# Supplements for “Multi-way block localization via sparse tensor clustering”

**Yuchen Zeng**
University of Wisconsin – Madison
yzeng58@wisc.edu

**Miaoyan Wang**
University of Wisconsin – Madison
miaoyan.wang@wisc.edu

## A Proofs

### A.1 Proof of Proposition 1

Let $\mathcal{S} = \{\mathbb{P}_\Theta \colon \Theta \in \mathcal{P}\}$ be the family of (either Gaussian or Bernoulli) tensor block models (2), where $\Theta = \mathcal{C} \times_1 \boldsymbol{M}_1 \times_2 \cdots \times_K \boldsymbol{M}_K$ parameterizes the mean block tensor. Since the mapping $\Theta \mapsto \mathbb{P}_\Theta$ is one-to-one, $\Theta$ is identifiable. Now suppose there are two decompositions of $\Theta = \Theta(\{\boldsymbol{M}_k\}, \mathcal{C}) = \Theta(\{\tilde{\boldsymbol{M}}_k\}, \tilde{\mathcal{C}})$. Based on the Assumption 1, we have

$$\Theta = \mathcal{C} \times_1 \boldsymbol{M}_1 \times_2 \cdots \times_K \boldsymbol{M}_K = \tilde{\mathcal{C}} \times_1 \tilde{\boldsymbol{M}}_1 \times_2 \cdots \times_K \tilde{\boldsymbol{M}}_K, \tag{1}$$

where $\mathcal{C}, \tilde{\mathcal{C}} \in \mathbb{R}^{R_1 \times \cdots \times R_K}$ are two irreducible cores, and $\boldsymbol{M}_k, \tilde{\boldsymbol{M}}_k \in \{0,1\}^{R_k \times d_k}$ are membership matrices for all $k \in [K]$. We will prove by contradiction that $\boldsymbol{M}_k$ and $\tilde{\boldsymbol{M}}_k$ induce the same partition of $[d_k]$, for all $k \in [K]$.

Suppose the above claim does not hold. Then there exists a mode $k \in [K]$ such that the $\boldsymbol{M}_k, \tilde{\boldsymbol{M}}_k$ induce two different partitions of $[d_k]$. Without loss of generality, we assume $k = 1$. The definition of partition implies that there exists a pair of indices $i \neq j, i, j \in [d_1]$, such that, $i, j$ belong to the same cluster based on $\boldsymbol{M}_k$, but they belong to different clusters based on $\boldsymbol{M}_k$. Let $\mathcal{C} \subset [d_1]$ denote the cluster that $i$ (or $j$) belong to based on $\boldsymbol{M}_k$, and $\mathcal{A}, \mathcal{B} \subset [d_1]$ denote the two different clusters that $i$, $j$ belongs to based on $\tilde{\boldsymbol{M}}_k$. Based on the left-hand side of (1)

$$\Theta(i, i_2, \ldots, i_K) = \Theta(j, i_2, \ldots, i_K), \quad \text{for all } (i_2, \ldots, i_K) \in [d_2] \times \cdots \times [d_K]. \tag{2}$$

On the other hand, (1) implies

$$\Theta(i, i_2, \ldots, i_K) = \Theta(k, i_2, \ldots, i_K), \quad \text{for all } k \in \mathcal{A} \text{ and } (i_2, \ldots, i_K) \in [d_2] \times \cdots \times [d_K], \tag{3}$$

and

$$\Theta(j, i_2, \ldots, i_K) = \Theta(k, i_2, \ldots, i_K), \quad \text{for all } k \in \mathcal{B} \text{ and } (i_2, \ldots, i_K) \in [d_2] \times \cdots \times [d_K]. \tag{4}$$

Combining (2), (3) and (4), we have

$$\Theta(i, i_2, \ldots, i_K) = \Theta(k, i_2, \ldots, i_K), \quad \text{for all } k \in \mathcal{A} \cup \mathcal{B} \text{ and } (i_2, \ldots, i_K) \in [d_2] \times \cdots \times [d_K].$$

Therefore, one can merge $\mathcal{A}, \mathcal{B}$ into one cluster along the mode 1. This contradicts the irreducibility of the core tensor $\tilde{\mathcal{C}}$. Therefore, $\boldsymbol{M}_1$ and $\tilde{\boldsymbol{M}}_1$ induce a same partition of $[d_1]$, and thus they are equal up to permutations. The proof is now complete.

