Machine Learning for Graphs and Sequential Data Exercise Sheet 6

Graphs: Embeddings and Classification

1 Node Embeddings

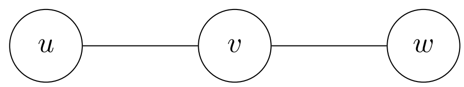

Figure 1: Undirected 3-chain for the Graph2Gauss problem

Problem 1: Consider an undirected 3-chain as in Figure 1 with three nodes u, v and w that we want to embed into $\mathbb{R}$, i.e. 1-dimensional, with Graph2Gauss. Find the embeddings analytically that we get by minimizing the training loss for a fixed embedding variance 1. So we are embedding each node as a 1-dimensional Gaussian with variance 1 by minimizing the loss

$$\mathcal{L} = E_{uv}^2 + e^{-E_{uw}} + E_{wv}^2 + e^{-E_{wu}}$$

where $E_{uv} = \mathrm{KL}(f(u)||f(v))$ is the KL divergence between the embeddings of node u and v.

Hint: The KL divergence between two normal distributions $\mathcal{N}(\mu, \sigma^2)$ and $\mathcal{N}(\nu, \tau^2)$ simplifies to

$$\mathrm{KL}\left(\mathcal{N}(\mu, \sigma^2)||\mathcal{N}(\nu, \tau^2)\right) = \log\frac{\tau}{\sigma} + \frac{\tau^2 + (\mu - \nu)^2}{2\sigma^2} - \frac{1}{2}.$$

Hint: Use the Lambert W-function to denote the inverse of $x\exp(x)$, i.e.

$$x\exp(x) = y \Rightarrow W(y) = x.$$

If you want to find a numerical solution, you can evaluate it for example on WolframAlpha with ProductLog(x).

2 Label Propagation

Problem 2: The goal in Label Propagation is to find a labeling $\boldsymbol{y} \in \{0,1\}^N$ that minimizes the energy $\min_{\boldsymbol{y}} \frac{1}{2}\sum_{ij} \boldsymbol{w}_{ij}(y_i - y_j)^2$ subject to $y_i = \hat{y}_i \ \forall i \in S$ where the set of nodes V has been partitioned into the labeled nodes S and the unlabeled nodes U, $w_{ij} \geq 0$ is the non-negative edge weight and $\hat{y}_i$ are the observed labels.

Following from the first observation regarding the Laplacian, the minimization problem can be rewritten and then relaxed to $\min_{\boldsymbol{y}\in\mathbb{R}^N} \boldsymbol{y}^T\boldsymbol{L}\boldsymbol{y}$ subject to the same constraints. Show that the closed form solution is

$$\boldsymbol{y}_U = -\boldsymbol{L}_{UU}^{-1} \cdot \boldsymbol{L}_{US} \cdot \hat{\boldsymbol{y}}_S$$

where w.l.o.g. we assume that the Laplacian matrix is partitioned into blocks for labeled and unlabeled nodes as

$$\boldsymbol{L} = \begin{pmatrix} \boldsymbol{L}_{SS} & \boldsymbol{L}_{SU} \\ \boldsymbol{L}_{US} & \boldsymbol{L}_{UU} \end{pmatrix}.$$

1 Node Embeddings

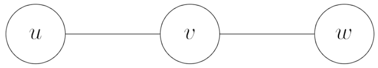

Figure 1: Undirected 3-chain for the Graph2Gauss problem

Problem 1: Consider an undirected 3-chain as in Figure 1 with three nodes u, v and w that we want to embed into $\mathbb{R}$, i.e. 1-dimensional, with Graph2Gauss. Find the embeddings analytically that we get by minimizing the training loss for a fixed embedding variance 1. So we are embedding each node as a 1-dimensional Gaussian with variance 1 by minimizing the loss

$$\mathcal{L} = E_{uv}^2 + e^{-E_{uw}} + E_{wv}^2 + e^{-E_{wu}}$$

$\sigma^2 = 1$

where $E_{uv} = \mathrm{KL}(f(u)||f(v))$ is the KL divergence between the embeddings of node u and v.

