Numerical on Graph Convolutional Networks

Problem Setup

Graph: 4 nodes

Adjacency matrix without self-loops:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Add self-loops:

$$\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

Degree matrix of $\hat{\mathbf{A}}$:

$$\mathbf{D} = \mathrm{diag}(3, 4, 3, 2)$$

Step 1: Compute normalized adjacency matrix $\tilde{\mathbf{A}}$

Calculate each element:

$$\tilde{A}_{i,j} = \frac{\hat{A}_{i,j}}{\sqrt{d_i d_j}}$$

For example:

- $\tilde{A}_{0,0} = \frac{1}{\sqrt{3 \times 3}} = \frac{1}{3} \approx 0.333$
- $\tilde{A}_{0,1} = \frac{1}{\sqrt{3 \times 4}} = \frac{1}{\sqrt{12}} \approx 0.289$
- And so on for all entries.

The full normalized adjacency matrix $\tilde{\mathbf{A}}$ is:

$$\begin{bmatrix} 0.333 & 0.289 & 0.333 & 0 \\ 0.289 & 0.25 & 0.289 & 0.354 \\ 0.333 & 0.289 & 0.333 & 0 \\ 0 & 0.354 & 0 & 0.5 \end{bmatrix}$$

Step 2: Node feature matrix $\mathbf{X}^{(0)}$

Each node has 3 features:

$$\mathbf{X}^{(0)} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Step 3: Weight matrices initialization

- Layer 1 weights $\mathbf{W}^{(1)} \in \mathbb{R}^{3\times 2}$:

$$\mathbf{W}^{(1)} = \begin{bmatrix} 0.2 & -0.1 \\ 0.3 & 0.4 \\ 0.5 & -0.2 \end{bmatrix}$$

- Layer 2 weights $\mathbf{W}^{(2)} \in \mathbb{R}^{2\times 2}$:

$$\mathbf{W}^{(2)} = \begin{bmatrix} 0.1 & 0.3 \\ -0.4 & 0.2 \end{bmatrix}$$

- Output layer weights $\mathbf{W}^{(out)} \in \mathbb{R}^{2\times 1}$ and bias $b^{(out)}$:

$$\mathbf{W}^{(out)} = \begin{bmatrix} 0.7 \\ -0.3 \end{bmatrix}, \quad b^{(out)} = 0.1$$

Step 4: Forward Pass Layer 1

1. Aggregate features:

$$\mathbf{Z}^{(1)} = \tilde{\mathbf{A}}\mathbf{X}^{(0)}$$

Calculate per node:

- Node 0:

$$\mathbf{0.333 \times +0.289 \times +0.333 \times +0 \times = [0.666, 0.622, 0.622][1]}$$

- Node 1:

$$\mathbf{0.289 \times +0.25 \times +0.289 \times +0.354 \times = [0.578, 0.828, 0.578][1]}$$

- Node 2:

$$\mathbf{0.333 \times +0.289 \times +0.333 \times +0 \times = [0.666, 0.622, 0.622][1]}$$

- Node 3:

$$\mathbf{0 \times +0.354 \times +0 \times +0.5 \times = [0, 0.354, 0.854][1]}$$

2. Transform features:

$$\mathbf{M}^{(1)} = \mathbf{Z}^{(1)}\mathbf{W}^{(1)}$$

Calculate per node using $\mathbf{W}^{(1)}$:

- Node 0:

$$[0.666, 0.622, 0.622] \times \begin{bmatrix} 0.2 & -0.1 \\ 0.3 & 0.4 \\ 0.5 & -0.2 \end{bmatrix} = [0.631, 0.058]$$

