Journal of Machine Learning Research 27 (2026) 1-54 Submitted 2/25; Revised 3/26; Published 4/26

High-dimensional Parameter Transfer With Fused-Regularizer

Zelin He ZBH5185@PSU.EDU
Department of Statistics, Pennsylvania State University, University Park, PA 16802, USA

Ying Sun YBS5190@PSU.EDU
School of EECS, Pennsylvania State University, University Park, PA 16802, USA

Jingyuan Liu* JINGYUAN@XMU.EDU.CN
Department of Statistics and Data Science, School of Economics, Xiamen University
Xiamen, Fujian 361102, China

Runze Li RZLI@PSU.EDU
Department of Statistics, Pennsylvania State University, University Park, PA 16802, USA

Editor: Kun Zhang

Abstract

Parameter transfer aims to improve parameter estimation accuracy by leveraging knowledge from related sources. This paper studies the parameter transfer problem from heterogeneous sources for high-dimensional M-estimators. Specifically, we propose a novel one-step estimator with a fused-regularizer and a target-data-oriented constraint, which can robustly capture parameter knowledge from source data in the presence of different types of data distribution shifts. Nonasymptotic bound is provided for the estimation error of target parameter, showing the proposed estimator could achieve effective parameter transfer under distribution shifts, and is guaranteed to perform no worse than any estimators learned only from the target data. We further show that the proposed estimator can achieve the minimax-optimal rate under much weaker conditions than existing methods. In addition, we extend the method to a distributed setting, requiring just one round of communication with source parameter estimators, while retaining the estimation accuracy of the centralized version. Extensive simulations and real data analysis further verify the effectiveness of the method.

Keywords: transfer learning, high-dimensional estimation, regularized M-estimator, nonasymptotic theory, sparsity.

1. Introduction

High-dimensional parameter estimation methods have gained increasing attention due to their wide application in areas including bioinformatics, astronomy, and information technology (Bühlmann and Van De Geer, 2011). With the dimension of unknown parameters being much larger than the sample size, the parameter estimation becomes much more challenging due to issues of error accumulation (Fan et al., 2020). Nonetheless, existing methods primarily rely on independent and identically distributed (i.i.d.) data. In many scenarios, obtaining such high-dimensional i.i.d. data is costly or infeasible, posing challenges to accurate parameter estimation and reliable decision-making.

*. corresponding author

Transfer learning is a promising framework that can improve the parameter estimation accuracy for the above low-data setting by incorporating side information from auxiliary source samples (Torrey and Shavlik, 2010). Specifically, this paper considers a parameter transfer setting, where we are given a target sample drawn from a target distribution $\mathbb{P}^{(0)}$ parameterized by a high-dimensional target parameter $\boldsymbol{\beta}_*^{(0)}$, along with some auxiliary samples drawn from K different source distributions $\{\mathbb{P}^{(k)}\}_{k=1}^K$ parameterized by $\{\boldsymbol{\beta}_*^{(k)}\}_{k=1}^K$. The goal is to improve the estimation accuracy of the target parameter by incorporating *parameter knowledge* from other source samples (Pan and Yang, 2009). We consider a challenging scenario where the similarity between the source and target is completely captured through the closeness of $\boldsymbol{\beta}_*^{(0)}$ and $\boldsymbol{\beta}_*^{(k)}$, with minimal additional structural similarity assumptions among the data distributions. Such a flexible setup encapsulates various types of distribution shifts of practical interests. For example, in a regression setting, it includes shifts in the marginal covariate distributions (He et al., 2024). In high-dimensional scenarios, such a covariate shift can be more severe due to a more complex covariate correlation structure (Nagel et al., 2018; Charney et al., 2017). Besides covariate shifts, the distribution shifts may also come from the shift in the conditional distribution of the responses. For example, in genomic studies, some source data may suffer from overdispersion due to a higher-than-expected variance (Wang et al., 2019). In the presence of distribution shifts, naively incorporating source samples can lead to estimation accuracy that is worse than using the target sample alone–a phenomenon known as "negative transfer" (Zhang et al., 2023). The challenge calls for the design of a high-dimensional parameter transfer procedure that can effectively exploit parameter similarity under these distribution shifts and can always prevent negative transfer.

Beyond challenges brought by distribution shifts, another challenge that often arises in transfer learning is the distributed nature of source samples. For example, for electronic health record data analysis in multicenter research studies, source samples are collected from different institutes or organizations, where direct data sharing is often impractical due to privacy and regulation constraints or the prohibitive communication cost (Kushida et al., 2012). This necessitates the development of privacy-preserving and communication-efficient solutions that can achieve comparable estimation accuracy as their centralized counterparts.

1.1 Contribution of this paper

To address the above challenges, we propose a parameter transfer framework for high-dimensional M-estimators, named *TransMission*, that introduces a series of fused-regularizers into a joint learning process combined with a novel target-data-oriented constraint. Joint learning with fused regularization captures transferable parameter knowledge across sources while maintaining robustness to a wide range of distribution shifts. The target-data-oriented constraint further narrows the search space for the target parameter using the target data, so the estimator will always perform no worse than the single-task estimator learned only on the target data, preventing the issue of negative transfer.

We further support the framework with a fine-grained, non-asymptotic theoretical analysis. We consider a p-dimensional M-estimation problem for an s-sparse target parameter $\boldsymbol{\beta}_*^{(0)}$. With a target sample of size n_T, and K source samples with each of size n_S, we show that *TransMission* yields a convergence rate of $O(\frac{s\log p}{n_T+Kn_S} + (h\sqrt{\frac{\log p}{n_T}}) \wedge (C_\Sigma^2 h^2) \wedge \frac{s\log p}{n_T})$,

where h measures difference between the source parameter $\boldsymbol{\beta}_*^{(k)}$ and the target parameter $\boldsymbol{\beta}_*^{(0)}$, and C_Σ measures the degree of distribution shifts. The result carries several merits. First, it shows *TransMission* is always no worse than the rate of the single-task estimator, i.e., $O(\frac{s\log p}{n_T})$, therefore preventing negative transfer thanks to the introduction of the target-data-oriented constraint. When the source parameters are sufficiently close to the target (h is small), *TransMission* is capable of taking full advantage of the source samples and attaining a much faster rate of $O(\frac{s\log p}{n_T+Kn_S})$. In addition, it reveals *TransMission* is robust to severe distribution shifts as reflected by a large value of C_Σ. Overall, the estimator achieves a sharper rate than many state-of-the-art methods under various scenarios, and the rate matches the existing minimax lower bound under a much weaker condition on h. We refer the readers to Section 3.3 for a more detailed comparison.

To address practical concerns in distributed settings, we also develop its variant termed *D-TransMission* that requires only one-shot transmission of the pre-estimated parameters from source tasks. We further show that *D-TransMission* achieves a statistical rate of the same order as *TransMission*, provided the source sample size is sufficiently large.

1.2 Related Work

Much progress has been made in parameter transfer in recent years. However, many only focus on capturing the shared parameter information, but either overlook other distribution shifts entirely or focus on mitigating specific shifts. For example, a line of approach is based on a two-step approach, which first pools source and target data for knowledge transfer and then applies a bias correction using only the target data (Bastani (2021); Li et al. (2022); Tian and Feng (2023)). However, such an approach does not explicitly address other distribution shifts, and the learning accuracy degrades quickly if the shifts are severe. Recently there have been a few pioneering works proposed to correct covariate shifts with imputation models (Liu et al., 2023; Zhou et al., 2025; Cai et al., 2025). However, the effectiveness of these methods depends on the availability of abundant unlabeled target data, and they do not account for other types of distribution shifts beyond covariate shifts.

Achieving a robust transfer of learnable parameter information has been investigated over the years in the multi-task setting, where regularization techniques are often employed to promote the exchange of parameter knowledge across different tasks (Pan and Yang, 2009), examples including ℓ_2-norm (Duan and Wang, 2023; Evgeniou et al., 2005), ℓ_1-norm (Li and Sang, 2019; Zhang et al., 2024), Euclidean-norm (Chen et al., 2023) and spectral-norm (Tian et al., 2025), to name a few. Most of these works borrow the idea of fused-lasso (Tibshirani et al., 2005) to promote parameter information sharing. These multi-task learning methods typically require all the tasks to have a comparable sample size and emphasize overall task performance. Therefore, they are not directly applicable to transfer learning problems where the target task, sometimes with far fewer samples, is the primary focus. Gao and Yang (2023), Liu (2024) and Long et al. (2013) utilize a similar fused-regularization structure to promote parameter transfer, yet they either only apply in the low-dimensional setting or require the source and target to have comparable sizes. Li et al. (2024) and Bai et al. (2024) propose robust methods for high-dimensional parameter transfer. However, the proposed methods either lack guarantees for preventing negative transfer or incur non-negligible costs to achieve such robustness. See Section 3.3 for a detailed comparison.

A closely related work is TransFusion (He et al., 2024). Compared with TransFusion, this paper makes several substantial new contributions. First, TransFusion is limited to linear regression and only guarantees robustness of parameter transfer under covariate shift, whereas this paper develops a general framework for parameter transfer applicable to M-estimators and accommodates a broader class of distribution shifts. Second, this paper proposes an entirely new one-step estimator with a novel target-data-oriented constraint for general M-estimators, which guarantees the estimator always performs no worse than estimators learned only from the target data; by contrast, TransFusion relies on a two-step approach and can fail when the source tasks are not transferable. Third, this paper develops several new theoretical results for the proposed estimator, improves the convergence rate of the estimated target parameter, and establishes minimax optimality under weaker conditions than those required in TransFusion.

