UNIVERSITY OF HERTFORDSHIRE
Hatfield - AL10 9AB

Individual Assignment: Machine Learning Tutorial

Understanding Oversmoothing in Graph Neural Networks: A Visual and Intuitive Tutorial

Submitted towards the completion requirements of the Master's program, for
the module **7PAM2021-0901-2025 - Machine Learning and Neural Networks**

MASTER OF SCIENCE

IN

DATA SCIENCE

Submitted By:

SUMAYYA
24069633

GitHub Repository: https://github.com/Sumayya22/GNN-Oversmoothing-Tutorial.git

ABSTRACT

Graph Neural Networks (GNNs) have become a foundational deep learning architecture for structured data such as molecules, citation networks, and social graphs. However, as message-passing depth increases, GNNs exhibit a critical limitation known as **oversmoothing**, where node representations converge towards indistinguishable values. This tutorial provides an intuitive, visual, and fully reproducible demonstration of oversmoothing. Through a synthetic community graph and a controlled depth experiment implemented using PyTorch Geometric, we show how node embeddings evolve as depth increases from 1 to 32 layers. Embedding plots and graph diagrams reveal the collapse phenomenon in a clear and accessible manner. All code and figures are provided in an accompanying Jupyter Notebook.

1. INTRODUCTION

Graph Neural Networks (GNNs) extend deep learning to graph-structured data by propagating information along edges. They excel in tasks where relationships between entities matter as much as the features themselves.

Despite their success, GNNs suffer from **oversmoothing**—a phenomenon where repeated message passing causes all node representations to become nearly identical. As depth increases, the network gradually loses its ability to discriminate between classes, severely limiting the usefulness of deep GNNs.

This tutorial aims to:

- Explain the mechanics of oversmoothing,
- Demonstrate it visually through embedding evolution,
- Provide code that allows readers to reproduce all results,
- Offer conceptual tools for understanding why oversmoothing occurs and how it is mitigated in modern GNN research.

This report pairs with a complete Jupyter Notebook containing all figures and experiments.

2. Background

2.1 Graph Neural Networks

A graph is defined as G = (V,E), where:

- V is the set of nodes,
- $E \subseteq V \times V$ is the set of edges.

Each node has a feature vector $h_v^{(0)} \in \mathbb{R}^d$

Most GNN architectures follow the **message-passing framework**:

$$h_v^{(k+1)} = \sigma\left(W^{(k)} \cdot \mathrm{AGG}(\{h_u^{(k)} : u \in N(v)\} \cup \{h_v^{(k)}\})\right),$$

where:

- $N(v)$ = neighbors of v,
- AGG is an aggregation function (mean, sum, attention),
- $W^{(k)}$ is a learnable weight matrix,
- σ is a nonlinearity such as ReLU.

This process is repeated across multiple layers, enabling nodes to integrate information from increasingly distant parts of the graph.

2.2 Oversmoothing Mechanism

Oversmoothing arises because message passing resembles repeated averaging of node features. After many layers, features from different regions of the graph become increasingly similar:

$$\lim_{k \to \infty} h_v^{(k)} = h_u^{(k)} \quad \forall u, v \in V$$

At this point:

- Class boundaries disappear,
- Node embeddings collapse into a single cluster,
- The GNN becomes unable to perform classification.

Oversmoothing is a fundamental challenge in GNN design and motivates architectural innovations such as residual connections, normalization layers, and decoupled propagation strategies.

