Simultaneous Estimation and Dataset Selection for Transfer Learning in High Dimensions by the Truncated Norm Penalty

Zeyu Li, Dong Liu, Yong He and Xinsheng Zhang

Southeast University, Nanyang Technological University,

Shandong University and Fudan University

Supplementary Material

In the supplementary material, we first provide extensional theoretical arguments concerning the two specific statistical models and remark on the optional fine-tuning step. Then, we present additional numerical details that further support our arguments. Finally, we provide the proofs of the theoretical results.

S.1 Extensional Theoretical Arguments

In this section, we present some extensional theoretical arguments. The preceding deterministic arguments are discussed under two specific statistical models, namely sparse linear regression and generalized low-rank trace regression, respectively. Thanks to Theorem 1, which guaranties an oracle local minimum of (2.5), it suffices to discuss the oracle pooling estimator $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$ thoroughly in these two specific cases.

S.1.1 Oracle pooling estimator under specific models

We first discuss the oracle pooling estimator under two specific statistical models, namely sparse linear regression and generalized low-rank trace regression, from a theoretical perspective, while providing some intuition on enlarging the regularization strength.

Sparse linear regression

We assume that the k-th dataset consists of n_k *i.i.d.* samples from the linear model $y_{k,i} = \langle \boldsymbol{\theta}_k^*, \boldsymbol{X}_{k,i} \rangle + \varepsilon_{k,i}$ and $k = 0, \ldots, K$. Let $\boldsymbol{\theta}_0^*$ be s-sparse with $s < \infty$. To cope with the sparsity structure of $\boldsymbol{\theta}_0^*$, we take the decomposable regularizer $\mathcal{R} = \|\cdot\|_1$. We follow Raskutti et al. (2010) and (a) assume that the $n_k \times p$ design matrices $\mathcal{X}_k = (\boldsymbol{X}_{k,1}, \ldots, \boldsymbol{X}_{k,n_k})^\top$ are formed by independently sampling n_k identical $\boldsymbol{X}_{k,i} \sim N(0, \boldsymbol{\Sigma}_k)$ with the covariance matrices $\boldsymbol{\Sigma}_k$ satisfying $M_1^{-1} \leq \lambda_{\min}(\boldsymbol{\Sigma}_k) \leq \lambda_{\max}(\boldsymbol{\Sigma}_k) \leq M_1$, which is often referred to as the $\boldsymbol{\Sigma}_k$-Gaussian ensembles; (b) assume that $\varepsilon_{k,i}$ are independently drawn from the same centered sub-Gaussian distribution; and (c) for $k = 1, \ldots, K$, we assume either $\|\boldsymbol{\delta}_k^*\|_1 \leq h$ or $\|\boldsymbol{\delta}_k^*\|_2 \leq h$ (if $\|\boldsymbol{\delta}_k^*\|_2 \leq h$, we further assume that $p/n_k \to c_k$ holds for some positive constant c_k) for $\boldsymbol{\delta}_k^* = \boldsymbol{\theta}_0^* - \boldsymbol{\theta}_k^*$.

Corollary 1 (Sparse linear regression)**.** *Given the above settings, we can solve the problem (2.4) with* $\lambda_{\mathcal{P}} \asymp \big[(\log p/n_{\mathcal{P}})^{1/2} + h\big]$. *As* $\min_{k \in \mathcal{P}} n_k \to \infty$, $p \to \infty$ *with*

$\max_{k\in\mathcal{P}} \log p/n_k \to 0$ *and* $h \to 0$*, the oracle pooling estimator satisfies*

$$\left\|\widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \boldsymbol{\theta}_0^*\right\|^2 = O_p\left[\frac{s\log p}{n_{\mathcal{P}}} + sh^2\right], \quad \left\|\widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \boldsymbol{\theta}_0^*\right\|_1 = O_p\left[s\left(\frac{\log p}{n_{\mathcal{P}}}\right)^{1/2} + sh\right].$$

In Li et al. (2022), the authors introduce the population oracle pooling parameter as $\boldsymbol{\theta}_{\mathcal{P}}^* = \operatorname{argmin} \sum_{k\in\mathcal{P}} n_k \mathbb{E}[\mathcal{L}_k(\boldsymbol{Z}_k;\boldsymbol{\theta})]/n_{\mathcal{P}}$, where $\mathcal{L}_k$ is the least squared error loss. In the first step, they acquire the estimator of $\boldsymbol{\theta}_{\mathcal{P}}^*$ as $\widetilde{\boldsymbol{\theta}}_{\mathcal{P}}$. In the second step, they plug-in $\widetilde{\boldsymbol{\theta}}_{\mathcal{P}}$ and estimate $\boldsymbol{\theta}_0^* - \boldsymbol{\theta}_{\mathcal{P}}^*$ by solving a fine-tuning optimization problem. When there is no model shift, namely $h = 0$, we have $\boldsymbol{\theta}_{\mathcal{P}}^* = \boldsymbol{\theta}_0^*$, and we shall solve (2.4) with the tuning parameter $\lambda_{\mathcal{P}} \asymp (\log p/n_{\mathcal{P}})^{1/2}$. We also call it the default rate of regularization, corresponding to the pooled variance term. In fact, in the pooling step of the classical two-step procedures (Bastani, 2021; Li et al., 2022; Tian and Feng, 2023), when $h \neq 0$, the typical rate for regularization is also $\lambda_{\mathcal{P}} \asymp (\log p/n_{\mathcal{P}})^{1/2}$. However, the second fine-tuning step of Li et al. (2022) relies heavily on the sparsity of $\boldsymbol{\theta}_{\mathcal{P}}^* - \boldsymbol{\theta}_0^*$, which, in turn, requires sparse contrast vectors and homogeneous Hessian matrices of the population loss functions. In the case of linear regression, the Hessian matrices are the population covariance matrices of the covariates, namely $\boldsymbol{\Sigma}_k := \mathbb{E}[\boldsymbol{X}_{k,i}\boldsymbol{X}_{k,i}^\top]$. Hence, these two-step methods are quite sensitive to covariate shift, as remarked in He et al. (2024). In fact, we now provide a toy example where the procedures might fail under nearly homogeneous covariates.

Example 1 (Almost homogeneous covariates)**.** For independent $\boldsymbol{X}_{k,i}$ such that $\mathbb{E}\boldsymbol{X}_{k,i} = 0$, $\boldsymbol{\Sigma}_k = \mathbb{E}\boldsymbol{X}_{k,i}\boldsymbol{X}_{k,i}^\top$, and independent noise $\varepsilon_{k,i} \sim N(0, \boldsymbol{I}_p)$, let $y_{k,i} = \langle \boldsymbol{\theta}_k^*, \boldsymbol{X}_{k,i}\rangle + \varepsilon_{k,i}$,

$k = 0, 1, 2$. In addition, assume that $n_0 = n_1 = n_2$ and $\boldsymbol{\theta}_0^*$ is s-sparse. It is shown in Li et al. (2022) that for $\boldsymbol{\delta}_k^* = \boldsymbol{\theta}_0^* - \boldsymbol{\theta}_k^*$, we have

$$\boldsymbol{\theta}_0^* - \boldsymbol{\theta}_{\mathcal{P}}^* = \left(\sum_{k=0}^{2} \boldsymbol{\Sigma}_k \right)^{-1} (\boldsymbol{\Sigma}_1 \boldsymbol{\delta}_1 + \boldsymbol{\Sigma}_2 \boldsymbol{\delta}_2) \,.$$

As for homogeneous covariates, such that the covariance matrices are set as $\boldsymbol{\Sigma}_0 = \boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$, we have $\boldsymbol{\theta}_0^* - \boldsymbol{\theta}_{\mathcal{P}}^* \asymp (\boldsymbol{\delta}_1 + \boldsymbol{\delta}_2)$, which should be well-bounded by $\|\cdot\|_1$ according to the triangular inequality, once we assume $\|\boldsymbol{\delta}_k^*\|_1 \leq h$. However, consider the nearly homogeneous covariates such that

$$\boldsymbol{\Sigma}_0 = \boldsymbol{I}_p, \quad \boldsymbol{\Sigma}_1 = \boldsymbol{I}_p + c\boldsymbol{Z}_p, \quad \boldsymbol{\Sigma}_2 = \boldsymbol{I}_p - c\boldsymbol{Z}_p,$$

where $\boldsymbol{Z}_p$ is a fixed realization from the standard Gaussian orthogonal ensemble (GOE), which is the symmetric random matrix with the diagonal elements taken independently from $N(0, 2p^{-1})$ and the off-diagonal elements taken independently from $N(0, p^{-1})$. As $p \to \infty$, the spectrum of $\boldsymbol{Z}_p$ is bounded within $[-2, 2]$ with high probability, so $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$ are positive definite with high probability for sufficiently small $c > 0$. Notably, $\|\mathrm{vec}(\boldsymbol{\Sigma}_1) - \mathrm{vec}(\boldsymbol{\Sigma}_2)\|_\infty \asymp \|\mathrm{vec}(\boldsymbol{Z}_p)\|_\infty \lesssim (\log p/p)^{1/2} \to 0$ as $p \to \infty$. In this case, we have $\boldsymbol{\theta}_0^* - \boldsymbol{\theta}_{\mathcal{P}}^* \asymp (\boldsymbol{\delta}_1 + \boldsymbol{\delta}_2) + c\boldsymbol{Z}_p(\boldsymbol{\delta}_1 - \boldsymbol{\delta}_2)$. Let $\boldsymbol{\delta}_1 = -\boldsymbol{\delta}_2 = (h, 0, \cdots, 0)^\top$, $\boldsymbol{Z}_p = (\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_p)$; we have

$$\|\boldsymbol{\theta}_0^* - \boldsymbol{\theta}_{\mathcal{P}}^*\|_1 \asymp ch\|\boldsymbol{z}_1\|_1 \gtrsim ch\sqrt{p} \to \infty,$$

by observing that $\|\boldsymbol{z}_1\|_1$ approximately equals the sum of p independent absolute values of $N(0, p^{-1})$.

That is to say, the population pooling parameter $\boldsymbol{\theta}^*_{\mathcal{P}}$ could be significantly different from $\boldsymbol{\theta}^*_0$ even if all $\boldsymbol{\theta}^*_k$ are close in terms of ℓ_1 norm, under heterogeneous covariates. On the other hand, if we instead impose the slightly stronger regularization $\lambda_{\mathcal{P}} \asymp (\log p/n_{\mathcal{P}})^{1/2} + h$, which could be viewed as the combination of bias and variance, we shall have $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$ close to $\boldsymbol{\theta}^*_0$, as ensured by Corollary 1 above.

Here, we slightly enlarge the regularization strength from the default rate to $\lambda_{\mathcal{P}} \asymp (\log p/n_{\mathcal{P}})^{1/2} + h$ to address any potential covariate shift. We would like to provide some intuitions on such enlarging regularization. It is well-known that $\mathcal{R} = \|\cdot\|_1$ is able to element-wise shrink the estimator towards 0. Recall that $\boldsymbol{\theta}^*_0$ is supported on some subset $S \subset \{0, 1, \ldots, p\}$ with cardinality s. However, given the potential model shift of sources, it is likely that we will have non-zero estimates in S^c, the magnitude of which is bounded above by h. By enlarging the regularization by h, we are likely to penalize those estimates in S^c to zero, as desired. The price we pay is that additional regularization is simultaneously enforced on each element in S, which also shrinks more towards 0 by h, resulting in the additional sh^2 term in Corollary 1. The very same intuition applies to the generalized low-rank trace regression below; except there, we use the matrix nuclear norm $\mathcal{R} = \|\cdot\|_N$ to shrink all the singular values.

In Li et al. (2022, 2024), for their oracle transfer learning estimators to be preferable, they need $h \lesssim (\log p/n_0)^{1/2}$, which results in the rate of $s \log p/n_{\mathcal{P}} + h^2$. Hence,

if s is finite, then the direct oracle pooling estimator is no worse than the minimax optimal oracle transfer learning estimators. This is also in the spirit of Chen et al. (2021), where the authors suggest that either the target estimator or the oracle pooling estimator is minimax optimal in low-dimensional knowledge transfer problems under certain distance similarities. If s is able to diverge as well, the uniform over-shrinkage discussed above might be sub-optimal, and we suggest considering more sophisticated methods to overcome the covariate shift (Li et al., 2024; He et al., 2024). Meanwhile, in many practical cases when s is not too large, we can consider using the more user-friendly $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$ after simply enlarging the regularization from the default rate.

