Network Lasso によるクラスタリングについて
–A Study of Clustering Based on the Network Lasso–

経営システム工学専攻 柳下翔太郎

1　はじめに

データ解析において，平均や回帰などの構造とクラスター構造を同時に推定したいという状況がしばしば生じる．近年，Hallac ら [1] が提案した Network Lasso（NL）はそれを実現する方法の 1 つであり，スパースモデリングのテクニックを利用した方法である．$\mathcal{V} = [n] := \{1, \ldots, n\}$ をサンプルのインデックスの集合，$\mathcal{E} \subset \{\{i, j\} : i, j \in [n]; i \neq j\}$ をサンプル間の隣接の有無を表す集合，$W = (w_{\{i,j\}})_{\{i,j\} \in \mathcal{E}} \in \mathbb{R}^{|\mathcal{E}|}_{\geq 0}$ を隣接したサンプル間の類似度を表す重みとし，重み付き無向グラフ $\mathcal{G} = (\mathcal{V}, \mathcal{E}, W)$ を考える．$f_i : \mathbb{R}^p \to \mathbb{R}$ をサンプル $i \in [n]$ に対する損失関数とし，NL は以下のように定式化される．

$$\underset{x_1,\ldots,x_n \in \mathbb{R}^p}{\text{minimize}} \quad \sum_{i \in \mathcal{V}} f_i(x_i) + \gamma \sum_{\{i,j\} \in \mathcal{E}} w_{\{i,j\}} \|x_i - x_j\|_2. \tag{1}$$

ただし，$\gamma > 0$ は定数，$\|\cdot\|_2$ は ℓ_2 ノルムである．直感的には，第 1 項はサンプルごとのフィッティングを強め，第 2 項は隣接したサンプルのパラメータを重みが大きいほど近づける効果を持つ．このことから，重み W はサンプル間の類似度に関する事前情報と解釈できる．さらに，Group Lasso のもつスパース性から，最適解 $(x_1^*, \ldots, x_n^*)$ において多くの $\{i, j\} \in \mathcal{E}$ に対して $x_i^* = x_j^*$ となることを期待し，そうなったとき，i と j が同じクラスターに属していると判定する．たとえば，$b_i \in \mathbb{R}$ を目的変数，$a_i \in \mathbb{R}^p$ を説明変数とし，$f_i(x_i) = \frac{1}{2}(b_i - a_i^\top x_i)^2$ とすれば，回帰構造とクラスター構造の同時推定，$a_i \in \mathbb{R}^p$ をデータ点とし，$f_i(x_i) = \frac{1}{2}\|x_i - a_i\|_2^2$ とすれば，平均構造とクラスター構造の同時推定を考えることができる．$\gamma \to \infty$ とし正則化パスを考えることで，階層クラスタリングにおけるデンドログラムのようなクラスターパスを生成できる（図 1）．

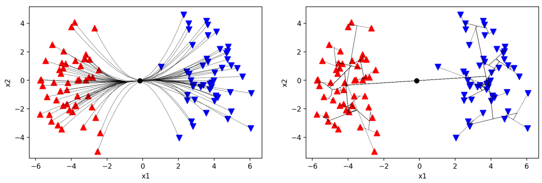

図 1　$\mathbb{R}^2$ 上での Network Lasso によるクラスタリングの失敗例（左）と提案する Network Trimmed Lasso による成功例（右）．三角の点がデータ点 a_i，細い曲線が x_i^* の軌跡，すなわちクラスターパスを表している．

本稿ではまず，NL が潜在的な真のクラスターを復元するための十分条件を与える．次に，NL の持つ弱点を克服するような定式化を提案する．さらに，それに対して交互方向乗数法（ADMM）（たとえば [2]）を適用しその収束性を示す．最後に，その定式化の有効性を数値実験により検討する．

2　Network Lasso のクラスター復元

この節では，NL が真のクラスターを復元するための十分条件を与える．これは Sun ら [3] が CC に対して与えた十分条件の NL に対する拡張である．

この節では，各サンプル $i \in \mathcal{V}$ が N 個の真のクラスター $C_1, \ldots, C_N (C_i \cap C_j = \emptyset, C_1 \cup \cdots \cup C_N = \mathcal{V})$ のうちの 1 つに入っていることを仮定する．