- Node 1:

$$[0.578, 0.828, 0.578] \times \mathbf{W}^{(1)} = [0.608, 0.150]$$

- Node 2:

$$[0.666, 0.622, 0.622] \times \mathbf{W}^{(1)} = [0.631, 0.058]$$

- Node 3:

$$[0, 0.354, 0.854] \times \mathbf{W}^{(1)} = [0.542, -0.224]$$

3. Apply ReLU:

$$\mathbf{H}^{(1)} = \max(0, \mathbf{M}^{(1)})$$

Result:

$$\begin{bmatrix} 0.631 & 0.058 \\ 0.608 & 0.150 \\ 0.631 & 0.058 \\ 0.542 & 0 \end{bmatrix}$$

Step 5: Forward Pass Layer 2

1. Aggregate features:

$$\mathbf{Z}^{(2)} = \tilde{\mathbf{A}}\mathbf{H}^{(1)}$$

Calculate per node:

- Node 0:

$$0.333 \times [0.631, 0.058] + 0.289 \times [0.608, 0.150] + 0.333 \times [0.631, 0.058] + 0 \times [0.542, 0] = [$$

- Node 1:

$$0.289 \times [0.631, 0.058] + 0.25 \times [0.608, 0.150] + 0.289 \times [0.631, 0.058] + 0.354 \times [0.542, 0]$$

- Node 2:

$$0.333 \times [0.631, 0.058] + 0.289 \times [0.608, 0.150] + 0.333 \times [0.631, 0.058] + 0 \times [0.542, 0] = [$$

- Node 3:

$$0 \times [0.631, 0.058] + 0.354 \times [0.608, 0.150] + 0 \times [0.631, 0.058] + 0.5 \times [0.542, 0] = [0.388,$$

2. Transform features:

$$\mathbf{M}^{(2)} = \mathbf{Z}^{(2)}\mathbf{W}^{(2)}$$

Calculate per node:

- Node 0:

$$[0.605, 0.091] \times \begin{bmatrix} 0.1 & 0.3 \\ -0.4 & 0.2 \end{bmatrix} = [0.028, 0.190]$$

- Node 1:

$$[0.536, 0.085] \times \mathbf{W}^{(2)} = [0.021, 0.172]$$

- Node 2:

$$[0.605, 0.091] \times \mathbf{W}^{(2)} = [0.028, 0.190]$$

- Node 3:

$$[0.388, 0.053] \times \mathbf{W}^{(2)} = [0.003, 0.130]$$

3. Apply ReLU:

$$\mathbf{H}^{(2)} = \max(0, \mathbf{M}^{(2)})$$

Result:

$$\begin{bmatrix} 0.028 & 0.190 \\ 0.021 & 0.172 \\ 0.028 & 0.190 \\ 0.003 & 0.130 \end{bmatrix}$$

Step 6: Output Layer - Node Classification Prediction

1. Compute logits:

$$\mathbf{Y} = \mathbf{H}^{(2)}\mathbf{W}^{(out)} + b^{(out)}$$

Calculate per node:

- Node 0:

$$0.028 \times 0.7 + 0.190 \times (-0.3) + 0.1 = 0.0196 - 0.057 + 0.1 = 0.0626$$

- Node 1:

$$0.021 \times 0.7 + 0.172 \times (-0.3) + 0.1 = 0.0147 - 0.0516 + 0.1 = 0.0631$$

- Node 2:

$$0.028 \times 0.7 + 0.190 \times (-0.3) + 0.1 = 0.0626 \quad \text{(same as node 0)}$$

- Node 3:

$$0.003 \times 0.7 + 0.130 \times (-0.3) + 0.1 = 0.0021 - 0.039 + 0.1 = 0.0631$$

2. Apply sigmoid to get class probabilities:

$$\hat{y}_i = \frac{1}{1 + e^{-Y_i}}$$

Examples:

- Node 0:

$$\hat{y}_0 = \frac{1}{1 + e^{-0.0626}} \approx 0.516$$

- Node 1:

$$\hat{y}_1 \approx 0.516$$

- Node 2:

$$\hat{y}_2 \approx 0.516$$

- Node 3:

$$\hat{y}_3 \approx 0.516$$