Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

FLAGSHIP: Enhancing Geant4 Monte Carlo Simulations through Machine Learning Integration

Project Status May 2025: Variational Autoencoder (VAE)

Gallo G., Cirrone G.A.P., Fattori S., Ientile V., Sciuto A., Tricomi A.

Spoke2 – WP6 meeting
May 07, 2025

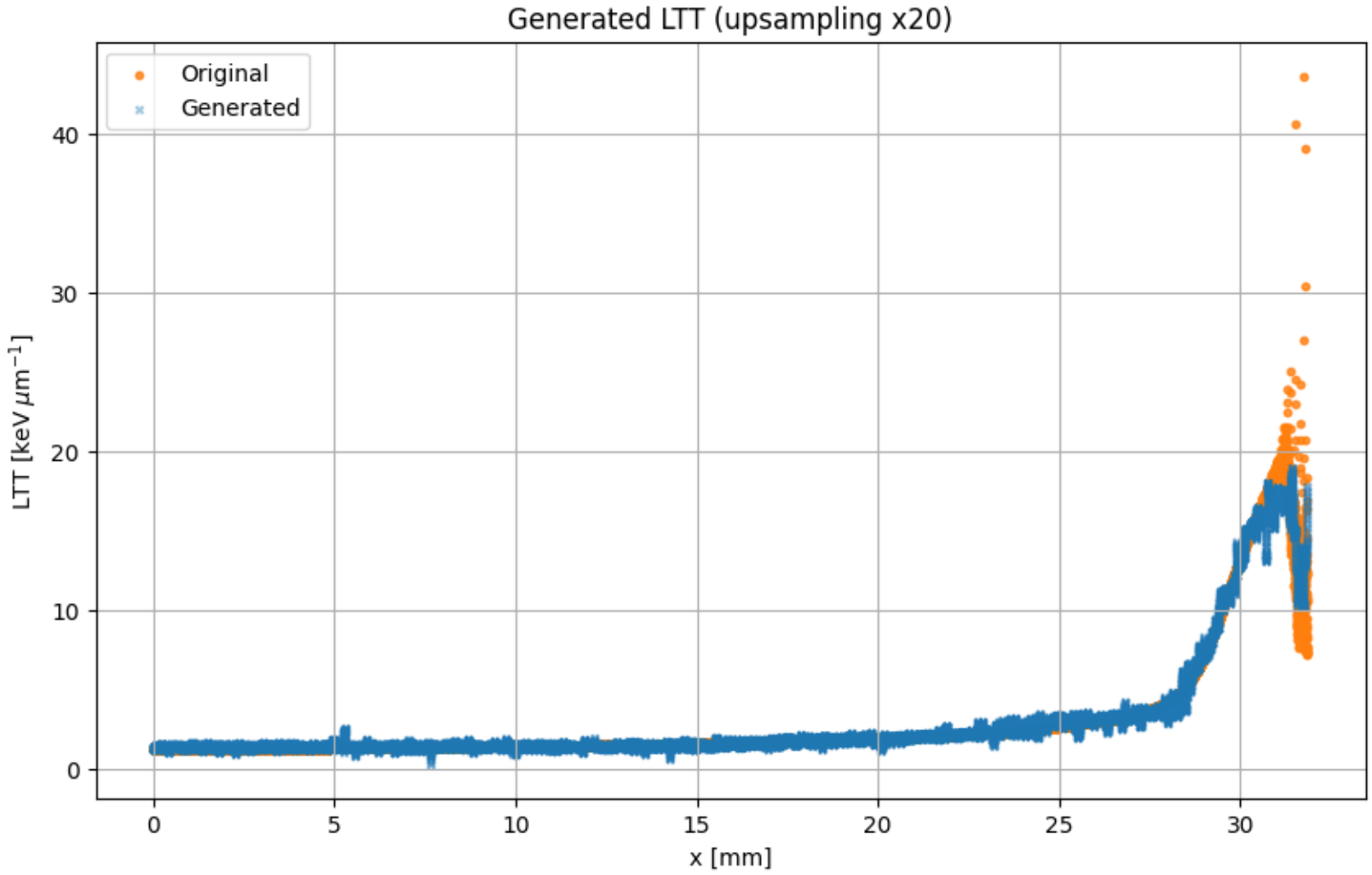
Project Summary

- **Objective**: Leverage a Variational Autoencoder (VAE) to generate high-resolution synthetic data for radiological features.
- **Task**: Address the *super-resolution* challenge — upsample Geant4 simulation results, e.g. from 1 mm to 1 μm granularity.
- **Approach**: Fill data gaps in sparse simulation output using learned feature distributions.
- **Expected Outcome**: Detailed, high-resolution spatial data that aligns with coarse simulation data.

