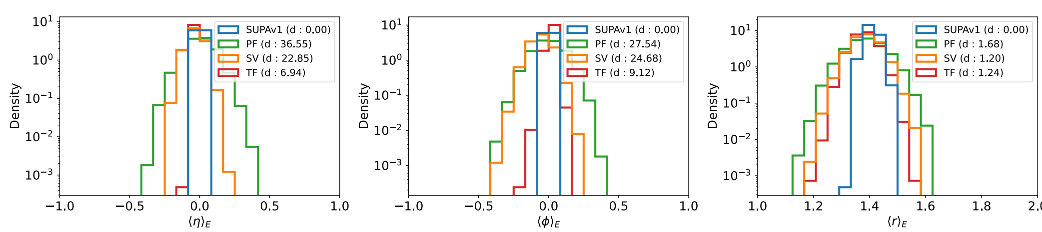


Figure 16: Histograms of energy weighted averages

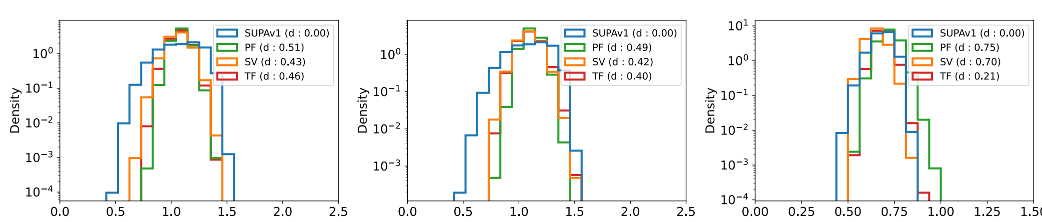
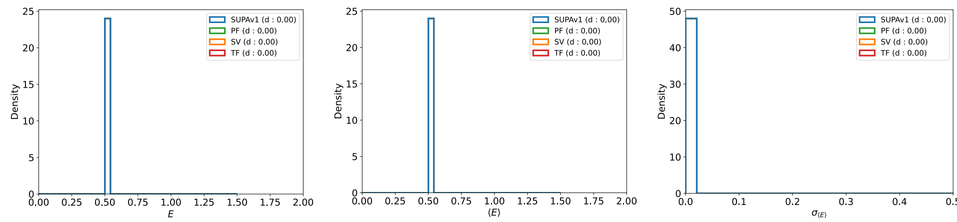
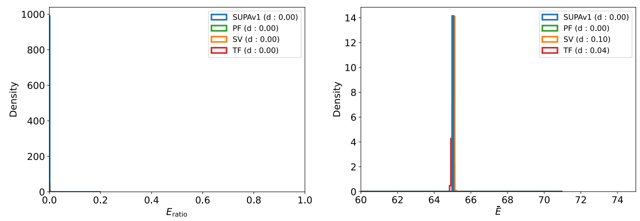


Figure 17: Histograms of lateral widths



(a) Point dist. for Energy feature (b) Energy Mean (c) Energy Variance



(d) Cell Energy Ratio (e) Layer Energy

Figure 18: Histograms of various shower shape variables

A.4.2 SUPAv2

Figs. 19 - 24 show the histograms of various shower shape variables for SUPAv2 and samples generated with PointFlow, SetVAE, and Transflowmer.

