Introduction to Machine Learning
Lab 1: Graph Sequence Simulator and Loader

Hongteng Xu

February 28, 2022

1 Motivation

- Graph is commonly-used to represent structured data like molecules and networks. Essentially, the images formulated as pixel matrices and the textual (or acoustic) data formulated as sequences are special cases of graph data format. Therefore, it is necessary for us to learn some basic graph generation and loading methods.
- Before testing on real-world data, we often need to do concept-proofs on synthetic dataset when developing a machine learning method. Given a model, how to simulate desired data is important for us to evaluate the rationality of the model and get some insights.

2 Tasks

1. Install PyTorch and PyTorch Geometric, which are widely used tools for graph modeling and learning.
2. Implement a graph sequence simulator based on the following auto-regressive manner:

$$\begin{aligned} &P_t = \sum_{k=1}^{K} \alpha_k G_{t-k}, \quad [\alpha_k] \in \Delta^{K-1} \\ &G_t \sim \text{Bernoulli}(P_t), \end{aligned} \tag{1}$$

   where $G_t \in \{0,1\}^{N\times N}$ represents the adjacency matrix of the graph at time t, P_t is the edge probability matrix used to sample G_t.
3. Implement a data loader for the generated graph sequence, which output each graph and its previous K graphs.
4. Implement the following nonlinear auto-regressive model:

$$\begin{aligned} &P_t = \text{Sigmoid}\left(\sum_{k=1}^{K} \alpha_k (G_{t-k} - 0.5)\right), \quad [\alpha_k] \in \Delta^{K-1} \\ &G_t \sim \text{Bernoulli}(P_t), \end{aligned} \tag{2}$$

   and observe the difference between the graph sequence generated by (2) and that generated by (1).