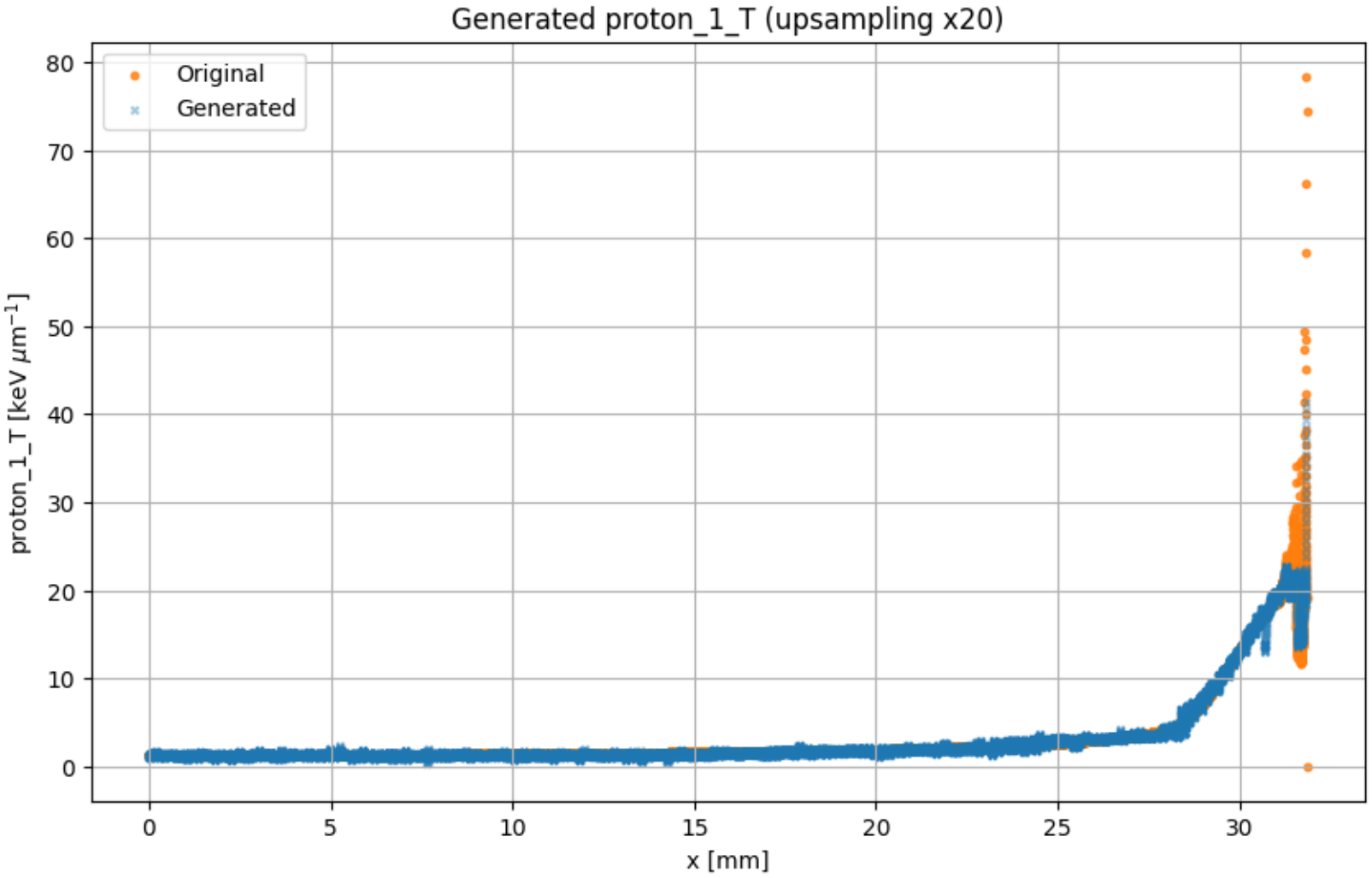
Current Status

- **Refactoring in progress**: `refactor/shared-utils-structure` branch. Remove duplicate code, such as similar utility-style functions that are implemented repeatedly in different classes or scripts, to reduce maintenance.
- **Implemented generative inference**: `src/vae_generate.py`.
- **Implemented optimization pipeline**: `src/vae_optimization.py`.
- Generated data upsampled by a factor of ×20.

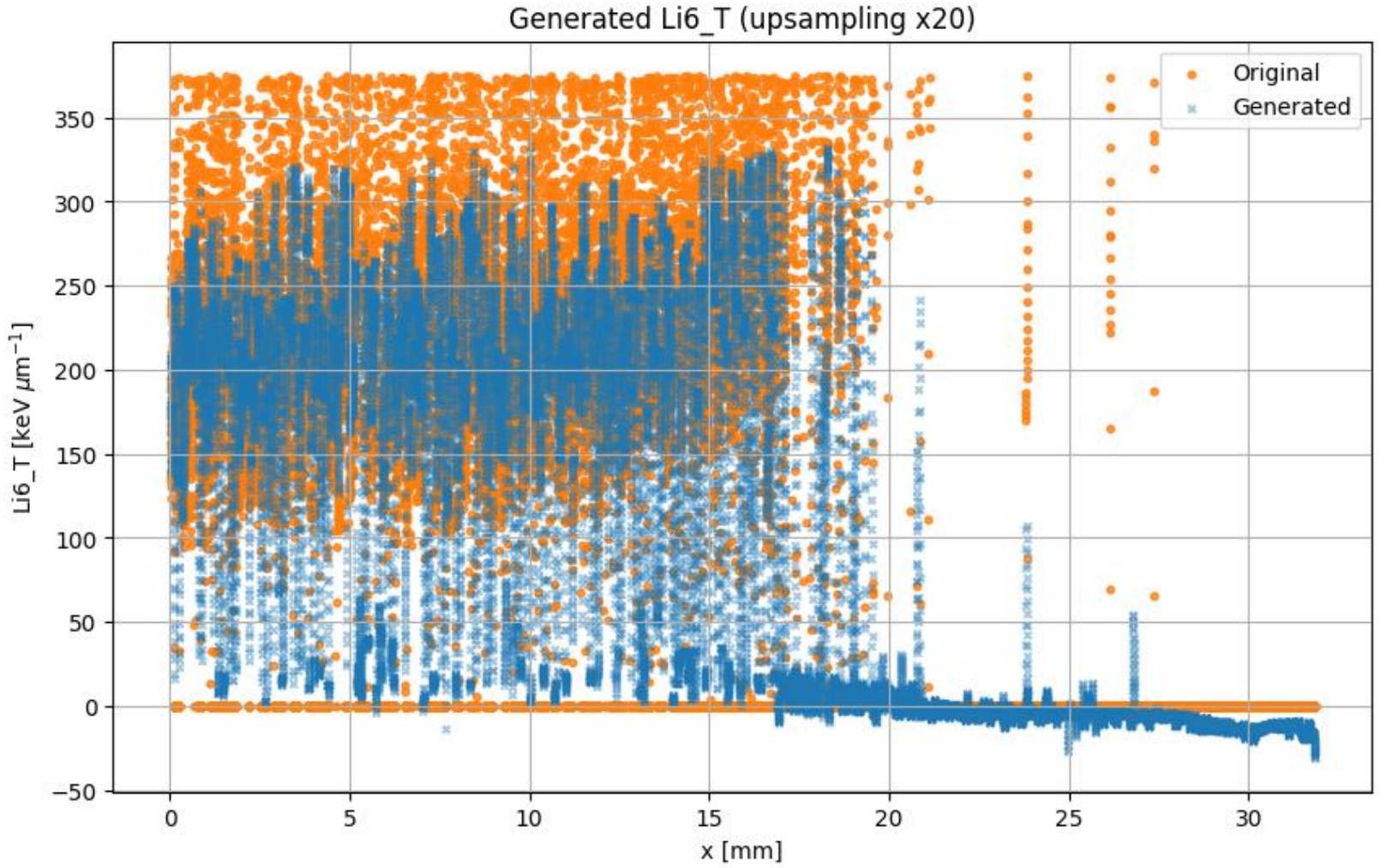
*Comparison of original (orange) and generated **total LET distributions** (blue).*
Granularity: - original data 20 um; generated data 1 um.