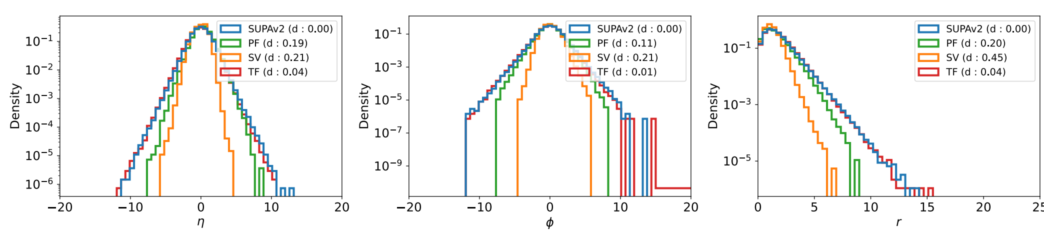


Figure 19: Histograms of point distributions for η, ϕ, and r

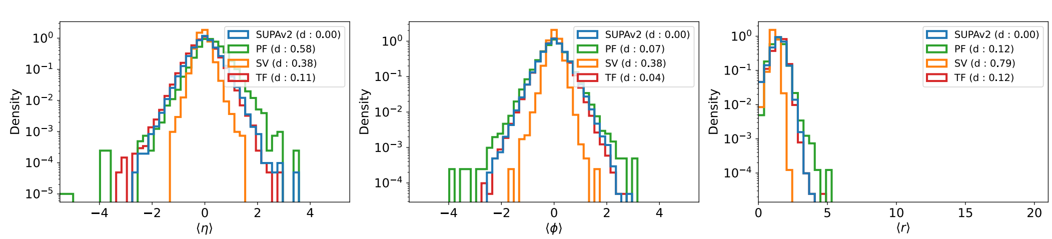


Figure 20: Histograms of sample means for different features

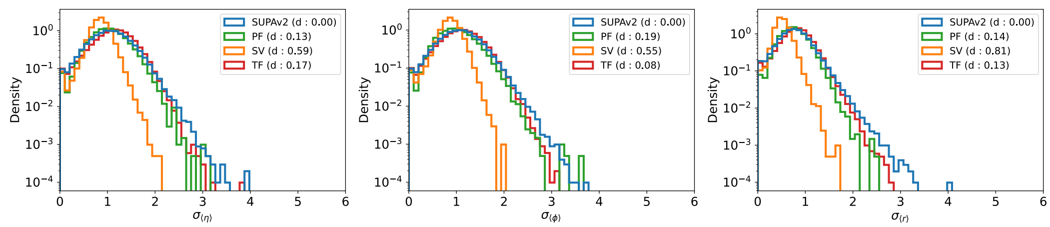


Figure 21: Histograms of sample variance for different features

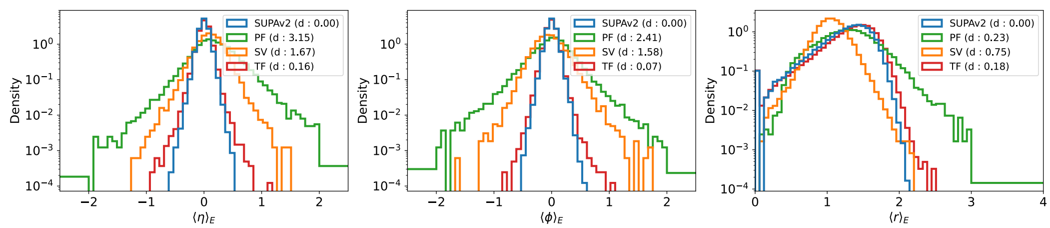


Figure 22: Histograms of energy weighted averages

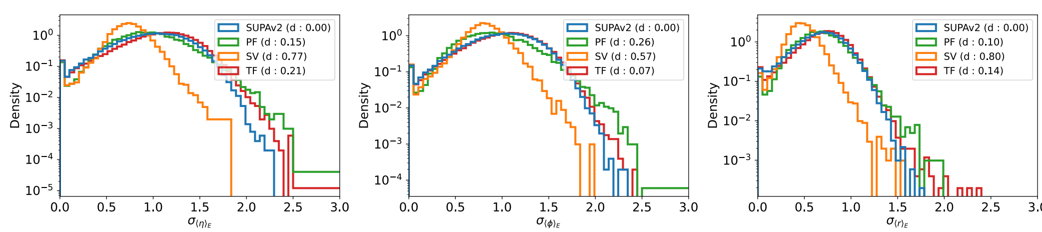
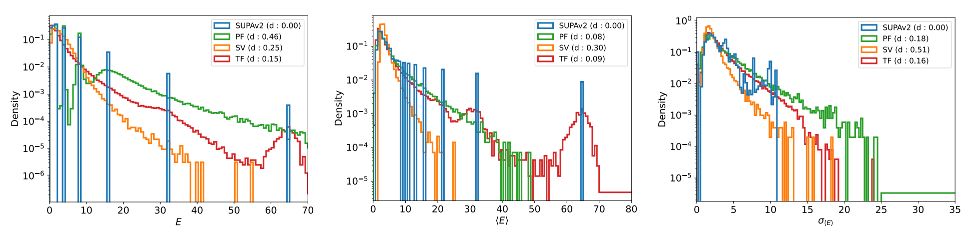
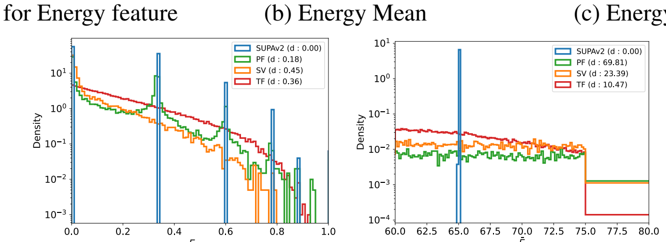


Figure 23: Histograms of lateral widths



(a) Point dist. for Energy feature (b) Energy Mean (c) Energy Variance



(d) Cell Energy Ratio (e) Layer Energy

Figure 24: Histograms of various shower shape variables

A.4.3 SUPAv3

Figs. 25 - 30 show the histograms of various shower shape variables for SUPAv3 and samples generated with PointFlow, SetVAE, and Transflowmer.

