Network Lasso via Splitting Methods

Chen-Wei Hua

University of Minnesota, Twinn Cities

May 2, 2025

Overview

1. Introduction

2. Splitting Methods for Network Lasso
   - ADMM (Alternating Direction Method of Multipliers)
   - AMA (Alternating Minimization Algorithm)

3. Simulation study : Tau Model

Motivation: Latent Space Model

We consider a set of N individuals (or nodes) :

- Each individual/node i has an latent position u_i to be estimated.
- Nodes are connected by a similarity structure.
- Similar nodes are expected to have similar latent positions.
- Resulting objective: fitting loss f plus weighted lasso penalty.

Network LASSO

We aim to estimate latent positions $\{u_i\}_{i=1}^N$ by solving the following convex optimization problem:

$$\min_{\{u_i\}} \sum_{i=1}^{N} f_i(u_i) + \gamma \sum_{(i,j)\in E} w_{ij} \|u_i - u_j\|$$

where:

- $u_i \in \mathbb{R}^d$: latent position of individual i in a d-dimensional space.
- $f_i(u_i)$: local fitting loss for individual i.
- $G = (V, E)$: known similarity graph, with $V = [N] = \{1, \ldots, N\}$.
- $w_{ij} \geq 0$: weight representing similarity between individuals i and j.
- $\gamma > 0$: regularization parameter controlling clustering strength.

Convex Clustering and Network Lasso

- When $f_i(u_i) = \frac{1}{2}\|x_i - u_i\|_2^2$, Network Lasso reduces to convex clustering:

$$\min_{\{u_i\}} \frac{1}{2}\sum_{i=1}^{n} \|x_i - u_i\|_2^2 + \gamma \sum_{(i,j)\in E} w_{ij}\|u_i - u_j\|$$

- Fitting values shrink together as $\gamma \to \infty$.
- To perform clustering, we sweep over different values of γ and observe how the (fitted) latent positions merge.

ADMM for Network Lasso: Setting [Hallac et al., 2015]

We consider the following reformulation:

$$\min_{\{u_i\},\{z_{ij}\}} \quad \sum_{i\in\mathcal{V}} f_i(u_i) + \gamma \sum_{(i,j)\in\mathcal{E}} w_{ij}\|z_{ij} - z_{ji}\|_2$$

$$\text{subject to} \quad u_i = z_{ij}, \quad \forall i \in \mathcal{V}, \quad j \in \mathcal{N}(i)$$

The augmented Lagrangian is:

$$\mathcal{L}_\rho(u, z, \lambda) = \sum_{i\in\mathcal{V}} f_i(u_i) + \sum_{(i,j)\in\mathcal{E}} \Bigg(\gamma w_{ij}\|z_{ij} - z_{ji}\|_2 - \frac{\rho}{2}\left(\|\lambda_{ij}\|_2^2 + \|\lambda_{ji}\|_2^2\right) + \frac{\rho}{2}\left(\|u_i - z_{ij} + \lambda_{ij}\|_2^2 + \|u_j - z_{ji} + \lambda_{ji}\|_2^2\right)\Bigg)$$

ADMM for Network Lasso: Update Steps

At iteration k, we perform:

- *u*-**update:**
$$\{u_i\}^{k+1} = \arg\min_{\{u_i\}} \mathcal{L}_\rho(u, z^k, \lambda^k)$$
- *z*-**update:**
$$\{z_{ij}\}^{k+1} = \arg\min_{\{z_{ij}\}} \mathcal{L}_\rho(u^{k+1}, z, \lambda^k)$$
- **Dual update:**
$$\lambda_{ij}^{k+1} = \lambda_{ij}^k + \rho(u_i^{k+1} - z_{ij}^{k+1})$$

ADMM for Network Lasso: Update Steps

There are closed form for each update:

- *u*-**update:**

$$u_i^{k+1} = \frac{x_i + \rho \sum_{j \in \mathcal{N}(i)} (z_{ij}^k - \lambda_{ij}^k)}{1 + \rho |\mathcal{N}(i)|}$$