Generalized low-rank trace regression

Then, we present the case for generalized low-rank trace regression; the results are analogous to those of the sparse regression case. Let the k-th dataset consist of n_k *i.i.d.* samples of $\boldsymbol{Z}_{k,i} = (\boldsymbol{X}_{k,i}, y_{k,i})$, where $\mathbb{P}(y_{k,i}|\boldsymbol{X}_{k,i}) \propto \exp\{y_{k,i}\eta_{k,i} - b_k(\eta_{k,i})\}$ for $\eta_{k,i} = \langle \boldsymbol{\theta}_k^*, \boldsymbol{X}_{k,i} \rangle$. Assume $\boldsymbol{\theta}_0^*$ is of rank r with $r < \infty$. Recall the subspaces defined in equations (1.1) and (1.2), and we take $\mathcal{R} = \|\cdot\|_N$ as the decomposable regularizer. We follow Fan et al. (2019) and assume $\boldsymbol{\theta}_k^*$ to be $d \times d$ square matrices. All analyzes could be readily extended to rectangular cases of dimensions $d_1 \times d_2$, with the rate replaced by $d = \max(d_1, d_2)$. In addition, assume all datasets share

the same $b_k(\cdot) = b(\cdot)$, where $b'(\eta_{k,i}) = \mathbb{E}(y_{k,i}|\boldsymbol{X}_{k,i})$ is called the inverse link function and $b''(\eta_{k,i}) = \mathrm{Var}(y_{k,i}|\boldsymbol{X}_{k,i})$. As also pointed out in Tian and Feng (2023), using different b_k is allowed; however, it is less practical.

We take $\boldsymbol{L}_k(\boldsymbol{\theta}) = \sum_{i\le n_k}\left[-y_{k,i}\langle\boldsymbol{\theta}, \boldsymbol{X}_{k,i}\rangle + b(\langle\boldsymbol{\theta}, \boldsymbol{X}_{k,i}\rangle)\right]/n_k$, whose gradient and Hessian matrices at $\boldsymbol{\theta}$ are, respectively, $\nabla\boldsymbol{L}_k(\boldsymbol{\theta}) = \sum_{i\le n_k}[b'(\langle\boldsymbol{\theta}, \boldsymbol{X}_{k,i}\rangle) - y_{k,i}]\boldsymbol{X}_{k,i}/n_k$ and $\widehat{\boldsymbol{H}}_k(\boldsymbol{\theta}) := \sum_{i\le n_k} b''(\langle\boldsymbol{\theta}, \boldsymbol{X}_{k,i}\rangle)\mathrm{vec}(\boldsymbol{X}_{k,i})\mathrm{vec}(\boldsymbol{X}_{k,i})^\top/n_k$. We make the following assumptions under the guidance of Fan et al. (2019): (a) for each $k \in \mathcal{P}$, the vectorized version of $\boldsymbol{X}_{k,i}$ is taken independently from a sub-Gaussian random vector with bounded Ψ_2-norm, namely $\|\mathrm{vec}(\boldsymbol{X}_{k,i})\|_{\Psi_2} \le M_1$; (b) we assume $|b''(\eta_{k,i})| \le M_2$ and $|b'''(\eta_{k,i})| \le M_3$ almost surely; (c) let $\boldsymbol{H}_k(\boldsymbol{\theta}_k^*) = \mathbb{E}\widehat{\boldsymbol{H}}_k(\boldsymbol{\theta}_k^*)$; assume that $\lambda_{\min}[\boldsymbol{H}_k(\boldsymbol{\theta}_k^*)] \ge \kappa_k$; (d) we assume either $\|\boldsymbol{\delta}_k^*\|_F \le h$, $\|\boldsymbol{\delta}_k^*\|_N \le h$, or $\|\mathrm{vec}(\boldsymbol{\delta}_k^*)\|_1 \le h$ for $\boldsymbol{\delta}_k^* = \boldsymbol{\theta}_0^* - \boldsymbol{\theta}_k^*$; and (e) assume that $\|\boldsymbol{\theta}_0^*\|_F \ge \alpha\sqrt{d}$ for some constant α. We claim the following rates of convergence.

Corollary 2 (Generalized low-rank trace regression)**.** *Given the above settings, if we solve the problem (2.4) with* $\lambda_{\mathcal{P}} \asymp (d/n_{\mathcal{P}})^{1/2} + h$, *as* $\min_{k\in\mathcal{P}} n_k \to \infty$, $d \to \infty$, *and* $h \to 0$ *with* $d \lesssim \min_{k\in\mathcal{P}} n_k$ *and* $(d/n_{\mathcal{P}})^{1/2} + h \to 0$, *the oracle pooling estimator satisfies*

$$\left\|\widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \boldsymbol{\theta}_0^*\right\|_F^2 = O_p\left[\frac{rd}{n_{\mathcal{P}}} + rh^2\right], \quad \left\|\widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \boldsymbol{\theta}_0^*\right\|_N = O_p\left[r\left(\frac{d}{n_{\mathcal{P}}}\right)^{1/2} + rh\right].$$

S.1.2 Optional fine-tuning

Recall that given the oracle pooling estimator $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$ (or $\widehat{\boldsymbol{\theta}}_0$ from the non-oracle method), we can choose to fine-tune the primal estimator using the target dataset by solving the following problem:

$$\widehat{\boldsymbol{\delta}} = \underset{\boldsymbol{\delta}\in\mathbb{R}^p}{\operatorname{argmin}} \frac{1}{n_0} \sum_{i\leq n_0} \mathcal{L}_0(\boldsymbol{Z}_{0,i}; \widehat{\boldsymbol{\theta}}_{\mathcal{P}} + \boldsymbol{\delta}) + \lambda_d \mathcal{R}(\boldsymbol{\delta}). \tag{S.1.1}$$

To some extent, the resulting fine-tuned estimator, denoted as $\widehat{\boldsymbol{\theta}}^{\star}_{\mathcal{P}} := \widehat{\boldsymbol{\theta}}_{\mathcal{P}} + \widehat{\boldsymbol{\delta}}$, could be viewed as an interpolation between the oracle pooling estimator and the target estimator, which places a greater emphasis on the personalized aspect of the target dataset.

When the sources are sufficiently informative, the fine-tuning step is, in fact, not statistically necessary; hence, the term is optional. For instance, in the extreme case that $h = 0$, any interpolation towards the target estimator only introduces additional variance to the oracle pooling estimator, leading to statistically sub-optimal performance. This is particularly likely to happen when the target sample size n_0 is also relatively small, which is common in practical transfer learning applications. Meanwhile, in certain cases when $\boldsymbol{\theta}_0^*$ has a low-dimensional model shift from the sources, as will be shown by numerical simulation and real data analysis in the appendix, the fine-tuning step is able to alleviate such influence in the same way as in Li et al. (2022); Tian and Feng (2023) and serves as the final assurance for satisfying

knowledge transfer performance.

Using interpolation between the knowledge transfer estimator and the target estimator for robustness against negative transfer is, in fact, common practice. Please refer to Duan and Wang (2023), where the authors use a similar procedure to (S.1.1) to avoid the negative impact caused by misusing the data pooling strategy. In the same spirit, Li et al. (2024) proposes splitting the samples to learn the best linear combination of the knowledge transfer estimator and the individual estimator using target data only, which has been proven to be no worse than the single-task estimator with high probability. Cross validation is suggested when choosing the tuning parameter for the fine-tuning step in practice.

S.2 Additional Numerical Details

Here, we provide additional numerical details that further support our arguments.

S.2.1 Implement details in Section 4

We first provide the implementation details of the competitors in Section 4 of the main article. Specifically, for $\widehat{\boldsymbol{\theta}}_{\mathrm{TF}}$, we apply the proposed truncated norm optimization on the datasets $\mathcal{P}$ with sufficiently large τ to obtain $(\widehat{\boldsymbol{\theta}}_0, \cdots, \widehat{\boldsymbol{\theta}}_K)$. Then $\widehat{\boldsymbol{\theta}}_{\mathrm{TF}}$ is defined as the weighted average of $\widehat{\boldsymbol{\theta}}_k$, i.e., $\widehat{\boldsymbol{\theta}}_{\mathrm{TF}} = \sum_{k\in\mathcal{P}} n_k \widehat{\boldsymbol{\theta}}_k / n_{\mathcal{P}}$. For $\widehat{\boldsymbol{\theta}}_{\mathrm{agg}}$, we first randomly divide the target data into two groups, $\mathcal{T}_1$ and $\mathcal{T}_2$, of equal size. We then obtain

the $\boldsymbol{\theta}_0$ estimator $\widehat{\boldsymbol{\theta}}^1_{\text{agg}}$ by truncated norm optimization on the datasets $\mathcal{T}_1 \cup [K]$ with sufficiently large τ. We can also obtain $\widehat{\boldsymbol{\theta}}^2_{\text{agg}}$ using the lasso estimator on the dataset $\mathcal{T}_1$. Then, for $\bar{\boldsymbol{\Theta}} = (\widehat{\boldsymbol{\theta}}^1_{\text{agg}}, \widehat{\boldsymbol{\theta}}^2_{\text{agg}})$, the aggregation estimator is defined as $\widehat{\boldsymbol{\theta}}_{\text{agg}} = \bar{\boldsymbol{\Theta}}\boldsymbol{\eta}$ where

$$\boldsymbol{\eta} = \underset{\boldsymbol{\eta} \in \text{positive simplex}}{\text{argmin}} \quad \mathcal{L}_0(\mathbf{Z}_{\mathcal{T}_2}, \bar{\boldsymbol{\Theta}}\boldsymbol{\eta}), \quad \mathbf{Z}_{\mathcal{T}_2} \text{ are samples from } \mathcal{T}_2.$$

For $\widehat{\boldsymbol{\theta}}_{\text{CV}}$, we first estimate $\widehat{\mathcal{A}}$, which is used to obtain the oracle pooling estimator $\widehat{\boldsymbol{\theta}}_{\text{CV}}$. We randomly divide the target data into three groups of equal size, denoted as $\mathcal{T}_r$ and $r \leq 3$. Next, for each r, we obtain the lasso estimator $\widehat{\boldsymbol{\theta}}^r_{\text{Target}}$ on the $\mathcal{T}_{-r}$, where $\mathcal{T}_{-r}$ is the target data without $\mathcal{T}_r$. Then we obtain $\widehat{\boldsymbol{\theta}}^r_k$ by truncated norm optimization on the datasets $k \cup \mathcal{T}_{-r}$ with sufficiently large τ. Accordingly, we can calculate the loss function $\mathcal{L}_0(\mathbf{Z}_{\mathcal{T}_r}, \widehat{\boldsymbol{\theta}}^r_k)$ for each k and r. Finally, we calculate $\mathcal{L}^k_0 = \sum_{r=1}^3 \mathcal{L}_0(\mathbf{Z}_{\mathcal{T}_r}, \widehat{\boldsymbol{\theta}}^r_k)/3$, $\mathcal{L}^0_0 = \sum_{r=1}^3 \mathcal{L}_0(\mathbf{Z}_{\mathcal{T}_r}, \widehat{\boldsymbol{\theta}}^r_0)/3$, and $\widehat{\sigma} = \sqrt{\sum_{r=1}^3 (\mathcal{L}_0(\mathbf{Z}_{\mathcal{T}_r}, \widehat{\boldsymbol{\theta}}^r_0) - \mathcal{L}^0_0)^2/2}$. Then we have $\widehat{\mathcal{A}} = \left\{k \neq 0, \mathcal{L}^k_0 - \mathcal{L}^0_0 \leq C_0(\widehat{\sigma} \vee 0.01)\right\}$. We set $C_0 = 1$ in the experiments.