Hint: The KL divergence between two normal distributions $\mathcal{N}(\mu, \sigma^2)$ and $\mathcal{N}(\nu, \tau^2)$ simplifies to

$$\mathrm{KL}\left(\mathcal{N}(\mu,\sigma^2)||\mathcal{N}(\nu,\tau^2)\right) = \log\frac{\tau}{\sigma} + \frac{\tau^2 + (\mu-\nu)^2}{2\sigma^2} - \frac{1}{2}.$$

Hint: Use the Lambert W-function to denote the inverse of $x\exp(x)$, i.e.

$$x\exp(x) = y \Rightarrow W(y) = x.$$

If you want to find a numerical solution, you can evaluate it for example on WolframAlpha with ProductLog(x).

$$\mathrm{KL}(\mathcal{N}(\mu,1)\,||\,\mathcal{N}(v,1)) = \frac{1+(\mu-v)^2}{2} - \frac{1}{2} = \frac{(\mu-v)^2}{2}$$

$$L = \frac{(\mu-v)^4}{4} + e^{-\frac{(\mu-w)^2}{2}} + \frac{(w-v)^4}{4} + e^{-\frac{(w-\mu)^2}{2}}$$

$$\frac{\partial L}{\partial v} = -(\mu-v)^3 - (w-v)^3 = 0$$

$$v - \mu = w - v$$

$$v = \frac{w-\mu}{2}$$

2 Label Propagation

Problem 2: The goal in Label Propagation is to find a labeling $\boldsymbol{y} \in \{0,1\}^N$ that minimizes the energy $\min_{\boldsymbol{y}} \frac{1}{2}\sum_{ij} \boldsymbol{w}_{ij}(y_i - y_j)^2$ subject to $y_i = \hat{y}_i \ \forall i \in S$ where the set of nodes V has been partitioned into the labeled nodes S and the unlabeled nodes U, $w_{ij} \geq 0$ is the non-negative edge weight and $\hat{y}_i$ are the observed labels.

Following from the first observation regarding the Laplacian, the minimization problem can be rewritten and then relaxed to $\min_{\boldsymbol{y}\in\mathbb{R}^N} \boldsymbol{y}^T \boldsymbol{L}\boldsymbol{y}$ subject to the same constraints. Show that the closed form solution is

$$\boldsymbol{y}_U = -\boldsymbol{L}_{UU}^{-1} \cdot \boldsymbol{L}_{US} \cdot \hat{\boldsymbol{y}}_S$$

where w.l.o.g. we assume that the Laplacian matrix is partitioned into blocks for labeled and unlabeled nodes as

$$\boldsymbol{L} = \begin{pmatrix} \boldsymbol{L}_{SS} & \boldsymbol{L}_{SU} \\ \boldsymbol{L}_{US} & \boldsymbol{L}_{UU} \end{pmatrix}.$$

$$y = \begin{pmatrix} \hat{y} \\ y \end{pmatrix}$$

$$\min \quad (\hat{y} \ \ y) \begin{pmatrix} L_{SS} & L_{SU} \\ L_{US} & L_{UU} \end{pmatrix} \begin{pmatrix} \hat{y} \\ y \end{pmatrix} = (\hat{y} \ \ y) \begin{pmatrix} \hat{y} L_{SS} + y L_{SU} \\ \hat{y} L_{US} + y L_{UU} \end{pmatrix}$$

$$= \hat{y}^2 L_{SS} + y\hat{y} L_{SU} + y\hat{y} L_{US} + y^2 L_{UU}$$

$$\frac{\partial \cdots}{\partial \hat{y}} = 2\hat{y} L_{SS} + 2y L_{US} = 0$$

$$\frac{\partial \cdots}{\partial y} = 2\hat{y} L_{US} + 2y L_{UU} = 0$$

$$y L_{UU} = -\hat{y} L_{US}$$

$$y = -$$

3 Spectral GNNs

Problem 3: Consider the spectral GNN given by

$$\boldsymbol{Z} = \phi(\boldsymbol{U} g(\boldsymbol{\Lambda}) \boldsymbol{U}^T \varphi(\boldsymbol{X})),$$

where ϕ and φ are non-linear, parametrized functions, e.g. multi-layer perceptrons. For this exercise we choose a polynomial filter of the form

$$g(\lambda) = \sum_{k=0}^{\infty} \theta_k \lambda^k.$$

Note that instead of parametrizing the spectral filter g we can also choose fixed coefficients θ_k, for example

$$\theta_k = \frac{(-t)^k}{k!}$$

where $t > 0$ is a hyperparameter that we can fine-tune.

