Physics-Informed Neural Networks using Curriculum Learning

Amel Vatic and Serge Kotchourko, 01.02.2024

Background

Background

What are PINNs?

PINNs are a class of neural networks which respect in their output the governing laws of physics described through differential equations. This can be achieved by introducing physical domain knowledge by PDEs into the network using the loss function [2].

Background

What is curriculum learning in the context of PINNs?

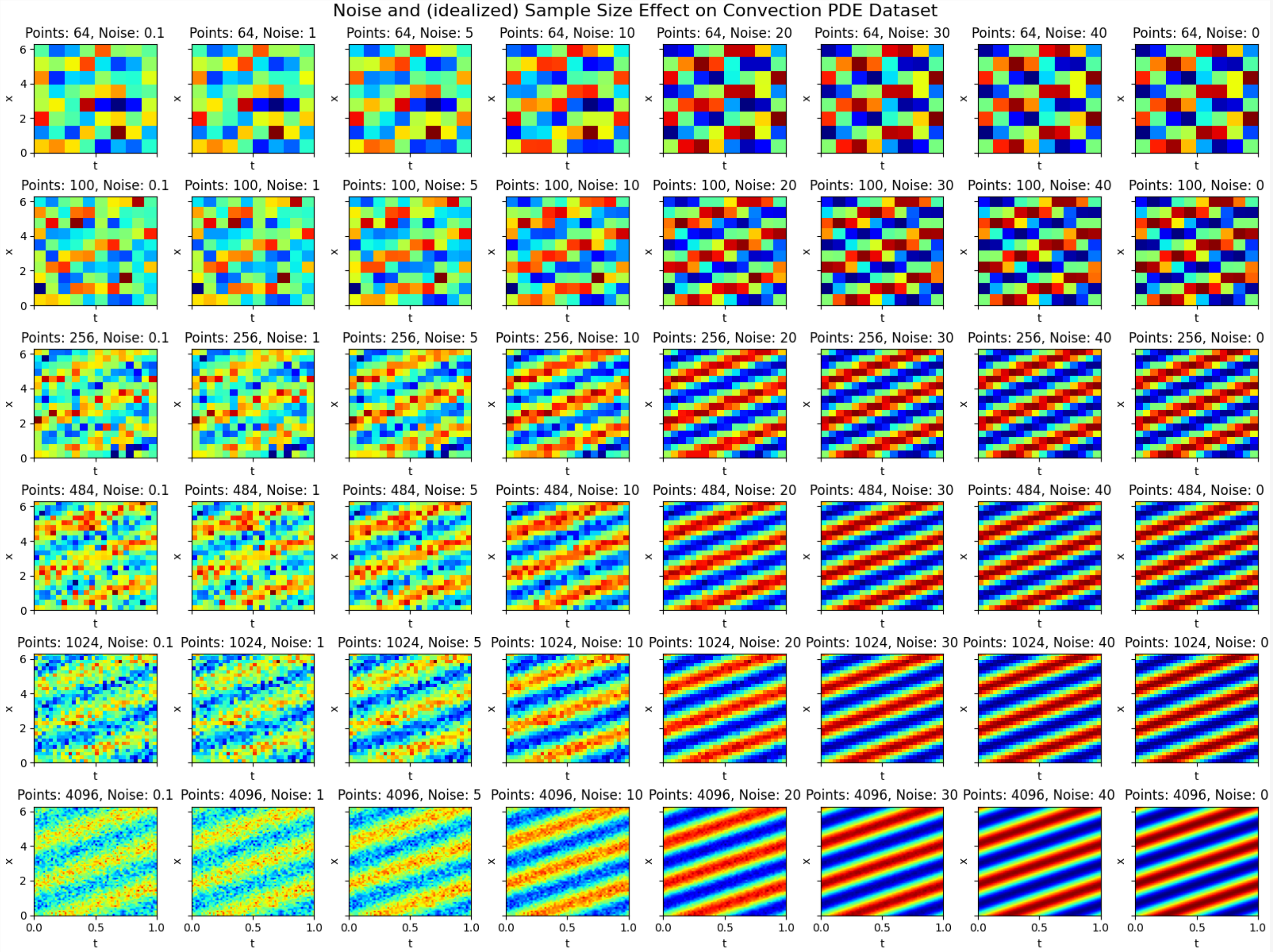
Within this context, curriculum learning is a training strategy that gradually increases the complexity of the training data, allowing the network to train on simpler examples first and then gradually increasing the complexity [1].

Motivation

Neural networks are data-driven models, learning from data. However, noisy and sparse data can have significant impact on the performance of neural networks and is a research topic in itself. In the context of PINNs, most studies assume mostly noise-free data.

Motivation

Visualizing the effect of sampling size and noise



Research Question

Krishnapriyan, Gholami, et al. (2021) analyzed the performance of PINNs and introduced and showed improvements through curriculum learning [1].

Our aim is to build on their work, specifically on the curriculum learning approach, and investigate the effects of sampling size and noise in the training data on the performance of PINNs.

The Classroom

Our first goal is to reproduce the results of Krishnapriyan, Gholami, et al. (2021) and verify against the baseline results, which does not use curriculum learning. Furthermore, we aim to find optimal hyperparameters for the optimizers used in the following experiments.

Setup

Convection-Diffusion Equation

The PDE in the experiments is the Convection–Diffusion Equation with no diffusion term and a scalar convection term given by

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0$$

with c being the convection coefficient and $u(x,t)$ is the primitive function of the CDE. The initial condition is given by a sine wave and the boundary condition is continuous.

Setup

Data Generation

The data is generated in the domain $\Omega = [0, 1] \times [0, 2\pi]$ consisting of 10000 equidistant points. The corresponding labels are the analytical solution of the CDE. Additionally, the sampled data point labels are augmented with Gaussian noise.