3. Dataset and Methodology

3.1 Synthetic Graph with Clear Community Structure

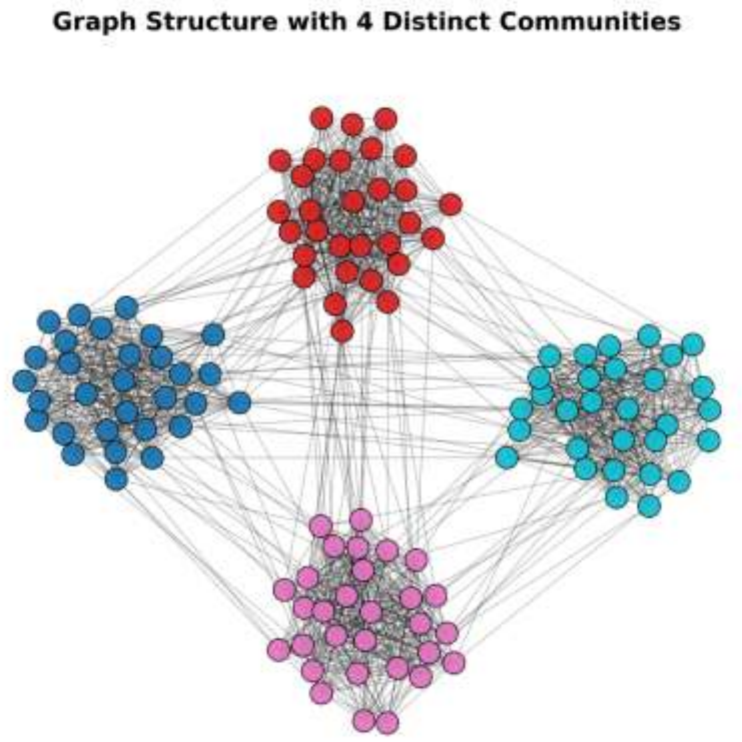
To illustrate oversmoothing effectively, we use a **Stochastic Block Model (SBM)** to generate a graph with four distinct communities.

Each community contains 30 nodes, with:

- High probability of intra-community edges,
- Low probability of inter-community edges.

This structure allows us to visually track whether embedding clusters remain separable as depth increases.

Figure 1. Graph Structure with 4 Communities



Alt-text: A graph with four visually distinct clusters of nodes, each cluster representing a community.

3.2 Model Architecture

We build:

- A **2-layer GCN** to learn meaningful initial embeddings,
- A **propagation-only deep GCN** to simulate depths up to 32 layers.

This design isolates oversmoothing due purely to message passing, independent of training.

3.3 Experimental Procedure

For depths
1, 2, 4, 8, 16, 32,

we:

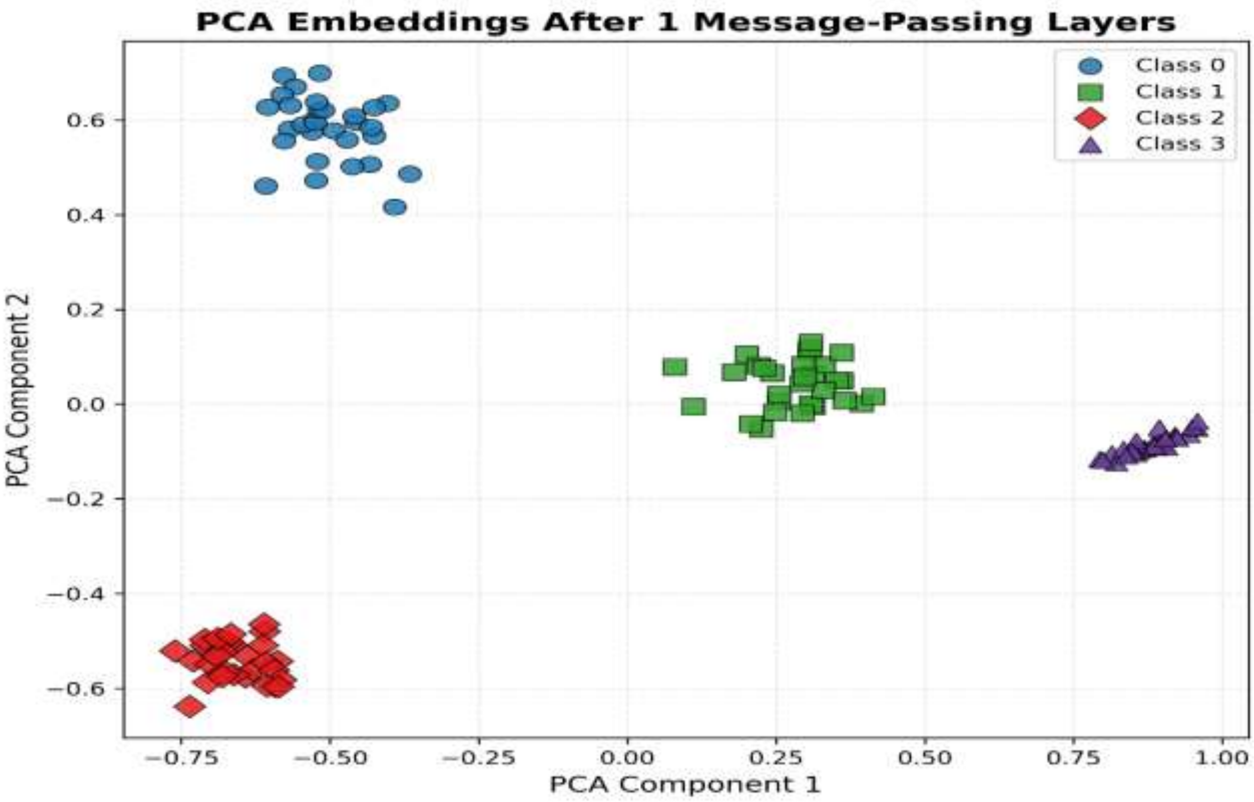
1. Apply k rounds of message passing,
2. Extract node embeddings,
3. Reduce dimensionality using PCA,
4. Plot embeddings in 2D.