DC-ADMM details

To solve the problem (2.6), the scaled augmented Lagrangian function is set as

$$\mathcal{L}_\rho(\Theta, \boldsymbol{\delta}) = S^{(m+1)}(\boldsymbol{\Theta}, \boldsymbol{\delta}) + \frac{\rho}{2} \sum_{k=1}^K ||\boldsymbol{\delta}_k + \boldsymbol{\theta}_k - \boldsymbol{\theta}_0 + \boldsymbol{\nu}_k||_2^2 - \frac{\rho}{2} \sum_{k=1}^K ||\boldsymbol{\nu}_k||_2^2.$$

Then, the standard ADMM procedure can be implemented as

$$\widehat{\boldsymbol{\theta}}_k^{l+1} = \underset{\boldsymbol{\theta}_k \in \mathbb{R}^p}{\operatorname{argmin}} \ \alpha_k \|\boldsymbol{y}_k - \mathcal{X}_k \boldsymbol{\theta}_k\|_2^2 + n_k \lambda_{\mathcal{P}} \|\boldsymbol{\theta}_k\|_1 + \frac{\rho}{2} \|\widehat{\boldsymbol{\delta}}_k^l + \boldsymbol{\theta}_k - \widehat{\boldsymbol{\theta}}_0^l + \widehat{\boldsymbol{\nu}}_k^l\|_2^2,$$

$$\widehat{\boldsymbol{\theta}}_0^{l+1} = \underset{\boldsymbol{\theta}_0 \in \mathbb{R}^p}{\operatorname{argmin}} \ \alpha_0 \|\boldsymbol{y}_0 - \mathcal{X}_0 \boldsymbol{\theta}_0\|_2^2 + n_0 \lambda_{\mathcal{P}} \|\boldsymbol{\theta}_0\|_1 + \frac{\rho}{2} \sum_{k=1}^{K} \|\widehat{\boldsymbol{\delta}}_k^l + \widehat{\boldsymbol{\theta}}_k^{l+1} - \boldsymbol{\theta}_0 + \widehat{\boldsymbol{\nu}}_k^l\|_2^2,$$

$$\widehat{\boldsymbol{\delta}}_k^{l+1} = \begin{cases} -\widehat{\boldsymbol{\theta}}_k^{l+1} + \widehat{\boldsymbol{\theta}}_0^{l+1} - \widehat{\boldsymbol{\nu}}_k^l, & \text{if } \|\widehat{\boldsymbol{\delta}}_k^{(m)}\|_1 \geq \tau, \\ \operatorname{prox}_{n_k \lambda_{\mathcal{Q}_k}/\rho}(-\widehat{\boldsymbol{\theta}}_k^{l+1} + \widehat{\boldsymbol{\theta}}_0^{l+1} - \widehat{\boldsymbol{\nu}}_k^l), & \text{if } \|\widehat{\boldsymbol{\delta}}_k^{(m)}\|_1 < \tau, \end{cases}$$

$$\widehat{\boldsymbol{\nu}}_k^{l+1} = \widehat{\boldsymbol{\nu}}_k^l + \widehat{\boldsymbol{\delta}}_k^{l+1} + \widehat{\boldsymbol{\theta}}_k^{l+1} - \widehat{\boldsymbol{\theta}}_0^{l+1},$$

where the superscript l denotes the l-th step of the ADMM iteration, $\boldsymbol{\nu}_k$ is the scaled dual variable, and the parameter ρ affects the speed of convergence The proximal operator under ℓ_1 penalty could be defined element-wise as $\operatorname{prox}_a(\boldsymbol{b})_i = (|\boldsymbol{b}_i| - a)_+ \operatorname{sign}(\boldsymbol{b}_i)$ (Parikh et al., 2014). To acquire $\widehat{\boldsymbol{\theta}}_k^{l+1}$, we construct artificial observations $(\mathcal{X}_k', \boldsymbol{y}_k')$ and solve standard lasso problems $\widehat{\boldsymbol{\theta}}_k^{l+1} = \operatorname{argmin}_{\boldsymbol{\theta}_k \in \mathbb{R}^p} \|\boldsymbol{y}_k' - \mathcal{X}_k' \boldsymbol{\theta}_k\|_2^2 + n_k \lambda_{\mathcal{P}} \|\boldsymbol{\theta}_k\|_1$ via the cyclic coordinate descent (Friedman et al., 2010). Let

$$(\mathcal{X}_0', \boldsymbol{y}_0') = \begin{pmatrix} \sqrt{\alpha_0} \mathcal{X}_0 & \sqrt{\alpha_0} \boldsymbol{y}_0 \\ \sqrt{\frac{\rho}{2}} \boldsymbol{I}_{p \times p} & \sqrt{\frac{\rho}{2}} \left(\widehat{\boldsymbol{\delta}}_1^l + \widehat{\boldsymbol{\theta}}_1^{l+1} + \widehat{\boldsymbol{\nu}}_1^l \right) \\ \vdots & \vdots \\ \sqrt{\frac{\rho}{2}} \boldsymbol{I}_{p \times p} & \sqrt{\frac{\rho}{2}} \left(\widehat{\boldsymbol{\delta}}_K^l + \widehat{\boldsymbol{\theta}}_K^{l+1} + \widehat{\boldsymbol{\nu}}_K^l \right) \end{pmatrix},$$

while for $k \in [K]$, let

$$(\mathcal{X}_k', \boldsymbol{y}_k') = \begin{pmatrix} \sqrt{\alpha_k} \mathcal{X}_k & \sqrt{\alpha_k} \boldsymbol{y}_k \\ -\sqrt{\frac{\rho}{2}} \boldsymbol{I}_{p \times p} & \sqrt{\frac{\rho}{2}} \left(\widehat{\boldsymbol{\delta}}_k^l - \widehat{\boldsymbol{\theta}}_0^l + \widehat{\boldsymbol{\nu}}_k^l \right) \end{pmatrix}.$$

For empirical realizations, we set $\widehat{\boldsymbol{\theta}}_k^0$ as the lasso solution for the k-th dataset, and we also set $\widehat{\boldsymbol{\theta}}_k^{(0)} = \widehat{\boldsymbol{\theta}}_k^0$, $\widehat{\boldsymbol{v}}_k^0 = \mathbf{0}$ for $k \in \{0, 1, \cdots, K\}$, and $\widehat{\boldsymbol{\delta}}_k^{(0)} = \widehat{\boldsymbol{\delta}}_k^0 = \widehat{\boldsymbol{\theta}}_0^{(0)} - \widehat{\boldsymbol{\theta}}_k^{(0)}$ for $k \in \{1, 2, \cdots, K\}$.

S.2.2 Generalized low-rank trace regression

For generalized low-rank trace regression, we generate datasets for $k = 0, \ldots, 4$ using both both identity link $b'(x) = x$ (corresponding to the linear model) and the logit link $b'(x) = 1/(1 + e^{-x})$ (corresponding to the logistic model), with $r = 3$, $p_1 = p_2 = 20$, $n_k = 400$ for $k \in \{0, \cdots, 4\}$. Similarly, $k = 0$ is the target, $k = 1, 2$ are the useful sources, while $k = 3, 4$ are the non-informative sources.

We report the results in Table 1 based on 100 replications. We draw the conclusion that the truncated-penalized algorithm still performs well in terms of simultaneously identifying the informative auxiliary datasets and recovering low-rank parameters under the current setting.

DC-ADMM details

Here, we provide the numerical implementation details concerning the generalized low-rank trace regression. We first present the standard ADMM procedure to solve (2.7). Then, we also discuss the fine-tuning step under generalized low-rank trace regression for the sake of completeness.

Table 1: The mean and standard error of the simulation results for generalized low-rank trace regression with unknown informative sources. For the column Datasets, by Target we mean only the target dataset is used, Oracle means we only use the useful datasets, and All means we use all the source datasets. In the case of using all datasets, if the method include dataset selection capability, we report the (TPR,TNR) of dataset selection instead of All.

Estimator	Link	$\|\cdot^\star - \boldsymbol{\theta}_0^*\|_F$	Datasets	Estimator	Link	$\|\cdot^\star - \boldsymbol{\theta}_0^*\|_F$	Datasets
$\widehat{\boldsymbol{\theta}}_{\text{target}}$	linear	0.914 (0.073)	All	$\widehat{\boldsymbol{\theta}}_{\text{target}}$	logit	1.473 (0.087)	Target
$\widehat{\boldsymbol{\theta}}_{\mathcal{P}\cup\mathcal{A}^c}$		1.283 (0.085)	All	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}\cup\mathcal{A}^c}$		1.429 (0.062)	All
$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$		0.889 (0.056)	Oracle	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$		**1.282 (0.061)**	Oracle
$\widehat{\boldsymbol{\theta}}_{\text{TN}}$		**0.581 (0.043)**	(1.00,1.00)	$\widehat{\boldsymbol{\theta}}_{\text{TN}}$		1.326 (0.074)	(1.00,0.94)

We first present the standard ADMM procedures that can solve (2.7) with the nuclear norm penalty. Under the generalized linear model setting, recall that

$$\nabla \boldsymbol{L}_k(\boldsymbol{\theta}_k) = \sum_{i=1}^{n_k} \left[b'(\eta_{k,i}) - y_{k,i}\right] \boldsymbol{X}_{k,i}/n_k,$$

$$\nabla^2 \boldsymbol{L}_k(\boldsymbol{\theta}_k) = \sum_{i=1}^{n_k} b''(\eta_{k,i}) \text{vec}(\boldsymbol{X}_{k,i}) \text{vec}^\top(\boldsymbol{X}_{k,i})/n_k.$$

We define the singular value shrinkage operator $\mathcal{S}_\lambda(\boldsymbol{Y})$ for $\boldsymbol{Y}$ of rank r. Let $\boldsymbol{Y} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ where $\boldsymbol{\Sigma} = \text{diag}\left\{(\sigma_i)_{1\le i\le r}\right\}$, $\boldsymbol{U}$, and $\boldsymbol{V}$ are column orthogonal by the singular value decomposition, then $\mathcal{S}_\lambda(\boldsymbol{Y}) := \boldsymbol{U}\boldsymbol{\Sigma}_\lambda\boldsymbol{V}^\top$, $\boldsymbol{\Sigma}_\lambda = \text{diag}\left\{(\sigma_i - \lambda)_+\right\}$.

the standard ADMM procedure could then be implemented as:

$$\widehat{\boldsymbol{\theta}}_k^{l+1} = \mathcal{S}_{\frac{n_k\lambda_{\mathcal{P}}}{\rho_1+\rho_2}}\left(\frac{\rho_1}{\rho_1+\rho_2}\left[-\widehat{\boldsymbol{\delta}}_k^l + \widehat{\boldsymbol{\theta}}_0^l - \widehat{\boldsymbol{\nu}}_k^l\right] + \frac{\rho_2}{\rho_1+\rho_2}\left[\widehat{\boldsymbol{\gamma}}_k^l + \widehat{\boldsymbol{\theta}}_k^{(m)} + \widehat{\boldsymbol{\mu}}_k^l\right]\right), \quad k = 1, 2, \cdots, K,$$

$$\widehat{\boldsymbol{\theta}}_0^{l+1} = \mathcal{S}_{\frac{n_0\lambda_{\mathcal{P}}}{K\rho_1+\rho_2}}\left(\frac{1}{K\rho_1+\rho_2}\left[\rho_1\sum_{k=1}^{K}\left(\widehat{\boldsymbol{\delta}}_k^l + \widehat{\boldsymbol{\theta}}_k^{l+1} + \widehat{\boldsymbol{\nu}}_k^l\right) + \rho_2\left(\widehat{\boldsymbol{\gamma}}_0^l + \widehat{\boldsymbol{\theta}}_0^{(m)} + \widehat{\boldsymbol{\mu}}_0^l\right)\right]\right),$$

$$\widehat{\boldsymbol{\delta}}_k^{l+1} = \begin{cases} -\widehat{\boldsymbol{\theta}}_k^{l+1} + \widehat{\boldsymbol{\theta}}_0^{l+1} - \widehat{\boldsymbol{\nu}}_k^l, & \text{if } \|\widehat{\boldsymbol{\delta}}_k^{(m)}\|_N \geq \tau, \\ \mathcal{S}_{n_k\lambda_{\mathcal{Q}_k}/\rho_1}\left(-\widehat{\boldsymbol{\theta}}_k^{l+1} + \widehat{\boldsymbol{\theta}}_0^{l+1} - \widehat{\boldsymbol{\nu}}_k^l\right), & \text{if } \|\widehat{\boldsymbol{\delta}}_k^{(m)}\|_N < \tau, \end{cases} \quad k = 1, 2, \cdots, K,$$

$$\widehat{\boldsymbol{\gamma}}_k^{l+1} = \underset{\boldsymbol{\gamma}_k}{\operatorname{argmin}}\ n_k\alpha_k\boldsymbol{Q}(\boldsymbol{\gamma}_k; \widehat{\boldsymbol{\theta}}_k^{(m)}) + \frac{\rho_2}{2}\|\boldsymbol{\gamma}_k - \widehat{\boldsymbol{\theta}}_k^{l+1} + \widehat{\boldsymbol{\theta}}_k^{(m)} + \widehat{\boldsymbol{\mu}}_k^l\|_F^2, \quad k = 0, 1, \cdots, K,$$

$$\widehat{\boldsymbol{\nu}}_k^{l+1} = \widehat{\boldsymbol{\nu}}_k^l + \widehat{\boldsymbol{\delta}}_k^{l+1} + \widehat{\boldsymbol{\theta}}_k^{l+1} - \widehat{\boldsymbol{\theta}}_0^{l+1}, \quad k = 1, 2, \cdots, K,$$

$$\widehat{\boldsymbol{\mu}}_k^{l+1} = \widehat{\boldsymbol{\mu}}_k^l + \widehat{\boldsymbol{\gamma}}_k^{l+1} - \widehat{\boldsymbol{\theta}}_k^{l+1} + \widehat{\boldsymbol{\theta}}_k^{(m)}, \quad k = 0, 1, \cdots, K,$$

where $\boldsymbol{\nu}_k$ and $\boldsymbol{\mu}_k$ are scaled dual variables, and ρ_1 and ρ_2 affect the speed of convergence. By some simple algebra, the updating formula for $\widehat{\boldsymbol{\gamma}}_k^{l+1}$ is

$$\operatorname{vec}\left(\widehat{\boldsymbol{\gamma}}_k^{l+1}\right) = \mathbf{A}^{-}\left\{\rho_2\operatorname{vec}\left(\widehat{\boldsymbol{\theta}}_k^{l+1} - \widehat{\boldsymbol{\theta}}_k^{(m)} - \widehat{\boldsymbol{\mu}}_k^l\right) - \alpha_k\operatorname{vec}\left(\sum_{i=1}^{n_k}\left[\left(b'(\eta_{k,i}^{(m)}) - y_{k,i}\right)\boldsymbol{X}_{k,i}\right]\right)\right\},$$

where $\mathbf{A} = \left[\alpha_k\sum_{i=1}^{n_k} b''(\eta_{k,i}^{(m)})\operatorname{vec}(\boldsymbol{X}_{k,i})\operatorname{vec}^\top(\boldsymbol{X}_{k,i}) + \rho_2\boldsymbol{I}\right]$ and $\eta_{k,i}^{(m)} = \langle\boldsymbol{X}_{k,i}, \widehat{\boldsymbol{\theta}}_k^{(m)}\rangle$.