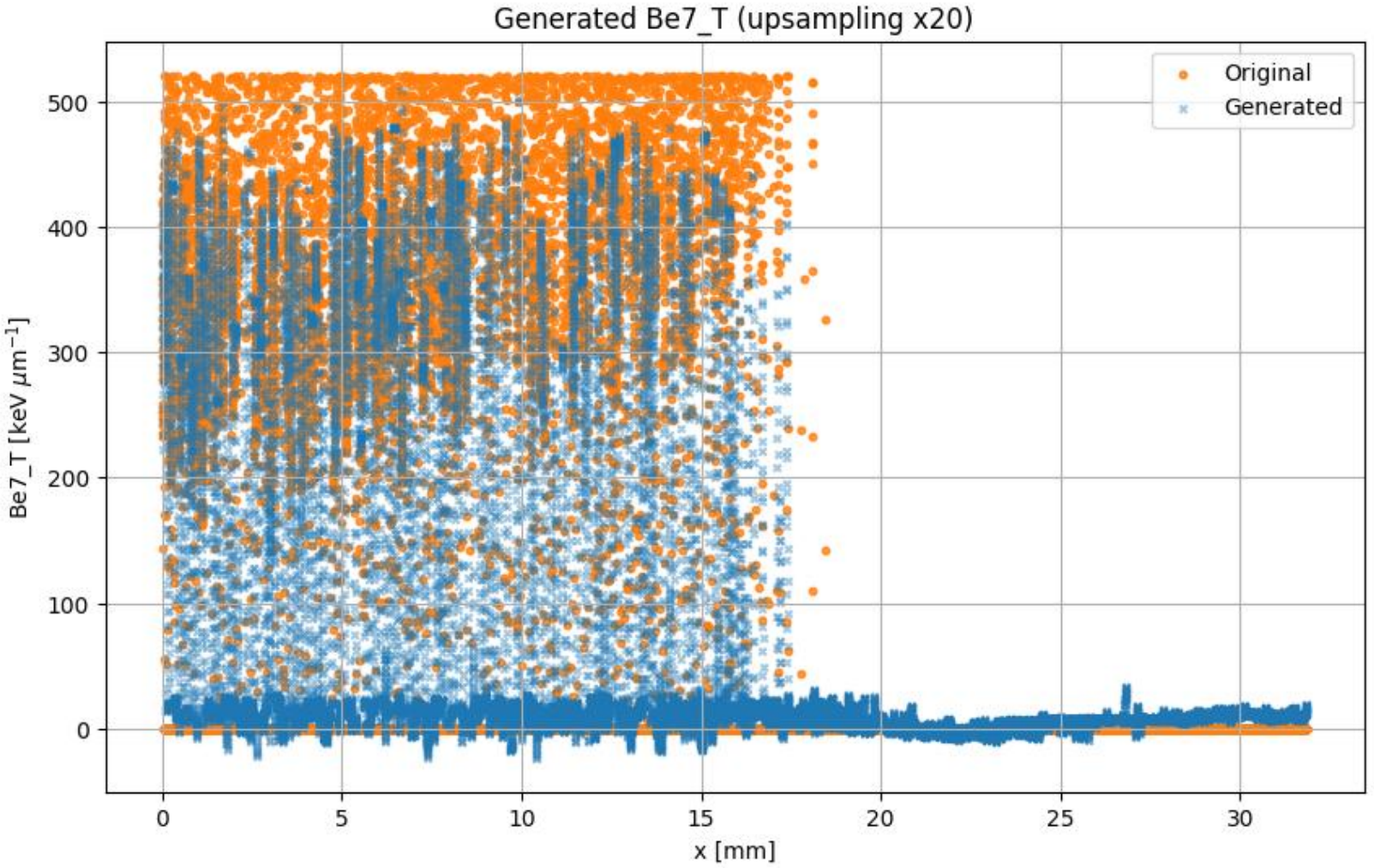
*Comparison of original (orange) and generated **primary proton LET distributions** (blue).*
Granularity: - original data 20 um; generated data 1 um.