Setup

Model

The model is a fully connected neural network with 3 hidden layers, 50 neurons per hidden layer and the hyperbolic tangent function as the activation function. The inputs to the network are the spatial and temporal coordinates (x, t) and the output of the network is $\hat{u}(x, t)$.

Loss Function

The loss function is the mean squared error (MSE) between the predicted and the analytical solution of the CDE and the PDE of the learned function. The loss function is given by

$$\mathcal{L}(\hat{u}, u \mid \theta) = \mathcal{L}_{\mathrm{PDE}}(\hat{u} \mid \theta) + \mathcal{L}_{\mathrm{MSE}}(\hat{u}, u \mid \theta)$$

$$\mathcal{L}_{\mathrm{PDE}}(\hat{u} \mid \theta) = \frac{\partial \hat{u}}{\partial t} + c\frac{\partial \hat{u}}{\partial x}$$

where $\hat{u}$ is the predicted value of the CDE, u is the analytical solution of the CDE, θ are the learned weights and biases of the network.

Optimizer

Optimizer	Learning Rate	Weight Decay	Momentum	Nesterov	Max Iterations	History Size
SGD	[.0001, .1]	[0, .1]	[.1, .9]	Yes, No	-	-
Adam	[.00001, .1]	[0, .1]	-	-	-	-
L-BFGS	[.01, 2]	-	-	-	5, 10, 20	25, 50, 100

The hyperparameters for the optimizers are found using sweeps with Bayesian optimization on the hyperparameter search space.

Setup

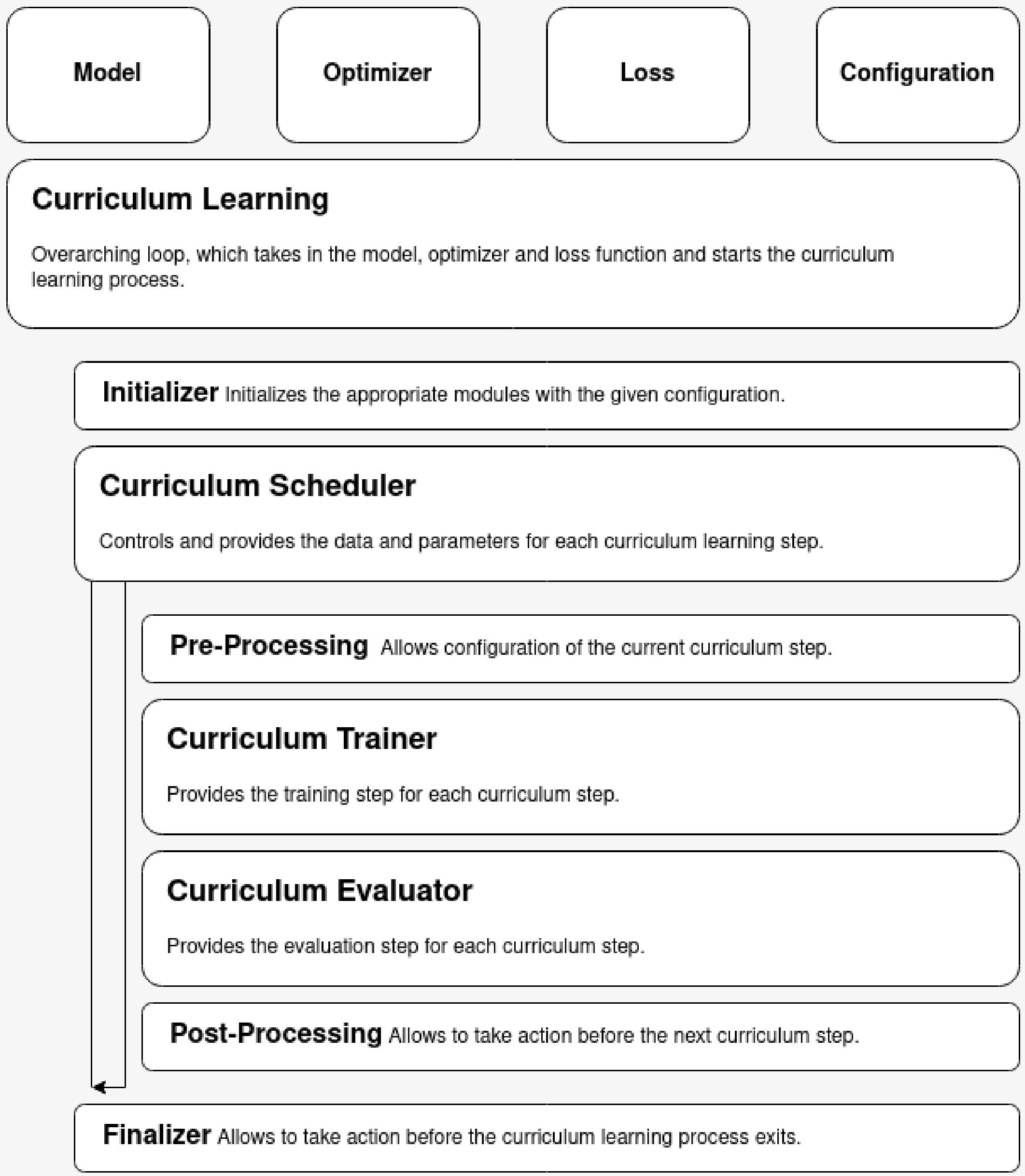
Curriculum Learning

During the curriculum learning, the convection coefficient c is increased from 1 to 30 in 30 steps. Each step consists of a common training routine with 250 epochs, 100 newly random sampled points from the domain with noise fixed at a SNR of 50.