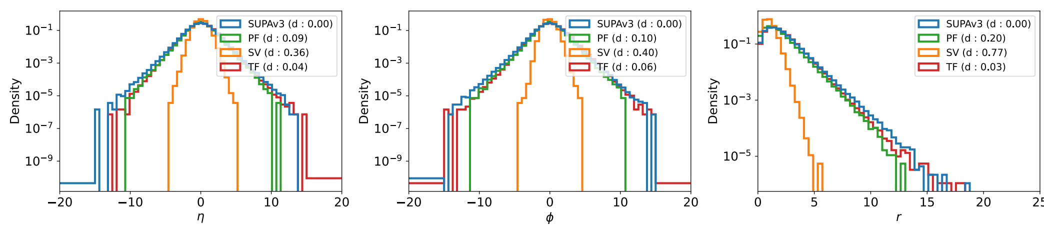


Figure 25: Histograms of point distributions for η, ϕ, and r

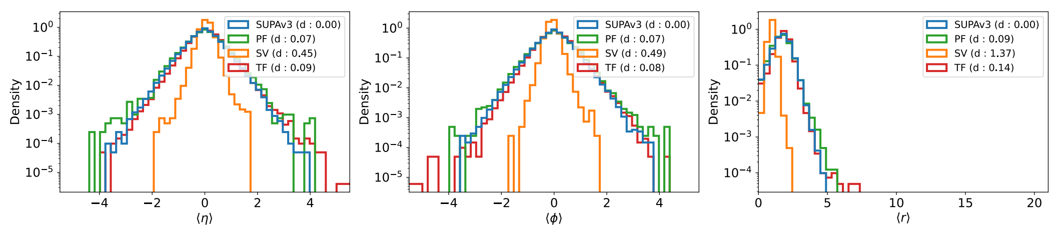


Figure 26: Histograms of sample means for different features

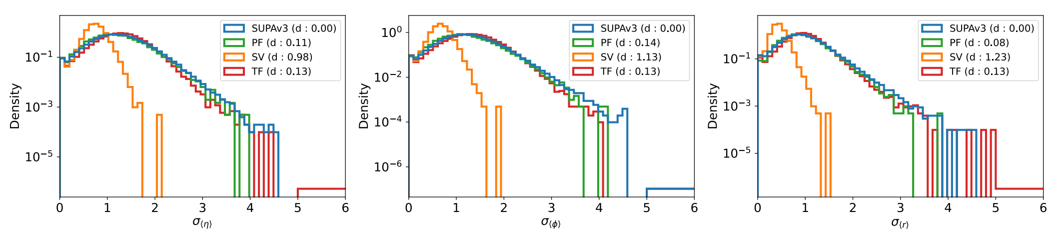


Figure 27: Histograms of sample variance for different features

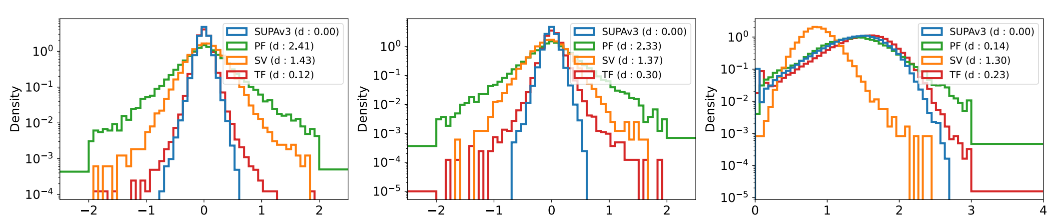


Figure 28: Histograms of energy weighted averages

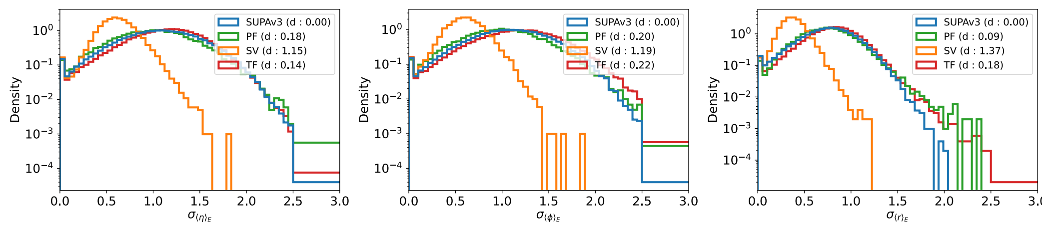
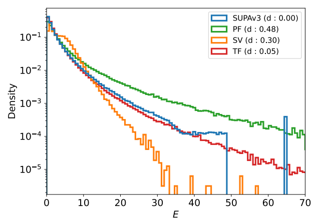
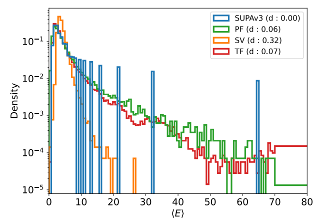


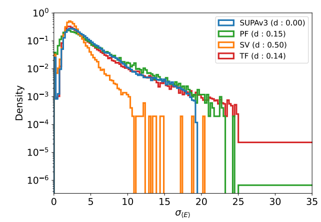
Figure 29: Histograms of lateral widths



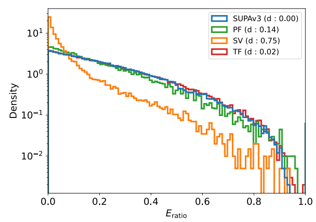
(a) Point dist. for Energy feature



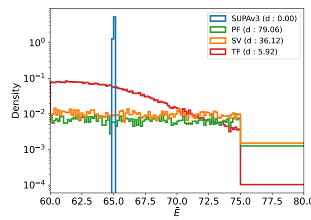
(b) Energy Mean



(c) Energy Variance



(d) Cell Energy Ratio



(e) Layer Energy

Figure 30: Histograms of various shower shape variables

A.4.4 SUPAv4

Figs. 31 - 36 show the histograms of various shower shape variables for SUPAv4 and samples generated with PointFlow, SetVAE, and Transflowmer.