- *z*-**update:**

$$z_{ij}^{k+1} = (1 - \theta) v_i + \theta v_j$$

  where

$$v_i = u_i^{k+1} + \lambda_{ij}^k$$

$$\theta = \max\left(1 - \frac{2\gamma w_{ij}}{\rho \|v_i - v_j\|_2}, 0.5\right)$$

- **Dual update:**

$$\lambda_{ij}^{k+1} = \lambda_{ij}^k + \rho(u_i^{k+1} - z_{ij}^{k+1})$$

AMA for Convex Clustering: Setting [Chi and Lange, 2015]

The convex clustering problem is formulated as:

$$\min_{\{u_i\}} \quad \frac{1}{2}\sum_{i=1}^{n} \|u_i - x_i\|_2^2 + \gamma \sum_{(i,j)\in E} w_{ij}\|v_{ij}\|_2$$

$$\text{subject to } v_{ij} = u_i - u_j \ \forall \ (i,j) \in E$$

The corresponding Lagrangian is:

$$\mathcal{L}(u, v, \lambda) = \frac{1}{2}\sum_{i=1}^{n} \|u_i - x_i\|_2^2 + \gamma \sum_{(i,j)\in E} w_{ij}\|v_{ij}\|_2 + \sum_{(i,j)\in E} \lambda_{ij}^\top (u_i - u_j - v_{ij})$$

AMA does not add augmented penalty term.

AMA for Convex Clustering: General Update Rules

At each iteration k, AMA performs:

(1) u**-update:**

$$\{u_i\}^{k+1} = \arg\min_{\{u_i\}} \mathcal{L}(u, v^k, \lambda^k)$$

(2) v**-update:**

$$\{v_{ij}\}^{k+1} = \arg\min_{\{v_{ij}\}} \mathcal{L}(u^{k+1}, v, \lambda^k)$$

(3) Dual ascent step:

$$\lambda_{ij}^{k+1} = \lambda_{ij}^{k} + \nu(u_i^{k+1} - u_j^{k+1} - v_{ij}^{k+1}), \quad \forall (i,j) \in E$$

where $\nu > 0$ is the step size.

AMA for Convex Clustering: *u*-update

At each iteration, we update u_i by solving:

$$\{u_i\}^{k+1} = \arg\min_{\{u_i\}} \left(\frac{1}{2}\|u_i - x_i\|_2^2 + \sum_{j \in \mathcal{N}(i)} (\lambda_{ij}^k - \lambda_{ji}^k)^\top u_i \right)$$

Setting the gradient to zero and solving yields:

$$(u_i - x_i) + \sum_{j \in \mathcal{N}(i)} (\lambda_{ij}^k - \lambda_{ji}^k) = 0$$

Thus, the closed-form update is:

$$u_i^{k+1} = x_i - \sum_{j \in \mathcal{N}(i)} (\lambda_{ij}^k - \lambda_{ji}^k)$$

Note: This update is simple; no linear system needs to be solved.

AMA for Convex Clustering: v-update

At each iteration, for each edge (i, j), we update v_{ij} by solving:

$$v_{ij}^{k+1} = \arg\min_{v_{ij}} \left(\gamma w_{ij} \|v_{ij}\|_2 - \lambda_{ij}^{k\top} v_{ij} \right)$$

The closed-form solution is:

$$v_{ij}^{k+1} = \left(1 - \frac{\gamma w_{ij}}{\|\lambda_{ij}^k\|_2} \right)_+ \lambda_{ij}^k$$

where $(\cdot)_+ = \max(\cdot, 0)$.

Summary: This is a group lasso-style shrinkage operator applied to λ_{ij}^k.

AMA for Convex Clustering: Dual Update

Dual update:

$$\lambda_{ij}^{k+1} = \lambda_{ij}^{k} + \nu(u_i^{k+1} - u_j^{k+1} - v_{ij}^{k+1}), \quad \forall (i,j) \in E$$

where $\nu > 0$ is the step size.

Convergence Guarantee ([Chi and Lange, 2015]Proposition 4.1)

If the step size satisfies:

$$\nu < \frac{2}{\rho(L)}$$

where $\rho(L)$ is the largest eigenvalue of the graph Laplacian L, then AMA converges to the unique minimizer of the convex clustering problem.

