

ReadMe

Thank you for your time and effort in reviewing our paper titled “An alternative approach to train neural networks using monotone variational inequality”! To reproduce results in the paper, please do the following

- `utils_gnn_VI_layer_NeurIPS.py` contains most of the utility functions for training and testing. Please keep it inside the folder as all other files before execution.
1. Classification with fully-connected networks: execute `simulation_NN_NeurIPS.py`
`lines 459` onward are used to generate tables with automatically saved .json results
 2. Small vs. Large random graph: in `simulation_layer_NeurIPS.py`, after running `lines 1-168`
 - a. Model recovery and prediction: run `lines 169-285`, with 286-370 for visualization using saved data
 - b. Neuron dynamics: `lines 371-587`, with 588 onward for visualization using saved data
 3. Real data (solar and traffic): in `real_data_test_NeurIPS.py`, after running `lines 1-138`
 - a. Solar: change `dataset='solar'` in line 139. Make sure the .mat data files are in the same folder as the .py file.
 - b. Traffic: change `dataset='traffic'` in line 139. Make sure the .p data files are in the same folder as the .py file.
 4. Real data (OGB large-scale dataset): in `real_data_test_OGB_NeurIPS.ipynb`
In particular, the attached `real_data_test_OGB_NeurIPS.html` includes results in the form of inline images.
 5. Real data (vision datasets): in `real_data_test_vision_FCNet/LeNet_NeurIPS.ipynb`
In particular, the attached `real_data_test_vision_FCNet/LeNet_NeurIPS.html` includes results in the form of inline images.

In particular, the code can be executed inline (e.g., via Jupyter Notebook) or simultaneously at once.