Naturally, we set $\widehat{\boldsymbol{\theta}}_k^0$ as the estimator by Fan et al. (2021), $\widehat{\boldsymbol{\theta}}_k^{(0)} = \widehat{\boldsymbol{\theta}}_k^0$, $\widehat{\boldsymbol{\mu}}_k^0 = \widehat{\boldsymbol{\gamma}}_k^0 = \mathbf{0}$, and $\widehat{\boldsymbol{v}}_k^0 = \mathbf{0}$ for $k \in \{0, 1, \cdots, K\}$ and set $\widehat{\boldsymbol{\delta}}_k^{(0)} = \widehat{\boldsymbol{\delta}}_k^0 = \widehat{\boldsymbol{\theta}}_0^{(0)} - \widehat{\boldsymbol{\theta}}_k^{(0)}$ hereafter.

Indeed, the fine-tuning step of the generalized low-rank trace regression is also not straightforward due to non-linearity. To fine-tune the generalized low-rank trace regression with a given $\widehat{\boldsymbol{\theta}}_{\mathcal{P}} \in \mathbb{R}^{d_1\times d_2}$, we rewrite the target problem (S.1.1) by omitting

the subscript 0 as:

$$\widehat{\boldsymbol{\delta}} = \underset{\boldsymbol{\delta}\in\mathbb{R}^{d_1\times d_2}}{\operatorname{argmin}} \ \frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(\boldsymbol{Z}_i, \widehat{\boldsymbol{\theta}}_{\mathcal{P}} + \boldsymbol{\delta}) + \lambda_d \mathcal{R}(\boldsymbol{\delta}), \tag{S.2.2}$$

where $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$ is known, $\mathcal{R} = \|\cdot\|_N$, and $\boldsymbol{L}(\boldsymbol{\delta}) = \sum_{i=1}^{n}\mathcal{L}\left(\boldsymbol{Z}_i, \widehat{\boldsymbol{\theta}}_{\mathcal{P}} + \boldsymbol{\delta}\right)/n$. Analogously, the local quadratic approximation of (S.2.2) is:

$$\left(\widehat{\boldsymbol{\delta}}^{(m+1)}, \widehat{\boldsymbol{\gamma}}\right) = \underset{\boldsymbol{\delta},\boldsymbol{\gamma}\in\mathbb{R}^{d_1\times d_2}}{\operatorname{argmin}} \ \boldsymbol{Q}(\boldsymbol{\gamma}; \widehat{\boldsymbol{\delta}}^{(m)}) + \lambda_d\|\boldsymbol{\delta}\|_N \quad \text{subject to} \quad \boldsymbol{\gamma} = \boldsymbol{\delta} - \widehat{\boldsymbol{\delta}}^{(m)},$$

where for $\eta_i^{(m)} = \langle \boldsymbol{X}_i, \widehat{\boldsymbol{\theta}}_{\mathcal{P}} + \widehat{\boldsymbol{\delta}}^{(m)}\rangle$:

$$\boldsymbol{Q}(\boldsymbol{\gamma}; \widehat{\boldsymbol{\delta}}^{(m)}) = \operatorname{vec}^{\top}(\boldsymbol{\gamma})\,\nabla^2\boldsymbol{L}(\widehat{\boldsymbol{\delta}}^{(m)})\operatorname{vec}(\boldsymbol{\gamma})/2 + \operatorname{vec}^{\top}(\boldsymbol{\gamma})\operatorname{vec}\left(\nabla\boldsymbol{L}(\widehat{\boldsymbol{\delta}}^{(m)})\right),$$

$$\nabla\boldsymbol{L}(\widehat{\boldsymbol{\delta}}^{(m)}) = \sum_{i=1}^{n}\left[b'\left(\eta_i^{(m)}\right) - y_i\right]\boldsymbol{X}_i/n,$$

$$\nabla^2\boldsymbol{L}(\widehat{\boldsymbol{\delta}}^{(m)}) = \sum_{i=1}^{n} b''\left(\eta_i^{(m)}\right)\operatorname{vec}(\boldsymbol{X}_i)\operatorname{vec}^{\top}(\boldsymbol{X}_i)/n.$$

Accordingly, we can apply the standard ADMM procedure:

$$\begin{aligned}
\widehat{\boldsymbol{\delta}}^{l+1} &= \mathcal{S}_{\lambda_d/\rho}\left(\widehat{\boldsymbol{\gamma}}^{l} + \widehat{\boldsymbol{\delta}}^{(m)} + \widehat{\boldsymbol{\nu}}^{l}\right),\\
\widehat{\boldsymbol{\gamma}}^{l+1} &= \underset{\boldsymbol{\gamma}\in\mathbb{R}^{d_1\times d_2}}{\operatorname{argmin}} \ \boldsymbol{Q}(\boldsymbol{\gamma}; \widehat{\boldsymbol{\delta}}^{(m)}) + \frac{\rho}{2}\|\gamma - \widehat{\boldsymbol{\delta}}^{l+1} + \widehat{\boldsymbol{\delta}}^{(m)} + \widehat{\boldsymbol{\nu}}^{l}\|_F^2,\\
\widehat{\boldsymbol{\nu}}^{l+1} &= \widehat{\boldsymbol{\nu}}^{l} + \widehat{\boldsymbol{\gamma}}^{l+1} - \widehat{\boldsymbol{\delta}}^{l+1} + \widehat{\boldsymbol{\delta}}^{(m)},
\end{aligned}$$

where $\boldsymbol{\nu}$ is the scaled dual variable and ρ affects the speed of convergence. We have:

$$\operatorname{vec}\left(\widehat{\boldsymbol{\gamma}}^{l+1}\right) = \mathbf{A}^{-}\left(\rho\operatorname{vec}\left(\widehat{\boldsymbol{\delta}}^{l+1} - \widehat{\boldsymbol{\delta}}^{(m)} - \widehat{\boldsymbol{\nu}}^{l}\right) - \operatorname{vec}\left(\sum_{i=1}^{n}\left[\left(b'\left(\eta_i^{(m)}\right) - y_i\right)\boldsymbol{X}_i\right]\right)/n\right),$$

where $\mathbf{A} = \rho \boldsymbol{I} + \sum_{i=1}^{n} b''\left(\eta_i^{(m)}\right) \mathrm{vec}(\boldsymbol{X}_i)\mathrm{vec}^\top(\boldsymbol{X}_i)/n$. Note that the ADMM algorithm is not affected by the initial values since the problem is a convex optimization problem. In practice, we could set $\widehat{\boldsymbol{v}}^0 = \mathbf{0}$, $\widehat{\boldsymbol{\delta}}^0 = \widehat{\boldsymbol{\delta}}^{(0)} = \mathbf{0}$, $\widehat{\boldsymbol{\gamma}}^0 = \mathbf{0}$.

S.2.3 Enlarging regularization

Here, we numerically validate our claims about enlarging the regularization. For numerical experiments, we generate useful datasets for $k \in \mathcal{P} = \{0, \ldots 5\}$ in the same way as in the main article on sparse linear regression, except that here we do not deliberately set $\boldsymbol{\theta}_{k1} = -0.4$ for all $k \in [K]$. Note that $\boldsymbol{\theta}_{01} = 0.4$ and the non-informative sources are not included.

We set $p = 500$, $n_0 = 250$, $n_1, \cdots, n_5 = 400$, and report the results in Table 2 based on 100 replications. We consider the following competitors (together with their fine-tuned versions denoted by the additional superscript $\cdot^\star$) and sketch how to select tuning parameters for different methods:

(a) the target lasso estimator $\widehat{\boldsymbol{\theta}}_{\text{target}}$ using target data and λ_{target} obtained through cross validation;

(b) the oracle pooling estimator $\widetilde{\boldsymbol{\theta}}_{\mathcal{P}}$ that targets the population parameter $\boldsymbol{\theta}^*_{\mathcal{P}}$, for which the tuning parameter $\lambda_{\widetilde{\mathcal{P}}}$ is naturally (and most commonly) chosen by cross validation using all pooling samples;

Table 2: The means (standard deviations) of losses under the same settings of Table 3 in the article, except here we do not deliberately set $\boldsymbol{\theta}_{k1} = -0.4$ for all $k \in [K]$. Note that $\boldsymbol{\theta}_{01} = 0.4$. Here the tuning parameter of $\widetilde{\boldsymbol{\theta}}$ is acquired by cross validation whose validation set consists of the pooled dataset (which is often the default choice). Meanwhile, the tuning parameter of $\widehat{\boldsymbol{\theta}}$ is acquired by cross validation whose validation set consists of target data only (ending up in a larger λ as expected).

Setting	Estimator	$\|\cdot - \boldsymbol{\theta}_0^*\|_2$	$\|\cdot^\star - \boldsymbol{\theta}_0^*\|_2$	Selected λ	$\|\boldsymbol{\theta}_{\mathcal{P}}^* - \boldsymbol{\theta}_0^*\|_1$	$\|\widehat{\boldsymbol{\theta}}_{\text{target}} - \boldsymbol{\theta}_0^*\|_2$
Ho_s	$\widetilde{\boldsymbol{\theta}}_{\mathcal{P}}$	0.320(0.026)	0.330(0.036)	0.038(0.003)	1.038(0.000)	0.550(0.125)
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$	0.322(0.027)	0.332(0.037)	0.049(0.009)		
He_s	$\widetilde{\boldsymbol{\theta}}_{\mathcal{P}}$	0.356(0.026)	0.363(0.033)	0.036(0.004)	5.001(0.147)	0.650(0.197)
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$	0.351(0.026)	0.358(0.035)	0.047(0.007)		
Ho_d	$\widetilde{\boldsymbol{\theta}}_{\mathcal{P}}$	0.365(0.035)	0.371(0.038)	0.038(0.004)	6.565(0.000)	0.550(0.125)
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$	0.348(0.034)	0.355(0.036)	0.056(0.008)		
He_d	$\widetilde{\boldsymbol{\theta}}_{\mathcal{P}}$	0.389(0.042)	0.398(0.049)	0.035(0.005)	8.618(0.196)	0.650(0.197)
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$	0.348(0.028)	0.359(0.040)	0.058(0.009)		

(c) the oracle pooling estimator $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$ with the tuning parameter $\lambda_{\widehat{\mathcal{P}}}$ selected by cross validation using target samples;

(d) the fine-tuned versions $\widetilde{\boldsymbol{\theta}}_{\mathcal{P}}^\star$ and $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}^\star$ (for the sake of fairness in comparisons, recall that $\widetilde{\boldsymbol{\theta}}_{\mathcal{P}}$ is originally used in two-step methods), with the tuning parameters in the second step selected by cross validation using only the target dataset.