1.3 Organization and Notation

The remainder of the paper is organized as follows. Section 2 introduces the setup of parameter transfer for high-dimensional M-estimator. In Section 3, we propose the *TransMission* framework, provide a theoretical characterization of its estimation error, and demonstrate its robustness by comparing against existing methods. Section 4 extends *TransMission* to the distributed setting. Section 5 validates the theoretical results through simulations and real-world data analysis. Section 6 concludes the paper. Proofs and additional details are provided in the appendix.

Throughout the paper, we use bold uppercase letters for matrices and bold lowercase letters for vectors. For a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, we denote the (i,j)-th element of $\boldsymbol{A}$ by $\boldsymbol{A}_{ij}$. For a p-dimensional vector $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)^\top$, we denote its ℓ_q norm as $\|\mathbf{x}\|_q = (\sum_{i=1}^p |\mathbf{x}_i|^q)^{1/q}$ for all $q > 0$, ℓ_0-norm as $\|\mathbf{x}\|_0 = \#\{j : \mathbf{x}_j \neq 0\}$ and ℓ_∞-norm as $\|\mathbf{x}\|_\infty = \max_{1\le j\le p} \mathbf{x}_j$. We use $\langle\cdot,\cdot\rangle$ to denote the standard Euclidean inner product. Let $a \vee b$ denote $\max\{a,b\}$ and $a \wedge b$ denote $\min\{a,b\}$. The notation $[K]$ represents the set $\{1,\ldots,K\}$. We use $c, c_0, c_1, \ldots$ to denote generic constants. Let $a_n = O(b_n)$ and $a_n \lesssim b_n$ denote $|a_n/b_n| \le c$ for some constant c when n is large enough; $a_n \asymp b_n$ if $a_n = O(b_n)$ and $b_n = O(a_n)$; $a_n = o(b_n)$ or $b_n \gg a_n$ if $a_n = O(c_n b_n)$ for some $c_n \to 0$. See Table A.1 for the full notation table.

2. Preliminaries

We consider a parameter transfer problem for M-estimation involving one target task and K source tasks. For the target task, we observe a target sample $\mathcal{S}^{(0)} = \{z_i^{(0)}\}_{i\in[n_T]}$, where each data point $z_i^{(0)} \in \mathcal{Z}$ is i.i.d. drawn from a target distribution $\mathbb{P}^{(0)}$. Define $\ell^{(0)} : \mathcal{Z}\times\mathbb{R}^p \to \mathbb{R}$ as the target loss function with $\ell^{(0)}(z^{(0)};\boldsymbol{\beta})$ measures the cost to any parameter $\boldsymbol{\beta} \in \mathbb{R}^p$ given the target data point $z^{(0)}$. Our goal is to estimate the unknown target parameter $\boldsymbol{\beta}_*^{(0)}$ defined as

$$\boldsymbol{\beta}_*^{(0)} \in \operatorname*{argmin}_{\boldsymbol{\beta}\in\mathbb{R}^p} \mathbb{E}_{z^{(0)}\sim\mathbb{P}^{(0)}}\left[\ell^{(0)}(z^{(0)};\boldsymbol{\beta})\right]. \tag{1}$$

We focus on a high-dimensional setting where the dimension p is larger than the target sample size n_T, yet the ground truth $\boldsymbol{\beta}_*^{(0)}$ is a sparse vector with $s := \|\boldsymbol{\beta}_*^{(0)}\|_0$ nonzero elements much smaller than p, i.e. $s \ll p$.

In addition to the target sample, we have access to K source samples $\{\mathcal{S}^{(k)}\}_{k\in[K]}$, where each source sample $\mathcal{S}^{(k)}$ consists of i.i.d. data points generated from the corresponding source distribution $\mathbb{P}^{(k)}$, which may differ among $k \in [K]$. Let $\ell^{(k)}$ be the loss function for each kth source task, we similarly define the source parameter as

$$\boldsymbol{\beta}_*^{(k)} \in \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}}\ \mathbb{E}_{z^{(k)}\sim\mathbb{P}^{(k)}}\left[\ell^{(k)}(z^{(k)};\boldsymbol{\beta})\right],\ k\in[K]. \tag{2}$$

For simplicity, we assume all source samples have the same size n_S, which is larger than the target sample size n_T, i.e., $0 < n_T < n_S$.

Our goal is to estimate $\boldsymbol{\beta}_*^{(0)}$ using both the target and source samples in a parameter transfer setting, where the source parameter $\boldsymbol{\beta}_*^{(k)}$ and the target parameter $\boldsymbol{\beta}_*^{(0)}$ share the same feature space and stay in the following parameter space:

$$\Theta(s,h) := \left\{\boldsymbol{\beta}_*^{(0)}\in\mathbb{R}^p, \boldsymbol{\beta}_*^{(k)}\in\mathbb{R}^p, k\in[K] : \|\boldsymbol{\beta}_*^{(0)}\|_0 \le s, \max_{k\in[K]}\|\boldsymbol{\beta}_*^{(k)}-\boldsymbol{\beta}_*^{(0)}\|_1 \le h\right\}. \tag{3}$$

In (3), the sparsity level of the target parameter $\boldsymbol{\beta}_*^{(0)}$ is upper bounded by s, and the informative level of each k-th source tasks is quantified by the ℓ_1-norm of $\boldsymbol{\delta}^{(k)} := \boldsymbol{\beta}_*^{(k)} - \boldsymbol{\beta}_*^{(0)}$, and is upper bounded by $h \ge 0$. We refer to $\boldsymbol{\delta}^{(k)}$ as "parameter contrast" or "source-specific signal". We choose an ℓ_1-sparse constraint for the high-dimensional contrast $\boldsymbol{\delta}^{(k)}$, as it aligns well with practical applications where parameter shifts typically spread over multiple dimensions but their overall magnitude does not grow too fast (Li et al., 2022; Fan et al., 2023). The results in the paper can be naturally extended to a general ℓ_q-sparse case for $q \in [0,1]$. The sources are considered informative for parameter transfer if h is relatively small. As in practice h is unknown in practice, one key challenge in parameter transfer is to capture the transferable information when h is small, whereas still being robust when h is large.

Another major challenge in parameter transfer is that though the sources may be transferable with a small h, other distribution characteristics may shift significantly in source populations compared with the target population. To be concrete, we provide some examples of such shifts in a supervised learning setting. Given data $(\mathbf{X}_i^{(k)}, \mathbf{y}_i^{(k)}) \in \mathcal{Z}$, $i = 0, 1, \ldots, n_k$, $k = 0, 1, \ldots, K$, where $k = 0$ refers to the *target data*, there are three typical *distribution shifts*: (1) *Covariate shift*: potential differences in marginal distributions of $\mathbf{X}_i^{(k)}$ among the $K+1$ populations; (2) *Conditional shift*: potential differences in conditional distributions of $(\mathbf{y}_i^{(k)}|\mathbf{X}_i^{(k)})$'s, typically exhibited by different parametric model forms between $\mathbf{y}_i^{(k)}$ and $\mathbf{X}_i^{(k)}$, or different nuisance parameters in the models; (3) *Label shift*: potential differences in marginal distributions of $\mathbf{y}_i^{(k)}$'s. In this paper, we refer to these differences collectively as *distribution shifts*. We want to remark that the conditional shift in our context does not include the differences in parameters of interest - they are the "parameter contrast" that we are modeling. For more intuitive illustration, refer to the following two special examples.

Example 1 (Normal linear model.) *Consider the simplest high-dimensional normal linear model:* $\mathbf{y}_i^{(k)} = \mathbf{X}_i^\top \boldsymbol{\beta}_*^{(k)} + \boldsymbol{\epsilon}_i^{(k)}$, *where* $\mathbf{X}_i^{(k)} \sim \mathrm{N}(\mathbf{0}, \boldsymbol{\Sigma}^{(k)})$ *and* $\boldsymbol{\epsilon}_i^{(k)} \sim \mathrm{N}(0, \sigma_k^2)$.

(1) **Covariate shift:** *heterogeneity in the covariance matrices across sources and the target, i.e.,* $\boldsymbol{\Sigma}^{(k)} \neq \boldsymbol{\Sigma}^{(l)}$ *for* $l \neq k \in \{0, \ldots, K\}$.

(2) **Conditional shift:** *heterogeneity in the error variances, i.e.,* $\sigma_k^2 \neq \sigma_l^2$.