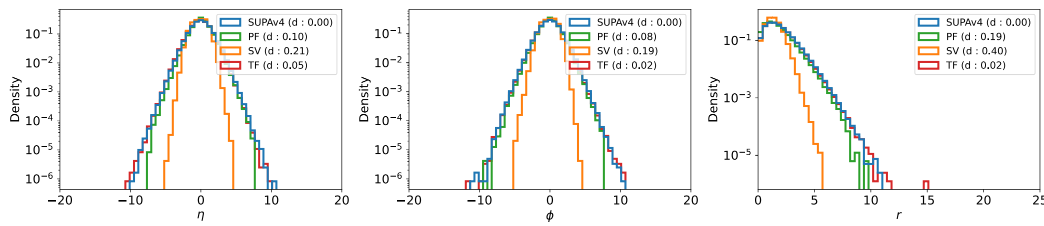


Figure 31: Histograms of point distributions for η, ϕ, and r

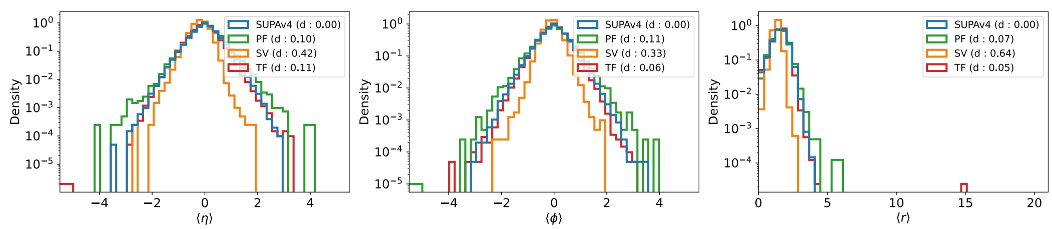


Figure 32: Histograms of sample means for different features

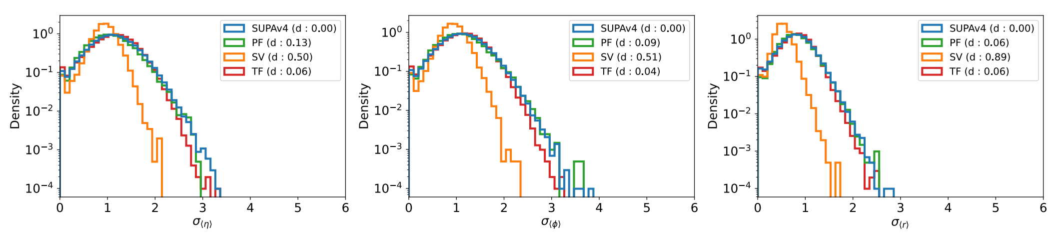


Figure 33: Histograms of sample variance for different features

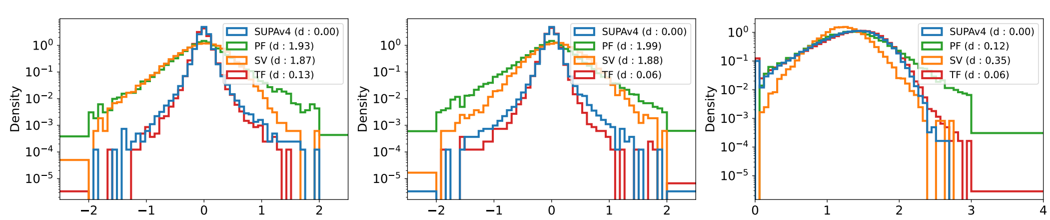


Figure 34: Histograms of energy weighted averages

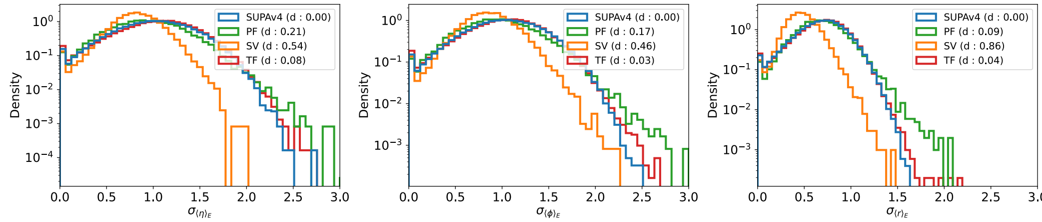
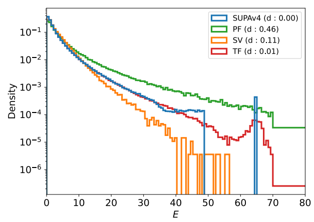
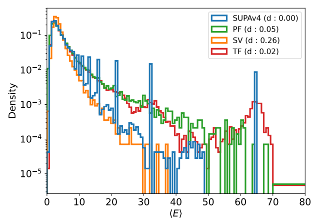