### A.2 Proof of Theorem 1

To study the performance of the least-square estimator $\hat{\Theta}$, we need to introduce some additional notations. The membership matrix $\boldsymbol{M}_k$ can be viewed as a onto function $\boldsymbol{M}_k \colon [d_k][R_k]$, and with a little abuse of notation, we still use $\boldsymbol{M}_k$ to denote the mapping function. Correspondingly, we use $\boldsymbol{M}_k(i) \in [R_k]$ to denote the cluster label for the element $i \in [d_k]$.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

### A.3 Sparse clustering

**Lemma 1** *Let $\mathbf{Y} \in \mathbb{R}^n$ be a response vector and $\mathbf{X} \in \mathbb{R}^{n \times p}$ the design matrix. Assume the response vector $\mathbf{Y}$ is mean-centered, i.e., $\sum_i Y_i = 0$. Suppose that $\mathbf{X}$ is an orthogonal design matrix with $X^T X = diag(n_1, ..., n_p)$. Define the ordinary least-square estimate $\hat{\boldsymbol{\beta}}_{ols} = (\hat{\beta}_{ols,1}, ..., \hat{\beta}^T_{ols,p})^T$. Consider the following constrained optimization:*

$$\hat{\boldsymbol{\beta}} = \arg\min\{\frac{1}{2}||\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}||_2^2 + \lambda pen(\boldsymbol{\beta})\}$$

*1. Case 1: L-0 penalization. $pen(\boldsymbol{\beta}) = ||\boldsymbol{\beta}||_0$:*
*Under the change of tuning parameter $\lambda' := f(\lambda) = \sqrt{2\lambda}$ such that $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, ..., \hat{\beta}_p)^T$ has a closed-form solution:*

$$\hat{\beta}_i = \hat{\beta}_{ols,i} \mathbb{I}_{|\hat{\beta}_{ols,i}| > \frac{\lambda'}{\sqrt{n_i}}} \; for \; all \; i = 1, ..., p$$

*2. Case 2: L-1 penalization. $pen(\boldsymbol{\beta}) = ||\boldsymbol{\beta}||_1$:*
*$\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, ..., \hat{\beta}_p)^T$ has a closed-form solution:*

$$\hat{\beta}_i = sign(\hat{\beta_{ols,i}})(|\hat{\beta_{ols,i}}| - \frac{\lambda}{n_i})_+ \; for \; all \; i = 1, 2, ..., p$$

**Proof 1** *The thing we want to minimize is*

$$L = \frac{1}{2}||\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}||_2^2 + \lambda||\boldsymbol{\beta}_0|| = \frac{1}{2}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \lambda||\boldsymbol{\beta}||_0 = L_1 + L_2$$

*where $L_1 = \frac{1}{2}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$, $L_2 = \lambda||\boldsymbol{\beta}||_0$.*