Example 2 (Generalized linear model) *Consider a high-dimensional generalized linear model defined for each task* $k \in \{0, 1, \ldots, K\}$ *by*

$$\mathbf{y}_i^{(k)} \mid \mathbf{X}_i^{(k)} \sim \mathbb{P}_{Y|X}^{(k)}(\mathbf{y}_i \mid \mathbf{X}_i; \boldsymbol{\beta}_*^{(k)}) = \rho_k(\mathbf{y}_i) \exp\left(\frac{\mathbf{y}_i(\mathbf{X}_i^\top \boldsymbol{\beta}_*^{(k)}) - \Psi_k(\mathbf{X}_i^\top \boldsymbol{\beta}_*^{(k)})}{\phi_k} \right), \quad \mathbf{X}_i^{(k)} \sim \mathbb{P}_X^{(k)},$$

where $\Psi_k(\cdot)$ *and* ϕ_k *are univariate functions, with* $\Psi_k'(\cdot)$ *being the inverse link function, and* ϕ_k *is the dispersion parameter. This family includes linear, logistic, Poisson, and multinomial regression models as special cases.*

(1) ***Covariate shift:*** *heterogeneity in the marginal covariate distributions* $\mathbb{P}_X^{(k)} \neq \mathbb{P}_X^{(l)}$ *across tasks.*

(2) ***Conditional shift:*** *heterogeneity in the conditional response models* $\mathbb{P}_{Y|X}^{(k)}$*, arising from differences in* $\Psi_k(\cdot)$*,* $\rho_k(\cdot)$*, or the dispersion parameter* ϕ_k*.*

Note that we only explicitly define covariate and conditional shifts in these examples, because due to the parametric model setup and the law of total probability, the **label shift** is automatically driven by covariate shift, conditional shift, or their combination.

3. Robust One-step Parameter Transfer

3.1 Transfer with Fused-regularizers and Target-data-oriented Constraint

The challenge of parameter transfer lies in tackling the distribution shifts while extracting the commonalities in parameters between source and target samples to estimate the sparse target parameter $\boldsymbol{\beta}_*^{(0)}$. To motivate our method, we first consider a baseline, which estimates $\boldsymbol{\beta}_*^{(0)}$ by pooling together all data and treating them as i.i.d. samples. Specifically, denote the empirical loss on the target and source tasks, respectively as $\mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)}) = (2n_T)^{-1} \sum_{i=1}^{n_T} \ell^{(0)}(z_i^{(0)}; \boldsymbol{\beta}^{(0)})$ and $\mathcal{L}^{(k)}(\boldsymbol{\beta}^{(k)}) = (2n_S)^{-1} \sum_{i=1}^{n_S} \ell^{(k)}(z_i^{(k)}; \boldsymbol{\beta}^{(k)})$. The pooling estimator is defined as

$$\hat{\boldsymbol{\beta}}_{\text{Pooling}} \in \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \sum_{k=0}^{K} \frac{n_k}{N} \mathcal{L}^{(k)}(\boldsymbol{\beta}) + \lambda \|\boldsymbol{\beta}\|_1 \right\}, \tag{4}$$

where $n_k = n_S$ for $k \in [K]$, $n_k = n_T$ for $k = 0$, $N = n_0 + \sum_{k=1}^K n_k$ is the total sample size and λ is a tuning parameter. Here we incorporate an additional ℓ_1-regularizer to promote a sparse solution. Such an estimator benefits from using a larger sample size to identify the model parameter. However, due to the existence of source-specific signal $\boldsymbol{\delta}^{(k)}$, such an estimator suffers from a non-negligible bias, and the distribution shifts across sources could further amplify such a bias. To avoid introducing such a bias, one may alternatively estimate the target parameter using solely the target data via solving:

$$\hat{\boldsymbol{\beta}}_T^{(0)} \in \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \mathcal{L}^{(0)}(\boldsymbol{\beta}) + \lambda \|\boldsymbol{\beta}\|_1 \right\}. \tag{5}$$

However, with such a single-task estimator, the source dataset will under no circumstance be helpful, even when the distributions are identical. It is therefore critical to find a method that can capture the transferable parameter knowledge, whereas being robust to the distribution shifts to improve the estimation of $\boldsymbol{\beta}_*^{(0)}$. To achieve such a goal, we introduce a new transfer learning framework for M-estimation with fused regularization, abbreviated as *TransMission*, which estimates $\boldsymbol{\beta}_*^{(0)}$ by solving the following problem:

$$\min_{\boldsymbol{\beta}^{(0)}\in\mathcal{C}_T,\boldsymbol{\beta}^{(k)}\in\mathbb{R}^p,k\in[K]} \left\{ \sum_{k=0}^{K} \frac{n_k}{N} \mathcal{L}^{(k)}(\boldsymbol{\beta}^{(k)}) + \lambda_0 \left\|\boldsymbol{\beta}^{(0)}\right\|_1 + \lambda_1 \sum_{k=1}^{K} \left\|\boldsymbol{\beta}^{(k)} - \boldsymbol{\beta}^{(0)}\right\|_1 \right\}, \tag{6a}$$

$$\text{with } \mathcal{C}_T = \{\boldsymbol{\beta}^{(0)} \in \mathbb{R}^p : \|\nabla \mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)})\|_\infty \le \lambda_T, \|\boldsymbol{\beta}^{(0)}\|_1 \le \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1 + \kappa_T\}, \tag{6b}$$

where $N = Kn_S + n_T$ is the total sample size, λ_0 and λ_1 are the tuning parameters, and $\mathcal{C}_T$ is a target-data oriented constraint that will be discussed shortly. In (6a), the first term measures the average fitness of the models $\{\boldsymbol{\beta}^{(k)}\}_{k=0}^K$, while the regularization terms simultaneously promote the sparsity of $\boldsymbol{\beta}^{(0)}$ and capture the shared knowledge between $\boldsymbol{\beta}^{(0)}$ and $\boldsymbol{\beta}^{(k)}$ penalizing their ℓ_1 difference. The non-transferable source-specific signal $\boldsymbol{\delta}^{(k)} = \boldsymbol{\beta}_*^{(k)} - \boldsymbol{\beta}_*^{(0)}$ is explicitly estimated and filtered out. Notice that unlike the pooling estimator $\hat{\boldsymbol{\beta}}_{\text{Pooling}}$ which transfers knowledge from the loss level, the *TransMission* estimator achieves the parameter transfer from the parameter level: transferable knowledge is first encoded into the parameter $\boldsymbol{\beta}^{(k)}$ and then passed to the estimation of target parameter $\boldsymbol{\beta}^{(0)}$. This allows *TransMission* to be robust to a variety of distribution shifts when transferring the parameter knowledge from heterogeneous sources.

In (6b), we introduce an additional constraint $\mathcal{C}_T$ on $\boldsymbol{\beta}^{(0)}$, where λ_T and κ_T are tuning parameters and $\hat{\boldsymbol{\beta}}_T^{(0)}$ is the single-task estimator defined in (5). The first term $\|\nabla \mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)})\|_\infty \le \lambda_T$ can be understood as a "confidence set" that summarizes the information on $\boldsymbol{\beta}^{(0)}$ provided by the target data (Fan, 2013). The second term, $\|\boldsymbol{\beta}^{(0)}\|_1 \le \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1 + \kappa_T$, constrains the size of the *TransMission* estimator relative to the single-task estimator. This constraint promotes sparsity within a feasible search space informed by the target data. Ideally, the sparsity level should be controlled by the true target parameter, i.e., $\|\boldsymbol{\beta}^{(0)}\|_1 \le \|\boldsymbol{\beta}_*^{(0)}\|_1$ (Fan, 2013; Hastie et al., 2015). However, since $\|\boldsymbol{\beta}_*^{(0)}\|_1$ is unknown in practice, so we propose to approximate it with the consistent estimator $\|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1$, and incorporate an additional slack term κ_T to account for the noise in the single-task estimator. Overall, the constraint $\mathcal{C}_T$ narrows down the search space for $\boldsymbol{\beta}^{(0)}$ using the target samples, and the objective function further selects within these candidates a good one assisted by the side information learned from the source samples. Thus, the resulting estimator is ensured to remain within the local neighborhood of the true target parameter, eliminating the need for any further debiasing steps while performing at least as well as the estimator based solely on the target sample.

In practice, implementing TransMission requires solving a single fused regularization problem together with a target-oriented constraint. At a high level, the joint optimization in (6a) can be reformulated as a standard ℓ_1-penalized regression through an appropriate reparameterization, allowing the use of off-the-shelf solvers. We then employ an adaptive tuning parameter selection strategy to ensure the solution satisfies the constraint (6b). Guided by the theoretical analysis, we further reduce the four tuning parameters λ_0, λ_1,

Algorithm 1 *TransMission*

Input: target sample $\mathcal{S}^{(0)}$, source samples $\{\mathcal{S}^{(k)}\}_{k=1}^{K}$, tuning parameters λ_0, λ_1, λ_T, κ_T.

Output: the estimated target parameter $\hat{\boldsymbol{\beta}}^{(0)}$.

1. Compute $\hat{\boldsymbol{\beta}}_T^{(0)}$ by solving $\min_{\boldsymbol{\beta}\in\mathbb{R}^p}\{\mathcal{L}^{(0)}(\boldsymbol{\beta}) + \lambda_T\|\boldsymbol{\beta}\|_1\}$.
2. Compute $(\hat{\boldsymbol{\beta}}^{(0)}, \hat{\boldsymbol{\beta}}^{(1)}, \ldots, \hat{\boldsymbol{\beta}}^{(K)})$ by solving
$$\min_{\boldsymbol{\beta}^{(0)}\in\mathbb{R}^p,\ \{\boldsymbol{\beta}^{(k)}\}_{k=1}^{K}\in(\mathbb{R}^p)^K} \sum_{k=0}^{K}\frac{n_k}{N}\mathcal{L}^{(k)}(\boldsymbol{\beta}^{(k)}) + \lambda_0\|\boldsymbol{\beta}^{(0)}\|_1 + \lambda_1\sum_{k=1}^{K}\|\boldsymbol{\beta}^{(k)} - \boldsymbol{\beta}^{(0)}\|_1,$$
with $\mathcal{C}_T = \{\boldsymbol{\beta}^{(0)}\in\mathbb{R}^p : \|\nabla\mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)})\|_\infty \le \lambda_T,\ \|\boldsymbol{\beta}^{(0)}\|_1 \le \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1 + \kappa_T\}$.
3. Output $\hat{\boldsymbol{\beta}}^{(0)}$.