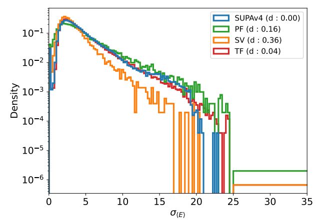
Figure 35: Histograms of lateral widths



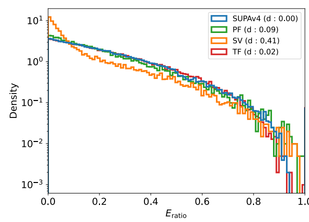
(a) Point dist. for Energy feature



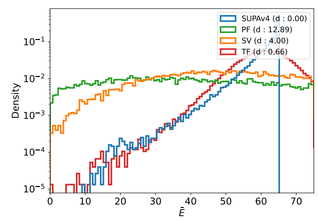
(b) Energy Mean



(c) Energy Variance



(d) Cell Energy Ratio



(e) Layer Energy

Figure 36: Histograms of various shower shape variables

A.4.5 SUPAv5

We only consider layer 0 for SUPAv5. Figs. 37 - 42 show the histograms of various shower shape variables for SUPAv5 and samples generated with PointFlow, SetVAE, and Transflowmer.

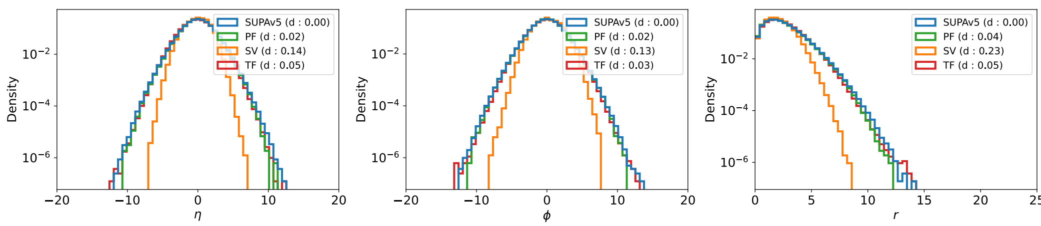


Figure 37: Histograms of point distributions for η, ϕ, and r

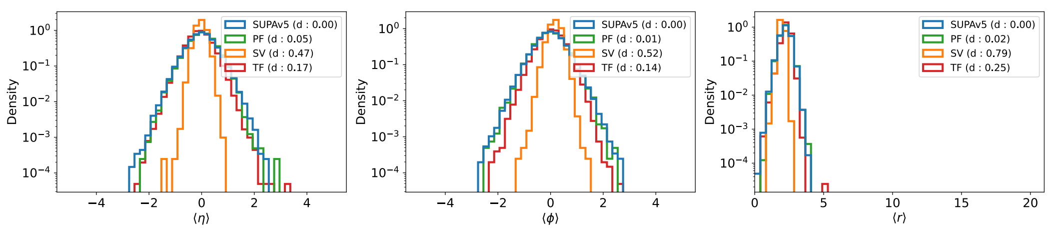


Figure 38: Histograms of sample means for different features

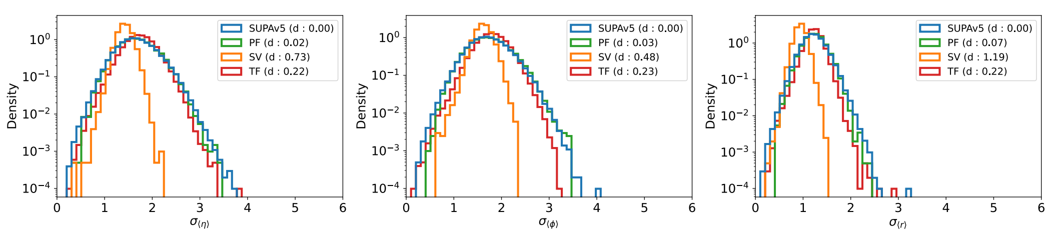


Figure 39: Histograms of sample variance for different features

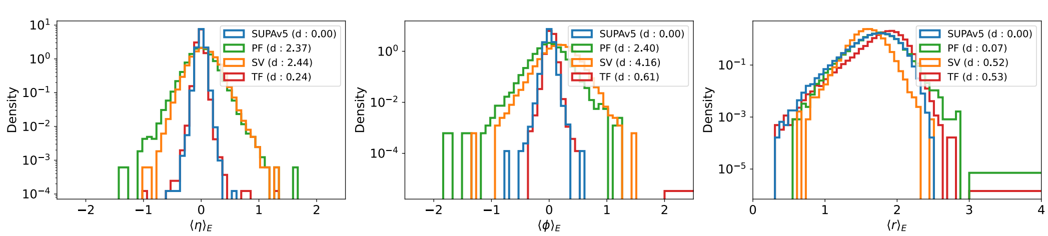


Figure 40: Histograms of energy weighted averages

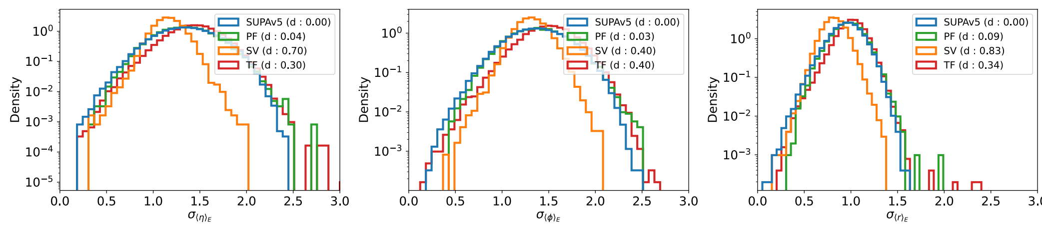
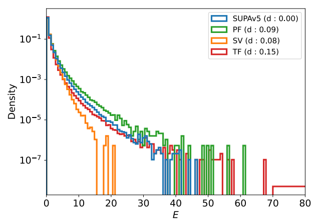
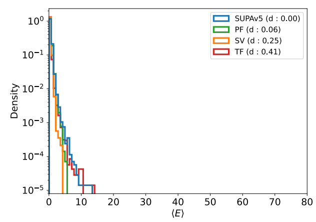


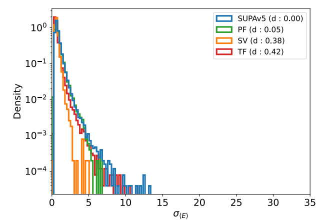
Figure 41: Histograms of lateral widths



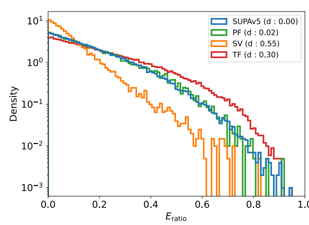
(a) Point dist. for Energy feature



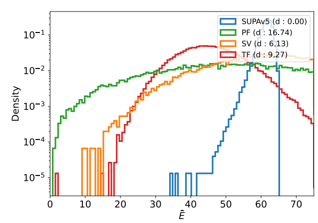
(b) Energy Mean



(c) Energy Variance



(d) Cell Energy Ratio



(e) Layer Energy

Figure 42: Histograms of various shower shape variables

A.5 Experiments on grid representation of data

In this section we will present some studies on generative modeling with the grid representation of data from SUPA. We discuss about how to downsample the point clouds below. For these studies, we generated another version of the dataset with SUPA such that it is similar to the CALOGAN dataset, i.e., with three layers and downsampled to a resolution in the multiples of 3×96, 12×12, and 12×6, for layer 0, 1, and 2, respectively.