The result is a sequence of plots showing the progressive collapse of embeddings.

4. Results and Analysis

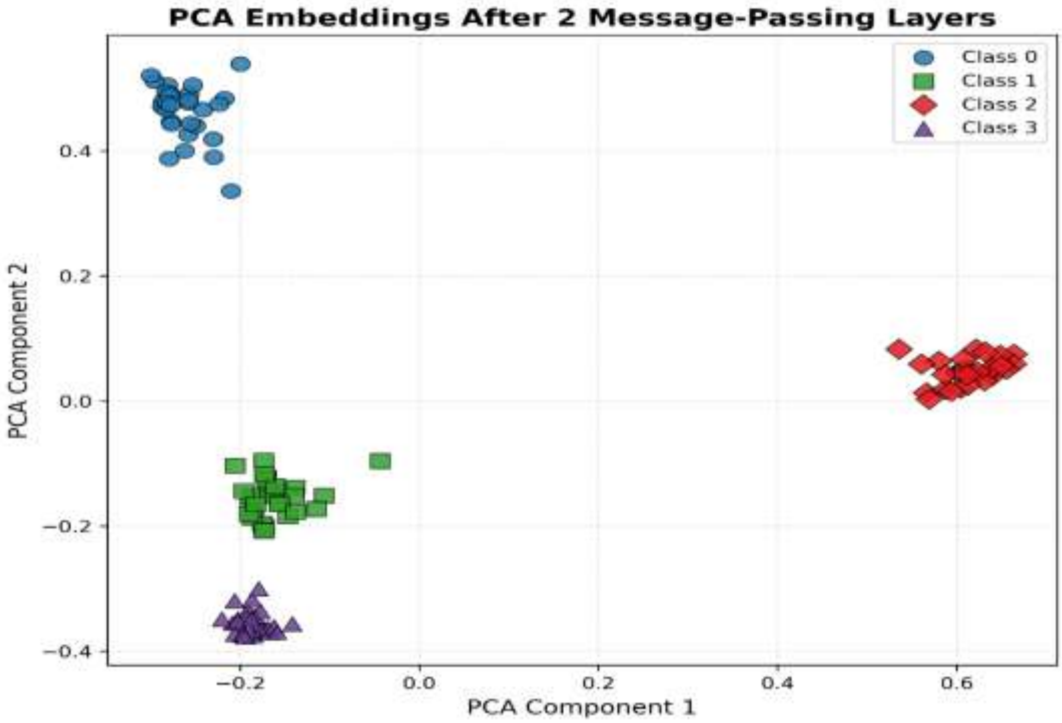
4.1 Early Depths (1–2 Layers)

Figure 2. Embeddings After 1 Layer



Distinct clusters remain clearly separated. Nodes are easily class-separable.