λ_T, and κ_T to two tuning parameters, both selected using cross-validation and a small grid search. A complete implementation guide is provided in Appendix A.2.

3.2 Theoretical Properties

We now demonstrate the robustness of the proposed *TransMission* estimator under distribution shifts by establishing and discussing its statistical convergence rate. To facilitate the analysis, we introduce some regularity conditions on the loss function. Since standard strong convexity and smoothness assumptions are overly restrictive in high-dimensional settings, we instead impose their relaxations that are widely adopted in the high-dimensional analysis of M-estimators (Negahban et al., 2012).

Assumption 1 (Restricted Strong Convexity and Smoothness) *For any $k = 0, \ldots, K$, $\boldsymbol{\beta}, \boldsymbol{\Delta} \in \mathbb{R}^p$, there exists constants $\alpha_k, \beta_k, \tau_k, \gamma_k > 0$ such that*

$$\left\langle \nabla\mathcal{L}^{(k)}\left(\boldsymbol{\beta}+\boldsymbol{\Delta}\right) - \nabla\mathcal{L}^{(k)}\left(\boldsymbol{\beta}\right), \boldsymbol{\Delta}\right\rangle \ge \alpha_k\|\boldsymbol{\Delta}\|_2^2 - \tau_k\frac{\log p}{n_k}\|\boldsymbol{\Delta}\|_1^2,$$
$$\left\langle \nabla\mathcal{L}^{(k)}\left(\boldsymbol{\beta}+\boldsymbol{\Delta}\right) - \nabla\mathcal{L}^{(k)}\left(\boldsymbol{\beta}\right), \boldsymbol{\Delta}\right\rangle \le \beta_k\|\boldsymbol{\Delta}\|_2^2 + \gamma_k\frac{\log p}{n_k}\|\boldsymbol{\Delta}\|_1^2,$$

with probability larger than $1 - c_1\exp(-c_2 n_k - c_3\log p)$, where $n_k = n_S$ for $k \in [K]$, $n_k = n_T$ for $k = 0$, and c_1, c_2, c_3 are generic constants.

Remark 2 *For linear regression models specified in Example 1, Assumption 1 holds with a broad class of anisotropic random design matrices (Rudelson and Zhou, 2012). For the broad class of generalized linear models specified in Example 2, Assumption 1 holds for $\boldsymbol{\beta}$, $\boldsymbol{\Delta} \in \mathbb{B}_2(R)$ for some large constant R (Agarwal et al., 2010). Here $\mathbb{B}_2(R) := \{\boldsymbol{\beta}\in\mathbb{R}^p : \|\boldsymbol{\beta}\|_2 \le R\}$ denotes the Euclidean ℓ_2-ball of radius $R > 0$. The restriction set $\mathbb{B}_2(R)$ is essential because for some generalized linear models, the Hessian function $\Psi_k''(\cdot)$ approaches zero as its argument diverges. For these models, our results hold with the restriction set imposed on the solution and parameter space. See a detailed discussion in Agarwal et al. (2010); Loh and Wainwright (2011, 2013).*

In the following, we define $\alpha_{\min} = \min_{0\le k\le K}\alpha_k, \alpha_{\max} = \max_{0\le k\le K}\alpha_k$ and $\beta_{\max} = \max_{0\le k\le K}\beta_k$, with α_k and β_k being the constants specified in Assumption 1. Notice that

in Assumption 1, we impose regularity conditions on each empirical loss function $\mathcal{L}^{(k)}$ individually but not any on their differences. This flexibility accounts for potential distribution shifts across tasks. Next, we state a theorem that shows with a proper choice of tuning parameters and sufficiently large sample size, the *TransMission* estimator $\hat{\boldsymbol{\beta}}^{(0)}$ is guaranteed to be close to the target parameter $\boldsymbol{\beta}_*^{(0)}$ under distribution shifts. The theorem is deterministic in nature, with corresponding probabilistic results provided in the subsequent corollaries.

Theorem 3 *Suppose Assumption 1 holds and the true parameters are in $\Theta(s,h)$ defined in (3). For TransMission problem (6), consider any choice of tuning parameters such that, $\lambda_T \geq 2\|\nabla\mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)})\|_\infty$, $\kappa_T = 24s\lambda_T/\alpha_0$,*

$$\lambda_0 \geq 2\Big\|\sum_{k=0}^{K}\frac{n_k}{N}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\Big\|_\infty, \lambda_1 \geq \frac{2}{K}\max_{k\in[K]}\Big\|\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\Big\|_\infty \vee \left(\frac{4\alpha_{\max}}{\gamma_0 K}\lambda_T\right).$$

Suppose $n_S > n_T > \left[\frac{C_0(2\lambda_0^2 s + K\lambda_1 h/\gamma_0)}{(K\lambda_1^2)\wedge\lambda_0^2} + C_1 s\right]\log p$, then any solution $\hat{\boldsymbol{\beta}}^{(0)}$ to the problem satisfies

$$\|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}_*^{(0)}\|_2^2 \leq \left(C_2\lambda_0^2 s + C_3 K\lambda_1 h\right) \wedge \left(C_4\lambda_T^2 s\right), \tag{7}$$

where $C_0 = \frac{64(\alpha_{\max}\tau_0 + \beta_{\max}\gamma_0)}{\alpha_{\min}^2}, C_1 = \frac{64(\tau_0^2+\alpha_0\tau_0)}{\alpha_0^2}, C_2 = \frac{9\gamma_0^2}{\alpha_{\min}^4}$, $C_3 = \frac{12\gamma_0}{\alpha_{\min}^2}$ and $C_4 = \frac{128}{\alpha_0^2}$.

The convergence rate established in (7) is given by the minimum of two terms, with the first term contributed by solving the objective (6a) and the second term coming from the constraint $\mathcal{C}_T$ in (6b). In the first term, $C_2\lambda_0^2 s$ reflects the rate of leveraging both target and source samples for estimating the s-sparse target parameter $\boldsymbol{\beta}_*^{(0)}$, whereas $C_3 K\lambda_1 h$ is the cost of estimating the source-target contrasts $\{\boldsymbol{\delta}^{(k)}\}_{k\in[K]}$, each of which is of magnitude h. The second term, $C_4\lambda_T^2 s$, corresponds to the rate obtained when using only the target sample for estimation. The bound shows that the *TransMission* estimator benefits from parameter transfer when the source tasks provide useful information, i.e., when h is small, and performs no worse than the single-task estimator when the source tasks are not beneficial. Further elaboration on this point will be given by Corollary 5 with a specific choice of tuning parameters. Notably, the results hold under a variety of distribution shifts, provided that the regularity conditions in Assumption 1 on each of the empirical loss functions $\mathcal{L}^{(k)}$ are satisfied.

To better interpret the result, we provide the probabilistic counterpart of Theorem 3 in the sequel, along with the choice of tuning parameters. We first introduce an additional assumption on the concentration property of loss functions.

Assumption 4 (Concentration) *$\{\nabla\ell^{(k)}(z_i^{(k)};\boldsymbol{\beta}_*^{(k)})\}_{k=0,\ldots,K,i\in[n_k]}$ are independent random vectors and for any $j\in[p]$, there are universal constants c_1, c_2 such that*

$$P(|\nabla_j\ell^{(k)}(z_i^{(k)};\boldsymbol{\beta}_*^{(k)})| \geq t) \leq 2\exp\left[-c_2 t^2/\sigma^2\right], \text{ for all } t > c_1,$$

where σ is a constant that is bounded above.

Assumption 4 requires the sample gradient evaluated at the true parameter to exhibit sub-Gaussian tails, a condition satisfied by various well-known models, including the generalized linear models discussed in Example 2, when the design matrix is sub-Gaussian (Agarwal et al., 2010). It's worth noting that Assumption 4, along with Assumption 1, is not the weakest possible for the method's broad application. We adopt these conditions to align with the literature, facilitating a direct comparison between the theoretical results of *TransMission* and existing methods. With this additional assumption, we are ready to establish the convergence rate of $\hat{\boldsymbol{\beta}}^{(0)}$ that holds with high probability.

Corollary 5 *Suppose Assumption 1 and 4 hold and the true parameters are in $\Theta(s,h)$ defined in (3). Suppose $n_S > n_T \gg Ks\log p$, if we choose*

$$\lambda_0 \asymp \left[\Big(\frac{h^2\log p}{s^2 n_T}\Big)^{1/4} + \Big(\frac{\log p}{N}\Big)^{1/2}\right], \quad \lambda_1 \asymp \frac{n_S}{N}\Big(\frac{\log p}{n_T}\Big)^{1/2}, \quad \textit{and} \quad \lambda_T \asymp \Big(\frac{\log p}{n_T}\Big)^{1/2},$$

then the solution of the TransMission problem (6) satisfies

$$\|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}_*^{(0)}\|_2^2 \lesssim \frac{s\log p}{N} + \left(h\sqrt{\frac{\log p}{n_T}}\right) \wedge \left(\frac{s\log p}{n_T}\right), \tag{8}$$

with probability larger than $1 - c_1\exp(-c_2\log p)$.

