Simultaneous Graph Learning and Matrix Recovery

Seyyid Emre Sofuoglu, Selin Aviyente

March 28, 2022

1 Methods

Our aim is to learn the underlying manifold of a data on a Cartesian Product Graph and recovery missing or grossly corrupted entries. The objective can be written in convex terms as:

$$\begin{aligned}&\text{minimize}_{L,S,\Phi_1,\Phi_2}\text{tr}(L^\top\Phi_1 L)+\text{tr}(L\Phi_2 L^\top)+\|S\|_1,\\&\text{s.t.}\mathcal{P}_\Omega[Y]=\mathcal{P}_\Omega[L+S],\ \Phi_1,\Phi_2\in\mathcal{S},\end{aligned}\tag{1}$$

where Φ_1,Φ_2 are Graph Laplacians for the row graph and column graph, respectively, and $\mathcal{S}$ is the space of undirected graph Laplacians. When the graph constraints are explicitely written, (1) becomes:

$$\begin{aligned}&\text{minimize}_{L,S,\Phi_1,\Phi_2}\alpha_1\text{tr}(L^\top\Phi_1 L)+\alpha_2\text{tr}(L\Phi_2 L^\top)+\|S\|_1+\|\Phi_1\|_F^2+\|\Phi_2\|_F^2,\\&\text{s.t.}\mathcal{P}_\Omega[Y]=\mathcal{P}_\Omega[L+S],\ \Phi_i=\Phi_i^\top,\ \Phi_i\mathbf{1}=0,\ \text{tr}(\Phi_i)=2I_i.\end{aligned}\tag{2}$$

Since the graph Laplacians are assumed to be symmetric, the upper triangle portion is enough to summarize the conditions on them. This allows us to rewrite (2) as:

$$\begin{aligned}&\text{minimize}_{L,S,\phi_i\leq 0,\mathbf{d}_{\phi_i}}\sum_{i=1}^{2}\alpha_i\left(2\mathbf{l}_i^\top\phi_i+\mathbf{d}_{\mathbf{l}_i}^\top\mathbf{d}_{\phi_i}+\beta_{1,i}f(\mathbf{d}_{\phi_i})+\beta_{2,i}g(\phi_i)\right)+\|S\|_1,\\&\text{s.t.}\mathcal{P}_\Omega[Y]=\mathcal{P}_\Omega[L+S],\ P_i\phi_i=-\mathbf{d}_{\phi_i},\end{aligned}\tag{3}$$

where $\mathbf{l}_i=\text{upper}(L_{(i)}L_{(i)}^\top)$ $\phi_i=\text{upper}(\Phi_i)$, $\mathbf{d}_{l_i}=\text{diag}(L_{(i)}L_{(i)}^\top)$, $\mathbf{d}_{\phi_i}=\text{diag}(\Phi_i)$, and $P_i\in\mathbb{R}^{I_i\times I_i(I_i-1)/2}$ is a matrix such that $P_i\text{upper}(W_i)=W_i\mathbf{1}-\text{diag}(W_i)$. $f(.)$ is a function that controls the degree distribution and $g(.)$ is a function that controls the sparsity of learned graph.

1.1 Optimization

(2) will be solved using ADMM. For (2) to be optimized, the graph variable L can be separated into two variables L_i, for $i\in\{1,2\}$ to avoid large inverses. Corresponding augmented Lagrangian is written as:

$$\begin{aligned}&\|S\|_1+\lambda_1\|\mathcal{P}_\Omega[Y-L-S-\Gamma_1]\|_F^2+\sum_{i=1}^{2}\alpha_i\text{tr}(L_i^\top\Phi_i L_i)+\beta_{1,i}f(\mathbf{d}_{\phi_i})+\beta_{2,i}g(\phi_i)+\lambda_{2,i}\|L_{(i)}-L_i-\Gamma_{2,i}\|_F^2\\&+\lambda_{3,i}\|P_i\phi_i+\mathbf{d}_{\phi_i}-\Gamma_{3,i}\|_F^2+\lambda_{4,i}(\mathbf{1}^\top\mathbf{d}_{\phi_i}-I_i-\gamma_{4,i})^2\end{aligned}\tag{4}$$

The variables are updated according to the following:

$$\mathcal{P}_\Omega[L^{t+1}]=\mathcal{P}_\Omega\left[\frac{1}{\lambda_1+\sum_{i=1}^2\lambda_{2,i}}\left(\lambda_1(Y-S^t-\Gamma_1^t)+\sum_{i=1}^{2}\lambda_{2,i}(L_{(i)}^t-\Gamma_{2,i}^t)\right)\right].\tag{5}$$

$$\mathcal{P}_{\Omega^\perp}[L^{t+1}]=\mathcal{P}_{\Omega^\perp}\left[\sum_{i=1}^{2}\frac{\lambda_{2,i}}{\sum_{i=1}^2\lambda_{2,i}}(L_{(i)}^t-\Gamma_{2,i}^t)\right]\tag{6}$$

$$S^{t+1} = \sigma(Y - L^{t+1} - \Gamma_1^t, 2/\lambda_1), \tag{7}$$

where $\sigma(.)$ is the soft thresholding operator.

$$L_i^{t+1} = L_{i,inv}^t(L_{(i)}^{t+1} - \Gamma_{2,i}^t), \tag{8}$$

where $L_{i,inv} = (\mathbf{I} + \frac{\alpha_i}{\lambda_{2,i}}\Phi_i^t)^{-1}$.

$$\phi_i^{t+1} = \Pi_{\mathbb{R}_-^{I_i(I_i-1)/2}}\left[-\left[\beta_{2,i}\mathbf{I} + \lambda_{3,i}P_i^\top P_i\right]^{-1}\left(2\alpha_i \mathbf{l}_i^{t+1} + \lambda_{3,i}P^\top\left(\mathbf{d}_{\phi_i}^t - \Gamma_{3,i}^t\right)\right)\right], \tag{9}$$

where $\Pi_{\mathbb{R}_-^{I_i(I_i-1)/2}}[.]$ is an operator that projects to the negative orthant.

$$\mathbf{d}_{\phi_i}^{t+1} = \left[(\beta_{1,i} + \lambda_{3,i})\mathbf{I} + \lambda_{4,i}\mathbf{1}\mathbf{1}^\top\right]^{-1}\left[\lambda_{4,i}(I_i + \gamma_{4,i}^t)\mathbf{1} - \alpha_i\mathbf{d}_{\mathbf{l}_i}^{t+1} - \lambda_{3,i}P_i\phi_i^{t+1} + \lambda_{3,i}\Gamma_{3,i}^t\right] \tag{10}$$

Finally, dual variables are updated as follows:

$$\Gamma_1^{t+1} = \Gamma_1^t - \mathcal{P}_\Omega[Y - L^{t+1} - S^{t+1}] \tag{11}$$
$$\Gamma_{2,i}^{t+1} = \Gamma_{2,i}^t - (L_{(i)}^{t+1} - L_i^{t+1}) \tag{12}$$
$$\Gamma_{3,i}^{t+1} = \Gamma_{3,i}^t - (P_i\phi_i + \mathbf{d}_{\phi_i}) \tag{13}$$
$$\gamma_{4,i}^{t+1} = \gamma_{4,i}^t - (\mathbf{1}^\top\mathbf{d}_{\phi_i} - I_i) \tag{14}$$