While both oracle transfer learning estimators outperform the original target estimation, as $\boldsymbol{\theta}_{\mathcal{P}}^*$ gets further away from $\boldsymbol{\theta}_0^*$ under the dense contrasts or heterogeneous covariate cases, indicated by larger $\|\boldsymbol{\theta}_{\mathcal{P}}^* - \boldsymbol{\theta}_0^*\|_1$, the default estimator $\widetilde{\boldsymbol{\theta}}_{\mathcal{P}}$ becomes less reliable. On the other hand, cross validation using only the target samples naturally

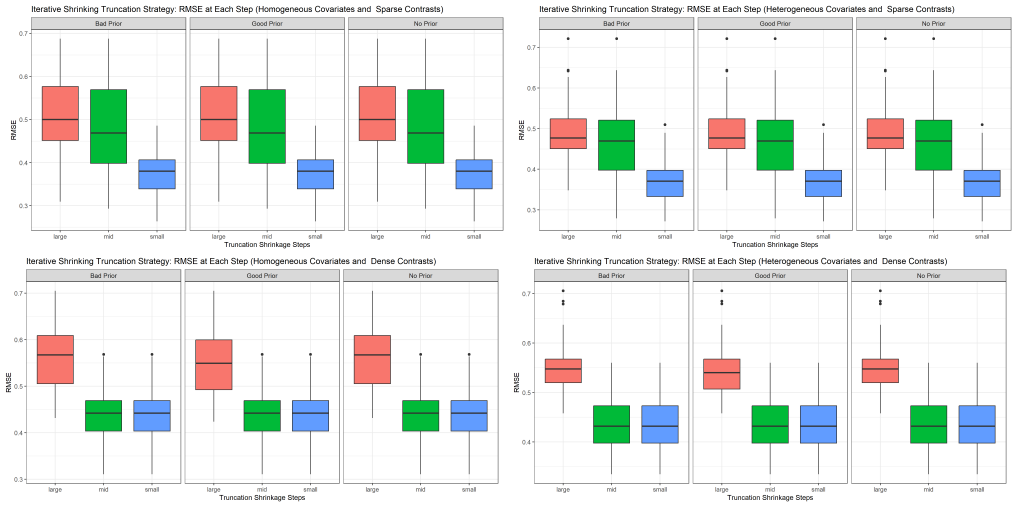

Figure 1: The boxplots of the RMSE with respect to each step from the shrinking τ strategy the same settings as Table 1 of the main article, based on 100 replications. The numerical stability of the non-convex algorithm is guaranteed under different initializations.

gives us larger λ, namely stronger regularization strength. Moreover, it leads to a more reliable estimator $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$. In addition, the fine-tuning step seems unnecessary in all these cases.

S.2.4 The shrinking τ strategy

To avoid the vicious cycle caused by poor initializations, we suggest a shrinking τ strategy that iteratively uses the output of our algorithm equipped with a larger τ as the input for our algorithm equipped with a smaller τ. The rationale behind the shrinking τ strategy is that if we take a sufficiently large τ and all sources are included blindly, then the optimization problem can almost be viewed as a convex one, and

the poor initialization does not matter much. The resulting output of this large τ algorithm is numerically stable but also biased due to the inclusion of non-informative sources. Then, we take this output to be the initialization of the algorithm with a smaller τ to filter out the useless sources. The procedure above can be carried out in an iterative manner (as τ shrinks), and the resulting estimator from the final round (with a suitable τ) is more numerically stable.

In the following, we report the numerical performance of the shrinking τ strategy. We work under almost the same settings as those in Table 1 of the main article. The only difference is that, instead of using the local lasso estimates as the initialization, we artificially construct three types of initialization: (1) bad prior: all $\boldsymbol{\theta}_k = \boldsymbol{\theta}_K^*$ (the K-th dataset is useless) for $k \in \{0\} \cup [K]$; (2) good prior: $\boldsymbol{\theta}_k = \boldsymbol{\theta}_k^*$ for $k \in \{0\} \cup [K]$; and (3) no prior: all $\boldsymbol{\theta}_k = \mathbf{0}$ for $k \in \{0\} \cup [K]$. We successively apply the proposed algorithm with $\tau = 60$ (large), $\tau = 30$ (mid), and $\tau = 15$ (small). The initialization of the large τ is taken from the three priors above, and the inputs of the smaller τ steps are the preceding outputs. We can see that the shrinking τ strategy is able to ensure the numerical stability of the non-convex algorithm, even with very poor initializations (bad priors).

S.2.5 Further numerical results

In this section, we report further numerical results (under 100 replications) to reflect the effects of high-dimensionality and the performance of different methods under more correlated covariates. We shall see that the proposed one-step algorithm remains numerically stable as the dimension increases or as the correlation grows.

For the effect of high-dimensionality, the settings in the main article ($n_0 = 250$, $p = 500$, $n_k = 400$ for $k \neq 0$) are extended to ($n_0 = 500$, $p = 1000$, $n_k = 800$ for $k \neq 0$) and ($n_0 = 750$, $p = 1500$, $n_k = 1200$ for $k \neq 0$). As the dimension p increases, we accordingly enlarge the candidate grid size for the tuning parameter τ, as the distance between the parameters of different datasets naturally increases with the growth of dimensionality in our simulation settings. We only report the case of sparse contrasts and heterogeneous covariates (He_s in the main article), as the results are almost identical in other settings. The results are collected in Table 3, where we can see that the performance of the proposed algorithm is numerically stable as the dimensions increase.

Meanwhile, to increase the correlation between the covariates, we adopt the following spiked covariance structure:

$$\boldsymbol{\Sigma}_k = \frac{\lambda_{\text{spike}}}{p} \boldsymbol{U}_k \boldsymbol{U}_k^\top + \frac{2}{3p} \boldsymbol{\Lambda}_k^\top \boldsymbol{\Lambda}_k,$$

where $\boldsymbol{\Lambda}_k$ is a matrix of $1.5p$ rows and p columns, and $\boldsymbol{U}_k$ is a matrix of p rows and

Table 3: The means (standard deviations) of the simulation results for different methods with the growing dimensionality and sample sizes. We only report the case of sparse contrasts and heterogeneous covariates (He_s in the main article), as the results are almost identical in other settings. For the column Datasets, by "Target" we mean only the target dataset is used, "Oracle" means we only use the useful datasets, and "All" means we use all the source datasets. In the case of using all datasets, if the method has dataset selection capability, we report the (TPR,TNR) of dataset selection instead of "All".

(n_k, n_0, p)	Estimator	$\|\cdot - \boldsymbol{\theta}_0^*\|_2$	$\|\cdot^\star - \boldsymbol{\theta}_0^*\|_2$	Datasets
$(400, 250, 500)$	$\widehat{\boldsymbol{\theta}}_{\text{target}}$	0.782(0.097)	NA	Target
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$	0.750(0.034)	0.406(0.063)	Oracle
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}\cup\mathcal{A}^c}$	1.190(0.092)	0.966(0.115)	All
	$\widehat{\boldsymbol{\theta}}_{\text{TN}}$	**0.397(0.105)**	**0.347(0.106)**	(1.000,0.000)
$(800, 500, 1000)$	$\widehat{\boldsymbol{\theta}}_{\text{target}}$	0.789(0.079)	NA	Target
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$	0.757(0.022)	0.374(0.035)	Oracle
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}\cup\mathcal{A}^c}$	1.621(0.078)	1.375(0.087)	All
	$\widehat{\boldsymbol{\theta}}_{\text{TN}}$	**0.314(0.034)**	**0.292(0.034)**	(1.000,0.000)
$(1200, 750, 1500)$	$\widehat{\boldsymbol{\theta}}_{\text{target}}$	0.775(0.059)	NA	Target
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$	0.764(0.016)	0.362(0.021)	Oracle
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}\cup\mathcal{A}^c}$	1.996(0.076)	1.704(0.077)	All
	$\widehat{\boldsymbol{\theta}}_{\text{TN}}$	**0.293(0.023)**	**0.277(0.022)**	(1.000,0.000)

r columns. The elements in both $\boldsymbol{\Lambda}_k$ and $\boldsymbol{U}_k$ are drawn independently from $N(0,1)$. The spiked covariance model is commonly used to model stronger correlations among the covariates (Paul, 2007; Baik and Silverstein, 2006; Bai et al., 2018), where the spike parameter λ_{spike} reflects the correlation strength. Note that $2\boldsymbol{\Lambda}_k^\top \boldsymbol{\Lambda}_k/3p$ is an unbiased estimator of $\boldsymbol{I}_p$, resulting in a weakly correlated $\boldsymbol{\Sigma}_k$ if $\lambda_{\text{spike}} = 0$ (which degenerates to the default heterogeneous setting in the main article). For $\lambda_{\text{spike}} > 0$, $\boldsymbol{\Sigma}_k$ admits a spiked covariance structure: a larger λ_{spike} implies a stronger cross-sectional correlation. For our experiments, we set $r = 10$ and let $\lambda_{\text{spike}} \in \{0, 5, 10\}$. As the $\boldsymbol{U}_k$'s are independent across different sources, we expect the covariates to be more heterogeneous as λ_{spike} increases.

In Table 4, we report the experimental results under growing λ_{spike} with ($n_0 = 500$, $p = 1000$, $n_k = 800$ for $k \neq 0$); we only report the case of sparse contrasts here, as the results are almost identical for the dense contrasts. The performance of the proposed one-round method remains stable. This is indeed foreseeable, as alluded to in the Introduction of the main article: our method is able to handle both model shift (slight differences in model parameters) and covariate shift (in this case, different covariance structures of the covariates). As a quick reminder, no covariate homogeneity condition is required for our method to converge theoretically.

In summary, the proposed one-step method adapts readily to higher dimensions and more correlated covariates, demonstrating its algorithmic robustness and poten-

Table 4: The means (standard deviations) of the simulation results for different methods under different covariance spikes, where $n_k = 800$, $n_0 = 500$, and $p = 1000$. We only report the case of sparse contrasts here, as the results are almost identical for the dense contrasts. For the column Datasets, by "Target" we mean only the target dataset is used, "Oracle" means we only use the useful datasets, and "All" means we use all the source datasets. In the case of using all datasets, if the method has dataset selection capability, we report the (TPR,TNR) of dataset selection instead of "All".

Correlation	Estimator	$\|\cdot - \boldsymbol{\theta}_0^*\|_2$	$\|\cdot^\star - \boldsymbol{\theta}_0^*\|_2$	Datasets
$\lambda_{\text{spike}} = 0$	$\widehat{\boldsymbol{\theta}}_{\text{target}}$	0.783(0.074)	NA	Target
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$	0.762(0.022)	0.379(0.031)	Oracle
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}\cup\mathcal{A}^c}$	1.625(0.078)	1.380(0.092)	All
	$\widehat{\boldsymbol{\theta}}_{\text{TN}}$	**0.320(0.033)**	**0.296(0.034)**	(1.000,0.000)
$\lambda_{\text{spike}} = 5$	$\widehat{\boldsymbol{\theta}}_{\text{target}}$	0.771(0.069)	NA	Target
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$	0.760(0.020)	0.375(0.029)	Oracle
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}\cup\mathcal{A}^c}$	1.651(0.082)	1.403(0.076)	All
	$\widehat{\boldsymbol{\theta}}_{\text{TN}}$	**0.310(0.031)**	**0.290(0.032)**	(1.000,0.000)
$\lambda_{\text{spike}} = 10$	$\widehat{\boldsymbol{\theta}}_{\text{target}}$	0.772(0.068)	NA	Target
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$	0.764(0.021)	0.388(0.030)	Oracle
	$\widehat{\boldsymbol{\theta}}_{\mathcal{P}\cup\mathcal{A}^c}$	1.709(0.079)	1.469(0.083)	All
	$\widehat{\boldsymbol{\theta}}_{\text{TN}}$	**0.311(0.033)**	**0.293(0.032)**	(1.000,0.000)

tial practicality in real world applications.

S.3 Proofs of Theoretical Results

Finally, we present the proofs of our theoretical results.

S.3.1 Proof of Proposition 1

Recall that $\widehat{\boldsymbol{\Theta}}^{(m+1)}$ and $\widehat{\boldsymbol{\delta}}^{(m+1)}$ are the minimizers of $S^{(m+1)}(\boldsymbol{\Theta}, \boldsymbol{\delta})$; then we obtain

$$\begin{aligned} 0 \leq S\left(\widehat{\boldsymbol{\Theta}}^{(m)}, \widehat{\boldsymbol{\delta}}^{(m)}\right) &= S^{(m+1)}\left(\widehat{\boldsymbol{\Theta}}^{(m)}, \widehat{\boldsymbol{\delta}}^{(m)}\right) \\ &\leq S^{(m)}\left(\widehat{\boldsymbol{\Theta}}^{(m)}, \widehat{\boldsymbol{\delta}}^{(m)}\right) \leq S^{(m)}\left(\widehat{\boldsymbol{\Theta}}^{(m-1)}, \widehat{\boldsymbol{\delta}}^{(m-1)}\right) \\ &= S\left(\widehat{\boldsymbol{\Theta}}^{(m-1)}, \widehat{\boldsymbol{\delta}}^{(m-1)}\right). \end{aligned} \tag{S.3.3}$$

The remaining parts can be obtained by following arguments similar to those in Wu et al. (2016); Liu et al. (2023).