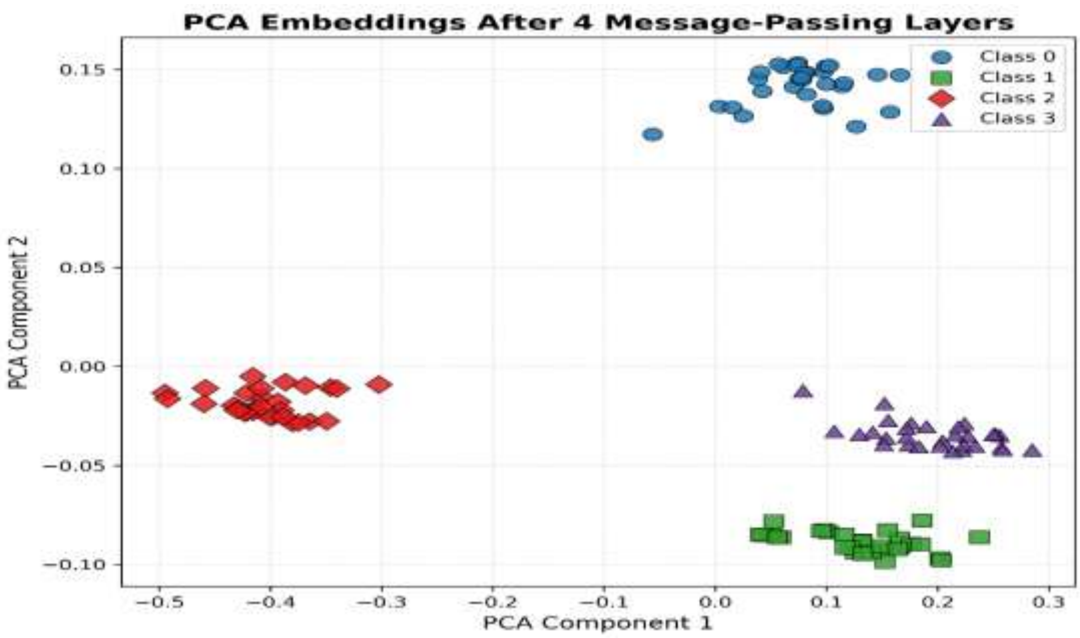
Figure 3. Embeddings After 2 Layers



Slight mixing occurs, but communities remain identifiable.

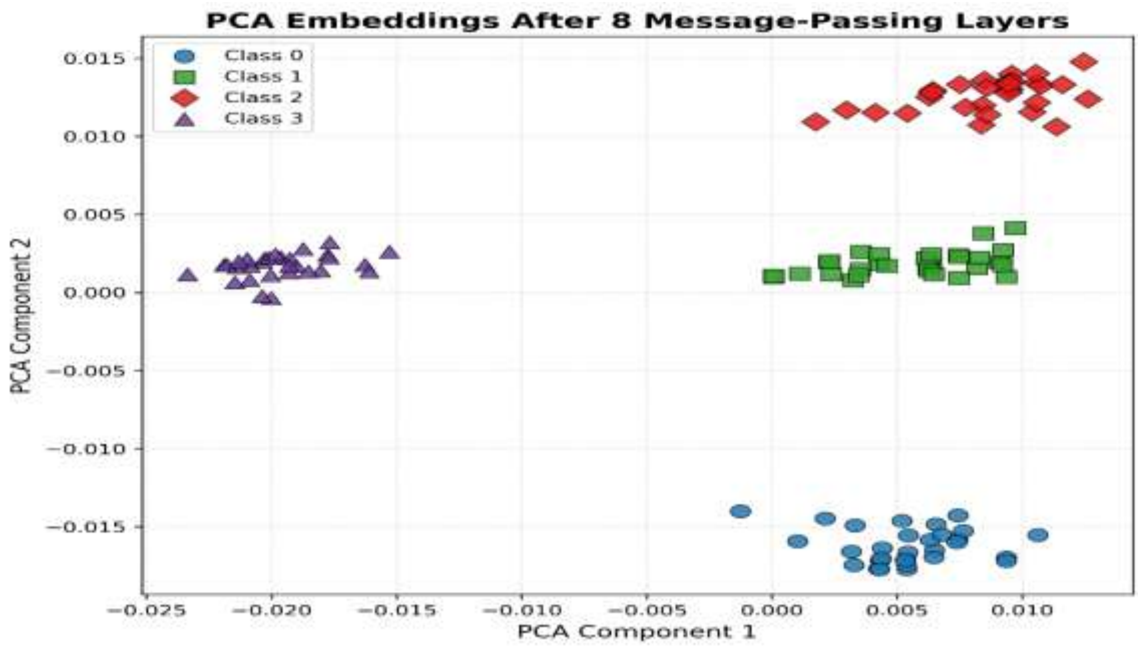
4.2 Medium Depths (4–8 Layers)

Figure 4. Embeddings After 4 Layers



Cluster boundaries begin to soften, signalling loss of discriminative structure.