The first term $s\log p/N$ in (8) is the rate of estimating an s-sparse parameter $\boldsymbol{\beta}_*^{(0)}$ based on N i.i.d. samples. This term reveals the benefit of using both the source and target datasets for estimating the target parameter $\boldsymbol{\beta}_*^{(0)}$. The term $h\sqrt{\log p/n_T}$ accounts for the rate of estimating $\boldsymbol{\delta}^{(k)}$ unique to each source task and thus is limited by the target dataset size n_T. The term $s\log p/n_T$ is the minimax optimal rate for estimating $\boldsymbol{\beta}_*^{(0)}$ using only the target data when $s \ll p$ (Raskutti et al., 2011). Taken together, the upper bound reveals that *TransMission* has a sharper convergence rate than estimators using the target sample alone when $h \lesssim s\sqrt{\log p/n_T}$, and always performs no worse than these estimators when $h \gtrsim s\sqrt{\log p/n_T}$. Notably, the rate depends only on the magnitude of the parameter contrast, i.e, h, and is unaffected by other forms of distribution shift. We explore this further in the next section.

3.3 Understanding the Robustness of the TransMission Method

In this section, we discuss the robustness of the *TransMission* method compared to the two-step method proposed in Li et al. (2022) and Tian and Feng (2023). While both methods aim to address the high-dimensional parameter transfer problem, we take a significantly different approach to achieve robustness to distribution shifts and guarantee no negative transfer. To understand this, we first demonstrate why the pooling estimator $\hat{\boldsymbol{\beta}}_{\text{Pooling}}$, defined in (4), will fail under distribution shifts. Define the population counterpart of the first-step pooling estimator as $\boldsymbol{\beta}_{\text{Pooling}} \in \operatorname{argmin}_{\boldsymbol{\beta}} \sum_{k=0}^{K} \frac{n_k}{N}\mathbb{E}\left[\mathcal{L}^{(k)}(\boldsymbol{\beta})\right]$. By the optimality of $\boldsymbol{\beta}_{\text{Pooling}}$ and the definition of $\boldsymbol{\beta}_*^{(k)}$ and assuming that $\mathcal{L}^{(k)}$ is second order differentiable, we have

$$\sum_{k=0}^{K}\frac{n_k}{N}\mathbb{E}\left(\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_{\text{Pooling}}) - \nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\right) = \sum_{k=0}^{K}\frac{n_k}{N}\mathbb{E}\left(\nabla^2\mathcal{L}_{\text{Int}}^{(k)}(\boldsymbol{\beta}_{\text{Pooling}} - \boldsymbol{\beta}_*^{(k)})\right) = \mathbf{0},$$

where we define $\nabla^2\mathcal{L}_{\text{Int}}^{(k)} := \nabla^2\mathcal{L}^{(k)}(\boldsymbol{\beta}_{\text{Int}}^{(k)})$ with $\boldsymbol{\beta}_{\text{Int}}^{(k)}$ being some intermediate point between $\boldsymbol{\beta}_{\text{Pooling}}$ and $\boldsymbol{\beta}_*^{(k)}$. Further assuming that the Hessian matrix is invertible, we can compute the asymptotic bias of the pooling estimator as

$$\boldsymbol{\delta}_{\text{Pooling}} := \boldsymbol{\beta}_{\text{Pooling}} - \boldsymbol{\beta}_*^{(0)} = \left[\sum_{k=0}^{K} \frac{n_k}{N}\mathbb{E}\left(\nabla^2\mathcal{L}_{\text{Int}}^{(k)}\right)\right]^{-1} \sum_{k=1}^{K} \frac{n_k}{N}\mathbb{E}\left(\nabla^2\mathcal{L}_{\text{Int}}^{(k)}\right)\boldsymbol{\delta}^{(k)}, \tag{9}$$

where recall $\boldsymbol{\delta}^{(k)} = \boldsymbol{\beta}_*^{(k)} - \boldsymbol{\beta}_*^{(0)}$ is the source-specific signal.

Recall that in the parameter space $\Theta(s, h)$, we assumed that $\boldsymbol{\delta}^{(k)}$ is ℓ_1-sparse in the sense that $\|\boldsymbol{\delta}^{(k)}\|_1 \leq h$ for all $k \in [K]$. Therefore, in a homogeneous setting where the expected Hessian $\mathbb{E}(\nabla^2\mathcal{L}_{\text{Int}}^{(k)})$ are the same across sources, $\boldsymbol{\delta}_{\text{Pooling}}$ is a convex combination of $\boldsymbol{\delta}^{(k)}$s with $\|\boldsymbol{\delta}_{\text{Pooling}}\|_1 \leq h$. However, under a distribution shift setting where the expected Hessian can vary arbitrarily across different sources, the sparsity pattern of $\boldsymbol{\delta}_{\text{Pooling}}$ could be destroyed when multiplying $\boldsymbol{\delta}^{(k)}$s by the expected Hessian matrix before combining. Consequently, the pooling bias $\boldsymbol{\delta}_{\text{Pooling}}$ may no longer have a bounded ℓ_1 norm, leading to a deterioration in the accuracy of sparse estimation.

The two-step methods proposed in Li et al. (2022) and Tian and Feng (2023) achieve parameter transfer by correcting the bias of $\boldsymbol{\delta}_{\text{Pooling}}$ via regularized regression using solely the target data. As shown in Table 1, these two-step estimator based on such a pooling estimator leads to a rate with an additional term C_Σ due to the distribution shifts, where

$$C_\Sigma := 1 + \max_{j\leq p}\max_{k}\left\| e_j^\top\left(\mathbb{E}\left(\nabla^2\mathcal{L}_{\text{Int}}^{(k)}\right) - \mathbb{E}\left(\nabla^2\mathcal{L}_{\text{Int}}^{(0)}\right)\right)\left(\sum_{k=1}^{K}\frac{1}{K}\mathbb{E}\left(\nabla^2\mathcal{L}_{\text{Int}}^{(k)}\right)\right)^{-1}\right\|_1. \tag{10}$$

with $e_j \in \mathbb{R}^p$ being the jth canonical basis vector. Such a factor C_Σ grows with the distribution shifts across tasks and can diverge in the rate of $O(\sqrt{p})$ even in the linear regression setting (He et al., 2024). Therefore their method suffers significantly from the high-dimensional distribution shifts, while *TransMission* is very robust to such shifts.

Remark 6 *Here we further clarify the distribution shifts from the perspective of parametric models. In statistical estimation,* $\mathbb{E}\left(\nabla^2\mathcal{L}_{Int}^{(k)}\right)$ *in (10) is closely related to the Fisher information matrix. The heterogeneity in Fisher information matrix can be understood along three shift types: (i)* covariate shift*, (ii)* conditional shift*, and (iii) the* label shift *induced by the first two. For example, in the linear model of Example 1, the Fisher information matrix reduces (up to scale) to the covariance* $\boldsymbol{\Sigma}^{(k)} = \frac{1}{n_k}\mathbb{E}((\mathbf{X}^{(k)})^\top\mathbf{X}^{(k)})$*, so heterogeneity in the Fisher information matrix directly corresponds to covariate shift, i.e.,* $\mathbb{P}_X^{(k)} \neq \mathbb{P}_X^{(l)}$ *for* $l \neq k \in \{0, \ldots, K\}$*; in the generalized linear models of Example 2, the Fisher information matrix is proportional to* $\frac{1}{n_k}\mathbb{E}\big((\mathbf{X}^{(k)})^\top\boldsymbol{W}_{k,\text{Int}}^2\mathbf{X}^{(k)}\big)$ *with diagonal weights* $(\boldsymbol{W}_{k,\text{Int}})_{ii} = \phi_k^{-1/2}\,\Psi_k''((\mathbf{X}_i^{(k)})^\top\boldsymbol{\beta}_{\text{Int}}^{(k)})^{1/2}$*; so heterogeneity in Fisher information matrix can arise from heterogeneity in the variance function* Ψ_k'' *or the dispersion* ϕ_k*, which corresponds to the conditional shift* $\mathbb{P}_{y|X}^{(k)} \neq \mathbb{P}_{y|X}^{(l)}$*. Since the marginal label distribution depends on both* $\mathbb{P}_X^{(k)}$ *and* $\mathbb{P}_{y|X}^{(k)}$*, label shift follows whenever either of these distributions changes. The factor* C_Σ *in* (10) *quantifies discrepancies induces from the above shifts.*

Remark 7 *We would like to clarify that our goal with the proposed TransMission estimator is not to explicitly tackle each type of distribution shifts individually, but to preserve effective and robust parameter transfer in the presence of distribution shifts. Concretely, of true interest is capturing the "parameter contrast" $\boldsymbol{\delta}^{(k)} = \boldsymbol{\beta}^{(k)} - \boldsymbol{\beta}^{(0)}$, while covariate and conditional distributions are allowed to vary across tasks. Such a robustness follows from the joint training with fused-regularization design in (6a), which achieves parameter transfer from the parameter level rather than loss level. As the loss functions are jointly considered in the objective function, with their respective parameters, and thus the losses (and hence the entire parametric models) are allowed to be different.*