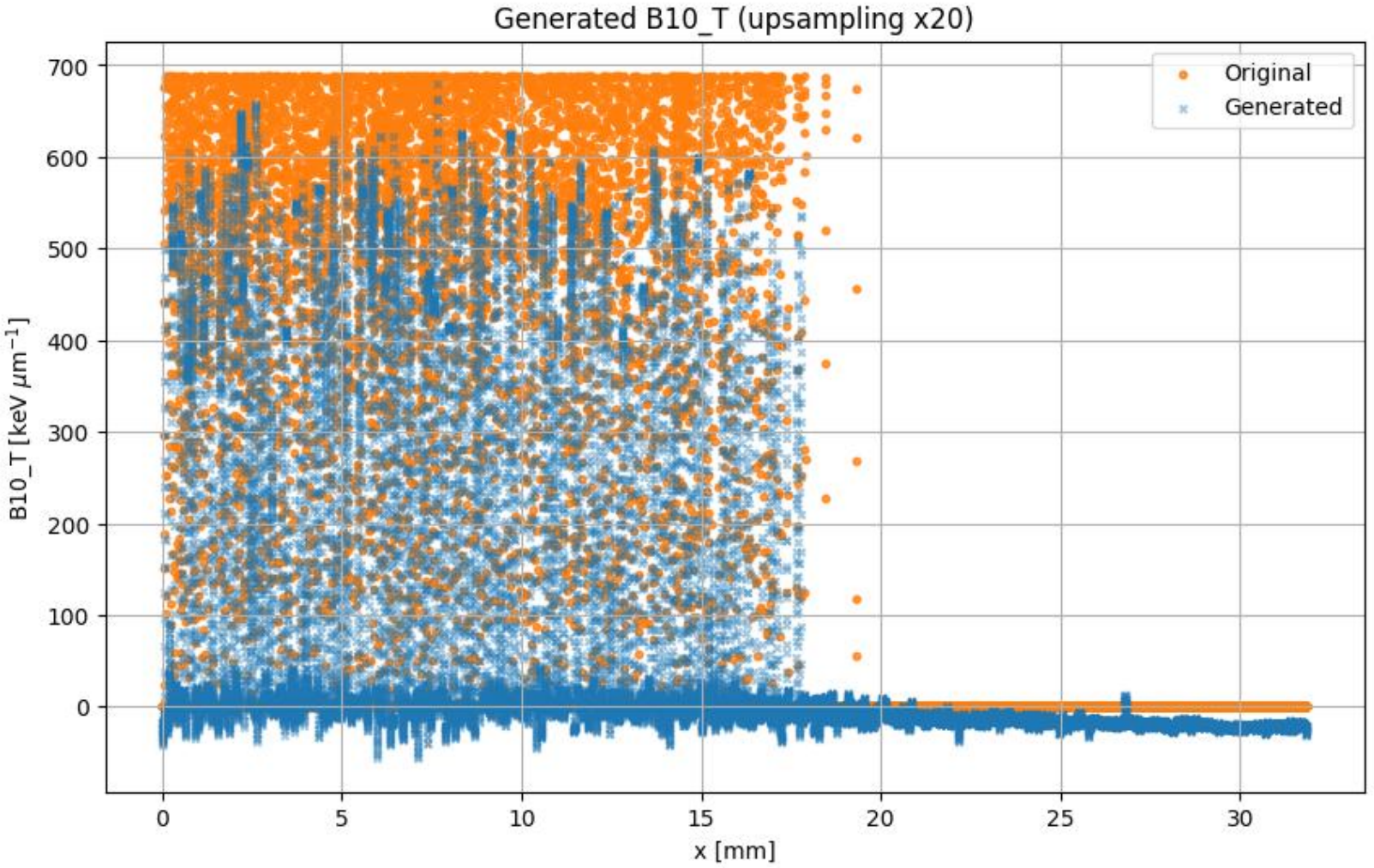
Comparison of original (orange) and generated ***^{6}Li LET distributions*** *(blue).*
Granularity: - original data 20 um; generated data 1 um.



Comparison of original (orange) and generated ***^{7}Be LET distributions*** *(blue).*
Granularity: - original data 20 um; generated data 1 um.



Comparison of original (orange) and generated ***^{10}B LET distributions*** *(blue).*
Granularity: - original data 20 um; generated data 1 um.



Observations

- Generated data generally mimics the structure of original distributions.
- Variation exists across different feature types.
- VAE performance consistent for LTT and proton_1_T; more variance in Li6_T, Be_7_T, B_10_T.
- Unexpected Result: Negative LET values have no physical meaning.

Challenges & Risks

- Edge cases in data generation require further evaluation.
- Sparsity in certain regions may affect fidelity.
- Need for thorough quantitative metrics.

Next Steps

- Finalize refactor of shared utility structure.
- Public release v2.0.
- Perform quantitative validation of generated data.
- Test higher upsampling factors.
- Force the VAE to generate strictly positive output.
- Run optimization on ICSC HPC resources.

Optimization Pipeline Summary

- Hyperparameter tuning using Optuna.
- Objective: Minimize reconstruction loss for improved generative quality.