***Case 1:***

*Here we view the optimization problem as a case in linear regression. The $L_1$ is exactly the $RSS/2$ in this case. So we compare the increment of $L_1$ when $L_2$ takes different values. We denote $z$ as the number of non-zero elements in $\boldsymbol{\beta}$.*
*(1) Consider the case we have no constraint on $z$. Thus we only have to minimize $L_1$. By the knowledge of linear regression, we know the unique minimizer is $\hat{\boldsymbol{\beta}}_{\boldsymbol{ols}} = (\mathbf{X^T X})^{-\mathbf{1}}\mathbf{XY}$. Assume there are $m$ zero elements in $\hat{\boldsymbol{\beta}}_{\boldsymbol{ols}}$ where $0 \leq m \leq p$*
*(2) Consider the case we have constraint on $z$: $z = i$, where $i = 0, 1, 2, ..., m$. Obviously, among these cases the $L$ can be minimized if and only if $i = m$. So, $z = m$ and $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}_{\boldsymbol{ols}}$ is the minimizer of $L$ when $0 \leq z \leq m$. (3) Consider the case that we have constraint on $x$: $z = m + 1$. Then we have to take one more non-zero element in $\boldsymbol{\beta}$ to be zero. Suppose we take $\hat{\beta}_l \neq 0$ to be 0. Then we obtain*

$$2L_1 - SSE(\beta_1, ..., \beta_{l-1}, \beta_{l+1}, ..., \beta_p) = SSR(\beta_l)$$

*by the columns in $\mathbf{X}$ are orthogonal to each other. Additionally,*

$$SSR(\beta_l) = \mathbf{Y}^T(\mathbf{H} - \mathbf{H_l})\mathbf{Y}$$

*where $\mathbf{H} = \mathbf{X}(\mathbf{X^T X})^{-\mathbf{1}}\mathbf{X} = \sum_{i=1}^p \frac{1}{n_i}\mathbf{x_{(i)}}\mathbf{x^T_{(i)}}$, $\mathbf{H_l} = \sum_{i \neq j} \mathbf{x_{(i)}}\mathbf{x^T_{(i)}}$, $\hat{\beta}_l = \frac{1}{n_l}\mathbf{x_l Y}$. Thus, we can simplify the second equation as:*

$$SSR(\beta_l) = n_l \hat{\beta}_l^2$$

*Thus, by taking $\hat{\beta}_l$ as 0, there is $\frac{n_l \hat{\beta}_l^2}{2}$ increment on $L_1$, $\lambda$ decrement on $L_2$. Obviously, if the increment of $L_1$ is larger than the decrement $L_2$, we should not take $\hat{\beta}_l$ as 0; conversely, if the increment of $L_1$ is less than the decrement of $L_2$, taking $\hat{\beta}_l$ as 0 can lessen the $L$.*
*(4) As we discussed, if there is still at least one element in $\boldsymbol{\beta_k}$ that satisfies that $\frac{n_k \hat{\beta}_k^2}{2} \leq \lambda$, we can keep reducing $L$ by taking $\boldsymbol{\beta_k}$ as 0 until all remain non-zero elements in $\hat{\beta}$ do not satisfy $\frac{n_k \hat{\beta}_k^2}{2} \leq \lambda$. Then we can minimize $L$.*

*Over all, the $\boldsymbol{\beta}$ that minimized $L$ is:*

$$\hat{\beta}_i = \hat{\beta}_{ols,i} \mathbb{I}_{|\hat{\beta}_{ols,i}| > \frac{\lambda'}{\sqrt{n_i}}} \; for \; all \; i = 1, ..., p$$

***Case 2:***

*Here we use the properties of subderivative. Taking subderivative of $L$, we obtain*

$$\frac{\partial L}{\partial \beta_j} = \begin{cases} \{n_j\beta_j - \mathbf{x}_{(\mathbf{j})}^{\mathbf{T}}\mathbf{Y} + \lambda\} & \text{if } \beta_j > 0 \\ [n_j\beta_j - \mathbf{x}_{(\mathbf{j})}^{\mathbf{T}}\mathbf{Y} - \lambda, n_j\beta_j - \mathbf{x}_{(\mathbf{j})}^{\mathbf{T}}\mathbf{Y} + \lambda] & \text{if } \beta_j = 0 \\ \{n_j\beta_j - \mathbf{x}_{(\mathbf{j})}^{\mathbf{T}}\mathbf{Y} - \lambda\} & \text{if } \beta_j < 0 \end{cases}$$