Method	Setting	Source Detection	Statistical Rate
TransLasso (Li et al., 2022)	LM	No	$\frac{s\log p}{N} + \left(C_\Sigma h\sqrt{\frac{\log p}{n_T}}\right) \wedge (C_\Sigma^2 h^2)$
		Yes	$\frac{s\log p}{N} + \left(C_\Sigma h\sqrt{\frac{\log p}{n_T}}\right) \wedge (C_\Sigma^2 h^2) \wedge \left(\frac{s\log p}{n_T}\right) + \frac{\log K}{n_T}$
TransGLM (Tian and Feng, 2023)	GLM	-	$\frac{s\log p}{N} + \left(C_\Sigma h\sqrt{\frac{\log p}{n_T}}\right) \wedge (C_\Sigma^2 h^2)$
TransMission (Corollary 9)	M-estimator	-	$\frac{s\log p}{N} + \left(h\sqrt{\frac{\log p}{n_T}}\right) \wedge (C_\Sigma^2 h^2) \wedge \left(\frac{s\log p}{n_T}\right)$

Table 1: Comparison of different parameter transfer methods. LM and GLM denote linear and generalized linear models, respectively. Source detection indicates whether an additional transferable source detection step is performed. C_Σ, defined in (10), quantifies the level of distribution shift.

Another advantage of our method is about preventing negative transfer, i.e., ensuring that incorporating source data does not degrade the estimation accuracy of the target parameter when h is large. As shown in Table 1, Li et al. (2022) proposes to perform an additional source detection step to address this negative transfer issue. However, such an additional detection step comes with a cost of order $O(\log K/n_T)$, therefore the overall convergence rate is always no faster than $O(\log K/n_T)$ even if h is small. In contrast, *TransMission* prevents negative transfer through a target data-oriented constraint during parameter transfer, without sacrificing the statistical convergence rate in the transferable regime, although it may incur additional computational cost due to an extra pilot estimator construction and hyperparameter tuning step.

Remark 8 *Several concurrent works, including TransHDGLM (Li et al., 2024), STRIFLE (Cai et al., 2025), TransFusion (He et al., 2024), and TransQR (Bai et al., 2024) have also demonstrated robustness in high-dimensional parameter transfer. However, they either fail to account for negative transfer or incur non-negligible costs, typically of order $O(1/n_T)$, to prevent the negative transfer.*

3.4 Sharper Convergence Rate and Minimax Optimality

We now show that with a more careful choice of tuning parameters, the *TransMission* estimator could achieve a faster convergence rate, and can match the existing lower bound, showing the advantage of the proposed method and the tightness of our bound.

Corollary 9 *Suppose Assumption 1, 4 hold and the true parameters are in $\Theta(s,h)$ defined in (3). If $n_S > n_T \gg Ks\log p$, and the tuning parameters are chosen as outlined in Appendix A.3, then the solution of the TransMission problem (6) satisfies*

$$\|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}_*^{(0)}\|_2^2 \lesssim \frac{s\log p}{N} + \left(\sqrt{\frac{\log p}{n_T}}h\right) \wedge \left(C_\Sigma^2 h^2\right) \wedge \frac{s\log p}{n_T},$$

with probability larger than $1 - c_1\exp(-c_2\log p)$.

Compared to Corollary 5, Corollary 9 establishes a sharper convergence rate by taking minimum over an additional term, $C_\Sigma^2 h^2$, in the second part of the bound. Recall that C_Σ, defined in (10), measures the degree of distribution shifts. This term enables a faster rate of convergence when $h \lesssim (C_\Sigma^{-2}\sqrt{\log p/n_T}) \wedge (C_\Sigma^{-1}\sqrt{s\log p/n_T})$, i.e., when the source and target distributions are highly similar. Appendix A.3 provides a detailed discussion of how such a tighter bound is achieved via a more careful choice of tuning parameters.

The upper bound established in Corollary 9 recovers the existing minimax lower bounds in the homogeneous setting. Specifically, prior works (Li et al., 2022; Tian and Feng, 2023) have shown that in linear regression and generalized linear models with a homogeneous design ($C_\Sigma = 1$), the ℓ_2-estimation error for $\boldsymbol{\beta}_*^{(0)}$ has the lower bound

$$\inf_{\check{\boldsymbol{\beta}}^{(0)}} \sup_{\boldsymbol{\beta}_*^{(0)}\in\Theta(s,h)} \|\check{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}_*^{(0)}\|_2^2 \gtrsim \frac{s\log p}{N} + \left(h\sqrt{\frac{\log p}{n_T}}\right) \wedge h^2 \wedge \frac{s\log p}{n_T},$$

with probability at least 1/2. In this setting, one can verify that $C_\Sigma = 1$, and our upper bound in Corollary 9 matches the minimax lower bound, showing the minimax optimality of *TransMission* under the conditions outlined in Corollary 9. Notably, even in this homogeneous setting ($C_\Sigma = 1$), our method achieves the minimax lower bound under significantly weaker conditions on h compared to existing works (Li et al., 2022; Tian and Feng, 2023; Li et al., 2024). For example, TransGLM requires $h \lesssim s\sqrt{\log p/n_T}$ and $h \ll \sqrt{n_T/\log p}$ to attain the minimax rate, which requires h to be sufficiently small and the target sample n_T to be neither too large nor too small. On the other hand, we only assume $h \ll \sqrt{n_T/(K^2\log p)}$. Given that the high-dimensional analysis typically requires $n_T \gg s\log p$, our assumption is much less restrictive on h compared to existing work.

Moreover, beyond the homogeneous case, our method remains robust to a broad class of distribution shifts, achieving a faster convergence in heterogeneous settings.

4. Distributed Parameter Transfer With Source Estimators

4.1 Distributed TransMission

In this section, we consider a distributed setting where the target and K source datasets are stored across different institutes or organizations, and direct data sharing is forbidden due to privacy and proprietary concerns. In such a setting, raw data cannot be pooled and multi-round communication-based distributed learning is often impractical due to the high dimensionality of model parameters. This motivates the development of a distributed version of *TransMission*, termed *D-TransMission*, which estimates the target parameter $\boldsymbol{\beta}_*^{(0)}$ using a *single round* of communication based on pre-fitted source parameters.

Our method integrates the idea of divide-and-conquer into the *TransMission* method described in Section 3, aiming to achieve a comparable estimation error as $\hat{\boldsymbol{\beta}}^{(0)}$ using only one-shot communication. Note that although divide-and-conquer has been well-studied for i.i.d. data (Lee et al., 2017), its application and analysis for high-dimensional parameter transfer is largely unexplored. Specifically, in *D-TransMission*, we let each source node k compute an estimator $\tilde{\boldsymbol{\beta}}^{(k)} \in \mathbb{R}^p$ locally based on source dataset $(\mathbf{X}^{(k)}, \mathbf{y}^{(k)})$ and send it to the target node. The target node then aggregates them with its own dataset $(\mathbf{X}^{(0)}, \mathbf{y}^{(0)})$ via solving the following joint learning problem:

$$\min_{\boldsymbol{\beta}^{(0)} \in \mathcal{C}_T, \boldsymbol{\beta}^{(k)} \in \mathbb{R}^p, k \in [K]} \left\{ \frac{n_T}{N} \mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)}) + \frac{n_S^2}{2N} \sum_{k=1}^{K} \left\| \tilde{\boldsymbol{\beta}}^{(k)} - \boldsymbol{\beta}^{(k)} \right\|_2^2 + \lambda_0 \left\| \boldsymbol{\beta}^{(0)} \right\|_1 + \sum_{k=1}^{K} \lambda_1 \left\| \boldsymbol{\beta}^{(k)} - \boldsymbol{\beta}^{(0)} \right\|_1 \right\}, \tag{11a}$$

$$\text{with } \mathcal{C}_T = \{\boldsymbol{\beta}^{(0)} \in \mathbb{R}^p : \|\nabla \mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)})\|_\infty \le \lambda_T, \|\boldsymbol{\beta}^{(0)}\|_1 \le \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1 + \kappa_T\}, \tag{11b}$$

Compared to its centralized counterpart, *D-TransMission* replaces the source empirical loss $\mathcal{L}^{(k)}(\cdot)$ with the mean squared error from the source parameter estimator $\tilde{\boldsymbol{\beta}}^{(k)}$ as a measure of model fitness. Such a modification makes *D-TransMission* no longer require transmitting the raw source data for transfer learning, reducing the communication cost from $O(n_S K p)$ to $O(Kp)$, which is a substantial improvement when either n_S or p is large. Algorithm 2 provides a pseudocode of the *D-TransMission* method.

Algorithm 2 *D-TransMission*

Input: target sample $\mathcal{S}^{(0)}$, source parameter estimators $\{\tilde{\boldsymbol{\beta}}^{(k)}\}_{k=1}^K$, tuning parameters λ_0, λ_1, λ_T, κ_T.

Output: the estimated target parameter $\hat{\boldsymbol{\beta}}_D^{(0)}$.