Tau Protein (Tubulin Associated Unit)

- Tau protein is a neuron-specific protein often used to study interactions between brain regions. There are p regions of interests(ROIs)
- Assume N patients is being studied, each patient k associated with a latent matrix $\Omega^{(k)} \in \mathbb{R}^{p \times p}$ representing ROI interactions.
- For each patient k, we observe:

$$\{y^{(k)}, \Gamma^{(k)}\}$$

  where:
  - $y^{(k)} \in \mathbb{R}^p$ indicates the tau contamination status.
  - $\Gamma^{(k)} \in \mathbb{R}^{p \times p}$ is a noisy observation of $\Omega^{(k)}$, additionally perturbed by $y^{(k)}$-dependent terms.
- Patients belonging to the same subtype are assumed to share the **same** Ω and have similar y values.

Tau Model – Optimization and Simulation Setup

Model:

$$\Gamma_{ij}^{(k)} = \Omega_{ij}^{(k)} + |y_i^{(k)} - y_j^{(k)}| + |y_i^{(k)} + y_j^{(k)}| + \epsilon_{ij}^{(k)}$$

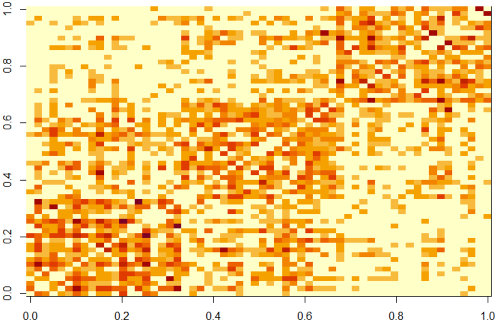
Network Lasso formulation:

$$\min_{\{\Omega^{(k)}\}} \frac{1}{2} \sum_k \sum_{i,j} \left(\Omega_{ij}^{(k)} + |y_i^{(k)} - y_j^{(k)}| + |y_i^{(k)} + y_j^{(k)}| - h(\Gamma_{ij}^{(k)}) \right)^2 + \lambda \sum_{k<k'} \rho(y^{(k)}, y^{(k')}) \|\Omega^{(k)} - \Omega^{(k')}\|_F$$

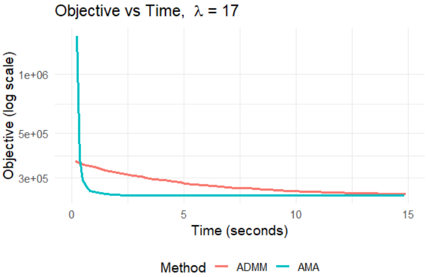
- Simulated Data is generated directly from the above model.
- Ground truth $\Omega^{(k)}$ is assigned based on subtype.
- $y^{(k)}$ is (randomly) generated using a SI model.
- ρ is the similarity of $y's$, used to determine LASSO penalty.

Tau Model – parameter/dimension setting

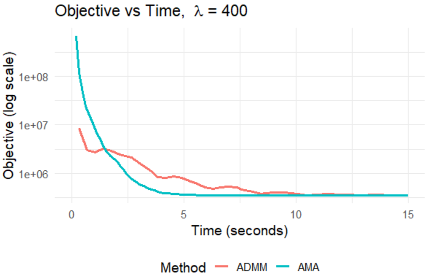
- We have 3 groups, 20 observations each, $N = 60$
- latent $\Omega \in \mathbb{R}^{p \times p}$ with $p = 68$
- ρ shown below is used for weight for LASSO penalty.