*Because $\beta_j$ minimize $L$ if and only if $0 \in \frac{\partial L}{\partial \beta_j}$ and $\mathbf{X}$ is orthogonal, we get:*

$$\hat{\beta}_j = \begin{cases} \frac{\mathbf{x}_{(\mathbf{j})}^{\mathbf{T}}\mathbf{Y} + \lambda}{n_j} & \text{if } \hat{\beta}_j < 0 \\ 0 & \text{if } \hat{\beta}_j = 0 \\ \frac{\mathbf{x}_{(\mathbf{j})}^{\mathbf{T}}\mathbf{Y} - \lambda}{n_j} & \text{if } \hat{\beta}_j > 0 \end{cases}$$

*Here, $\hat{\boldsymbol{\beta}}_{ols} = (\mathbf{X}^{\mathbf{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathbf{T}}\mathbf{Y} = diag(1/n_1, ..., 1/n_p)X^TY$, so $\hat{\beta}_{ols,j} = \frac{\mathbf{x}_{(\mathbf{j})}^{\mathbf{T}}\mathbf{Y}}{n_j}$. Then the solution of $\hat{\beta}_j$ can be simplified as:*

$$\hat{\beta}_i = sign(\hat{\beta_{ols,i}})(|\hat{\beta_{ols,i}}| - \frac{\lambda}{n_i})_+ \; for \; all \; i = 1, 2, ..., p$$

| $n_1$ | $n_2$ | $n_3$ | $d_1$ | $d_2$ | $d_3$ | noise | CER (mode 1) | CER (mode 2) | CER (mode 3) |
|---|---|---|---|---|---|---|---|---|---|
| 40 | 40 | 40 | 3 | 5 | 4 | 4 | **0(0)** | **0(0)** | **0(0)** |
| 40 | 40 | 40 | 3 | 5 | 4 | 8 | **0(0)** | 0.0095(0.0247) | 0.0021(0.0145) |
| 40 | 40 | 40 | 3 | 5 | 4 | 12 | 0.0038(0.0138) | 0.0331(0.0453) | 0.0222(0.0520) |
| 40 | 40 | 80 | 3 | 5 | 4 | 4 | **0(0)** | 0.0017(0.0121) | **0(0)** |
| 40 | 40 | 80 | 3 | 5 | 4 | 8 | **0(0)** | **0(0)** | **0(0)** |
| 40 | 40 | 80 | 3 | 5 | 4 | 12 | **0(0)** | 0.0257(0.0380) | 0.0026(0.0064) |
| 40 | 40 | 40 | 4 | 4 | 4 | 4 | **0(0)** | **0(0)** | **0(0)** |
| 40 | 40 | 40 | 4 | 4 | 4 | 8 | 0.0023(0.0165) | 0.0034(0.0239) | **0(0)** |
| 40 | 40 | 40 | 4 | 4 | 4 | 12 | 0.0519(0.0744) | 0.0414(0.0697) | 0.0297(0.0644) |
| 40 | 40 | 80 | 4 | 4 | 4 | 4 | **0(0)** | **0(0)** | **0(0)** |
| 40 | 40 | 80 | 4 | 4 | 4 | 8 | **0(0)** | **0(0)** | **0(0)** |
| 40 | 40 | 80 | 4 | 4 | 4 | 12 | 0.0132(0.0405) | 0.0106(0.0366) | 0.0043(0.0168) |

Table 1: Given the true $d_1, d_2, d_3$, the simulation results is calculated across 50 tensors each time.