S.3.2 Proof of Theorem 1

Note that the truncated norm penalty makes the problem (2.5) non-convex, so all minima here are discussed in a local manner. It helps to decompose (2.5) into sub-problems. First, for any $\boldsymbol{\theta}_0'$ in some set $\mathcal{O}_{\mathcal{P}}$, we acquire the best response of $\widehat{\boldsymbol{\theta}}_k$ as

$$\begin{aligned} \widehat{\boldsymbol{\theta}}_k(\boldsymbol{\theta}_0') &= \underset{\boldsymbol{\theta}_k \in \mathbb{R}^p}{\operatorname{argmin}} \widehat{\boldsymbol{L}}_k(\boldsymbol{\theta}_k; \boldsymbol{\theta}_0') \\ &= \underset{\boldsymbol{\theta}_k \in \mathbb{R}^p}{\operatorname{argmin}} \underbrace{\boldsymbol{L}_k\left(\boldsymbol{\theta}_k\right) + \lambda_{\mathcal{P}} \mathcal{R}\left(\boldsymbol{\theta}_k\right)}_{\text{single-task}(k)} + \underbrace{\lambda_{\mathcal{Q}_k} \min\left[\mathcal{R}\left(\boldsymbol{\theta}_k - \boldsymbol{\theta}_0'\right), \tau\right]}_{\text{TNP}(k)}. \end{aligned} \tag{S.3.4}$$

We then plug-in the best responses and solve locally for $\widehat{\boldsymbol{\theta}}_0$ by

$$\widehat{\boldsymbol{\theta}}_0 = \underset{\boldsymbol{\theta}_0' \in \mathcal{O}_{\mathcal{P}}}{\operatorname{argmin}} \left[\frac{n_0}{N} \boldsymbol{L}_0(\boldsymbol{\theta}_0') + \frac{n_0}{N} \lambda_{\mathcal{P}} \mathcal{R}\left(\boldsymbol{\theta}_0'\right) \right] + \sum_{1 \le k \le K} \frac{n_k}{N} \widehat{\boldsymbol{L}}_k \left(\widehat{\boldsymbol{\theta}}_k(\boldsymbol{\theta}_0'); \boldsymbol{\theta}_0' \right). \tag{S.3.5}$$

For the informative datasets $k \in \mathcal{A}$, recalling the variance-bias decomposition in (3.10), we have $\mathcal{R}^*\left(\nabla \boldsymbol{L}_{\mathcal{P}}(\boldsymbol{\theta}_0^*)\right) \lesssim v_{\mathcal{P}} + h$ since $\|\nabla^2 \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)\|_{\mathcal{B}_k \to \mathcal{R}^*} \le M$ for $k \in \mathcal{A}$ and $h \to 0$. According to Proposition 2, for $\lambda_{\mathcal{P}} \gtrsim v_{\mathcal{P}} + h$, we have $\mathcal{R}(\widehat{\boldsymbol{\theta}}_{\mathcal{P}} - \boldsymbol{\theta}_0) \lesssim \lambda_{\mathcal{P}}$ for the oracle pooling estimator $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$. Let $l_{\mathcal{P}} \asymp v_{\mathcal{P}} + h$ be sufficiently large, so that the open set $\mathcal{O}_{\mathcal{P}} = \{\boldsymbol{\theta} | \mathcal{R}(\boldsymbol{\theta} - \boldsymbol{\theta}_0^*) < l_{\mathcal{P}}\}$ contains $\widehat{\boldsymbol{\theta}}_{\mathcal{P}}$. Now, for any $\boldsymbol{\theta}_0' \in \mathcal{O}_{\mathcal{P}}$, define $\widehat{\boldsymbol{\delta}}_k'(\boldsymbol{\theta}_0') = \boldsymbol{\theta}_0' - \widehat{\boldsymbol{\theta}}_k(\boldsymbol{\theta}_0')$; we rewrite (S.3.4) in the open set $\mathcal{R}(\widehat{\boldsymbol{\delta}}_k') < \tau$ as

$$\widehat{\boldsymbol{\delta}}_k'(\boldsymbol{\theta}_0') = \underset{\mathcal{R}(\boldsymbol{\delta}_k) < \tau}{\operatorname{argmin}} \boldsymbol{L}_k\left(\boldsymbol{\theta}_0' - \boldsymbol{\delta}_k\right) + \lambda_{\mathcal{P}} \mathcal{R}\left(\boldsymbol{\theta}_0' - \boldsymbol{\delta}_k\right) + \lambda_{\mathcal{Q}_k} \mathcal{R}\left(\boldsymbol{\delta}_k\right).$$

By the convexity of $\boldsymbol{L}_k$ and the triangular inequality, we have

$$\begin{aligned}
&\boldsymbol{L}_k\left(\boldsymbol{\theta}_0' - \boldsymbol{\delta}_k\right) + \lambda_{\mathcal{P}} \mathcal{R}\left(\boldsymbol{\theta}_0' - \boldsymbol{\delta}_k\right) + \lambda_{\mathcal{Q}_k} \mathcal{R}\left(\boldsymbol{\delta}_k\right) \\
&\quad \ge \boldsymbol{L}_k\left(\boldsymbol{\theta}_0'\right) - \mathcal{R}^*\left(\nabla \boldsymbol{L}_k(\boldsymbol{\theta}_0')\right) \mathcal{R}(\boldsymbol{\delta}_k) + \lambda_{\mathcal{P}} \mathcal{R}(\boldsymbol{\theta}_0') - \lambda_{\mathcal{P}} \mathcal{R}(\boldsymbol{\delta}_k) + \lambda_{\mathcal{Q}_k} \mathcal{R}(\boldsymbol{\delta}_k) \\
&\quad \ge \boldsymbol{L}_k\left(\boldsymbol{\theta}_0'\right) + \lambda_{\mathcal{P}} \mathcal{R}\left(\boldsymbol{\theta}_0'\right),
\end{aligned} \tag{S.3.6}$$

as long as $\lambda_{\mathcal{Q}_k} \ge \lambda_{\mathcal{P}} + \mathcal{R}^*\left(\nabla \boldsymbol{L}_k(\boldsymbol{\theta}_0')\right)$. For $v_k = \mathcal{R}^*(\nabla \boldsymbol{L}_k(\boldsymbol{\theta}_k^*))$, we have

$$\begin{aligned}
\mathcal{R}^*\left(\nabla \boldsymbol{L}_k(\boldsymbol{\theta}_0')\right) &= \mathcal{R}^*\left(\nabla \boldsymbol{L}_k(\boldsymbol{\theta}_k^* + \boldsymbol{\theta}_0' - \boldsymbol{\theta}_k^*)\right) \\
&\lesssim v_k + \mathcal{R}^*\left(\nabla^2 \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)(\boldsymbol{\theta}_0' - \boldsymbol{\theta}_k^*)\right) \\
&\le v_k + \mathcal{R}^*\left(\nabla^2 \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)(\boldsymbol{\theta}_0' - \boldsymbol{\theta}_0^*)\right) + \mathcal{R}^*\left(\nabla^2 \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)(\boldsymbol{\theta}_0^* - \boldsymbol{\theta}_k^*)\right) \\
&\lesssim v_k + h,
\end{aligned}$$

since $\max(\|\nabla^2 \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)\|_{\mathcal{R}\to\mathcal{R}^*}, \|\nabla^2 \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)\|_{\mathcal{B}_k\to\mathcal{R}^*}) \leq M$, $\mathcal{R}(\boldsymbol{\theta}_0' - \boldsymbol{\theta}_0^*) \lesssim v_p + h \to 0$, $\mathcal{B}(\boldsymbol{\theta}_0^* - \boldsymbol{\theta}_k^*) \leq h$, and $v_{\mathcal{P}} \lesssim v_k$. That is to say, for $\lambda_{\mathcal{Q}_k} \gtrsim v_k + h$, we have $\widehat{\boldsymbol{\delta}}_k'(\boldsymbol{\theta}_0') = \mathbf{0}$ for all $\boldsymbol{\theta}_0' \in \mathcal{O}_{\mathcal{P}}$ according to (S.3.6).

Then, for non-informative datasets $k \in \mathcal{A}^c$, the first part of the problem (S.3.4) is essentially the single-task estimation using the k-th dataset only, whose minimizer is denoted by $\widehat{\boldsymbol{\theta}}_k' = \operatorname{argmin}_{\boldsymbol{\theta}_k \in \mathbb{R}^p} \boldsymbol{L}_k(\boldsymbol{\theta}_k) + \lambda_{\mathcal{P}} \mathcal{R}(\boldsymbol{\theta}_k)$. For the non-informative study k, since $\mathcal{R}(\widehat{\boldsymbol{\theta}}_k' - \boldsymbol{\theta}_0^*) > 2\tau$ by assumption, we have $\mathcal{R}(\widehat{\boldsymbol{\theta}}_k' - \boldsymbol{\theta}_0') > \tau$ for $\boldsymbol{\theta}_0' \in \mathcal{O}_{\mathcal{P}}$. The second part of (S.3.4) is then fixed as $\lambda_{\mathcal{Q}_k}\tau$ in an open neighborhood of $\widehat{\boldsymbol{\theta}}_k'$, so that $\widehat{\boldsymbol{\theta}}_k'$ is indeed a local minimum of (S.3.4) and $\widehat{\boldsymbol{\theta}}_k(\boldsymbol{\theta}_0') = \widehat{\boldsymbol{\theta}}_k'$ for all $\boldsymbol{\theta}_0' \in \mathcal{O}_{\mathcal{P}}$.

In the end, plug the best responses into the problem (S.3.5); the resulting problem is then equivalent to oracle pooling by (2.4), and the solution is $\widehat{\boldsymbol{\theta}}_0 = \widehat{\boldsymbol{\theta}}_{\mathcal{P}}$. The proof is complete.

S.3.3 Proof of Corollary 1

First, for the restricted strong convexity (RSC) condition, we take the following result from Raskutti et al. (2010), such that there are positive constants $(\kappa_{k,1}, \kappa_{k,2})$, depending only on $\boldsymbol{\Sigma}_k$, for $\widehat{\boldsymbol{\Sigma}}_k = \mathcal{X}_k^\top \mathcal{X}_k / n_k$,

$$\left\langle \boldsymbol{\Delta}, \widehat{\boldsymbol{\Sigma}}_k \boldsymbol{\Delta} \right\rangle \geq \kappa_{k,1} \|\boldsymbol{\Delta}\|^2 - \kappa_{k,2} \frac{\log p}{n_k} \|\boldsymbol{\Delta}\|_1^2, \quad \text{for all } \boldsymbol{\Delta} \in \mathbb{R}^p. \tag{S.3.7}$$

with a probability greater than $1 - c_{k,1} \exp(-c_{k,2} n_k)$. We then focus on the Hessian matrix of $\boldsymbol{L}_{\mathcal{P}}$, which is $\sum_{k\in\mathcal{P}} n_k \widehat{\boldsymbol{\Sigma}}_k / n_{\mathcal{P}}$.

Lemma 1 (RSC Conditions)**.** *Under the settings of Corollary 1, let* $\boldsymbol{\Delta}_s$ *(or* $\boldsymbol{\Delta}_{s^c}$*) be the projection of* $\boldsymbol{\Delta}$ *onto the s-sparse support (or its complement). There exists a positive constant* $\kappa_{\mathcal{P}}$ *such that for all* $\|\boldsymbol{\Delta}_{s^c}\|_1 \leq 3\|\boldsymbol{\Delta}_s\|_1$*,*

$$\frac{1}{n_{\mathcal{P}}}\left\langle \boldsymbol{\Delta}, \sum_{k\in\mathcal{P}} n_k \widehat{\boldsymbol{\Sigma}}_k \boldsymbol{\Delta} \right\rangle \geq \left[\kappa_{\mathcal{P}} - \sum_{k\in\mathcal{P}} 16 s \kappa_{k,2} \frac{\log p}{n_k}\right] \|\boldsymbol{\Delta}\|^2, \tag{S.3.8}$$

with a probability greater than $1 - \sum_{k\in\mathcal{P}} c_{k,1} \exp\left(-c_{k,2} n_k\right)$*.*

Then, we give the following result concerning the rate of $\mathcal{R}^*\left(\nabla \boldsymbol{L}_{\mathcal{P}}(\boldsymbol{\theta}_0^*)\right)$.