1. Compute $\hat{\boldsymbol{\beta}}_T^{(0)}$ by solving $\min_{\boldsymbol{\beta} \in \mathbb{R}^p} \{\mathcal{L}^{(0)}(\boldsymbol{\beta}) + \lambda_T \|\boldsymbol{\beta}\|_1\}$.
2. Compute $(\hat{\boldsymbol{\beta}}_D^{(0)}, \hat{\boldsymbol{\beta}}_D^{(1)}, \ldots, \hat{\boldsymbol{\beta}}_D^{(K)})$ by solving

$$\min_{\substack{\boldsymbol{\beta}^{(0)} \in \mathbb{R}^p, \\ \{\boldsymbol{\beta}^{(k)}\}_{k=1}^K \in (\mathbb{R}^p)^K}} \frac{n_T}{N} \mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)}) + \frac{n_S^2}{2N} \sum_{k=1}^{K} \|\tilde{\boldsymbol{\beta}}^{(k)} - \boldsymbol{\beta}^{(k)}\|_2^2 + \lambda_0 \|\boldsymbol{\beta}^{(0)}\|_1 + \lambda_1 \sum_{k=1}^{K} \|\boldsymbol{\beta}^{(k)} - \boldsymbol{\beta}^{(0)}\|_1,$$

   with $\mathcal{C}_T = \{\boldsymbol{\beta}^{(0)} \in \mathbb{R}^p : \|\nabla \mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)})\|_\infty \le \lambda_T,\ \|\boldsymbol{\beta}^{(0)}\|_1 \le \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1 + \kappa_T\}$.
3. Output $\hat{\boldsymbol{\beta}}_D^{(0)}$.

Next, we specify how to choose $\tilde{\boldsymbol{\beta}}^{(k)}$ to preserve the convergence rate of *TransMission*. Under the parameter space $\Theta(s, h)$, the $\boldsymbol{\beta}_*^{(k)}$'s are also sparse. Therefore, a natural choice of $\tilde{\boldsymbol{\beta}}^{(k)}$ is the ℓ_1-regularized estimator computed purely based on source dataset given by:

$$\hat{\boldsymbol{\beta}}_L^{(k)} \in \operatorname*{argmin}_{\boldsymbol{\beta}} \left\{ \mathcal{L}^{(k)}(\boldsymbol{\beta}) + \tilde{\lambda}_k \|\boldsymbol{\beta}\|_1 \right\}. \tag{12}$$

However, it is well known that the regularization term introduces a bias in $\hat{\boldsymbol{\beta}}_L^{(k)}$, and aggregating the local $\hat{\boldsymbol{\beta}}_L^{(k)}$s can only reduce the variance and has almost no effects on such

bias (Mcdonald et al., 2009). Resolving the issue necessarily requires us to first "correct" the bias at the level of each local source node before transmitting it to the target node for information aggregation.

Various approaches have been proposed to achieve such a bias correction (Javanmard and Montanari, 2014; van de Geer et al., 2014; Ning and Liu, 2017), many of which can be expressed in the form

$$\tilde{\boldsymbol{\beta}}^{(k)} = \hat{\boldsymbol{\beta}}_L^{(k)} - \hat{\boldsymbol{\Theta}}^{(k)} \nabla \mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_L^{(k)}), \tag{13}$$

where $\hat{\boldsymbol{\Theta}}^{(k)}$ will be specified later. To understand how such an estimator can achieve a smaller bias than $\hat{\boldsymbol{\beta}}_L^{(k)}$, we may rewrite (13) by subtracting $\boldsymbol{\beta}_*^{(k)}$ from both sides and applying the mean value theorem to obtain

$$\tilde{\boldsymbol{\beta}}^{(k)} - \boldsymbol{\beta}_*^{(k)} = -\hat{\boldsymbol{\Theta}}^{(k)} \nabla \mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)}) + \left(\boldsymbol{I} - \hat{\boldsymbol{\Theta}}^{(k)} \nabla^2 \mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{Int}^{(k)}) \right) \left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)} \right). \tag{14}$$

where $\hat{\boldsymbol{\beta}}_{Int}^{(k)}$ is a vector intermediating $\hat{\boldsymbol{\beta}}_L^{(k)}$ and $\boldsymbol{\beta}_*^{(k)}$. On the right-hand side of (14), the first term, $\hat{\boldsymbol{\Theta}}^{(k)} \nabla \mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})$, is expected to exhibit a similar concentration property as $\nabla \mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})$ thus contributes to the variance of $\tilde{\boldsymbol{\beta}}^{(k)}$; the second term is a product of the error of $\hat{\boldsymbol{\beta}}_L^{(k)}$ and $\boldsymbol{I} - \hat{\boldsymbol{\Theta}}^{(k)} \nabla^2 \mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{Int}^{(k)})$, which is affected by the bias of $\hat{\boldsymbol{\beta}}_L^{(k)}$. Therefore, to minimize the effect of bias in the subsequent knowledge transfer step, one should choose a $\hat{\boldsymbol{\Theta}}^{(k)}$ that is a good approximation of $[\nabla^2 \mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{Int}^{(k)})]^{-1}$. With a careful choice of $\tilde{\boldsymbol{\beta}}$, the *D-TransMission* framework could greatly reduce communication overhead while maintaining minimal performance loss compared to its centralized counterpart. In this paper, we follow the work of van de Geer et al. (2014) to construct $\hat{\boldsymbol{\Theta}}^{(k)}$, with details specified in Appendix A.4. Under mild conditions, it can be shown that the chosen $\hat{\boldsymbol{\Theta}}^{(k)}$ satisfies $\max_{j \in p} \| \boldsymbol{e}_j - \hat{\boldsymbol{\Theta}}_{j\cdot}^{(k)} \nabla^2 \mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{Int}^{(k)}) \|_\infty \leq \lambda_j$, with $\hat{\boldsymbol{\Theta}}_j^{(k)}$ being the jth row of $\hat{\boldsymbol{\Theta}}^{(k)}$. In this way, the bias term in (14) can be controlled by λ_j, achieving the goal of bias reduction.

Remark 10 *We note that while constructing the estimator in (13) typically involves solving p different regularized M-estimation problems, with computational complexity $O(p^2)$ per node, they can be performed locally and offline. These pre-computed estimators can then be stored as pre-trained models and transmitted to the central server only when a target task arises. Moreover, the computation involving source data is decoupled from the transfer step involving the target, which automatically avoids redoing all the computations whenever the target task changes. In fact, we found that in practice $\tilde{\boldsymbol{\beta}}^{(k)}$ can also be chosen to be asymptotically unbiased estimators, such as the SCAD estimator, which achieves similar performance but is much more computationally efficient. In this paper, we stick to the proposed approach as it is better suited for establishing a non-asymptotic statistical convergence guarantee. We would like to also note that the primary objective of D-TransMission is to achieve a comparable statistical rate as the centralized estimator under the communication constraint, rather than to distribute the total computational workload across nodes. Designing computationally lighter alternatives with comparable theoretical guarantees is an important direction for future research.*

4.2 Theoretical Guarantee

We next demonstrate that, given a sufficiently large source sample size, the *D-TransMission* estimator $\hat{\boldsymbol{\beta}}_D^{(0)}$ can achieve the same statistical convergence rate as the *TransMission* estimator $\hat{\boldsymbol{\beta}}^{(0)}$ with just one-shot communication based on pre-estimated source parameters.

We first introduce a set of regularity assumptions listed in Appendix A.4, where conditions (D1), (D3), and (D4) are standard in high-dimensional sparse regression (Lee et al., 2017). Condition (D2) assumes row-wise sparsity on $\mathbb{E}[(\nabla^2 \mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)}))]^{-1}$, which is also commonly imposed in related literature (van de Geer et al., 2014; Zhang and Zhang, 2014). With these additional regularity conditions, we now establish the statistical convergence rate of the *D-TransMission* estimator.

Theorem 11 *Under Assumption 1, 4 and 13 in Appendix A.4, assume* $n_T \gg K^2 s \log p$ *and* $n_S \gg K^2 s^2 \log p$. *If we construct* $\{\tilde{\boldsymbol{\beta}}^{(k)}\}_{k\in[K]}$ *through (13) and (A.5) with parameters* $\tilde{\lambda}_k = \mu_{jk} \asymp \sqrt{\log p / n_S}$, *then the solution of the D-TransMission problem (11) satisfies*

$$\|\hat{\boldsymbol{\beta}}_D^{(0)} - \boldsymbol{\beta}_*^{(0)}\|_2^2 \lesssim \frac{s \log p}{N} + \left(\sqrt{\frac{\log p}{n_T}} h\right) \wedge \frac{s \log p}{n_T}, \tag{15}$$

with probability at least $1 - c_1 \exp(-c_2 \log p)$.

Theorem 11 demonstrates with a sufficiently large n_S, $\hat{\boldsymbol{\beta}}_D^{(0)}$ achieves a statistical rate of the same order as the centralized counterpart established in Corollary 5, while requiring only a single round of communication. Notice that the theorem imposes a more restrictive growth condition on the source sample size n_S with the number of source tasks K, a requirement common to divide-and-conquer type methods. Using arguments similar to Section 3.4, one can further establish the minimax optimality of the *D-TransMission* estimator under similar conditions.