定義 1. $\mathcal{P} := \{C_1, \ldots, C_N\}$ と $\overline{\mathcal{P}} := \{\overline{C}_1, \ldots, \overline{C}_M\}$ を $\mathcal{V}$ の分割とする．

1. $\overline{\mathcal{P}} = \mathcal{P}$ のとき，$\overline{\mathcal{P}}$ は $\mathcal{P}$ を完全に復元する，という．
2. 任意の $\overline{C} \in \overline{\mathcal{P}}$ に対して，ある $I \subset \{1, \ldots, N\}$ が存在して，$\overline{C} = \cup_{l \in I} C_l$ となるとき，$\overline{\mathcal{P}}$ は $\mathcal{P}$ の粗化である，という．さらに $\overline{\mathcal{P}} = \{\mathcal{V}\}$ のとき，自明な粗化である，という．そうでないとき，非自明な粗化である，という．

真のクラスター $\mathcal{P} = \{C_1, \ldots, C_N\}$ と (1) の重み $w_{\{i,j\}}$ に対して，いくつかの記号を定義する．

$$\begin{aligned} n_k &:= |C_k|, & k &\in [N], \\ w_i^{(k)} &:= \sum_{j \in C_k} w_{\{i,j\}}, & i &\in \mathcal{V}, k \in [N], \\ w^{(k,k')} &:= \sum_{i \in C_k} \sum_{j \in C_{k'}} w_{\{i,j\}}, & k, k' &\in [N]. \end{aligned}$$

簡単のために，$\{i,j\} \notin \mathcal{E}$ に対しては $w_{\{i,j\}} = 0$ とした．

定理 1. 任意の $i \in \mathcal{V}$ に対して，f_i は狭義凸かつ L_i-平滑であるとする．$f^{(k)} := \sum_{i \in C_k} f_i$ とし，任意の $k \in [N]$ に対し，$f^{(k)}$ は α_k-強凸であるとし，$\overline{x}^{(k)} = \operatorname*{argmin}_{x \in \mathbb{R}^p} f^{(k)}(x)$ とおく．任意の $k \neq k'$ に対し，$\overline{x}^{(k)} \neq \overline{x}^{(k')}$ を仮定する．任意の $i, j \in C_k, k \in [N]$ に対し

$$\mu_{ij}^{(k)} = \sum_{l \neq k} \left| w_i^{(l)} - w_j^{(l)} \right| + \frac{2\,(L_i + L_j)}{\alpha_k} \sum_{l \neq k} w^{(k,l)}$$

とし，$n_k w_{\{i,j\}} > \mu_{ij}^{(k)}$ を仮定する．

$$\gamma_{\max} := \min_{k \neq k'} \left\{ \frac{\|\overline{x}^{(k)} - \overline{x}^{(k')}\|_2}{\frac{2}{\alpha_k} \sum_{l \neq k} w^{(k,l)} + \frac{2}{\alpha_{k'}} \sum_{l \neq k'} w^{(k',l)}} \right\},$$

$$\gamma_{\min} := \max_{k} \max_{i,j \in C_k} \left\{ \frac{\left\| \nabla f_j(\overline{x}^{(k)}) - \nabla f_i(\overline{x}^{(k)}) \right\|_2}{n_k w_{\{i,j\}} - \mu_{ij}^{(k)}} \right\},$$

とする．ただし，$a > 0$ に対して $\frac{a}{0} = \infty$ とする．

$(x_1^*, ..., x_n^*)$ を (1) の最適解とし，$\overline{\mathcal{P}}$ を同値関係 $x_i^* = x_j^*$ から定まる $\mathcal{V}$ の商集合とする．

1. $\gamma_{\min} \leq \gamma < \gamma_{\max}$ ならば，$\overline{\mathcal{P}}$ は $\mathcal{P}$ を完全に復元する．
2. $\gamma_{\min} \leq \gamma < \max_k \frac{\|\nabla f^{(k)}(\overline{x})\|_2}{\sum_{l \neq k} w^{(k,l)}}$ ならば，$\overline{\mathcal{P}}$ は $\mathcal{P}$ の非自明な粗化である．ただし，

$$\overline{x} := \operatorname*{argmin}_{x \in \mathbb{R}^p} \sum_{k \in [N]} f^{(k)}(x) = \operatorname*{argmin}_{x \in \mathbb{R}^p} \sum_{i \in \mathcal{V}} f_i(x).$$

