Semi-supervised classification with graph convolutional networks by Kipf and Welling

Samuel Unicomb

MACSI, Department of Mathematics and Statistics, University of Limerick

December 4, 2023

Node representations $H^{(l)}$ are embedded recursively following

$$H^{(l+1)} = \sigma\left(W^{(l)}H^{(l)}\hat{A}\right), \tag{1}$$

where

1. column i of $H^{(l)}$ is the l-th embedding $h_i^{(l)}$ of node i
2. $H^{(0)} = X$ where X are graph feature vectors
3. $W^{(l)}$ is a matrix of adjustable weights
4. σ is a nonlinear function such as ReLU or tanh
5. $\hat{A}$ is a symmetrically normalised Laplacian with entries

$$\hat{A}_{ij} = \frac{A_{ij} + I_{ij}}{\sqrt{(k_i + 1)(k_j + 1)}}.$$

$$H^{(0)} = X \tag{2a}$$

$$M^{(0)} = H^{(0)}\hat{A} \tag{2b}$$

$$H^{(1)} = \text{ReLU}\left(W^{(0)}M^{(0)}\right) \tag{2c}$$

$$M^{(1)} = H^{(1)}\hat{A} \tag{2d}$$

$$H^{(2)} = \text{softmax}\left(W^{(1)}M^{(1)}\right) \tag{2e}$$

Figure 1: Zachary karate club coloured using Louvain method in *Gephi*.

Figure 2: Unsupervised embedding of Zachary karate club.

Figure 3: Comparing $\sigma(\tilde{H}^{(1)}) = \tanh(\tilde{H}^{(1)})$ and $\mathrm{id}(\tilde{H}^{(1)})$.
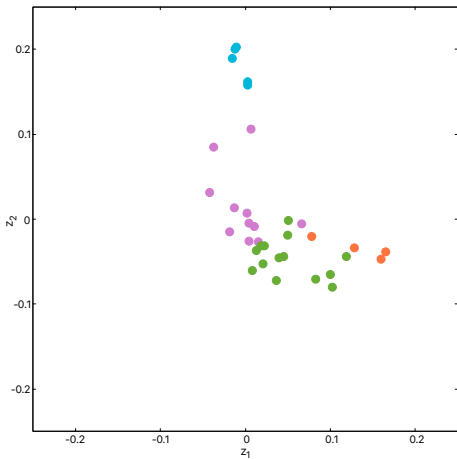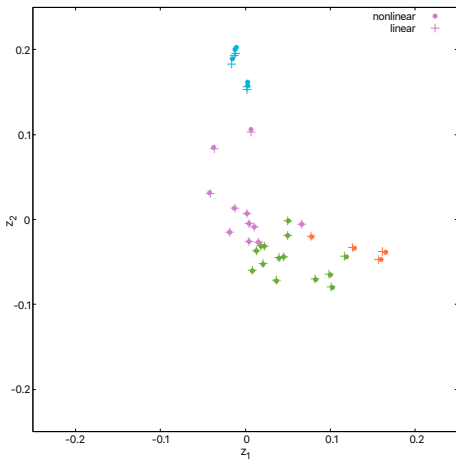
$$H^{(0)} = X \tag{3a}$$

$$M^{(0)} = H^{(0)}\hat{A} \tag{3b}$$

$$H^{(1)} = \tanh\left(W^{(0)}M^{(0)}\right) \tag{3c}$$

$$M^{(1)} = H^{(1)}\hat{A} \tag{3d}$$

$$H^{(2)} = \tanh\left(W^{(1)}M^{(1)}\right) \tag{3e}$$

$$M^{(2)} = H^{(2)}\hat{A} \tag{3f}$$

$$H^{(3)} = \tanh\left(W^{(2)}M^{(2)}\right) \tag{3g}$$

$$M^{(3)} = H^{(3)}\hat{A} \tag{3h}$$

$$H^{(4)} = \text{softmax}\left(W^{(4)}M^{(4)}\right) \tag{3i}$$

We add a softmax layer and apply the cross-entropy loss

$$\mathcal{L} = -\sum_{l \in \mathcal{Y}_L} \sum_f Y_{fl} \ln Z_{fl} \qquad (4)$$