A baseline experiment is also run, where the convection coefficient is fixed at 30 and the curriculum learning is not used.

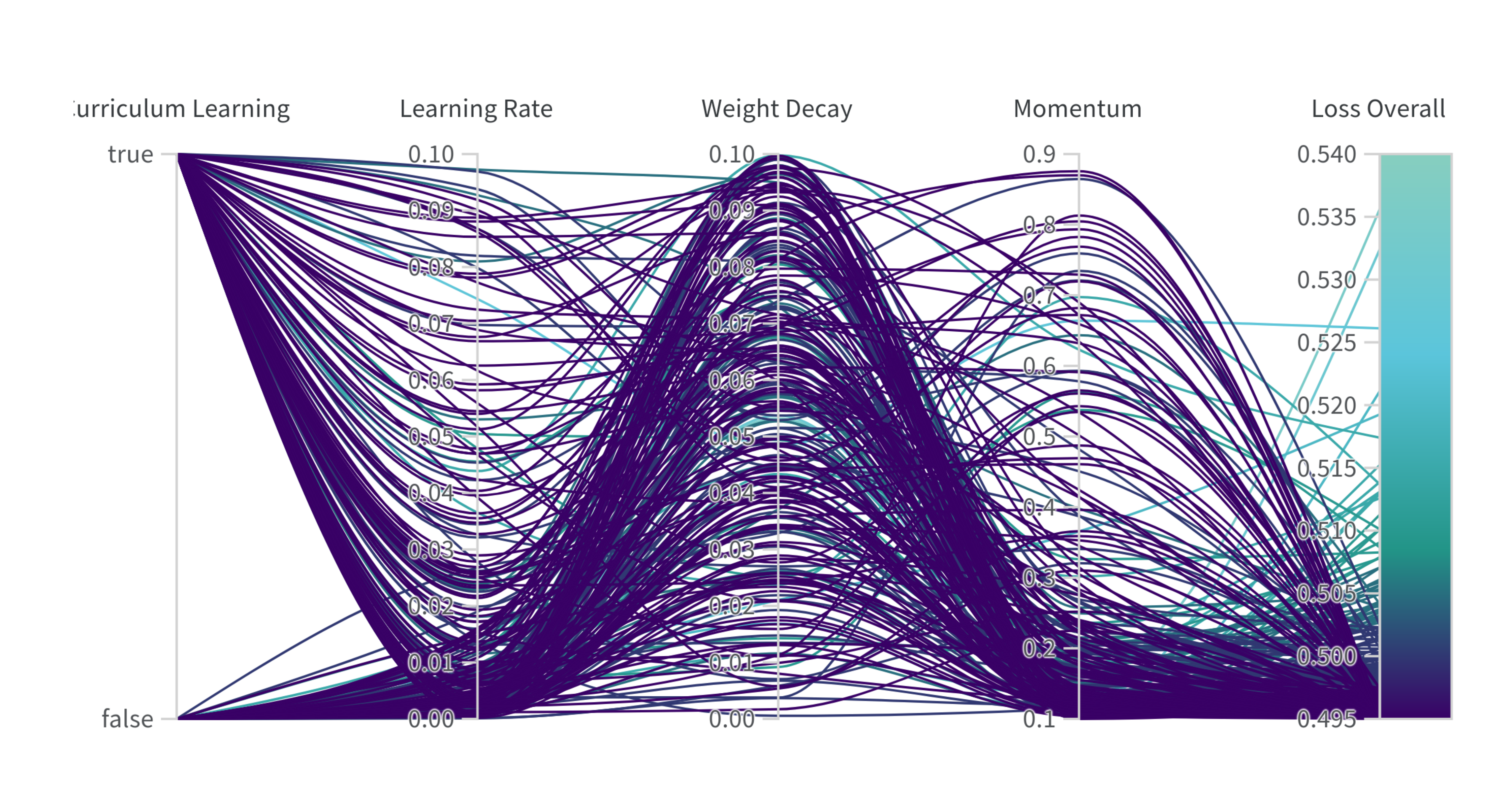
Setup

Curriculum Learning

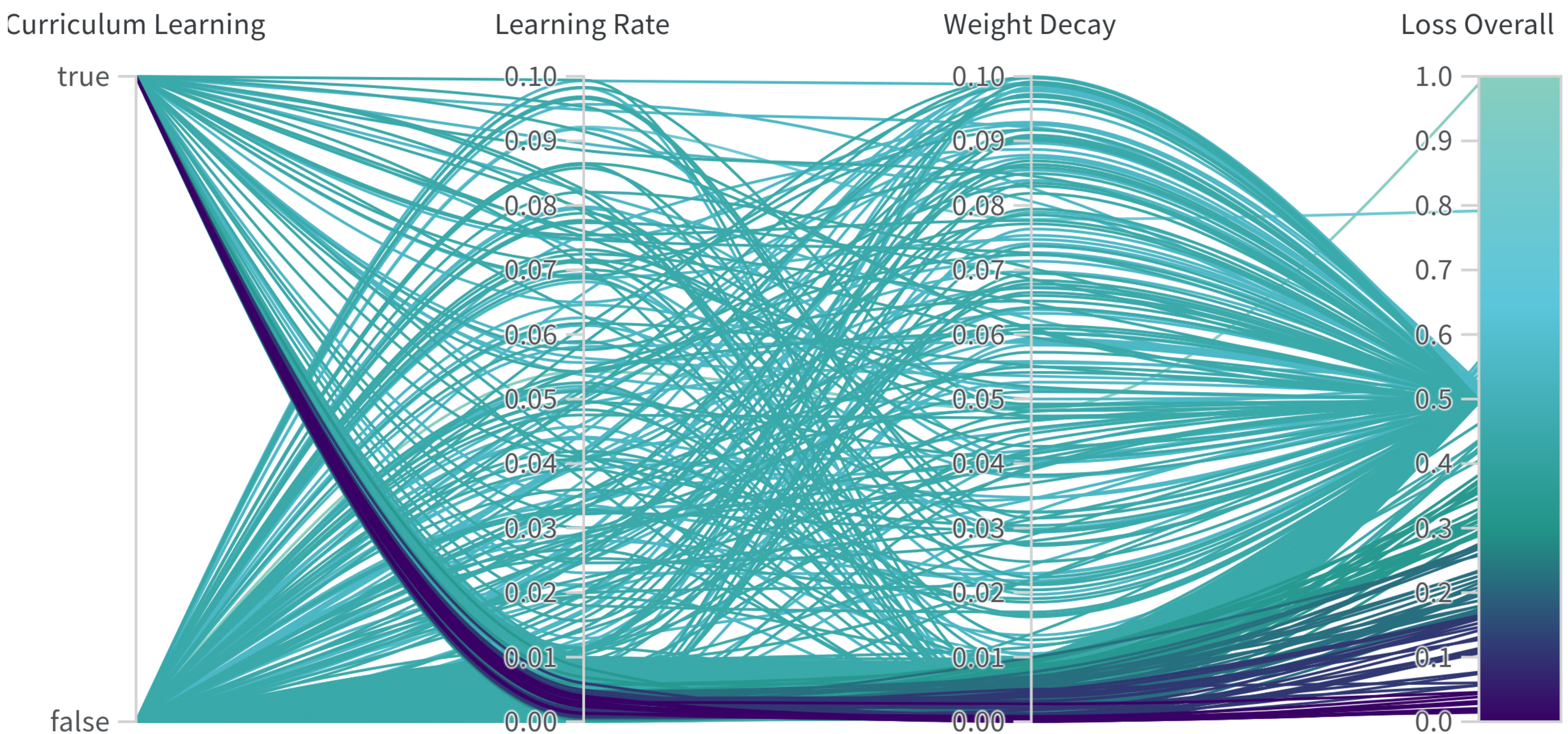


Results

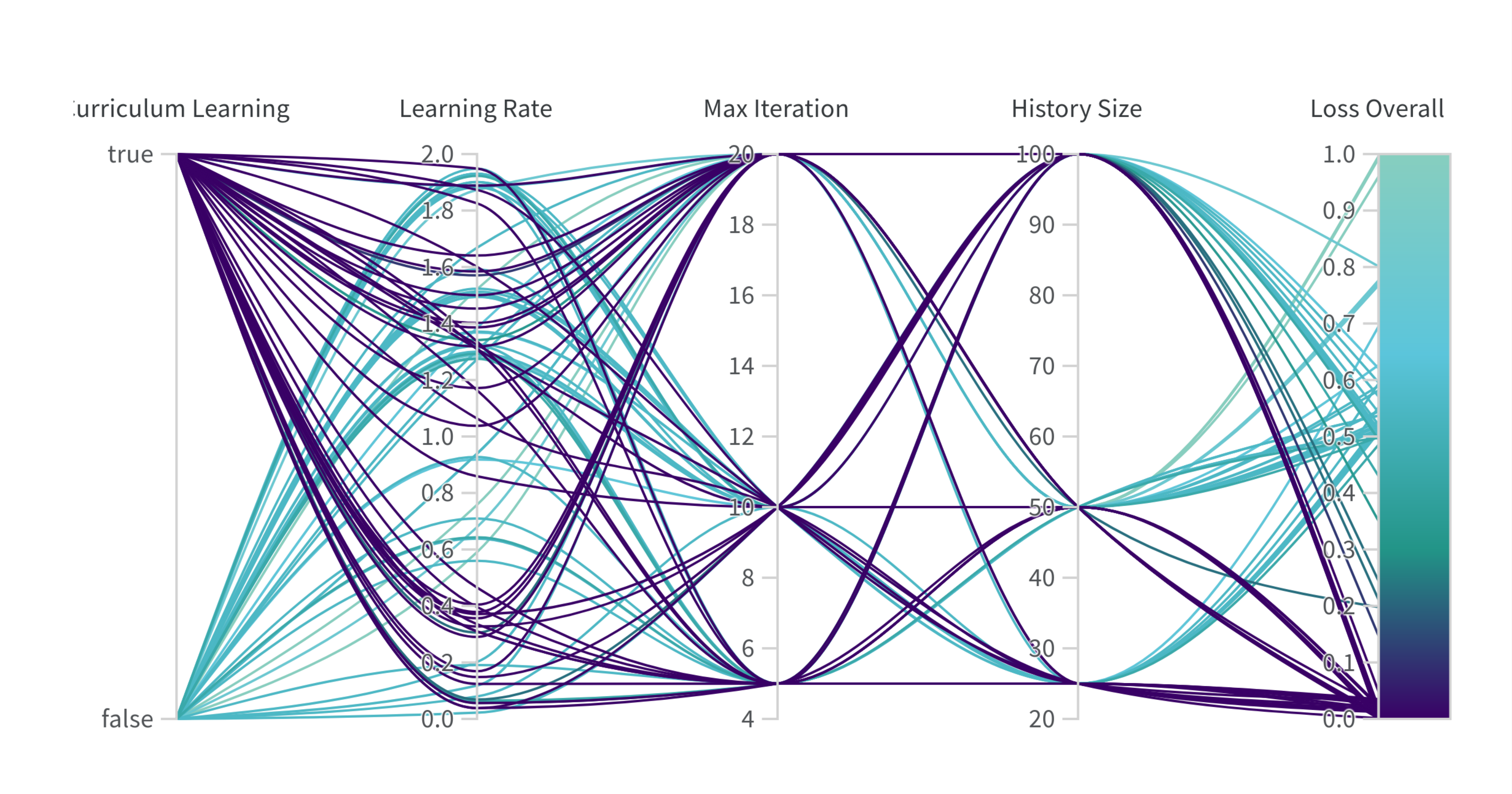
Stochastic Gradient Descent Hyperparameter Sweep



Adam Hyperparameter Sweep



L-BFGS Hyperparameter Sweep



A Crowded and Noisy Classroom

The previous experiment is an idealized scenario, where the data is almost noise-free. However, our goal is to investigate the effects of sampling size and noise in the training data on the performance of PINNs. Therefore, using the optimal hyperparameters, we vary the sample size and noise to see how they affect the ability of PINNs to learn the CDE.