である．

定理 1 から，事前情報がうまく与えられれば，γ を $[\gamma_{\min}, \gamma_{\max})$ 上でとることで，NL が真のクラスター $\{C_1, ..., C_N\}$ を復元できることがわかる．

3 Network Trimmed Lasso

NL では，上述の十分条件が成立しない，特に，重み $w_{\{i,j\}}$ が「うまく」与えられていないとき，図 1 左のように，単純な数値例においてさえ厳密に $x_i^* = x_j^*$ とならず，真のクラスターどころかクラスターを形成することさえできない．これを解決するために，以下のような基数制約による定式化を考える．

$$\begin{aligned} &\underset{x_1, \ldots, x_n}{\text{minimize}} & &\sum_{i \in \mathcal{V}} f_i(x_i) & (2) \\ &\text{subject to} & &\left| \left\{ \{i,j\} \in \mathcal{E} : \|x_i - x_j\|_2 > 0 \right\} \right| \leq K. \end{aligned}$$

ただし，K は $K \leq |\mathcal{E}|$ であるような非負整数である．$x_i^* = x_j^*$ となるような制約を直接組み込むことで, クラスターが形成される．さらに，K の減少につれて，$x_i^* = x_j^*$ となるエッジが増え, クラスターパスの生成が期待できる．しかし, (2) の基数制約の左辺は $(x_1, ..., x_n)$ の不連続関数であり，一般に，問題 (2) は大域的な解を得ることが難しい．このような事情から，まず基数制約を連続関数による等価な制約に書き換える．$\boldsymbol{z} = (z_{\{i,j\}})_{\{i,j\} \in \mathcal{E}} \in \mathbb{R}^{p|\mathcal{E}|}$ とし，$\|z_{\{i,j\}}\|_2$ の小さいほうから $|\mathcal{E}| - K$ 個の要素の和を

$$T_K(\boldsymbol{z}) = \|z_{(K+1)}\|_2 + \cdots + \|z_{(|\mathcal{E}|)}\|_2$$

とおく．ここで，$\|z_{(i)}\|_2$ は $(\|z_1\|_2, ..., \|z_{|\mathcal{E}|}\|_2) \in \mathbb{R}^{|\mathcal{E}|}$ の大きいほうから i 番目の要素を表す．$\boldsymbol{x} := (x_i)_{i \in \mathcal{V}}$ とし，D を $z_{\{i,j\}} = x_i - x_j$ となるような

$p|\mathcal{E}| \times pn$ 行列とすれば，(2) は以下の最適化問題と等価である [4]．

$$\begin{aligned} \underset{x_1,\ldots,x_n}{\text{minimize}} \quad & \sum_{i\in\mathcal{V}} f_i(x_i) \\ \text{subject to} \quad & T_K(D\boldsymbol{x}) = 0. \end{aligned} \tag{3}$$

$T_K(D\boldsymbol{x})$ は非凸であるが，$\boldsymbol{x}$ の連続関数である．さらに，(3) に対して，$\gamma > 0$ として，以下の無制約問題を考える．

$$\underset{x_1,\ldots,x_n}{\text{minimize}} \quad \sum_{i\in\mathcal{V}} f_i(x_i) + \gamma T_K(D\boldsymbol{x}). \tag{4}$$

$T_K(D\boldsymbol{x}) \geq 0$ であり，$T_K(D\boldsymbol{x}) > 0$ と $\left|\left\{\{i,j\} \in \mathcal{E} : \|x_i - x_j\|_2 > 0\right\}\right| > K$ が等価であるので，(4) の第二項は基数制約に対するペナルティ関数の役割を果たす．本稿では，(4) を Network Trimmed Lasso（NTL）と呼ぶ．次に示すように，適当な仮定の下で γ を十分大きくとれば，NTL は (3) と等価であることが示される．