Comparison : Objective vs runtime

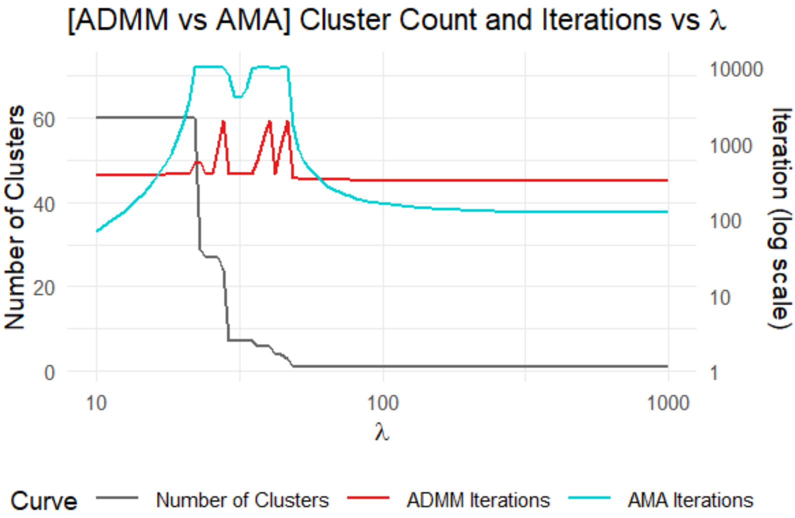


(a) Objective with $\lambda = 17$

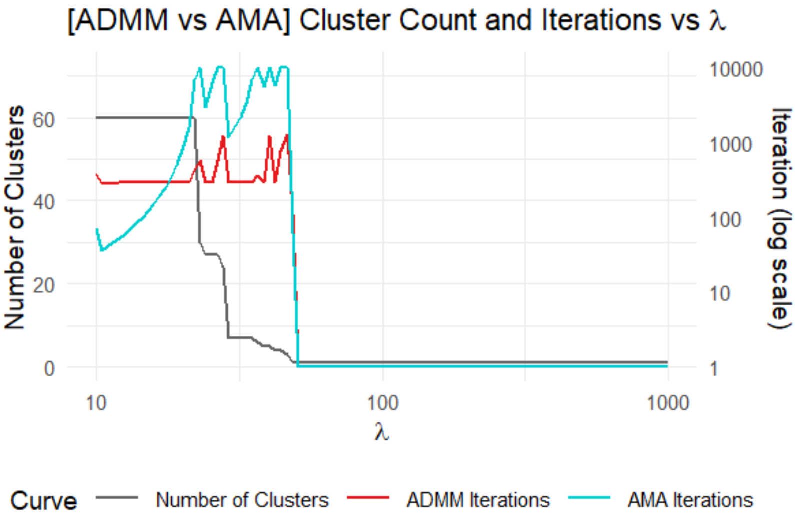


(b) Objective with $\lambda = 400$

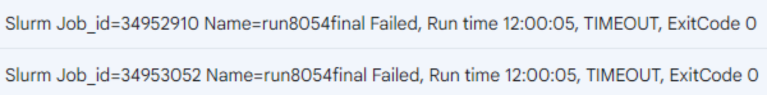
ADMM : clusters and iterations over λ



ADMM : clusters and iterations over λ(warm start)



Summary

1. For smaller λ, AMA converges faster than ADMM.
2. In the merging phase, AMA shows convergence issues, with significantly more iterations or failure to converge.
3. Essentially, Splitting method is doing coordinate decent and update on (Primal, Axillary, dual). When doing warm start, consider using all of them.
4. 

```
Slurm Job_id=34952910 Name=run8054final Failed, Run time 12:00:05, TIMEOUT, ExitCode 0
Slurm Job_id=34953052 Name=run8054final Failed, Run time 12:00:05, TIMEOUT, ExitCode 0
```

References

Chi, E. C. and Lange, K. (2015).
Splitting methods for convex clustering.
Journal of Computational and Graphical Statistics, 24(4):994–1020.

Hallac, D., Leskovec, J., and Boyd, S. (2015).
Network lasso: Clustering and optimization in large graphs.
In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 387–396. ACM.

Thank you!