| $n_1$ | $n_2$ | $n_3$ | $d_1$ | $d_2$ | $d_3$ | noise | overall accuracy | estimated $d_1$ | estimated $d_2$ | estimated $d_3$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 40 | 40 | 40 | 3 | 5 | 4 | 4 | **1** | 3(0) | 5(0) | 4(0) |
| 40 | 40 | 40 | 3 | 5 | 4 | 8 | 0.74 | 3(0) | 4.76(0.0610) | 3.98(0.02) |
| 40 | 40 | 40 | 3 | 5 | 4 | 12 | 0.02 | 2.8(0.0571) | 3.58(0.1072) | 3.3(0.0915) |
| 40 | 40 | 40 | 4 | 4 | 4 | 4 | **1** | 4(0) | 4(0) | 4(0) |
| 40 | 40 | 40 | 4 | 4 | 4 | 8 | 0.88 | 3.94(0.0339) | 3.96(0.0280) | 3.96(0.0280) |
| 40 | 40 | 40 | 4 | 4 | 4 | 12 | 0.04 | 3.08(0.0983) | 3.12(0.1016) | 3.12(0.0975) |
| 40 | 40 | 80 | 4 | 4 | 4 | 4 | **1** | 4(0) | 4(0) | 4(0) |
| 40 | 40 | 80 | 4 | 4 | 4 | 8 | 1 | 4(0) | 4(0) | 4(0) |
| 40 | 40 | 80 | 4 | 4 | 4 | 12 | 0.78 | 3.9(0.0429) | 3.92(0.0388) | 3.96(0.04) |

Table 2: The simulation results across 50 tensors each time from estimating the $d_1, d_2, d_3$.

| sparsity rate | noise | method | estimated sparsity Rate | Correct Zero Rate | Correct One Rate | Total Correct Rate |
|---|---|---|---|---|---|---|
| 0.5 | 4 | $\lambda = 0$ | 0(0) | 0(0) | 1(0) | 0.5075(0.0676) |
| 0.5 | 4 | $\lambda = 100$ | 0.5677(0.0667) | 1(0) | 0.8519(0.0678) | 0.9248(0.0377) |
| 0.5 | 4 | $\lambda = 200$ | 0.5952(0.0688) | 1(0) | 0.7975(0.0787) | 0.8973(0.0433) |
| 0.5 | 4 | $\bar{\lambda} = 86.61396$ | 0.5606(0.0668) | $\mathbf{0.9993}(\mathbf{0.0035})$ | $\mathbf{0.8655}(\mathbf{0.0685})$ | $\mathbf{0.9312}(\mathbf{0.0377})$ |
| 0.5 | 8 | $\lambda = 0$ | 0(0) | 0(0) | 1(0) | 0.5075(0.0676) |
| 0.5 | 8 | $\lambda = 100$ | 0.5072(0.068) | 0.879(0.0898) | 0.8554(0.0634) | 0.8665(0.0559) |
| 0.5 | 8 | $\lambda = 200$ | 0.5884(0.0618) | 0.9753(0.034) | 0.7877(0.0776) | 0.8794(0.0492) |
| 0.5 | 8 | $\bar{\lambda} = 344.3656$ | 0.6298(0.0652) | 0.9956(0.0128) | 0.7259(0.0873) | 0.8586(0.0518) |
| 0.8 | 8 | $\lambda = 0$ | 0(0) | 0(0) | 1(0) | 0.2029(0.0541) |
| 0.8 | 8 | $\lambda = 100$ | 0.6458(0.0646) | 0.7453(0.0616) | 0.7136(0.2017) | 0.7435(0.0668) |
| 0.8 | 8 | $\lambda = 200$ | 0.7947(0.0627) | 0.9119(0.0601) | 0.6259(0.2376) | 0.8589(0.0698) |
| 0.8 | 8 | $\bar{\lambda} = 246.9212$ | 0.826(0.0622) | 0.9462(0.0412) | 0.6077(0.2495) | 0.8841(0.0602) |

Table 3: Results for Simulation 6 over 50 simulated data sets ($n_1 = 40, n_2 = 40, n_3 = 40, d_1 = 3, d_2 = 5, d_3 = 4$.