Tuned Hyperparameters in VAE Optimization

🧠 Network Architecture:

- `num_layers` – Number of hidden layers
- `layer_0_size, layer_1_size, ...` – Hidden layer sizes
- `latent_dim` – Size of latent space
- `use_dropout` – Wheter to apply dropout (can be set to default)
- `dropout_rate` – Dropout probability

📤 Output Layer (Exit Activation):

- `use_exit_activation` - Toggle for activation on the output (Can be set to default)
- `exit_activation_type` – One of: shifted_softplus, elu_offset, `pelu`
  - If `shifted_softplus`: `beta_softplus`
  - If `elu_offset`: `offset_init`
  - If `pelu`: `a_init, b_init`

Tuned Hyperparameters in VAE Optimization

Loss Function:

- `loss_type` - One of: `standard`, `inverse`, `sigmoid`
- `use_neg_penalty` – Apply penalty on negative values
- `neg_penalty_weight` – Penalty scaling factor
- `beta` – Loss coefficient
- `beta_scale` – Loss scale (inverse/sigmoid)
- `recon_target` – Target for sigmoid loss

Optimizer:

- `optimizer` – One of: `Adam`, `RMSprop`
- `learning_rate` – Optimizer learning rate
- `weight_decay` – Weight decay regularization

Batch Processing:

- `batch_size` – Batch size used during training

VAE Optimization Test

The user can also set other parameters through a configuration file.

- The type of data scaler and its parameters.
- Number of Optuna trials.
- `val_loss_threshold` is checked during the model's fit() method (in AutoEncoder) to:
  - Decide whether to prune a trial (e.g., with Optuna)
  - Warn or stop early if validation loss is too high

```
"scaler": {
  "type": "standard",
  "with_mean": false,
  "with_std": true
},
"n_trials": 500,
"val_loss_threshold": 2,
"identity_features": ["x"],
"default_training_epochs": 100,
```

Best Model Summary (Optimization Test)

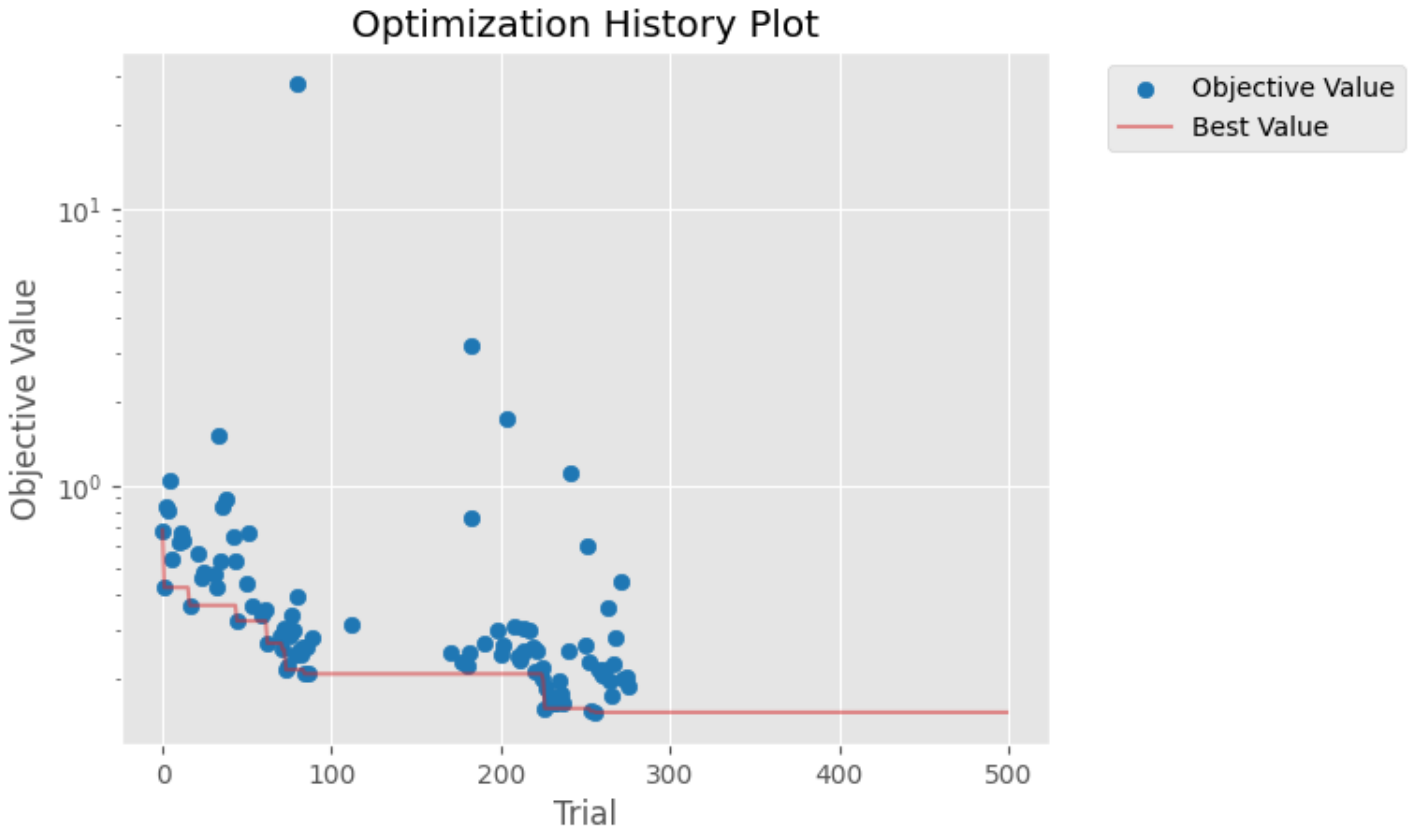
🧠 Model Architecture: AutoEncoder

Component	Input/Output Shape	Layers	Parameters
Encoder	[36, 24]	Linear → BatchNorm1d → Dropout → ReLU (x1), Linear (x2)	14,316
Decoder	[24, 36]	Linear → BatchNorm1d → ReLU, Linear → PELU	10,370
Total Params	—	—	**24,686**

📏 Model Characteristics:

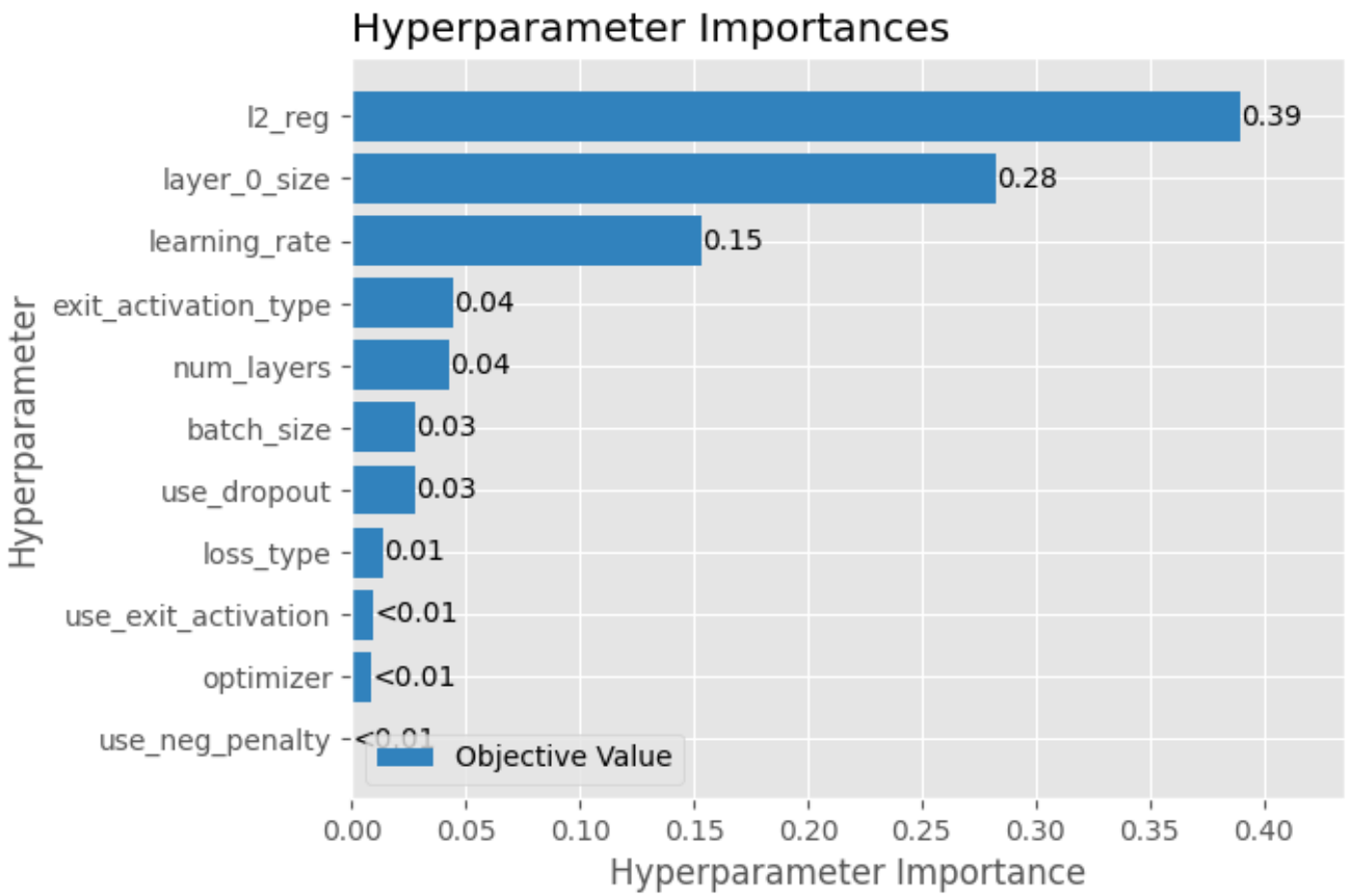
- Input Features: 37 (36 + 1 identity)
- Latent Dimension: 24
- Hidden Layer Size: 164
- Nonlinearities: ReLU (encoder), PELU (decoder output)
- Dropout + BatchNorm applied

