April 7, 2020

Before: "unsupervised" learning on graphs

$L = D - A \qquad N = D^{-1/2} L D^{-1/2} = I - D^{-1/2} A D^{-1/2}$

$$\left.\begin{aligned} Lx &= \lambda_2 x \\ Nx &= \lambda_2 x \end{aligned}\right\}$$

clusters based on sign

"sweep cut"

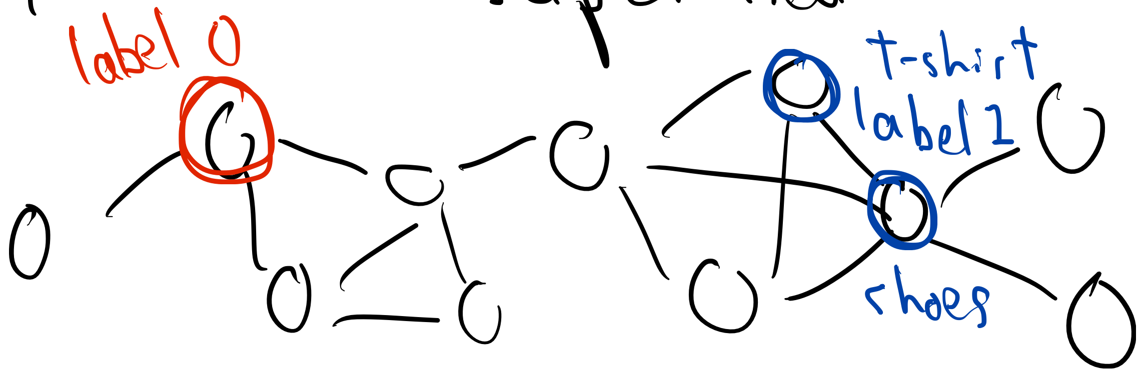
$$\left.\begin{aligned} LX_k &= \Lambda_k X_k \\ NX_k &= \Lambda_k X_k \end{aligned}\right\}$$

$n \overbrace{\boxed{X_k}}^{k}$

node $\to$ point in $\mathbb{R}^k$

cuts, volumes as motivation

Today: "semi-supervised" learning



$G = (V, E)$

$H \subseteq V$ labelled

0/1

Goal: For $U = V \setminus H$ assign 0/1

Remember: $x^T L x = \sum_{(i,j) \in E} (x_i - x_j)^2$

$$\min_{x \in \mathbb{R}^n} \quad x^T L x \Big\} \text{ smoothness}$$

s.t. $\quad x_h = \text{label}(x_h), \; h \in H$

$\text{label}(x_h) \rightarrow \ell_h$

(Zhu, Ghahramani, Lafferty '03)

Lagrangian: $\mathcal{L}(x, \lambda) = x^T L x - \sum_{h \in H} \lambda_h (x_h - l_h)$

$$\nabla \mathcal{L}(x^*, \lambda^*) = 0$$

$$\nabla_{x_u} \mathcal{L}(x, \lambda) = 2(Lx)_u$$

$$= 2[(Dx)_u - (Ax)_u]$$

$$= 0 \Rightarrow d_u x_u - \sum_v A_{uv} x_v = 0$$

$$x_u = \frac{1}{d_u} \sum_{(u,v) \in E} x_v$$