Setup Changes

Setup Changes

Optimization

Optimizer	Learning Rate	Weight Decay	Max Iterations	History Size
Adam	.0025	.0005	-	-
L-BFGS$_{\mathrm{high}}$	1.5	-	10	25
L-BFGS$_{\mathrm{low}}$	.15	-	10	100

The hyperparameters for the optimizers are fixed to the optimal hyperparameters found in the previous experiment.

Sample Size

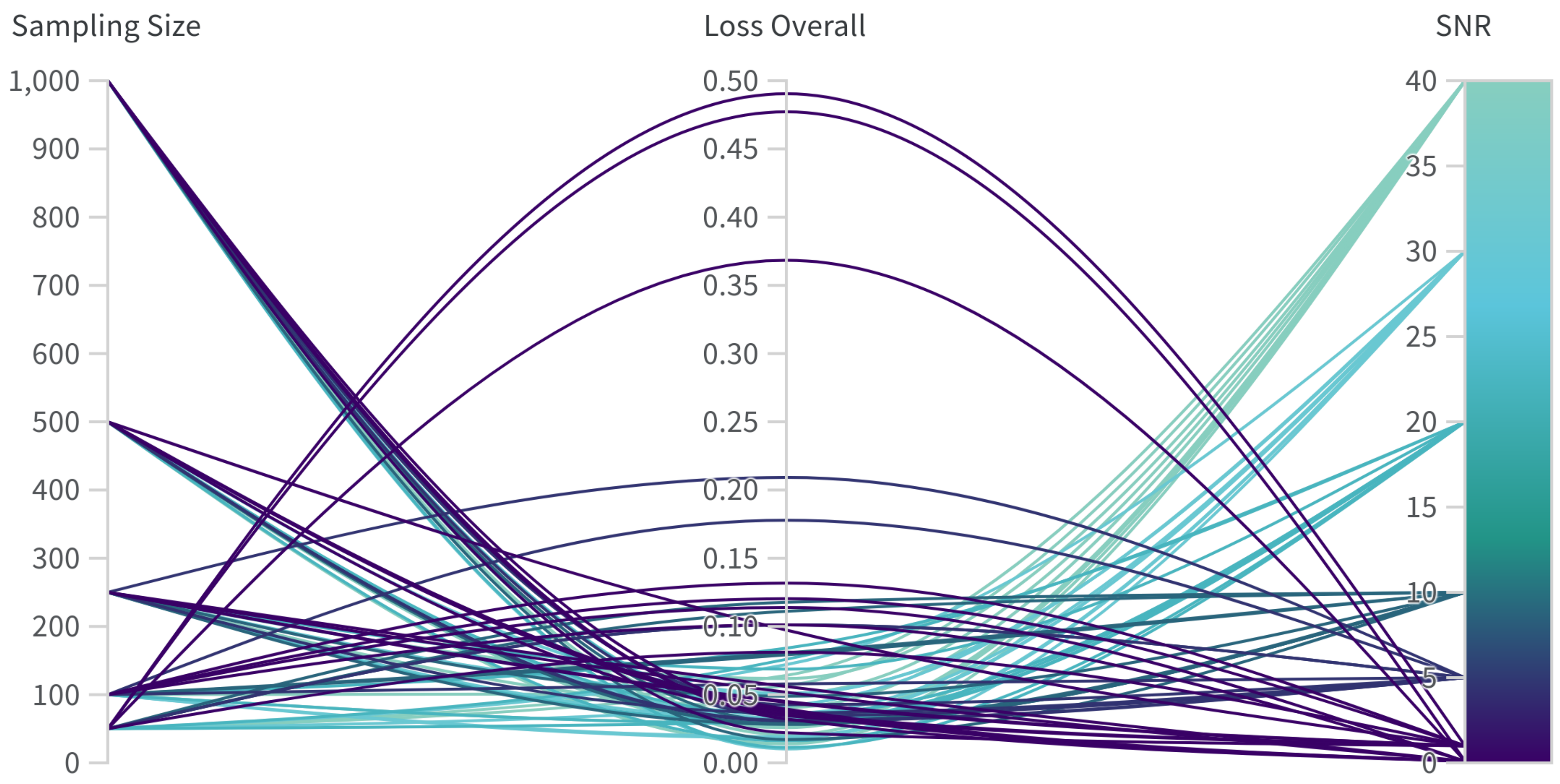
The sample size is varied between 10, 50, 100, 500 and 1000 samples and randomly sampled from the domain $\Omega = [0, 1] \times [0, 2\pi]$ consisting of 10000 equidistant points.

Noise

The signal-to-noise ratio (SNR) is varied from 0.1, 1, 5, 10, 20, 30 and 40 and added as Gaussian noise to the training data.

