ch1: Graph Theory

理解:

$$G = \{V, E\}$$
$$V = \{v_1, v_2, v_3, v_4\}$$
$$E = \{e_1, e_2, e_3\}$$

$$D = \begin{bmatrix} 3 & & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \qquad A = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & & \\ 1 & & 0 & \\ 1 & & & 0 \end{bmatrix}$$

$$L = D - A = \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 1 & & \\ -1 & & 1 & \\ -1 & & & 1 \end{bmatrix}$$

$$L_{sym} = D^{-1/2} L D^{-1/2} = I - D^{-1/2} A D^{-1/2}$$

$$L_{rw} = D^{-1} L = I - D^{-1} A \qquad \text{random walk Laplacian ?}$$

$$|\lambda I - L| = 0$$
$$Lu = \lambda u$$
$$L = U \Lambda U^T = \sum_{1}^{N} \lambda_k u_k \cdot u_k^T$$

ch2: Graph Signal Processing

$$\forall s \in R^N$$

$$u_k^T \cdot s = \frac{u_k^T \cdot s}{|u_k^T|} = |s| \cos\theta_k = p_k$$

$$p = U^T s \qquad\qquad s = Up$$

$$TV(s) = s^T L s = s^T U \Lambda U^T s = (U^T s)^T \Lambda (U^T s) = p^T \Lambda p = \sum_1^N p_k^2 \lambda_k \geq 0$$

$$E(s) = |s|^2 = (Up)^T \cdot (Up) = p^T p = \sum_1^N p_k^2$$

$$s_{in}\left(\sum_1^N p_k u_k\right) \to H \to s_{out}\left(\sum_1^N p_k^{'} u_k\right)$$

$$s_{in}\left(\sum_1^N p_k u_k\right) \to H(filter) \to s_{out}\left(\sum_1^N h(\lambda_k) p_k u_k\right)$$

$$s_{out} = H s_{in} = \sum_1^N h(\lambda_k) p_k u_k$$

$$= \begin{bmatrix} u_1 & u_2 & ... & u_n \end{bmatrix} \begin{bmatrix} h\left(\lambda_1\right) p_1 \\ h\left(\lambda_2\right) p_2 \\ ... \\ h\left(\lambda_n\right) p_n \end{bmatrix} = U \begin{bmatrix} h\left(\lambda_1\right) & & & \\ & h\left(\lambda_2\right) & & \\ & & ... & \\ & & & h\left(\lambda_n\right) \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ ... \\ p_n \end{bmatrix}$$

$$= U \begin{bmatrix} h\left(\lambda_1\right) & & & \\ & h\left(\lambda_2\right) & & \\ & & ... & \\ & & & h\left(\lambda_n\right) \end{bmatrix} U^T \overrightarrow{s_{in}}$$

$$H = U \begin{bmatrix} h(\lambda_1) & & & \\ & h(\lambda_2) & & \\ & & ... & \\ & & & h(\lambda_n) \end{bmatrix} U^T = U \Lambda_h U^T$$

$$\Lambda_h = \lim_{K \to \infty} \sum_0^K h_k \Lambda^k \qquad K \ll N$$

$$H = U \left(h_0 \Lambda^0 + h_1 \Lambda^1 + h_2 \Lambda^2 + ... + h_K \Lambda^K \right) U^T$$

$$= U \left(h_0 \Lambda^0 \right) U^T + U \left(h_1 \Lambda^1 \right) U^T + U \left(h_2 \Lambda^2 \right) U^T + ... + U \left(h_K \Lambda^K \right) U^T$$

$$= h_0 L^0 + h_1 L^1 + h_2 L^2 + ... + h_K L^K = \sum_0^K h_k L^k$$

$$K = 1 \qquad\qquad H = h_0 L^0 + h_1 L^1$$

ch2: Hypergraph Learning Architecture

(2) Transformation

Reductive Transformation

$(E, X, Y) \Rightarrow A$ hyperedges to edges

clique expansion + adaptive expansion

Non-reductive Transformation

star/line/tensor expansion

(3) Message

whose: v-v v-e e-v

what : e-consistent + e-dependent

how : fixed-pooling + learnable-pooling

(4) Training