Lemma 2 (Convergence Rates)**.** *Under the settings of Corollary 1, we have as* $\min_{k\in\mathcal{P}} n_k \to \infty$*,* $p \to \infty$*, and* $h \to 0$*,*

$$\mathcal{R}^*\left(\nabla \boldsymbol{L}_{\mathcal{P}}(\boldsymbol{\theta}_0^*)\right) = O_p(\underbrace{\sqrt{\frac{\log p}{n_{\mathcal{P}}}}}_{v_{\mathcal{P}}} + \underbrace{h}_{b_{\mathcal{P}}}). \tag{S.3.9}$$

With the help of Lemma 1 and Lemma 2, while $\max_{k\in\mathcal{P}} \log p/n_k \to 0$ and $h \to 0$, Corollary 1 holds according to Proposition 2. Then we give the proofs of Lemma 1 and Lemma 2.

Proof of Lemma 1

It is the straightforward consequence of (S.3.7). For $\|\boldsymbol{\Delta}_{s^c}\|_1 \leq 3\|\boldsymbol{\Delta}_s\|_1$, we have $\|\boldsymbol{\Delta}\|_1 = \|\boldsymbol{\Delta}_s\|_1 + \|\boldsymbol{\Delta}_{s^c}\|_1 \leq 4\|\boldsymbol{\Delta}_s\|_1 \leq 4\sqrt{s}\|\boldsymbol{\Delta}\|$; then (S.3.8) holds with $\kappa_{\mathcal{P}} := \sum_{k\in\mathcal{P}} n_k \kappa_{k,1}/n_{\mathcal{P}}$ by the union bound of probability.

Proof of Lemma 2

Controlling $\mathcal{R}^*(\nabla \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)) \asymp \|\mathcal{X}_k^\top \boldsymbol{\epsilon}_k\|_\infty / n_k$ is straightforward by noticing that the maximum of a p-dimensional vector with zero mean sub-Gaussian elements, with variance proxies of order n_k, is controlled by $(n_k \log p)^{1/2}$ using standard union bound arguments. As for $\mathcal{R}^*\left(\nabla \boldsymbol{L}_{\mathcal{P}}(\boldsymbol{\theta}_0^*)\right)$, for h sufficiently small, it suffices to bound $v_{\mathcal{P}} = \mathcal{R}^*\left(\sum_{k\in\mathcal{P}} n_k \nabla \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)\right)/n_{\mathcal{P}}$ and $b_{\mathcal{P}} = \sum_{k\in\mathcal{P}} n_k \mathcal{R}^*\left(\nabla^2 \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)\boldsymbol{\delta}_k^*\right)/n_{\mathcal{P}}$. First, $v_{\mathcal{P}} \asymp \|\sum_{k\in\mathcal{P}} \mathcal{X}_k^\top \boldsymbol{\epsilon}_k\|_\infty / n_{\mathcal{P}} = O_p(\sqrt{\log p / n_{\mathcal{P}}})$ by noticing that each element of $\sum_{k\in\mathcal{P}} \mathcal{X}_k^\top \boldsymbol{\epsilon}_k$ is the sum of $n_{\mathcal{P}}$ independent centered random variables and is of order $n_{\mathcal{P}}^{1/2}$, we obtain the result through similar union bound arguments. In the end, for $b_{\mathcal{P}} = \sum_{k\in\mathcal{P}} n_k \mathcal{R}^*\left(\nabla^2 \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)\boldsymbol{\delta}_k^*\right)/n_{\mathcal{P}}$, we proceed by controlling each term $\|\mathcal{X}_k^\top \mathcal{X}_k \boldsymbol{\delta}_k^*\|_\infty / n_{\mathcal{P}}$. Recall that for the $m \times n$ matrix $\boldsymbol{A} = (a_{ij})$ and its transpose $\boldsymbol{A}^\top = \left((\boldsymbol{A}^\top)_1, \cdots, (\boldsymbol{A}^\top)_m\right)$, we have

$$\|\boldsymbol{A}\|_{1\to\infty} = \sup_{\|v\|_1 \le 1} \|\boldsymbol{A}v\|_\infty = \max_{(i,j)\in\{1,\ldots,m\}\times\{1,\ldots,n\}} |a_{ij}|,$$

$$\|\boldsymbol{A}\|_{2\to\infty} = \sup_{\|\boldsymbol{v}\|_2 \le 1} \|\boldsymbol{A}\boldsymbol{v}\|_\infty = \max_{i=1,\ldots,m} \left\|(\boldsymbol{A}^\top)_i\right\|_2,$$

so $\|\mathcal{X}_k^\top \mathcal{X}_k \mathcal{X}_k \boldsymbol{\delta}_k^*\|_\infty \le \|\mathcal{X}_k^\top \mathcal{X}_k\|_{1\to\infty} \|\boldsymbol{\delta}_k^*\|_1$ or $\|\mathcal{X}_k^\top \mathcal{X}_k \boldsymbol{\delta}_k^*\|_\infty \le \|\mathcal{X}_k^\top \mathcal{X}_k\|_{2\to\infty} \|\boldsymbol{\delta}_k^*\|_2$. Note that, by the Cauchy-Schwarz inequality, the maximum of $|(\mathcal{X}_k^\top \mathcal{X}_k)_{ij}|$ is obtained on the diagonal, where $(\mathcal{X}_k^\top \mathcal{X}_k)_{ii} = n_k(\boldsymbol{\Sigma}_k)_{i,i} + O_p(n_k^{1/2})$ occurs, so that $\|\mathcal{X}_k^\top \mathcal{X}_k \boldsymbol{\delta}_k^*\|_\infty \lesssim n_k h$ has a probability tending to 1 using the union bound again, as $\max_{k\in\mathcal{P}} \log p / n_k \to 0$ and $\|\boldsymbol{\delta}_k^*\|_1 \le h$. On the other hand, $\|\mathcal{X}_k^\top \mathcal{X}_k\|_{2\to\infty} \le \lambda_{\max}(\mathcal{X}_k^\top \mathcal{X}_k) \le M_3 n_k$ almost surely if $p/n_k \to c_k$ as $n_k, p \to \infty$ according to Yin et al. (1988); Bai and Silverstein

(1998, 2010) for $\|\boldsymbol{\delta}_k^*\|_2 \leq h$. The proof is complete.

S.3.4 Proof of Corollary 2

We first verify the RSC conditions under the settings of Corollary 2. Since $\|\text{vec}(\boldsymbol{X}_{k,i})\|_{\Psi_2} \leq M_1$ and $\lambda_{\min}[\boldsymbol{H}_k(\boldsymbol{\theta}_k^*)] \geq \kappa_k$ for $k \in \mathcal{P}$, according to (6.11) of Fan et al. (2019), with probability $1 - \exp(-c_k d)$,

$$\left\langle \text{vec}(\boldsymbol{\Delta}), \widehat{\boldsymbol{H}}_k(\boldsymbol{\theta}_k^*)\text{vec}(\boldsymbol{\Delta}) \right\rangle \geq \kappa_{k,1}\|\boldsymbol{\Delta}\|_F^2 - \kappa_{k,2}\sqrt{\frac{d}{n_k}}\|\boldsymbol{\Delta}\|_N^2, \quad \text{for all } \boldsymbol{\Delta} \in \mathbb{R}^{d\times d}. \tag{S.3.10}$$

As for the rate of convergence, given $\|\text{vec}(\boldsymbol{X}_{k,i})\|_{\Psi_2} \leq M_1$ and $|b''(\eta_{k,i})| \leq M_2$ almost surely, according to Lemma 1 of Fan et al. (2019), for $d \lesssim n_k$, as $d \to \infty$,

$$\left\| \frac{1}{n_k} \sum_{i \leq n_k} \left[b'\left(\langle \boldsymbol{\theta}_k^*, \boldsymbol{X}_{k,i} \rangle \right) - y_{k,i} \right] \boldsymbol{X}_{k,i} \right\|_{op} = O_p(\sqrt{\frac{d}{n_k}}). \tag{S.3.11}$$

Then, we establish Lemma 3 and Lemma 4 based on (S.3.10) and (S.3.11).

Lemma 3 (RSC Conditions)**.** *Under the settings of Corollary 2, for $\boldsymbol{\delta}_k^* = \boldsymbol{\theta}_0^* - \boldsymbol{\theta}_k^*$, let $\boldsymbol{\Delta}_{\overline{\mathcal{M}}}$ (or $\boldsymbol{\Delta}_{\overline{\mathcal{M}}^\perp}$) be the projection of $\boldsymbol{\Delta}$ onto $\overline{\mathcal{M}}$ (or $\overline{\mathcal{M}}^\perp$). Denote $\widehat{\boldsymbol{H}}_{\mathcal{P}} = \nabla^2 \boldsymbol{L}_{\mathcal{P}} = \sum_{k\in\mathcal{P}} n_k \widehat{\boldsymbol{H}}_k / n_{\mathcal{P}}$. There exists a positive constant $\kappa_{\mathcal{P}}$. as $\min_{k\in\mathcal{P}} n_k \to \infty$, $d \to \infty$, and $h \to 0$ with $(d/n_{\mathcal{P}})^{1/2} + h \to 0$, we have*

$$\left\langle \text{vec}(\boldsymbol{\Delta}), \widehat{\boldsymbol{H}}_{\mathcal{P}}(\boldsymbol{\theta}_0^*)\text{vec}(\boldsymbol{\Delta}) \right\rangle \geq \kappa_{\mathcal{P}}\|\boldsymbol{\Delta}\|_F^2, \quad \textit{for all } \|\boldsymbol{\Delta}_{\overline{\mathcal{M}}^\perp}\|_N \leq 3\|\boldsymbol{\Delta}_{\overline{\mathcal{M}}}\|_N. \tag{S.3.12}$$

with a probability tending to 1.

Lemma 4 (Convergence Rates)**.** *Under the settings of Corollary 2, as* $\min_{k\in\mathcal{P}} n_k \to \infty$*,* $d \to \infty$*, and* $h \to 0$ *with* $d \lesssim \min_{k\in\mathcal{P}} n_k$*, we have*

$$\mathcal{R}^* \left(\nabla \boldsymbol{L}_{\mathcal{P}}(\boldsymbol{\theta}_0^*) \right) = O_p \left[\underbrace{\sqrt{\frac{d}{n_{\mathcal{P}}}}}_{v_{\mathcal{P}}} + \underbrace{h}_{b_{\mathcal{P}}} \right].$$

As the proofs in Negahban et al. (2012) clarify, we require the RSC condition only on the intersection of $\mathbb{C}$ with a local ball $\{\|\boldsymbol{\Delta}\| \leq R\}$, where $R \asymp (d/n_{\mathcal{P}})^{1/2} + h \to 0$ is the error radius according to Lemma 4. Given sufficiently small R, we have $\delta \boldsymbol{L}_{\mathcal{P}}(\boldsymbol{\Delta}; \boldsymbol{\theta}_0^*) \gtrsim \left\langle \mathrm{vec}(\boldsymbol{\Delta}), \widehat{\boldsymbol{H}}_{\mathcal{P}}(\boldsymbol{\theta}_0^*) \mathrm{vec}(\boldsymbol{\Delta}) \right\rangle$, so that the RSC conditions hold according to Lemma 3, and Theorem 2 follows naturally from Proposition 2. Finally, we give the proofs of Lemma 3 and Lemma 4.

Proof of Lemma 3

It is the straightforward consequence of (S.3.10). For (S.3.12), we first focus on each term, for $\eta_{k,i} = \langle \boldsymbol{\theta}_k^*, \boldsymbol{X}_{k,i} \rangle$,

$$\begin{aligned}
n_k \left\langle \mathrm{vec}(\boldsymbol{\Delta}), \widehat{\boldsymbol{H}}_k(\boldsymbol{\theta}_0^*) \mathrm{vec}(\boldsymbol{\Delta}) \right\rangle &= \sum_{i \leq n_k} b'' \left(\langle \boldsymbol{\theta}_0^*, \boldsymbol{X}_{k,i} \rangle \right) \left(\langle \boldsymbol{X}_{k,i}, \boldsymbol{\Delta} \rangle \right)^2 \\
&= \sum_{i \leq n_k} b'' \left(\langle \boldsymbol{\theta}_k^* + \boldsymbol{\delta}_k^*, \boldsymbol{X}_{k,i} \rangle \right) \left(\langle \boldsymbol{X}_{k,i}, \boldsymbol{\Delta} \rangle \right)^2 \\
&= \sum_{i \leq n_k} \left[b'' \left(\eta_{k,i} \right) \left(\langle \boldsymbol{X}_{k,i}, \boldsymbol{\Delta} \rangle \right)^2 + b'''(\eta_{k,i}) \langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^* \rangle \left(\langle \boldsymbol{X}_{k,i}, \boldsymbol{\Delta} \rangle \right)^2 \right. \\
&\quad \left. + r_{k,i}(\langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^* \rangle) \left(\langle \boldsymbol{X}_{k,i}, \boldsymbol{\Delta} \rangle \right)^2 \right].
\end{aligned} \tag{S.3.13}$$

First, notice that both $\|\boldsymbol{\delta}_k^*\|_N \leq h$ and $\|\mathrm{vec}(\boldsymbol{\delta}_k^*)\|_1 \leq h$ imply that $\|\boldsymbol{\delta}_k^*\|_F \leq h$.