Appendix 1 : ADMM for Network Lasso: u-update

At each iteration, we update u_i by solving:

$$u_i^{k+1} = \arg\min_{u_i} \left(\frac{1}{2}\|u_i - x_i\|_2^2 + \frac{\rho}{2} \sum_{j \in \mathcal{N}(i)} \|u_i - z_{ij}^k + \lambda_{ij}^k\|_2^2 \right)$$

Setting the gradient to zero and solving for u_i yields:

$$(1 + \rho|\mathcal{N}(i)|)\, u_i = x_i + \rho \sum_{j \in \mathcal{N}(i)} (z_{ij}^k - \lambda_{ij}^k)$$

Thus, the closed-form update is:

$$u_i^{k+1} = \frac{x_i + \rho \sum_{j \in \mathcal{N}(i)} (z_{ij}^k - \lambda_{ij}^k)}{1 + \rho|\mathcal{N}(i)|}$$

Appendix 2 : ADMM for Network Lasso: z-update

At each iteration, for each edge (i, j), define:

$$v_i = u_i^{k+1} + \lambda_{ij}^k, \quad v_j = u_j^{k+1} + \lambda_{ji}^k$$

Compute the shrinkage factor:

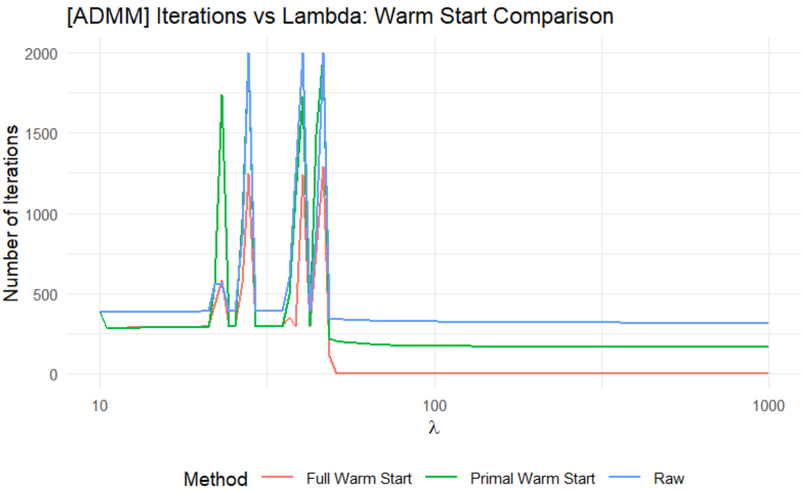
$$\theta = \max\left(1 - \frac{2\gamma w_{ij}}{\rho \|v_i - v_j\|_2}, 0.5\right)$$

Then update:

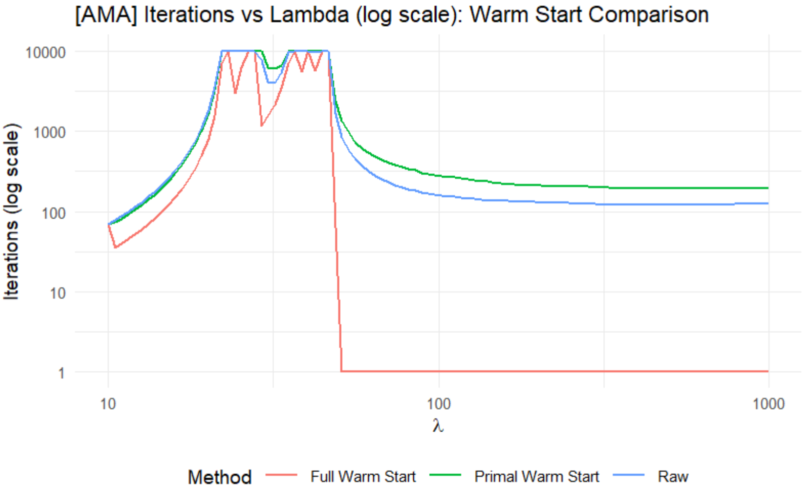
$$z_{ij}^{k+1} = (1-\theta)v_i + \theta v_j$$
$$z_{ji}^{k+1} = (1-\theta)v_j + \theta v_i$$

Summary: If v_i and v_j are close, merge together; otherwise, shrink towards each other with factor θ.

ADMM : warmstart



AMA : warmstart


[AMA] Iterations vs Lambda (log scale): Warm Start Comparison
Iterations (log scale)
λ
Method
Full Warm Start
Primal Warm Start
Raw