**SE4050 – Deep Learning  
DL Lab 6 – Answers**

1. **Observation** Graph Density:

* As the number of nodes N increased while keeping the edge ratio E = 2\*N constant, the density of the graph decreased.

Degree Distribution (Histogram Plot):

* Observation: The degree distribution, which started as irregular and somewhat right skewed for small N (20), rapidly converged towards a Poisson distribution as N increased.

1. **Differences between supervised learning, self-supervised learning and semi-supervised learning methods**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Supervised learning | Self-supervised learning | Semi-supervised learning |
| Data Requirement | Requires a large dataset of both inputs and their corresponding correct outputs (labels). | Relies exclusively on unlabeled data, with no external labels provided. | Uses a mix of a small set of labeled data and a large amount of unlabeled data. |
| Learning Process | The algorithm learns a mapping from inputs to outputs by comparing its predictions against the known correct outputs. | The model generates its own supervisory signals by solving "pretext tasks" defined by the data itself. | The small labeled dataset provides initial guidance, and the model then uses the large unlabeled dataset to learn underlying patterns and improve its understanding. |

**Differences between transductive learning and inductive learning**

|  |  |  |
| --- | --- | --- |
|  | Transductive learning | Inductive learning |
| **Data** | Uses both labeled training data and the specific unlabeled test data during the training phase. | Uses only the labeled training data to learn the model. |
| **Model Creation** | Does not build a general model and may require re-training from scratch if new test data is introduced later. | Builds a reusable predictive model. |
| **Process** | A single-stage process where the model uses the information from both labeled and unlabeled data to infer labels for the unlabeled test set. | Typically has two stages: a training stage where a model is learned from labeled data, followed by a testing stage where the learned model makes predictions on a separate, unseen test set. |

1. **After increasing the number of epochs from 50 to 500,**

Observations: The loss decreased.  
 Validation accuracy improved.

**After experimenting without self-loops added to GCNConv() layers in the GCN() model,**

Observations: There’s a significant decrease in model accuracy.

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **Key Idea** | **How It Aggregates Neighbors** | **Special Feature** |
| **Message Passing GNN (MPNN)** | General framework for GNNs | Nodes exchange “messages” with neighbors, then update their states | Covers many GNN types including GCN, GAT, GraphSAGE |
| **Graph Convolutional Network (GCN)** | Extends CNN to graphs | Weighted average of neighbor features using normalized adjacency | Simple, efficient, but fixed aggregation |
| **Graph Attention Network (GAT)** | Uses attention on neighbors | Learns importance (weights) of neighbors via attention mechanism | Adaptive weights → captures varying importance of neighbors |
| **GraphSAGE** | Inductive learning on graphs | Samples neighbors and aggregates (mean, LSTM, max-pool) | Can generalize to unseen nodes/graphs (inductive capability) |