定理 2. *1.* 任意の $i \in \mathcal{V}$ に対し, f_i が L_i-平滑であるとし，$\boldsymbol{x}^\gamma := (x_1^\gamma, \ldots, x_n^\gamma)$ を (4) の最適解とする．ある $C > 0$ が存在し，任意の $i \in \mathcal{V}, \gamma > 0$ に対して $\|x_i^\gamma\|_2 \leq C$ であることを仮定する．このとき，

$$\gamma > \sum_{i\in\mathcal{V}} (\|\nabla f_i(0)\|_2 + 2L_i C) \tag{5}$$

ならば，$\boldsymbol{x}^\gamma$ は (3) の最適解である．

2. f_i に L_i-平滑性に加えて凸性を仮定し，$\boldsymbol{x}^\gamma := (x_1^\gamma, \ldots, x_n^\gamma)$ を (4) の局所最適解とする．ある $C > 0$ が存在し，任意の $i \in \mathcal{V}, \gamma > 0$ に対して $\|x_i^\gamma\|_2 \leq C$ であることを仮定する．このとき，(5) が成立するならば，$\boldsymbol{x}^\gamma$ は (3) の局所最適解である．

この定理に動機を得て，基数制約付き問題 (2) の代わりに NTL (4) を解くことを考える．

4 アルゴリズム

最適化問題 (4) を解くために，ADMM の適用を考える．まず，(4) を以下の等価な線型等式制約付きの問題に書き換える．

$$\begin{aligned} \underset{\boldsymbol{x},\boldsymbol{z}}{\text{minimize}} \quad & \sum_{i\in\mathcal{V}} f_i(x_i) + \gamma T_K(\boldsymbol{z}) \\ \text{subject to} \quad & \boldsymbol{z} = D\boldsymbol{x}. \end{aligned} \tag{6}$$

双対変数を $\boldsymbol{y} \in \mathbb{R}^{p|\mathcal{E}|}$，$\rho > 0$ を定数として，(6) の拡張ラグランジュ関数を，

$$\begin{aligned} L_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y}) = & \sum_{i\in\mathcal{V}} f_i(x_i) + \gamma T_K(\boldsymbol{z}) \\ & + \boldsymbol{y}^\top(\boldsymbol{z} - D\boldsymbol{x}) + \frac{\rho}{2}\|\boldsymbol{z} - D\boldsymbol{x}\|_2^2 \end{aligned}$$

で定義する．ADMM は各反復 t で

$$\boldsymbol{z}^{t+1} \in \underset{\boldsymbol{z}}{\operatorname{argmin}}\, L_\rho(\boldsymbol{x}^t, \boldsymbol{z}, \boldsymbol{y}^t), \tag{7}$$

$$\boldsymbol{x}^{t+1} \in \underset{\boldsymbol{x}}{\operatorname{argmin}}\, L_\rho(\boldsymbol{x}, \boldsymbol{z}^{t+1}, \boldsymbol{y}^t), \tag{8}$$

$$\boldsymbol{y}^{t+1} = \boldsymbol{y}^t + \rho(\boldsymbol{z}^{t+1} - D\boldsymbol{x}^{t+1})$$

のように点列を生成する反復法である．(7) は非凸であるが解析解を導出できる．(8) が解析解を持たない場合には，$L > 0$ として，(8) を以下に置き換える．

$$\begin{aligned} \boldsymbol{x}^{t+1} \in \underset{\boldsymbol{x}}{\operatorname{argmin}} \Big\{ & \nabla f(\boldsymbol{x}^t)^\top(\boldsymbol{x} - \boldsymbol{x}^t) + \frac{L}{2}\|\boldsymbol{x} - \boldsymbol{x}^t\|_2^2 \\ & + (\boldsymbol{y}^t)^\top(\boldsymbol{z}^{t+1} - D\boldsymbol{x}) + \frac{\rho}{2}\|\boldsymbol{z}^{t+1} - D\boldsymbol{x}\|_2^2 \Big\}. \end{aligned}$$

ただし，$f(\boldsymbol{x}) = \sum_{i\in\mathcal{V}} f_i(x_i)$ である．この修正を施したものは，近接 ADMM [5] と呼ばれる．これらのアルゴリズムの収束性に関して，以下が成り立つ．

定理 3. 任意の $i \in \mathcal{V}$ に対して f_i は連続的微分可能かつ凸であると仮定する．*ADMM*（もしくは，近接 *ADMM*）によって生成された点列 $(\boldsymbol{x}^t, \boldsymbol{z}^t, \boldsymbol{y}^t)$ が $(\boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{y}^*)$ に収束することを仮定する．このとき，$\boldsymbol{x}^*$ は (4) の局所最適解である．

