Question 1

- For the first node, the second-layer representation is:

$$z_1^{(2)} = \alpha_{12}^{(2)} W^{(2)} z_2^{(1)} + \alpha_{13}^{(2)} W^{(2)} z_3^{(1)}$$

- For the fourth node, since $z_2^{(1)} = z_6^{(1)}$ and $z_3^{(1)} = z_5^{(1)}$, the second-layer representation is:

$$z_4^{(2)} = (\alpha_{42}^{(2)} + \alpha_{46}^{(2)}) W^{(2)} z_2^{(1)} + (\alpha_{43}^{(2)} + \alpha_{45}^{(2)}) W^{(2)} z_3^{(1)}$$

Therefore, contributions for v_1 and v_4 come from the same hidden representations. They share contributions from the same neighbors, but the magnitude differ due to the larger neighborhood size, leading to smaller normalized attention coefficients per neighbor.

Question 2

If all nodes have identical features, the model will struggle to achieve high classification accuracy because it cannot distinguish nodes based on their input features. In this case, the GNN relies solely on the graph structure to differentiate nodes, as the initial features provide no unique information. This is problematic in the karate network, where nodes from different classes can have similar neighborhood structures. Without meaningful feature distinctions, the model cannot effectively separate nodes into classes, leading to poor performance.

Question 3

Sum Readout

For each graph, the graph-level representation is the sum of the node representations:

$$z_{G1} = sum(\begin{bmatrix} 2.2 & -0.6 & 1.4 \\ 0.2 & 1.8 & 1.5 \\ 0.5 & 1.1 & -1.0 \end{bmatrix}) = \begin{bmatrix} 2.9 & 2.3 & 1.9 \end{bmatrix}$$

$$z_{G2} = sum(\begin{bmatrix} 0.7 & 0.1 & 1.3 \\ 1.2 & -0.9 & 0.3 \\ 2.2 & 0.9 & 1.2 \\ -0.7 & 1.8 & 1.5 \end{bmatrix}) = \begin{bmatrix} 3.4 & 1.9 & 4.3 \end{bmatrix}$$

$$z_{G3} = sum(\begin{bmatrix} -0.4 & 1.8 & 0.1 \\ 2.2 & -0.6 & 1.5 \end{bmatrix}) = \begin{bmatrix} 1.8 & 1.2 & 1.6 \end{bmatrix}$$

Mean Readout

The same way, we get:

$$z_{G1} = \frac{1}{3} \begin{bmatrix} 2.9 & 2.3 & 1.9 \end{bmatrix} = \begin{bmatrix} 0.97 & 0.77 & 0.63 \end{bmatrix}$$

$$z_{G2} = \frac{1}{4} \begin{bmatrix} 3.4 & 1.9 & 4.3 \end{bmatrix} = \begin{bmatrix} 0.85 & 0.475 & 1.075 \end{bmatrix}$$

$$z_{G3} = \frac{1}{2} \begin{bmatrix} 1.8 & 1.2 & 1.6 \end{bmatrix} = \begin{bmatrix} 0.9 & 0.6 & 0.8 \end{bmatrix}$$

Max Readout

Finally,

$$z_{G1} = \begin{bmatrix} 2.2 & 1.8 & 1.5 \end{bmatrix}$$

$$z_{G2} = \begin{bmatrix} 2.2 & 1.8 & 1.5 \end{bmatrix}$$

$$z_{G3} = \begin{bmatrix} 2.2 & 1.8 & 1.5 \end{bmatrix}$$

Sum distinguishes the graphs well, as each graph has a unique representation Same for mean except that it normalizes by the number of nodes, reducing the range of the representations. However, max fails to distinguish the graphs, as the maximum values are the same across all graphs.

Question 4

When the C_4 and C_8 graphs are fed into the model with all node features initialized to 1, the resulting graph-level representations z_{G_1} and z_{G_2} will differ in magnitude but not in their underlying structure. Since the sum readout function aggregates the node representations by summing them, and C_8 has twice as many nodes as C_4, the representation z_{G_2} will be approximately twice the magnitude of z_{G_1}. This difference arises because both graphs are regular cycles with uniform degree, resulting in nearly identical node representations after aggregation. Thus, the graph size directly influences the magnitude of the sum readout, leading to $z_{G_2} \approx 2 \cdot z_{G_1}$.

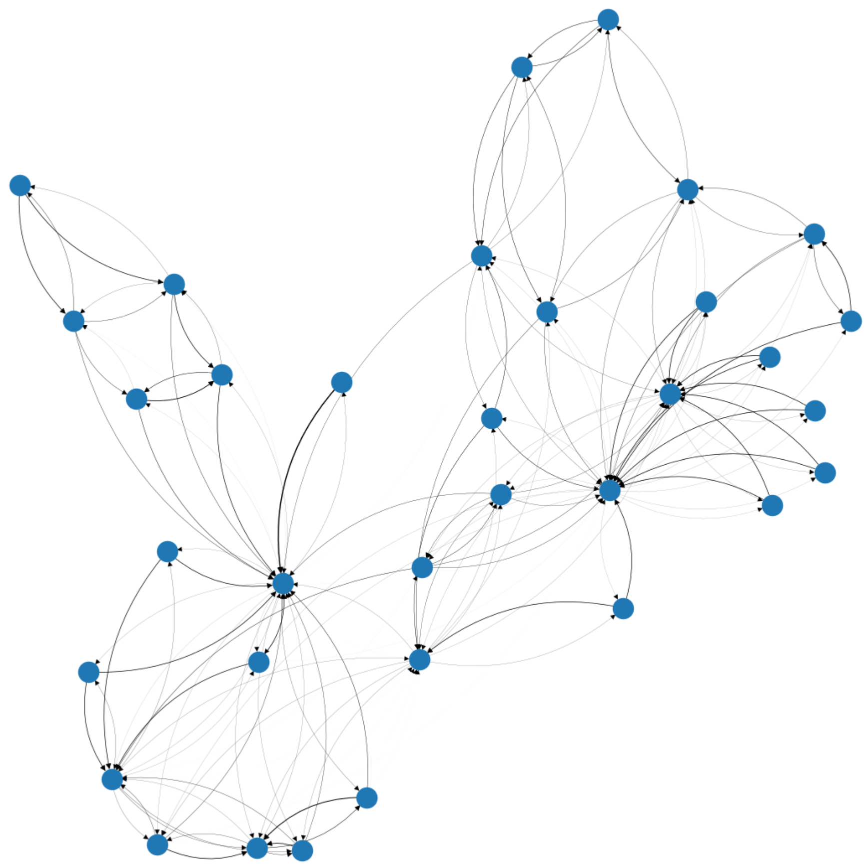

Figure 1: Visualization of Attention Scores