Show that this choice of g constraints the possible graph filters.

4 PPNP

Problem 4: The iterative equation of PPNP is given by

$$\boldsymbol{H}^{(l+1)} = (1-\alpha)\hat{\boldsymbol{A}}\boldsymbol{H}^{(l)} + \alpha \boldsymbol{H}^{(0)}$$

where $\hat{\boldsymbol{A}} = \tilde{\boldsymbol{D}}^{-\frac{1}{2}} \tilde{\boldsymbol{A}} \tilde{\boldsymbol{D}}^{-\frac{1}{2}}$ is the propagation matrix. Derive the closed form solution for infinitely many propagation steps.

Hint: If we have for a matrix $\boldsymbol{T}$ that all its eigenvalues λ are strictly between -1 and 1, an equivalent matrix formulation of the geometric series formula holds and

$$\sum_{k=0}^{\infty} \boldsymbol{T}^k = (\boldsymbol{I} - \boldsymbol{T})^{-1}.$$

Hint: The eigenvalues λ_i of any normalized Laplacian $\boldsymbol{L} = \boldsymbol{I} - \boldsymbol{D}^{-\frac{1}{2}} \boldsymbol{A} \boldsymbol{D}^{-\frac{1}{2}}$ are $0 \leq \lambda_i \leq 2$.

3 Spectral GNNs

Problem 3: Consider the spectral GNN given by

$$\boldsymbol{Z} = \phi(\boldsymbol{U} g(\boldsymbol{\Lambda}) \boldsymbol{U}^T \varphi(\boldsymbol{X})),$$

where ϕ and φ are non-linear, parametrized functions, e.g. multi-layer perceptrons. For this exercise we choose a polynomial filter of the form

$$g(\lambda) = \sum_{k=0}^{\infty} \theta_k \lambda^k.$$

Note that instead of parametrizing the spectral filter g we can also choose fixed coefficients θ_k, for example

$$\theta_k = \frac{(-t)^k}{k!}$$

where $t > 0$ is a hyperparameter that we can fine-tune.

Show that this choice of g constraints the possible graph filters.

$$g(\lambda) = \sum_{k=0}^{\infty} \frac{(-t\lambda)^k}{k!} = e^{-t\lambda} \Rightarrow$$

$$\lambda_i < \lambda_j \qquad g(\lambda_i) > g(\lambda_j) \Rightarrow \text{low pass}$$

4 PPNP

Problem 4: The iterative equation of PPNP is given by

$$\boldsymbol{H}^{(l+1)} = (1-\alpha)\hat{\boldsymbol{A}}\boldsymbol{H}^{(l)} + \alpha\boldsymbol{H}^{(0)}$$

where $\hat{\boldsymbol{A}} = \tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}$ is the propagation matrix. Derive the closed form solution for infinitely many propagation steps.

Hint: If we have for a matrix $\boldsymbol{T}$ that all its eigenvalues λ are strictly between -1 and 1, an equivalent matrix formulation of the geometric series formula holds and

$$\sum_{k=0}^{\infty} \boldsymbol{T}^k = (\boldsymbol{I} - \boldsymbol{T})^{-1}.$$

Hint: The eigenvalues λ_i of any normalized Laplacian $\boldsymbol{L} = \boldsymbol{I} - \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{A}\boldsymbol{D}^{-\frac{1}{2}}$ are $0 \leq \lambda_i \leq 2$.

$$H^{(1)} = (1-\alpha)\hat{A}H^0 + \alpha H^0$$

$$H^{(2)} = (1-\alpha)\hat{A}H^1 + \alpha H^0 = (1-\alpha)^2\hat{A}^2H^0 + (1-\alpha)\alpha\hat{A}H^0 + \alpha H^0$$

$$H^{(3)} = (1-\alpha)\hat{A}H^2 + \alpha H^0 = (1-\alpha)^3\hat{A}^3H^0 + (1-\alpha)^2\alpha\hat{A}^2H^0 + (1-\alpha)\alpha\hat{A}H^0 + \alpha H_0$$

$$H^n = (1-\alpha)^n\hat{A}^nH^0 + \alpha\sum_{i=0}^{n-1}(1-\alpha)^i\hat{A}^iH^0$$

$$= [(1-\alpha)\hat{A}]^nH^0 + \sum_{i=0}^{n-1}[(1-\alpha)\hat{A}]^i\alpha H^0$$