これらのことから，f_i が L_i-平滑かつ凸であり，ADMM（もしくは，近接 ADMM）によって生成された点列が収束すれば，基数制約付き問題 (2) の局所最適解を得ることができる．

5 数値実験

NTL (4) を $K \searrow 0$ として解のパスを考えることで，図 1 右のようなクラスターパスが生成でき

る．この節では，実験結果の 1 つを提示する．図 2 は 2 種類の単回帰の人工データに対する，NL，NTL，NL+NTL によるクラスターパスである．クラスターパスの初期値は $(x_{i,1}^0, x_{i,2}^0) = (b_i, 0) = \operatorname*{argmin}_{x \in \mathbb{R}^2} f_i(x)$ とし，NL+NTL では途中から NL から NTL に切り替えた．NL とデータセット 2 に対する NTL ではクラスタリングが失敗しているが，NL+NTL ではどちらのデータセットに対してもクラスタリングがうまくいっている．この NTL の初期値に NL の解を用いる方法は，どのような損失関数に対しても適用できる．このほかの数値実験の結果は，論文本体を参照されたい．

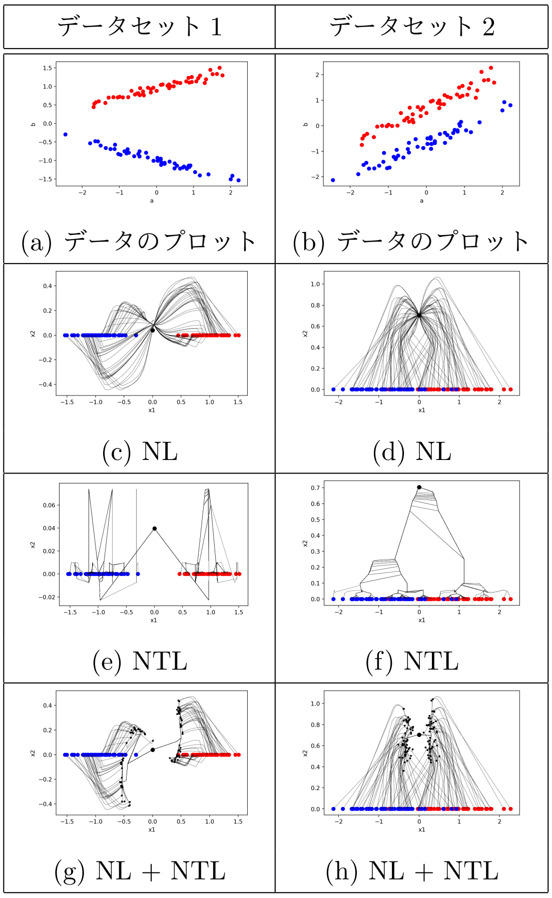

図 2　単回帰データとクラスターパス．

6　まとめ

本稿ではまず，Network Lasso が真のクラスターを復元する十分条件を与え，事前情報をうまく与えられた Network Lasso がクラスタリングに有効であることを理論的に示した．次に，Network Trimmed Lasso を提案し，それが基数制約付き問題と等価であることを示した．さらに，ADMM に基づくアルゴリズムを提案し，その収束性を示した．最後に，数値実験により Network Trimmed Lasso の有効性を示した．しかしながら，ADMM の収束性は点列の収束を仮定しており，あまり強い結果ではないので，この仮定を弱めることや，大域的収束性を示すことが今後の課題として残る．

参考文献

[1] D. Hallac, J. Leskovec, and S. Boyd, Network lasso: Clustering and optimization in large graphs, KDD, (2015), 387–396.

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., Distributed optimization and statistical learning via the alternating direction method of multipliers, Found. Trends Mach. Learn., 3 (2011), 1–122.

[3] D. Sun, K. C. Toh, and Y. Yuan. Convex clustering: Model, theoretical guarantee and efficient algorithm. arXiv preprint 1810.02677, (2018).

[4] J. Gotoh, A. Takeda, and K. Tono, DC formulations and algorithms for sparse optimization problems, Math. Program., 169 (2018), 141–176.

[5] G. Li and T. K. Pong, Global convergence of splitting methods for nonconvex composite optimization, SIAM J. Optim., 25 (2015), 2434–2460.