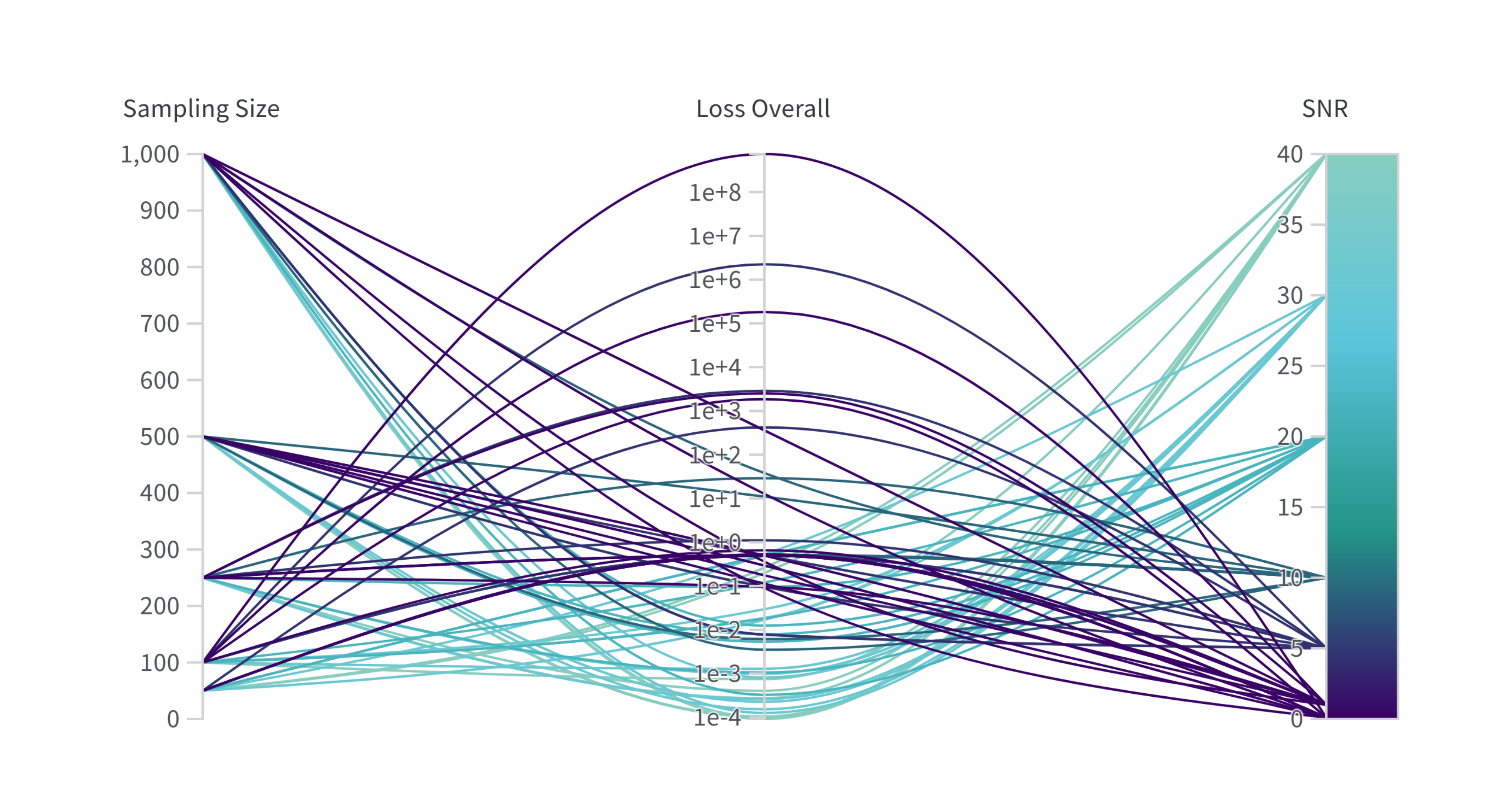
Results

Adam

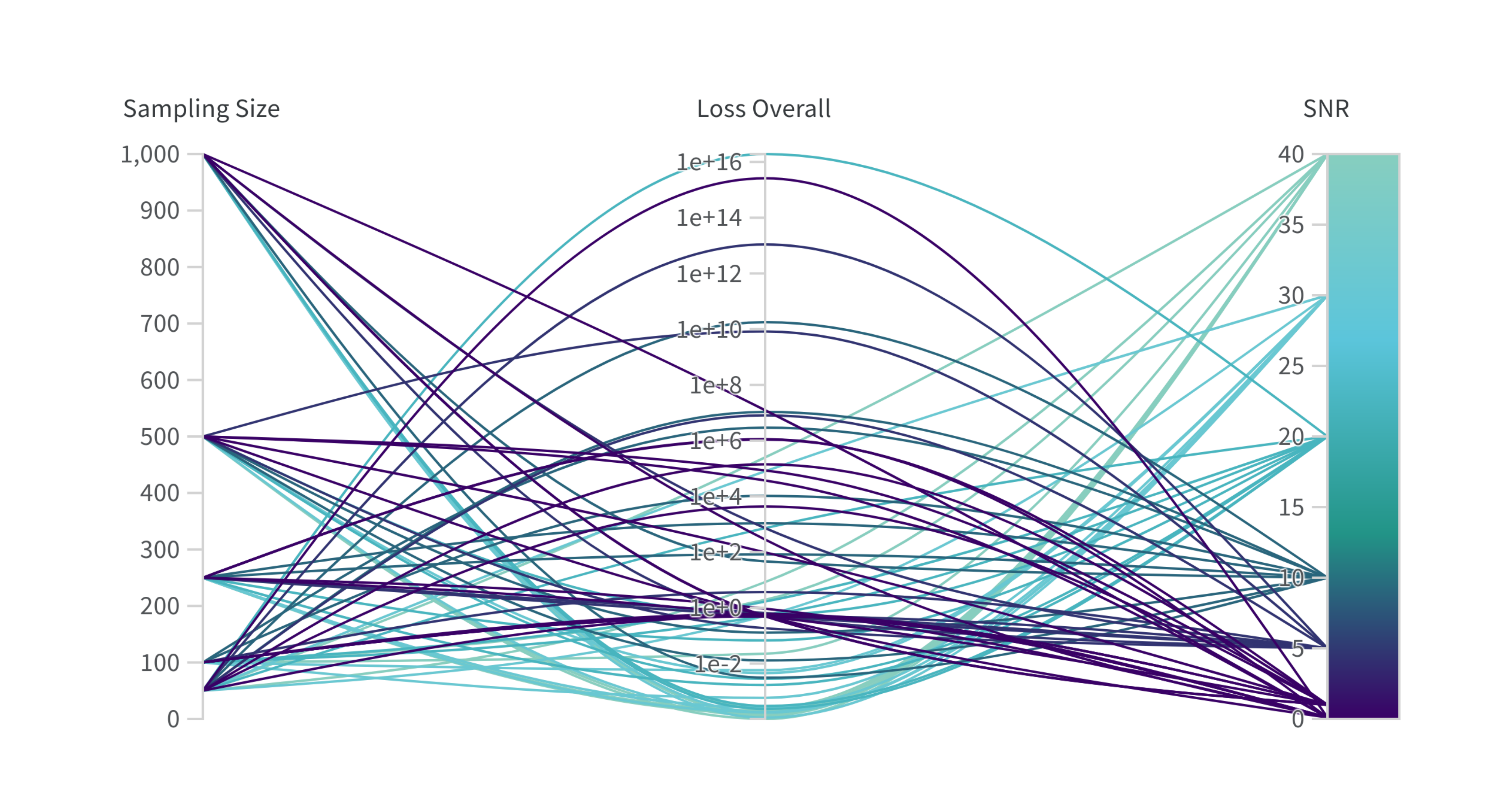


Sampling Size
Loss Overall
SNR
1,000
900
800
700
600
500
400
300
200
100
0
0.50
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
40
35
30
25
20
15
10
5
0

L-BFGS$_{\text{high}}$



L-BFGS$_{\mathrm{low}}$



A Slower Approach on Learning the Subject

Introducing noise can have a negative effect on the performance, which can