Deep Learning for Modeling: Concepts, Tools, and Techniques

ASSIGNMENTS PART TWO

This part of the assignment contains two competitions. The first competition is about generative models. A provided template "generative.py" should be finished to work with the provided automatic checker script "scoreChecker4.py". The second competition is about reinforcement learning tasks. You should also complete the provided template "reinfrocement.py" to work with the provided automatic checker script "scoreChecker5.py".

1. **Competition Two**: Please implement a PyTorch generative model for a random 2D XY model, where variables are within the range of $[-\pi, \pi]$. A dataset named "xyData.bz2" is provided. You should implement your model based on the template "generative.py" provided. To evaluate your performance, an automatic checker script "scoreChecker4.py" is provided. The script will estimate your model's samples' mean energy, a lower mean energy means better samples. Your rank will be based on this mean energy. For the base points, your model should reach a mean energy of less than -15.0. All submissions will be ranked, and the higher your rank, the more points you will receive. And, we'll check your model implementation manually to make sure no dataset samples are used in the sampling.
   Due Date: May 25th 2025

2. **Competition Three**: Please implement a PyTorch reinforcement learning model for the Atari Assault video game. Use the Gymnasium environments in "ram" mode. To properly install the gymnasium, you should run `pip install "gymnasium[atari, accept-rom-license]"`. Detailed introduction and instructions about the game can be found at the Docs of Gymnasium and the AtariAge page. You should implement your model based on the template "reinforcement.py" provided. To evaluate your performance, an automatic checker script "scoreChecker5.py" is provided. We set a limit of 4000 time steps for one simulation. A total of 20 simulations will be run. The best total score is used as your final score. For the base points, your model should reach a score of greater than 462. All submissions will be ranked, and the higher your rank, the more points you will receive.
   Due Date: May 25th 2025

3. **Competition Four**: Please implement a PyTorch sequential model for the dynamic prediction of a protein's structure. The dataset "trainDipeptide.npy" (containing 70000 time-step coordinates of backbone atoms) is provided. You should implement your model based on the template "sequential.py" provided. To evaluate your performance, an automatic checker script "scoreChecker6.py" is provided. For the base points, your model must reach a MSE loss of less than 0.5. All submissions will be ranked, and the higher your rank, the more points you will receive.
   Due Date: May 25th 2025