Downsampling. For comparison, we downsample the point clouds to their corresponding image representation (see Figure 1) by first defining the region of interest i.e. a rectangular region for each layer and the number of bins/cells/pixels in both the horizontal (or η) and vertical (or ϕ) directions. Finally, for each cell, we sum the energy of all the points falling within it to get the pixel intensity. We can increase the number of cells in order to get higher resolutions. Figure 1b, 1c, and 1d show the downsampled image representations at resolutions of 3x, 2x and 1x respectively for the shower shown in Figure 1a. We choose 1x to be the same resolution as used in CaloGAN [Paganini et al., 2018] (i.e. 12×12 for Layer 1).

A.5.1 Validity of SUPA as a benchmark with grid representation

We show the comparison of performance of generative models trained over data generated with SUPA and Geant4 in § 5.3. In this section, we extend those studies with more analysis and plots. Figure 43 shows the scatter plot of the average ranks of those models. The average rank for a model on a dataset is obtained by first ranking them with respect to each marginal's discrepancy and then averaging over all the marginals.

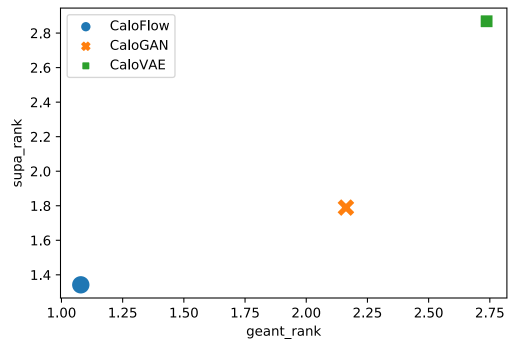


Figure 43: Scatter Plot for ranks over different models. Ranking of the models are consistent over both, SUPA and GEANT4, showing the validity of SUPA as a benchmark.

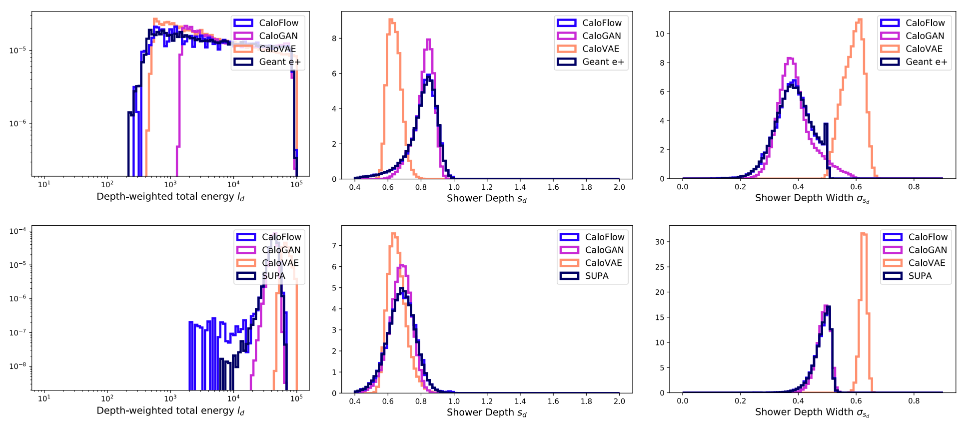


Figure 44: Histogram for various marginals for GEANT4 e+ (top) and SUPA (bottom) vs. showers generated from different trained models