We have $\| \langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^* \rangle \|_{\Psi_2} \leq M_1 h$ by the definition of sub-Gaussian random vectors, so that $\langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^* \rangle = o_p(1)$ as $h \to 0$. The third term in the last line of (S.3.13) hence vanishes, and it suffices to control the first two terms. We then control the first term directly by (S.3.10), for $\|\boldsymbol{\Delta}_{\overline{\mathcal{M}}^\perp}\|_N \leq 3\|\boldsymbol{\Delta}_{\overline{\mathcal{M}}}\|_N$, with a probability greater than $1 - \exp(-c_k d)$ we have

$$\begin{aligned}
\sum_{i \leq n_k} b''\left(\eta_{k,i}\right) \left(\langle \boldsymbol{X}_{k,i}, \boldsymbol{\Delta} \rangle\right)^2 &\geq n_k \kappa_{k,1} \|\boldsymbol{\Delta}\|_F^2 - \kappa_{k,2} \sqrt{n_k d} \|\boldsymbol{\Delta}\|_N^2 \\
&\geq \left[n_k \kappa_{k,1} - 32 r \kappa_{k,2} \sqrt{n_k d} \right] \|\boldsymbol{\Delta}\|_F^2 \\
&\geq \left[n_k \kappa_{k,1} - 32 r \kappa_{k,2} \sqrt{n_{\mathcal{P}} d} \right] \|\boldsymbol{\Delta}\|_F^2,
\end{aligned}$$

due to the fact that $\|\boldsymbol{\Delta}\|_N \leq 4\|\boldsymbol{\Delta}_{\overline{\mathcal{M}}}\|_N \leq 4\sqrt{2r}\|\boldsymbol{\Delta}\|_F$. Then, to control the second term of (S.3.13), we have $|b'''(\eta_{k,i})| \leq M_3$ almost surely by assumption, $|\langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^* \rangle| = O_p(h)$ as shown earlier, and $(\langle \boldsymbol{X}_{k,i}, \boldsymbol{\Delta} \rangle)^2 = O_p(\|\boldsymbol{\Delta}\|_F^2)$. Combining these results, as $n_{\mathcal{P}} \to \infty$, $d \to \infty$ with $d/n_{\mathcal{P}} \to 0$, we have, by the union bound of probability, that

$$\left\langle \mathrm{vec}(\boldsymbol{\Delta}), \widehat{\boldsymbol{H}}_{\mathcal{P}}(\boldsymbol{\theta}_0^*) \mathrm{vec}(\boldsymbol{\Delta}) \right\rangle \geq \kappa_{\mathcal{P}} \|\boldsymbol{\Delta}\|_F^2, \quad \text{for all } \|\boldsymbol{\Delta}_{\overline{\mathcal{M}}^\perp}\|_N \leq 3\|\boldsymbol{\Delta}_{\overline{\mathcal{M}}}\|_N,$$

with probability tending to 1, where $\kappa_{\mathcal{P}} = c_1 \sum_{k \in \mathcal{P}} n_k \kappa_{k,1} / n_{\mathcal{P}}$ for some constant c_1.

Proof of Lemma 4

We recall that $\mathcal{R}^*(\nabla \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)) = \| \sum_{i \leq n_k} \left[b'\left(\langle \boldsymbol{\theta}_k^*, \boldsymbol{X}_{k,i} \rangle\right) - y_{k,i} \right] \boldsymbol{X}_{k,i} / n_k \|_{op} = O_p(\sqrt{d/n_k})$ directly from (S.3.11). We focus on $\mathcal{R}^*\left(\nabla \boldsymbol{L}_{\mathcal{P}}(\boldsymbol{\theta}_0^*)\right)$ for h sufficiently small. Again, it

suffices to bound $v_{\mathcal{P}} = \mathcal{R}^*\left(\sum_{k\in\mathcal{P}} n_k \nabla \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)\right)/n_{\mathcal{P}}$ and $b_{\mathcal{P}} = \sum_{k\in\mathcal{P}} n_k \mathcal{R}^*\left(\nabla^2 \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)\boldsymbol{\delta}_k^*\right)/n_{\mathcal{P}}$. We could use the standard ε-net argument to control $v_{\mathcal{P}}$ as in Lemma 1 of Fan et al. (2019), which gives $v_{\mathcal{P}} = O_p(\sqrt{d/n_{\mathcal{P}}})$. As for $b_{\mathcal{P}}$, we control each term

$$\mathcal{R}^*\left(\nabla^2 \boldsymbol{L}_k(\boldsymbol{\theta}_k^*)\boldsymbol{\delta}_k^*\right) = \frac{1}{n_k}\left\|\sum_{i\le n_k} b''\left(\langle \boldsymbol{\theta}_k^*, \boldsymbol{X}_{k,i}\rangle\right)\langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^*\rangle \boldsymbol{X}_{k,i}\right\|_{op}.$$

By the definition of $\|\cdot\|_{op}$ and the standard ϵ-net arguments as in (S.3.11), we have

$$\begin{aligned}
&\left\|\sum_{i\le n_k} b''\left(\langle \boldsymbol{\theta}_k^*, \boldsymbol{X}_{k,i}\rangle\right)\langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^*\rangle \boldsymbol{X}_{k,i}\right\|_{op} \\
&= \sup_{\boldsymbol{u}\in\mathcal{S}^{d-1},\boldsymbol{v}\in\mathcal{S}^{d-1}}\left|\sum_{i\le n_k} b''\left(\langle \boldsymbol{\theta}_k^*, \boldsymbol{X}_{k,i}\rangle\right)\langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^*\rangle \boldsymbol{u}^\top \boldsymbol{X}_{k,i}\boldsymbol{v}\right| \\
&\le \frac{16}{7}\max_{\boldsymbol{u}\in\mathcal{N}^d,\boldsymbol{v}\in\mathcal{N}^d}\left|\sum_{i\le n_k} b''\left(\langle \boldsymbol{\theta}_k^*, \boldsymbol{X}_{k,i}\rangle\right)\langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^*\rangle \boldsymbol{u}^\top \boldsymbol{X}_{k,i}\boldsymbol{v}\right|,
\end{aligned}$$

where $\mathcal{S}^{d-1}$ is the $(d-1)$-dimensional sphere and $\mathcal{N}^d$ is a 1/4-net on $\mathcal{S}^{d-1}$. Then, notice that $|b''\left(\langle \boldsymbol{\theta}_k^*, \boldsymbol{X}_{k,i}\rangle\right)| \le M_2$ by definition, while $\|\langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^*\rangle\|_{\Psi_2} \le M_1 h$ and $\|\boldsymbol{u}^\top \boldsymbol{X}_{k,i}\boldsymbol{v}\|_{\Psi_2} \le M_1$ are due to the fact that $\|\mathrm{vec}(\boldsymbol{X}_{k,i})\|_{\Psi_2} \le M_1$. Since the product of two sub-Gaussian random variables is sub-exponential, we have for all $\boldsymbol{u}\in\mathcal{N}^d$ and $\boldsymbol{v}\in\mathcal{N}^d$,

$$\|\langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^*\rangle \boldsymbol{u}^\top \boldsymbol{X}_{k,i}\boldsymbol{v}\|_{\Psi_1} \le \|\langle \boldsymbol{X}_{k,i}, \boldsymbol{\delta}_k^*\rangle\|_{\Psi_2}\|\boldsymbol{u}^\top \boldsymbol{X}_{k,i}\boldsymbol{v}\|_{\Psi_2} \le M_1^2 h,$$

for the sub-exponential norm $\|\cdot\|_{\Psi_1}$. Hence, we are facing the sum of n_k independent sub-exponential random variables, with the sub-exponential norm (controlling both both mean and standard error of the random variables) bounded above by h. Then,

by the union bound over all points on $\mathcal{N}^d \times \mathcal{N}^d$ following (6.9) of Fan et al. (2019), we obtain $b_{\mathcal{P}} = O_p[h + h \max_{k \in \mathcal{P}} (d/n_k)^{1/2}]$. The proof is then complete by noticing that $d \lesssim \min_{k \in \mathcal{P}} n_k$, by assumption.

Bibliography

Bai, Z., K. P. Choi, and Y. Fujikoshi (2018). Consistency of aic and bic in estimating the number of significant components in high-dimensional principal component analysis. *The Annals of Statistics 46*(3), 1050–1076.

Bai, Z. and J. W. Silverstein (2010). *Spectral analysis of large dimensional random matrices*, Volume 20. Springer.

Bai, Z.-D. and J. W. Silverstein (1998). No eigenvalues outside the support of the limiting spectral distribution of large-dimensional sample covariance matrices. *The Annals of Probability 26*(1), 316–345.

Baik, J. and J. W. Silverstein (2006). Eigenvalues of large sample covariance matrices of spiked population models. *Journal of multivariate analysis 97*(6), 1382–1408.

Bastani, H. (2021). Predicting with proxies: Transfer learning in high dimension. *Management Science 67*(5), 2964–2984.

Chen, S., Q. Zheng, Q. Long, and W. J. Su (2021). A theorem of the alternative for personalized federated learning. *arXiv preprint arXiv:2103.01901*.

Duan, Y. and K. Wang (2023). Adaptive and robust multi-task learning. *The Annals of Statistics 51*(5), 2015–2039.

Fan, J., W. Gong, and Z. Zhu (2019). Generalized high-dimensional trace regression via nuclear norm regularization. *Journal of Econometrics 212*(1), 177–202.

Fan, J., W. Wang, and Z. Zhu (2021). A shrinkage principle for heavy-tailed data: High-dimensional robust low-rank matrix recovery. *The Annals of Statistics 49*(3), 1239 – 1266.

Friedman, J., T. Hastie, and R. Tibshirani (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software 33*(1), 1–22.

He, Z., Y. Sun, and R. Li (2024). Transfusion: Covariate-shift robust transfer learning for high-dimensional regression. In *International Conference on Artificial Intelligence and Statistics*, pp. 703–711. PMLR.

Li, S., T. T. Cai, and H. Li (2022). Transfer learning for high-dimensional linear regression: Prediction, estimation and minimax optimality. *Journal of the Royal Statistical Society Series B: Statistical Methodology 84*(1), 149–173.

Li, S., L. Zhang, T. T. Cai, and H. Li (2024). Estimation and inference for high-dimensional generalized linear models with knowledge transfer. *Journal of the American Statistical Association 119*(546), 1274–1285.

Liu, D., C. Zhao, Y. He, L. Liu, Y. Guo, and X. Zhang (2023). Simultaneous cluster structure learning and estimation of heterogeneous graphs for matrix-variate fmri data. *Biometrics 79*(3), 2246–2259.

Negahban, S. N., P. Ravikumar, M. J. Wainwright, and B. Yu (2012). A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Statistical Science 27*(4), 538–557.

Parikh, N., S. Boyd, et al. (2014). Proximal algorithms. *Foundations and trends® in Optimization 1*(3), 127–239.

Paul, D. (2007). Asymptotics of sample eigenstructure for a large dimensional spiked covariance model. *Statistica Sinica 17*(4), 1617–1642.

Raskutti, G., M. J. Wainwright, and B. Yu (2010). Restricted eigenvalue properties for correlated gaussian designs. *The Journal of Machine Learning Research 11*, 2241–2259.

Tian, Y. and Y. Feng (2023). Transfer learning under high-dimensional generalized linear models. *Journal of the American Statistical Association 118*(544), 2684–2697.

Wu, C., S. Kwon, X. Shen, and W. Pan (2016). A new algorithm and theory for penalized regression-based clustering. *Journal of Machine Learning Research 17*(188), 1–25.

Yin, Y.-Q., Z.-D. Bai, and P. R. Krishnaiah (1988). On the limit of the largest eigenvalue of the large dimensional sample covariance matrix. *Probability theory and related fields 78*, 509–521.