where

1. $\mathcal{Y}_L$ contains indices of nodes that are labelled
2. $f \in \{0, 1, 2, 3\}$ indexes the node classes in Zachary
3. $Y_{fl} \in \{0, 1\}$ indicates class membership of node l
4. $Z_{fl} = H_{fl}^{(4)} = \mathsf{softmax}(h_{fl}^{(4)})$ is normalised output data.
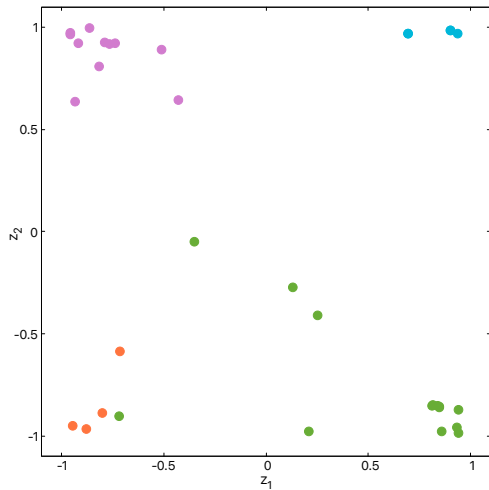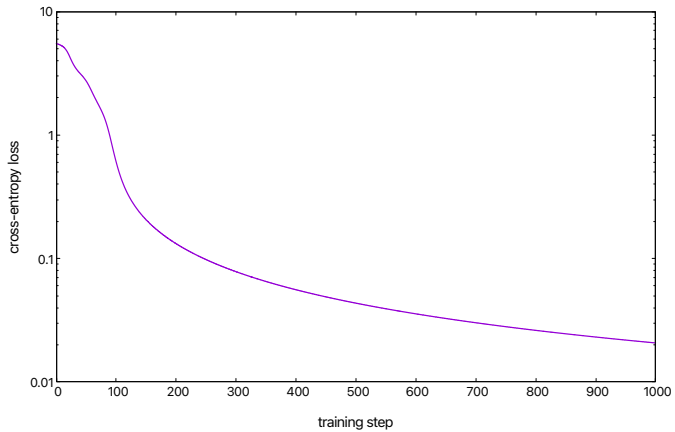
Figure 4: Semi-supervised embedding of Zachary in two dimensions

Change in $\mathcal{L}$ due to change in w_{ji} is

$$\frac{\partial \mathcal{L}}{\partial w_{ji}} = \sum_\alpha \frac{\partial \mathcal{L}}{\partial h_j^\alpha} \frac{\partial h_j^\alpha}{\partial w_{ji}} = \sum_\alpha \delta_j^\alpha m_i^\alpha, \tag{5}$$

where α runs over node indices $\{1, \ldots, |\mathcal{V}|\}$. Then

$$\begin{aligned}
\delta_j^\alpha \equiv \frac{\partial \mathcal{L}}{\partial h_j^\alpha} &= \sum_k \frac{\partial \mathcal{L}}{\partial h_k^\alpha} \frac{\partial h_k^\alpha}{\partial h_j^\alpha} + \sum_\beta \sum_k \frac{\partial \mathcal{L}}{\partial h_k^\beta} \frac{\partial h_k^\beta}{\partial h_j^\alpha} \\
&= \sum_k \delta_k^\alpha \frac{\partial h_k^\alpha}{\partial h_j^\alpha} + \sum_\beta \sum_k \delta_k^\beta \frac{\partial h_k^\beta}{\partial h_j^\alpha}
\end{aligned} \tag{6}$$

where k runs over all cells from j, and $\beta \in \text{neighbours}(\alpha)$, with

$$\frac{\partial h_k^\alpha}{\partial h_j^\alpha} = w_{kj} \frac{\sigma'(h_j^\alpha)}{k_\alpha + 1} \quad \text{and} \quad \frac{\partial h_k^\beta}{\partial h_j^\alpha} = w_{kj} \frac{\sigma'(h_j^\alpha)}{\sqrt{(k_\alpha + 1)(k_\beta + 1)}}. \tag{7}$$