Further, in Figures 44-49, we show a subset of the marginals (see § 5 for a detailed explanation on the marginals and Paganini et al. [2018] for the grid representation based marginals) for GEANT4 and SUPA and also the showers generated with different models trained on them. These marginal plots illustrate the diversity in various distributions present in data from GEANT4, and, more importantly in SUPA. Further, the distributions of the generated showers from different models behave similarly on both datasets, reinstating the proposition that a better model on SUPA implies a better model on the detailed GEANT4.

A.5.2 High-resolution experiments

In this section, we show the utility of SUPA beyond using it for training at low resolution (similar to the resolution used in CaloGAN, which we call 1x), as well as the limitation of the current models.

We train CaloFlow [Krause and Shih, 2021] with SUPA by downsampling the point clouds at the higher resolutions of 2x and 3x. Table 4 shows the mean discrepancy metric (see § A.5.1) for the models. We observe the trend that training at higher resolutions result in poorer performance (diagonal terms) in general. Further, when the generated samples from the trained models are downsampled to 1x, the performance deteriorates as compared to samples generated from models trained directly with data at 1x resolution.

A.6 Extended Results.

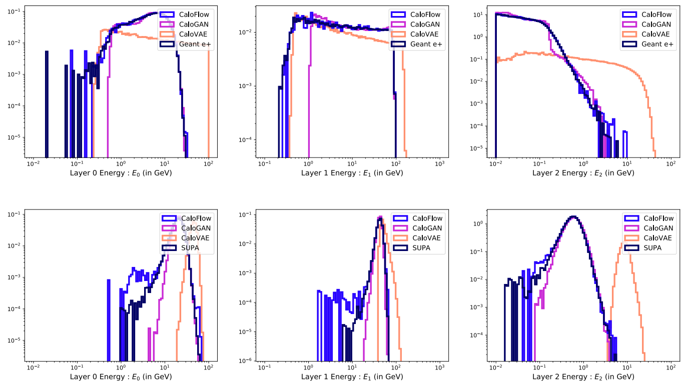


Figure 45: Histogram for Layer Energy for GEANT4 e+ (top) and SUPA (bottom) vs. showers generated with different trained models.

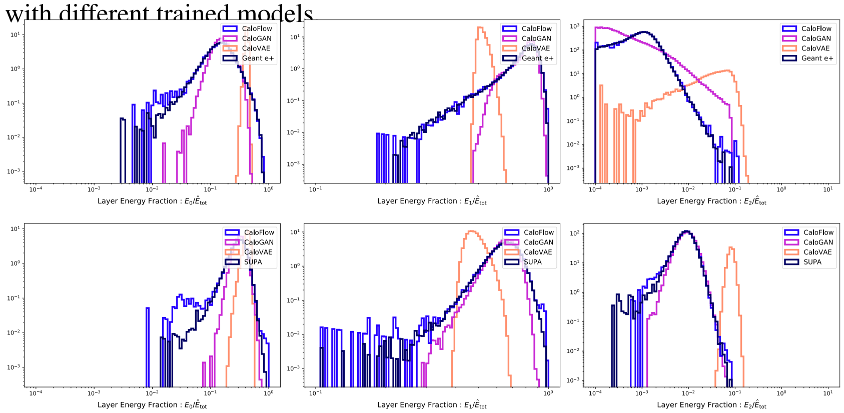


Figure 46: Histogram for Layer energy fraction for GEANT4 e+ (top) and SUPA (bottom) vs. showers generated with different trained models.

	1x	2x	3x
1x	3.57	6.35	7.20
2x	-	6.78	-
3x	-	-	8.29

Table 4: Mean discrepancy metric (see § A.5.1) for CaloFlow model when trained and tested over different resolutions. Columns correspond to the training resolution and rows to the test resolution. The results on the diagonal show that CaloFlow's performance degrades when resolution increases, and the top row shows that it is not simply due to the sheer dimensionality of the signal since the model does not leverage structure at high resolution to perform better at low resolution.

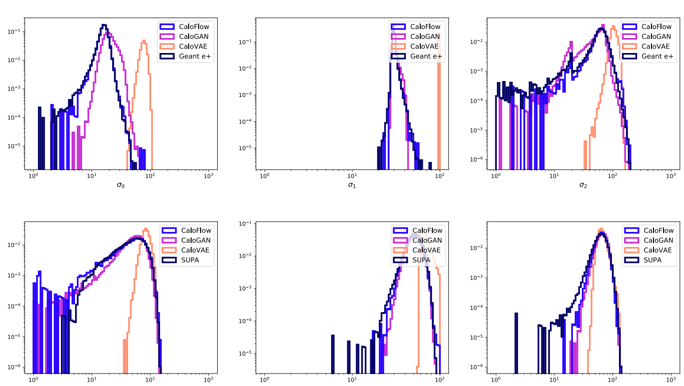


Figure 47: Histogram for Layer lateral width for GEANT4 e+ (top) and SUPA (bottom) vs. showers generated with different trained models.