be reduced to some degree by increasing the sample size. Another approach to possibly reduce the negative effects of noise is to increase the number of curriculum steps.

Setup Changes

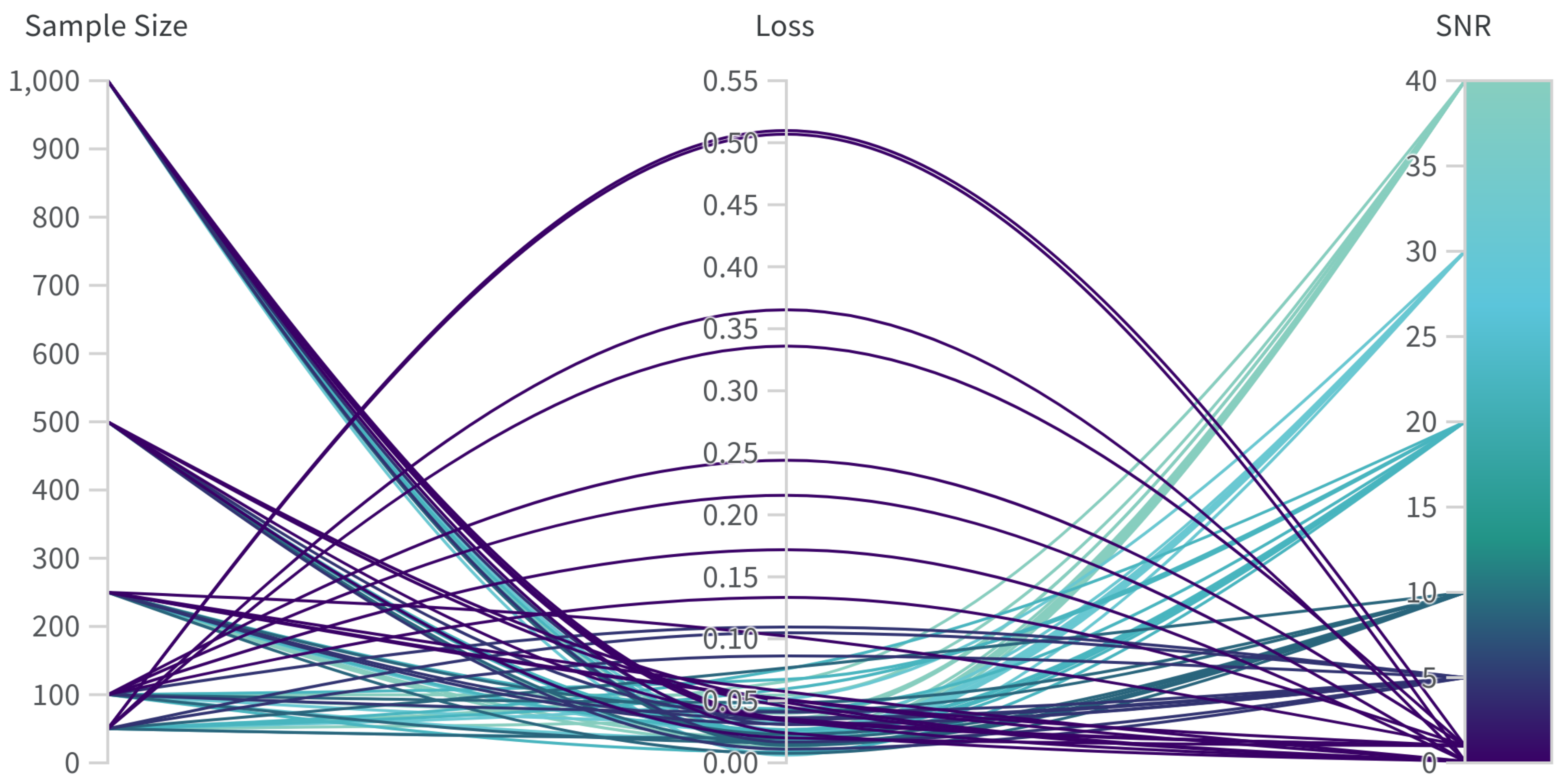
Setup Changes

Curriculum Learning

The curriculum learning steps are increased from 30 to 60. The convection coefficient c is increased by 0.5 per curriculum step, starting at 0.5 and ending at 30.