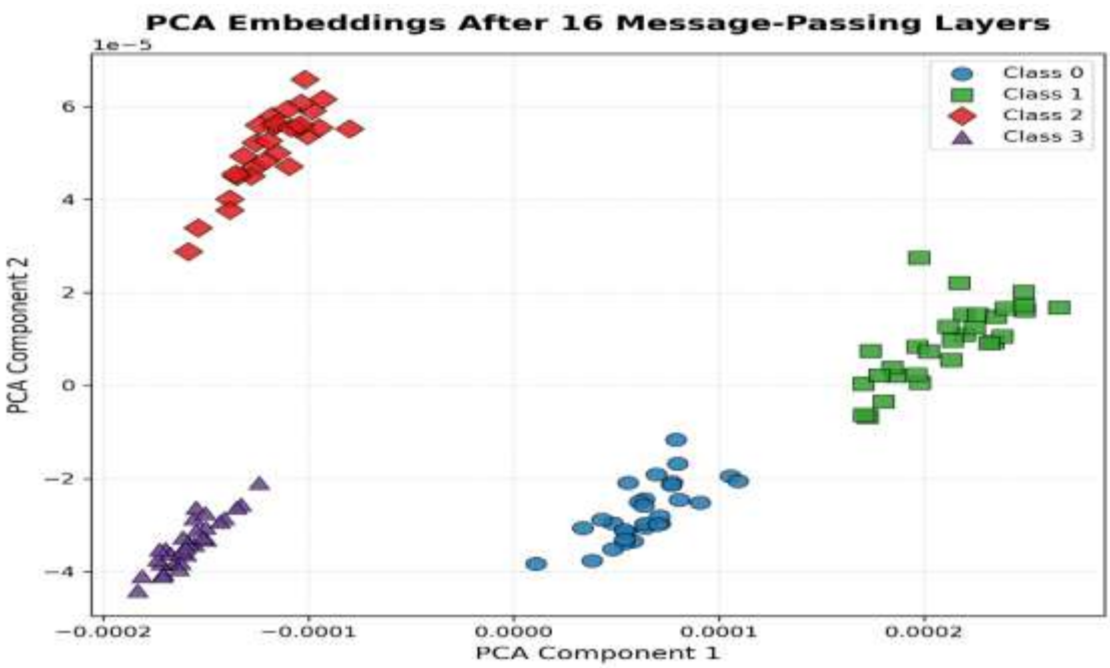
Figure 5. Embeddings After 8 Layers



Significant overlap appears; embedding structure degrades rapidly.