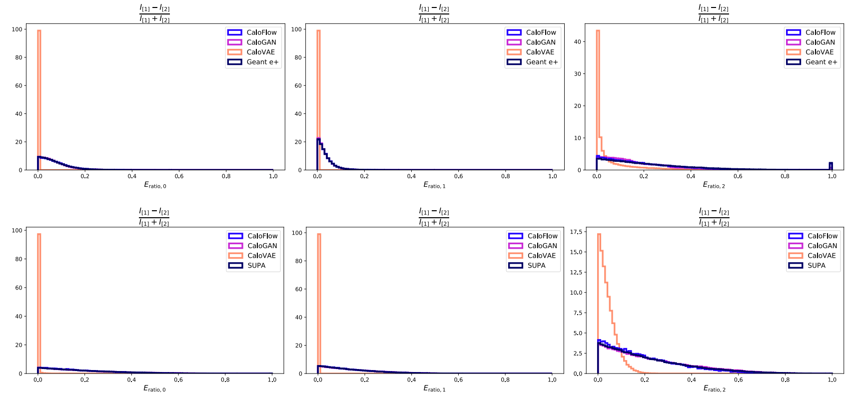


Figure 48: Histogram for $E_{\text{ratio},i}$ for GEANT4 e+ (top) and SUPA (bottom) vs. showers generated with different trained models.

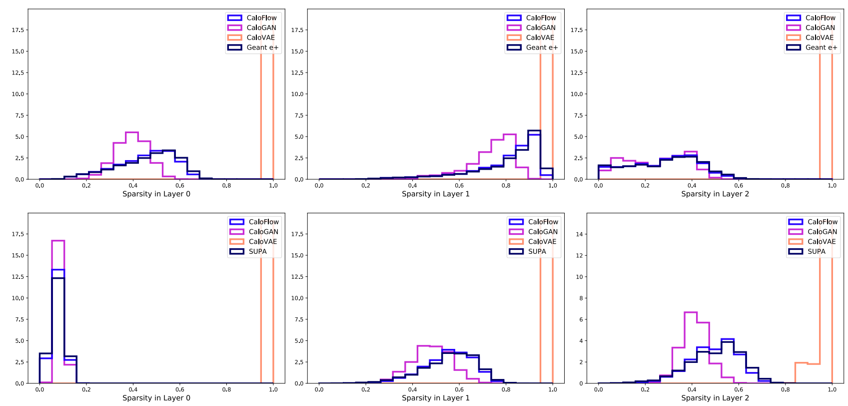


Figure 49: Histogram for Layer sparsity for GEANT4 e+ (top) and SUPA (bottom) vs. showers generated with different trained models.

Table 5: Performance benchmarks across different datasets with SetVAE, PointFlow and Transflowmer. The distance metric is Wasserstein-1. The reported numbers are averages over a group of marginals as indicated in the top row. Lower numbers are better.

	$\sigma_{\langle\eta_i\rangle}, \sigma_{\langle\phi_i\rangle}, \sigma_{\langle r_i\rangle}$			$\langle E\rangle$			$\sigma_{\langle E\rangle}$		
Dataset	SV	PF	TF	SV	PF	TF	SV	PF	TF
SUPAv1	0.513	0.585	**0.359**	0.001	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
SUPAv2	0.648	0.154	**0.130**	0.302	**0.077**	0.087	0.513	0.177	**0.165**
SUPAv3	1.114	**0.109**	0.126	0.320	**0.064**	0.071	0.500	0.152	**0.144**
SUPAv4	0.634	0.092	**0.051**	0.263	0.047	**0.022**	0.355	0.156	**0.038**
SUPAv5	0.799	**0.040**	0.223	0.251	**0.059**	0.414	0.377	**0.047**	0.421

	$\langle\eta_i\rangle_E, \langle\phi_i\rangle_E, \langle r_i\rangle_E$			$\sigma_{\langle\eta_i\rangle_E}, \sigma_{\langle\phi_i\rangle_E}, \sigma_{\langle r_i\rangle_E}$			$\bar{E}$		
Dataset	SV	PF	TF	SV	PF	TF	SV	PF	TF
SUPAv1	16.244	21.921	**5.766**	0.517	0.591	**0.379**	0.101	**0.000**	0.039
SUPAv2	1.336	1.933	**0.137**	0.779	**0.134**	0.190	23.387	69.814	**10.468**
SUPAv3	1.365	1.627	**0.217**	1.226	**0.147**	0.151	36.116	79.062	**5.916**
SUPAv4	1.369	1.346	**0.083**	0.645	0.169	**0.067**	3.997	12.895	**0.659**
SUPAv5	2.373	1.615	**0.463**	0.740	**0.058**	0.317	**6.132**	16.742	9.273