

Supplementary Materials

Supplementary File S1 — Part of the code used to reproduce the results

We provide a part of the code used to reproduce the results presented in the paper. The code consists of five Python scripts, each prefixed with a number to indicate the suggested order of execution. The scripts are as follows:

1. 1.1 `train_network.py`: Responsible for training the neural network. This script also saves the model weights at specified iteration intervals to allow for intermediate analysis.
2. 2.1 `compute_Betti_curves.py`: Calculates the persistent homology of the convolutional kernel space, providing insights into the topological features that emerge during training.
3. 2.2 `compute_loss.py`: Computes and plots the loss curve over the course of the network's training process, helping to visualize the learning progression.
4. 2.3 `compute_ROC.py`: Calculates and plots the variation of the ROC curve at different training iterations, which is crucial for assessing the model's performance over time.
5. 3.1 `plot_Betti_curves.py`: Generates and visualizes the Betti curves at various training iterations. This script is instrumental in analyzing the evolution of topological features in the convolutional kernel space across the training process.

The execution sequence is as follows: start with 1.1 `train_network.py`, then proceed with the series of scripts numbered 2.* (which include 2.1 `compute_Betti_curves.py`, 2.2 `compute_loss.py`, and 2.3 `compute_ROC.py`), and finally, end with 3.1 `plot_Betti_curves.py`. Each script serves a specific purpose within the research framework, from training the neural network to analyzing and visualizing the topology of convolutional kernel spaces and performance metrics.