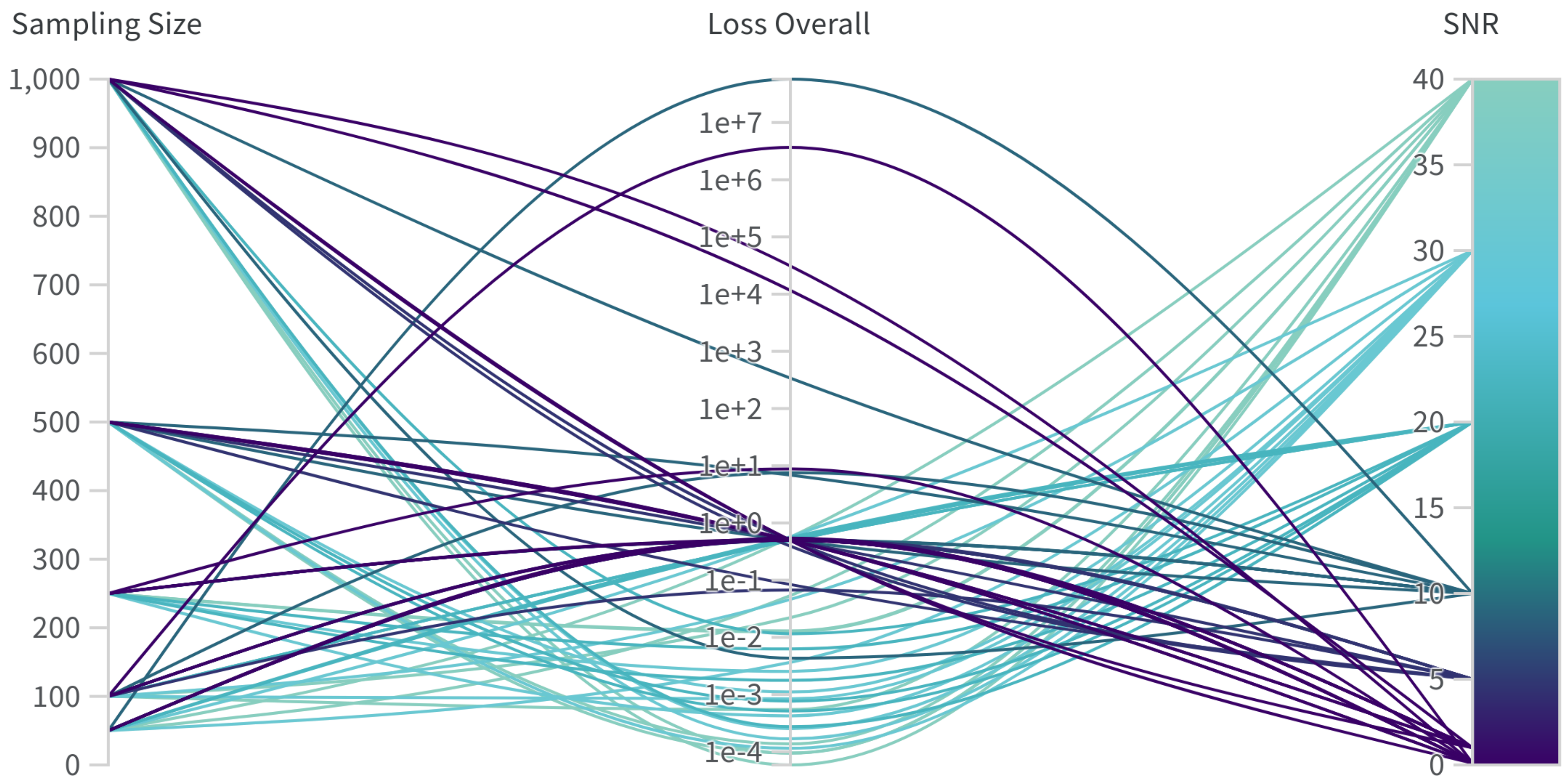
Results

Adam



Sample Size
Loss
SNR
1,000
900
800
700
600
500
400
300
200
100
0
0.55
0.50
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
40
35
30
25
20
15
10
5
0

L-BFGS$_{\text{high}}$



Key Takeaways

- Ability to reproduce the results of Krishnapriyan, Gholami, et al. (2021) [1] and verify against the baseline results.
- Ability to learn the CDE under noisy and sparse data.
    - Adam is able to reconstruct the co-domain of the CDE for even high noise levels and sparse data.
    - L-BFGS methods are less robust, unable to reconstruct the co-domain of the CDE for high noise levels.
- Increasing the number of curriculum steps did not have a significant effect on the performance of PINNs.

Thank you for your attention!

Questions?

References

[1] Aditi Krishnapriyan, Amir Gholami, Shandian Zhe, Robert Kirby, and Michael W Mahoney. Characterizing possible failure modes in physics-informed neural networks. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, Advances in Neural Information Processing Systems, volume 34, pages 26548–26560. Curran Associates, Inc., 2021.

[2] M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics, 378:686–707, 2019.