Optimization History

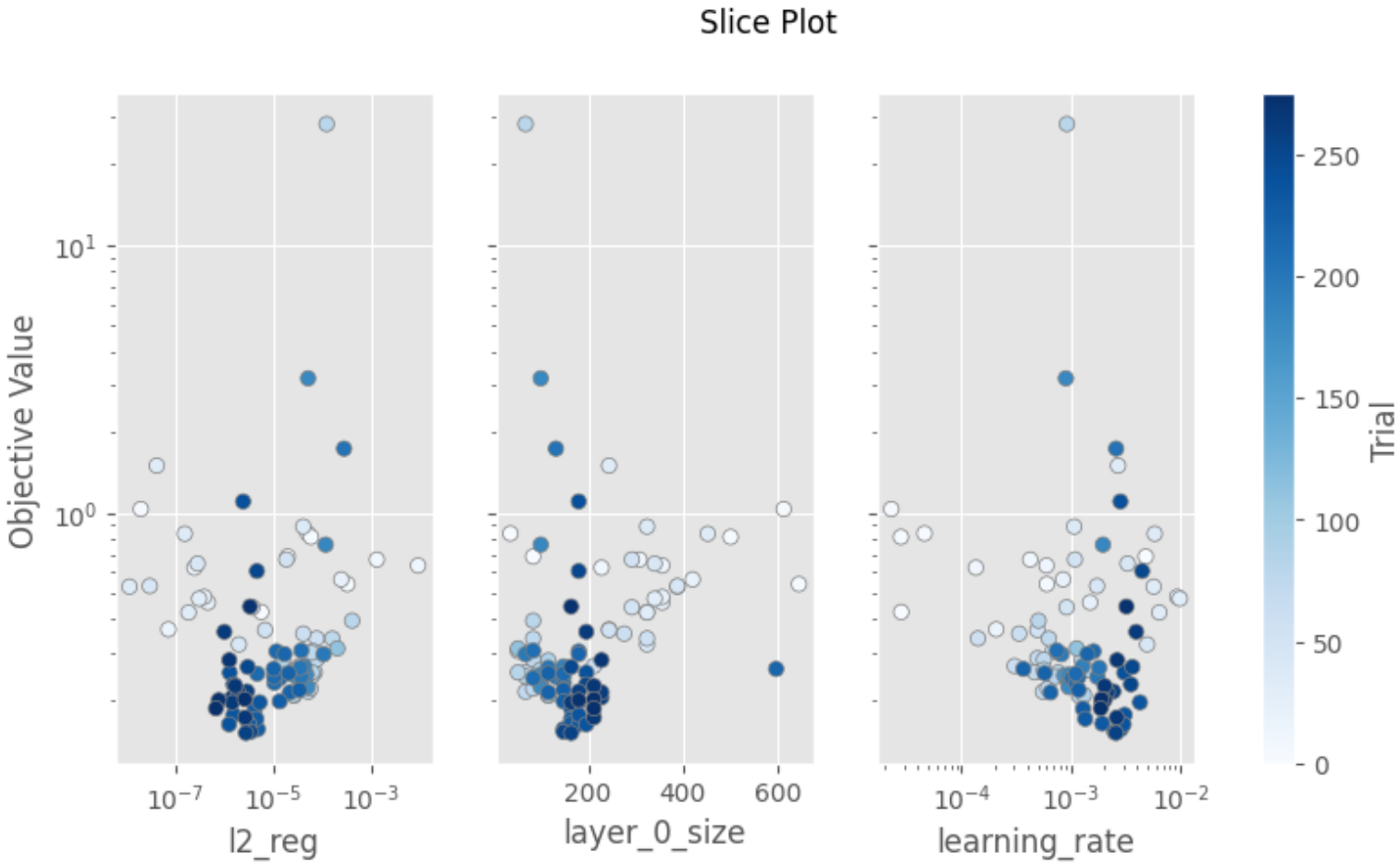


Gradual convergence observed, best values stabilized after ~200 trials

Hyperparameter Importance

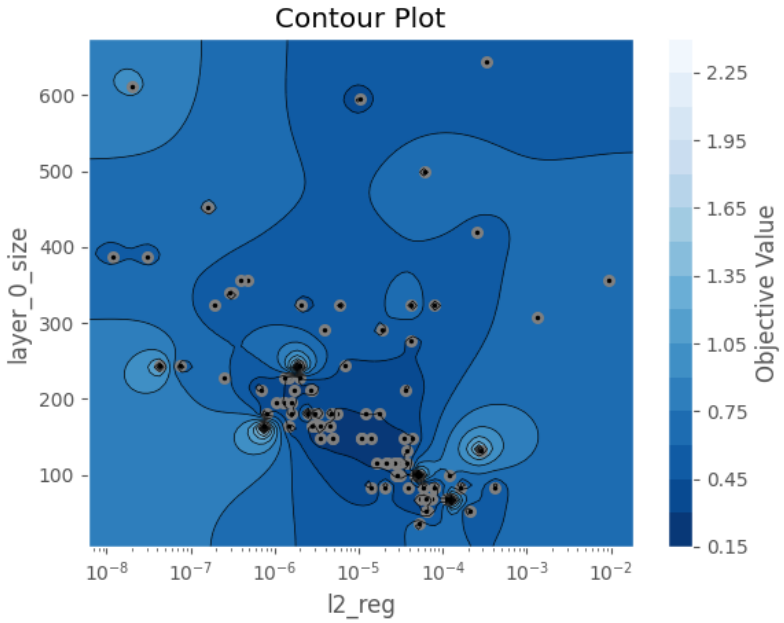


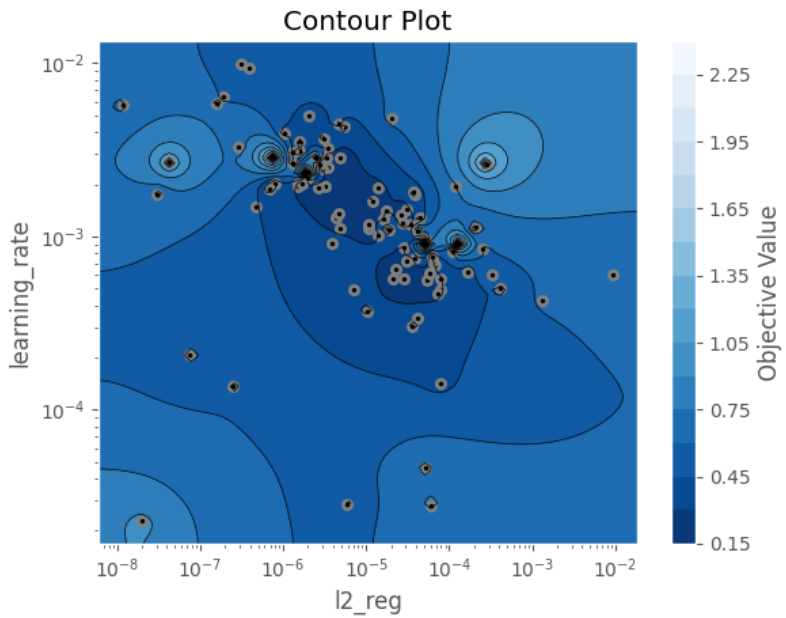
Top parameters

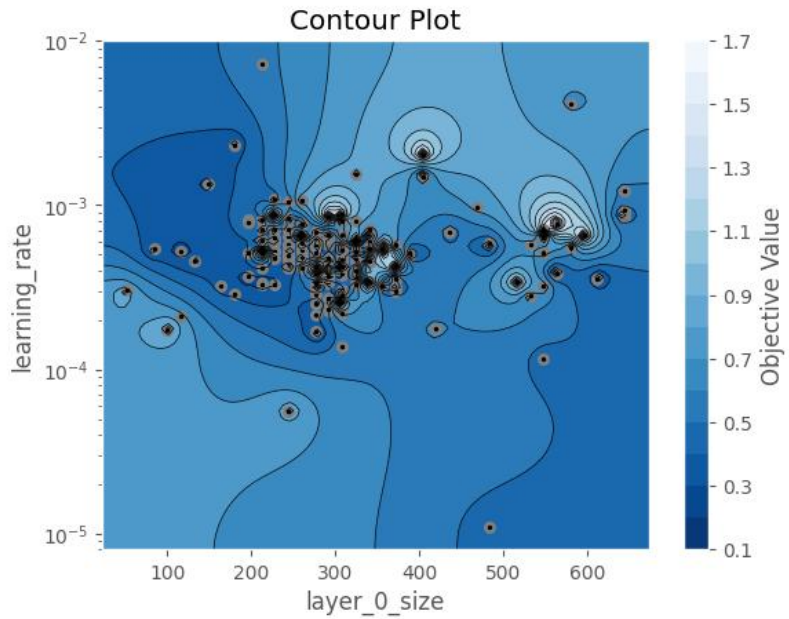


Best performance clusters in specific ranges

Top parameters

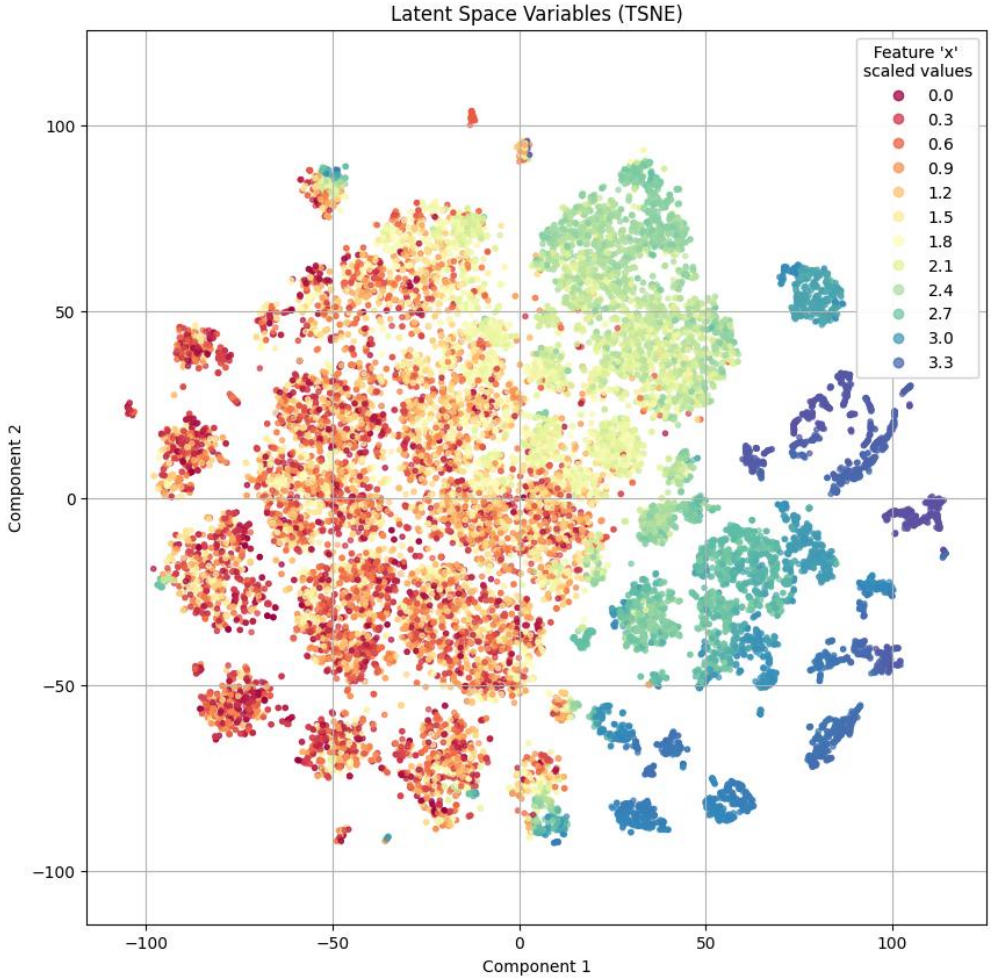






Observation: Nonlinear interactions across top parameters

Latent Space Structure (t-SNE Projection)



Visualization Summary

- Each point represents a data sample projected into 2D from the VAE's latent space using t-SNE.
- Colors correspond to scaled values of feature 'x'.

Interpretation

- The latent space shows clear structure: samples with similar 'x' values cluster together.
- A smooth gradient of colors from left to right indicates that the latent space encodes feature 'x' in an organized, continuous manner.
- Distinct clusters (especially on the right) suggest that the model captures non-linear separations related to 'x'.

Implication

- The VAE latent space effectively preserves semantic similarity and is well-suited for generative tasks like interpolation or conditional synthesis based on 'x'.

Reconstruction Performance