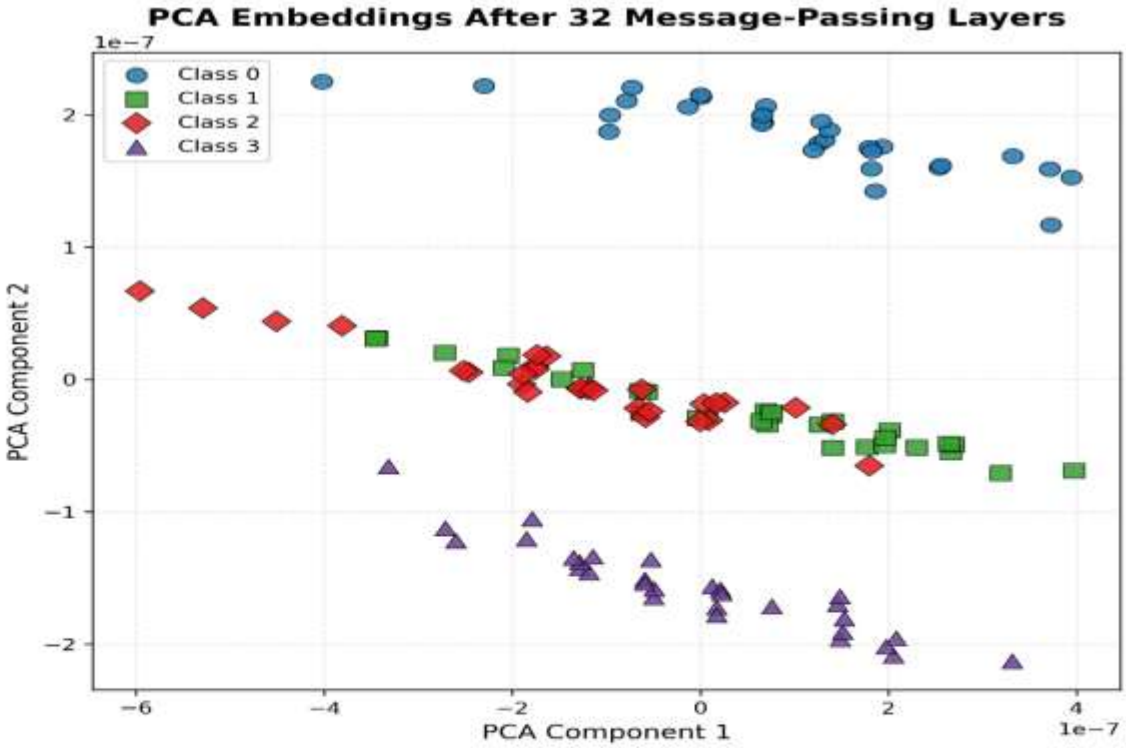
4.3 Deep GNNs (16–32 Layers)

Figure 6. Embeddings After 16 Layers



Representations show near-complete collapse toward a single region.

Figure 7. Embeddings After 32 Layers



All embeddings converge into a single point.
This is complete oversmoothing.

5. Why Oversmoothing Happens

GCNs implement propagation using a normalized adjacency matrix AAA.
A k-layer GCN approximates:

$$H^{(k)} \approx A^k XW.$$

As k grows large:

- A^k converges to a rank-1 matrix,
- All rows become identical,
- Therefore all node embeddings become identical.

Oversmoothing is thus a mathematical inevitability for standard GNNs with many layers.

6. Mitigation Strategies in Research

Modern GNN architectures address oversmoothing through:

- **Residual/Skip Connections**

  Preserve information from earlier layers.

- **Normalization Techniques**

  Such as PairNorm or BatchNorm to prevent embedding collapse.

- **DropEdge**

  Reduces graph density during training, slowing smoothing effects.

- **Decoupled GNNs**

  Separate feature transformation from propagation.

  Each of these strategies is active research.

7. Accessibility and Reproducibility

Accessibility Considerations

- Figures use distinct markers and high-contrast colors.
- Alt-text descriptions accompany all figures.
- Section headings support screen-reader navigation.

Reproducibility

- All experiments implemented in a Jupyter Notebook.
- All plots saved automatically.
- Dependencies listed in `requirements.txt`.
- MIT License ensures reusability.

8. Conclusion

Oversmoothing fundamentally limits the depth and expressive power of GNNs. Through controlled experiments and clear visualizations, this tutorial demonstrated how increasing message-passing depth leads to embedding collapse. The results illustrate why many practical GNNs use only 2–3 layers and why current research focuses on mitigating oversmoothing.

The complete experiment, including code and figures, is available in the accompanying GitHub repository.

9. References

Cited foundational works:

- Kipf, T. & Welling, M. (2017). *Semi-Supervised Classification with Graph Convolutional Networks.*
- Li, Q., Han, Z., & Wu, X. (2018). *Deeper Insights into GCNs: Understanding GCNs in the Context of Graph Signal Denoising.*
- Oono, K. & Suzuki, T. (2019). *Graph Neural Networks Exponentially Lose Expressive Power for Node Classification.*
- Hamilton, W. (2020). *Graph Representation Learning.*
- PyTorch Geometric Documentation: https://pytorch-geometric.readthedocs.io