Summary Questions of the last lecture

- What is the key idea to mitigate the scalability problem of SSL for millions of data?

→ We approximate the graph signal by a linear combination of only the k smoothest eigenvectors with k Fourier coefficients and then we the k Fourier coefficients are used for optimization variables instead of the N elements of the graph signal.

$$\boldsymbol{f}^* = \underset{\boldsymbol{f}\in\mathbb{R}^{n_l+n_u}}{\operatorname{argmin}} (\boldsymbol{f}-\boldsymbol{y})^T \boldsymbol{C}(\boldsymbol{f}-\boldsymbol{y}) + \boldsymbol{f}^T \boldsymbol{L}\boldsymbol{f}$$

$$\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T \text{ and } \boldsymbol{f} = \boldsymbol{U}\boldsymbol{\alpha}$$

$$\boldsymbol{\alpha}^* = \underset{\boldsymbol{\alpha}\in\mathbb{R}^{n_l+n_u}}{\operatorname{argmin}} (\boldsymbol{U}\boldsymbol{\alpha}-\boldsymbol{y})^T \boldsymbol{C}(\boldsymbol{U}\boldsymbol{\alpha}-\boldsymbol{y}) + \boldsymbol{\alpha}^T \boldsymbol{\Lambda}\boldsymbol{\alpha}$$

only first k eigenvectors in $\boldsymbol{f} = \bar{\boldsymbol{U}}\bar{\boldsymbol{\alpha}}$!

$$\bar{\boldsymbol{\alpha}}^* = (\bar{\boldsymbol{\Lambda}} + \bar{\boldsymbol{U}}^T \boldsymbol{C}\bar{\boldsymbol{U}})^{-1}\bar{\boldsymbol{U}}^T \boldsymbol{C}\boldsymbol{y}$$

It requires $k \times k$ *matrix inversion*

Summary Questions of the lecture

- What happens to $\boldsymbol{L}$ when $N \to \infty$?

→ As $N \to \infty$, Laplacian ($\boldsymbol{L}$) smoothness normalized by N^2 over a graph signal $\boldsymbol{f}$ defined on data set $\{x_i\}$ sampled from $p(x)$ converges to a weighted smoothness operator on a continuous function $F(x)$ defined over a measure space $\mathcal{X}$.

$$\mathcal{L}_p(F) = \frac{1}{2}\int \big(F(\boldsymbol{x}_1) - F(\boldsymbol{x}_2)\big)^2 W(\boldsymbol{x}_1, \boldsymbol{x}_2) p(\boldsymbol{x}_1) p(\boldsymbol{x}_2) d\boldsymbol{x}_1 d\boldsymbol{x}_2$$

where $W(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp(-\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2)/(2\sigma^2)$

As $N \to \infty$, $\frac{1}{N^2}\boldsymbol{f}^T\boldsymbol{L}\boldsymbol{f} = \frac{1}{2N^2}\sum_{i,j} W_{ij}\big(f_i - f_j\big)^2 \to \mathcal{L}_p(F)$

Summary Questions of the lecture

- What does the weighted smoothness operator in Hilbert space define, instead of eigenvectors of Laplacian L?

→ It defines eigenfunctions, each of which represents a function having a unique frequency (eigenvalue) in the Hilbert space.

First eigenfunction: $\phi_1(\boldsymbol{x}) = \min_{F:\int F^2(\boldsymbol{x})p(\boldsymbol{x})D(\boldsymbol{x})d\boldsymbol{x}=1} \mathcal{L}_p(F)$

What is the solution? $\phi_1(\boldsymbol{x}) = 1$ because $\mathcal{L}_p(1) = 0$

How to define $\phi_2(\boldsymbol{x})$? same, constraining to be orthogonal to $\phi_1(\boldsymbol{x})$

$$\int F(\boldsymbol{x})\phi_1(\boldsymbol{x})\, p(\boldsymbol{x})D(\boldsymbol{x})d\boldsymbol{x} = 0$$

How to define eigenvalues? $\lambda_k = \mathcal{L}_p(\phi_k)$

Summary Questions of the lecture

- How can we obtain numerical eigenfunctions with sampled large dada?

→ After rotating the coordinate of the sample space using PCA, we divide each coordinate into B number of bins, and determine the discrete probability distribution by counting the number of nodes (samples) that fall into each bin. Then, we solve the generalized eigenvalue problem to obtain an approximate eigenfunction having a constant in each bin.

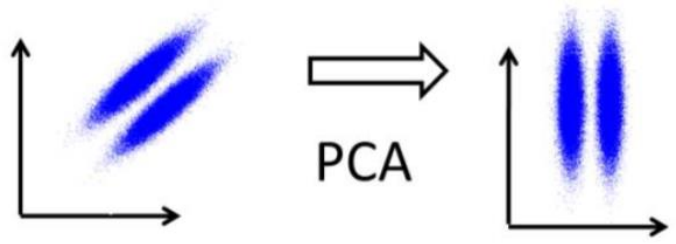
Find $\boldsymbol{R}$ such that $\boldsymbol{s} = \boldsymbol{Rx}$

For each "independent" s_k approximate $p(s_k)$

Given $p(s_k)$ numerically solve for eigensystem of $\mathcal{L}_{p_k}$

$$\left(\widetilde{\boldsymbol{D}} - \boldsymbol{P}\widetilde{\boldsymbol{W}}\boldsymbol{P}\right)g = \lambda\widetilde{\boldsymbol{D}}g$$

g – vector of length B ≡ number of bins



Summary Questions of the lecture

- How can we determine eigenvectors from the numerical eigenfunctions?

→ Each element value of an eigenvector is assigned by the eigenfunction value of the bin where the corresponding sample (node) falls into.