5. Simulation and Real Data Analysis

5.1 Simulation with Synthetic Data

To showcase the effectiveness of *TransMission* method and support our theoretical results, we conduct simulations under different forms of distribution shifts. We compare the empirical performance of *TransMission* and *D-TransMission* against several established baselines. Specifically, we include *Single-Lasso*, which applies ℓ_1-regularized regression solely on the target task, and *Agg-Lasso*, which pools all source and target samples and applies ℓ_1-regularized regression on the combined dataset. Additionally, we compare with *TransGLM* (Tian and Feng, 2023), which performs an additional de-bias step on the pooling estimator, and *TransHDGLM* (Li et al., 2024), an iterative algorithm that alternates between updating the target parameter and estimating the source-target contrasts.

We follow a similar experimental setup as in Li et al. (2022, 2024); He et al. (2024) by considering a high-dimensional regression problem with $p = 500$ and sparsity level $s = 10$. The true target parameter is set as $(\boldsymbol{\beta}_*^{(0)})_j = 0.3 \cdot \mathbf{1}_{\{1 \le j \le s\}}$. We then generate $n_T = 150$

independent target samples $\{(\mathbf{X}_i^{(0)}, \mathbf{y}_i^{(0)})\}_{i\in[n_T]}$ with

$$\mathbf{X}_i^{(0)} \sim \mathrm{N}(\mathbf{0}, \boldsymbol{I}),\ \mathbf{y}_i^{(0)} \sim \begin{cases} \mathrm{N}((\mathbf{X}_i^{(0)})^\top \boldsymbol{\beta}_*^{(0)}, 1) \text{ for linear regression,} \\ \mathrm{Ber}\left(\left(1+\exp(-(\mathbf{X}_i^{(0)})^\top \boldsymbol{\beta}_*^{(0)})\right)^{-1}\right) \text{ for logistic regression,} \end{cases}$$

where $\mathrm{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, $\mathrm{Ber}(p)$ represents a Bernoulli distribution with success probability p and $\boldsymbol{I}$ stands for the identity matrix. For the source sample, we generate the true source parameters $\boldsymbol{\beta}_*^{(k)} = \boldsymbol{\beta}_*^{(0)} + \boldsymbol{\delta}^{(k)}$ with $\boldsymbol{\delta}_j^{(k)} \sim N(0, (h/50)^2)$ for $1 \le j \le 50$ and $\boldsymbol{\delta}_j^{(k)} = 0$ otherwise. Unless otherwise specified, we choose the source sample size $n_S = 200$. When generating the source samples $\{(\mathbf{X}_i^{(k)}, \mathbf{y}_i^{(k)})\}_{i\in[n_S]}$, we simulate two types of distribution shifts-covariate shift and parametric model shift:

Covariate Shift. To assess robustness against covariate shifts, we generate independent covariates $\mathbf{X}_i^{(k)}$ under two settings.

(a) *Homogeneous design.* Each $\mathbf{X}_i^{(k)} \sim \mathrm{N}(0, \boldsymbol{I})$.

(b) *Heterogeneous design.* Each $\mathbf{X}_i^{(k)} \sim \mathrm{N}(0, \boldsymbol{\Sigma}^{(k)})$ with $\boldsymbol{\Sigma}^{(k)} = (\boldsymbol{A}^{(k)})^\top(\boldsymbol{A}^{(k)}) + \boldsymbol{I}$. Here $\boldsymbol{A}^{(k)}$ is a random matrix with each entry equals ρ with probability ρ and equals 0 with probability $1-\rho$. A larger ρ implies a more severe covariate shift.

Conditional Shift. To investigate the robustness against parametric model shifts, we generate independent response $\mathbf{y}_i^{(k)}$ under two settings.

(a) *Homogeneous design.* Each $\mathbf{y}_i^{(k)}$ is generated with $\boldsymbol{\beta}_*^{(k)}$ and $\mathbf{X}_i^{(k)}$ from the same parametric model as the target task.

(b) *Heterogeneous design.* For logistic regression, we set p_{ki} to follow a Beta distribution with the same mean as the target task but varying variance across tasks. The binary response $\mathbf{y}_i^{(k)}$ is then drawn from $\mathrm{Ber}(p_{ki})$, introducing task-specific overdispersion (See Appendix A.5 for details).

For linear regression, we evaluate performance using prediction error on a test target dataset, while for logistic regression classification accuracy is used. Table 2 and Table 3 compare the performance of different methods on the linear and logistic regression tasks, respectively. The results lead to the following observations.

In Table 2(a), when both h and ρ are small, indicating a weak parameter contrast and covariate shift (a setting favorable to *Agg-Lasso* and *TransGLM*), *TransMission* achieves comparable prediction error to other methods. As the covariate shift level increases, i.e., under more severe distribution shifts, the prediction error of other methods increases rapidly, whereas *TransMission* exhibits only a slight degradation in performance. This effect is even more pronounced as h grows, as both *Agg-Lasso* and *TransGLM* eventually underperform the single-task Lasso baseline, while *TransMission* consistently outperforms all other methods. A similar trend is observed in the logistic regression setting (Table 3). Under homogeneous conditions, *TransMission* performs on par with existing methods, yet it demonstrates a clear advantage when covariate and parametric model shifts are present. These findings align with the theoretical results in Section 3, confirming that *TransMission* maintains comparable performance in homogeneous settings while achieving superior robustness under distribution shifts.

h	ρ	Agg-Lasso	Single-Lasso	Trans-GLM	TransHDGLM	TransMission
5	0.1	0.132	0.894	0.171	**0.123**	0.134
5	0.2	0.141	0.891	0.162	**0.127**	0.136
5	0.4	0.172	0.896	0.194	**0.158**	0.160
10	0.1	0.320	0.895	0.329	0.239	**0.197**
10	0.2	0.372	0.894	0.308	0.281	**0.206**
10	0.4	0.433	0.893	0.374	0.335	**0.254**
15	0.1	0.651	0.894	0.534	0.440	**0.251**
15	0.2	0.782	0.894	0.540	0.497	**0.274**
15	0.4	0.884	0.885	0.615	0.586	**0.324**
20	0.1	1.095	0.894	0.874	0.663	**0.301**
20	0.2	1.339	0.887	0.882	0.704	**0.326**
20	0.4	1.439	0.886	0.927	0.805	**0.404**

(a)

ρ	n_S	TransMission	D-TransMission
0.0	50	0.698	1.054
0.0	100	0.537	0.937
0.0	150	0.357	0.617
0.0	200	0.297	0.438
0.0	250	0.258	0.310
0.0	300	0.224	0.260
0.3	50	0.665	1.021
0.3	100	0.515	0.845
0.3	150	0.414	0.610
0.3	200	0.348	0.440
0.3	250	0.326	0.355
0.3	300	0.300	0.303

(b)

Table 2: Prediction error comparison in the linear regression case. (a) comparison of different methods across varying levels of covariate shift (ρ) and parameter contrast (h). The best performance for each setting is highlighted in bold. (b) comparison of *TransMission* and *D-TransMission* errors across different values of ρ and source sample sizes (n_S) with $h = 20$.

h	Shift Type	Agg-Lasso	Single-Lasso	Trans-GLM	TransHDGLM	TransMission
5	none	**0.662**	0.534	0.644	0.655	0.647
5	covariate	0.675	0.537	0.657	0.671	**0.676**
5	both	0.664	0.524	0.650	0.653	**0.664**
15	none	**0.622**	0.540	0.606	0.618	0.621
15	covariate	0.603	0.538	0.591	0.575	**0.621**
15	both	0.610	0.524	0.591	0.560	**0.627**
25	none	0.572	0.533	0.561	0.560	**0.586**
25	covariate	0.559	0.533	0.559	0.533	**0.579**
25	both	0.564	0.519	0.554	0.524	**0.588**

Table 3: Classification accuracy comparison in the logistic regression case. Comparison of different methods across different heterogeneity levels (h) and shift types (no shifts, covariate shifts, both covariate and parametric model shifts). The best performance for each setting is highlighted in bold.

Table 2(b) evaluates *D-TransMission*, the communication-efficient variant of *TransMission*, in comparison to its centralized counterpart *TransMission*. As the source sample size n_S increases, the performance of *D-TransMission* progressively approaches that of the centralized *TransMission*. This aligns with the theoretical result that with sufficiently large source data, the distributed method can achieve comparable accuracy using only a single round of communication with source parameter estimators.

We have included several additional experiments to further examine the robustness and computational behavior of the proposed method. Specifically, we evaluate performance under heavy-tailed error distributions, where the noise follows a Student-t distribution with varying degrees of freedom. We also evaluate performance under varying feature dimensions p, to assess scalability in high-dimensional regimes. The corresponding results and detailed discussions are provided in Appendix C.

5.2 Real-Data Studies

We evaluate the proposed TransMission method on two real-data applications: a high-dimensional gene expression prediction task under a linear regression model and a handwritten-digit classification task under a logistic regression model. Together, these studies assess the method in both biological and image-based settings with heterogeneous source-target relationships.

Gene expression analysis. We study a real-world high-dimensional gene expression prediction problem based on the GTEx Portal dataset. Following established practice in the genomics literature (e.g., Li et al., 2022), we investigate how genes associated with the central nervous system (CNS) influence the expression of a protein-coding gene across multiple tissues. After preprocessing and filtering, we obtain approximately $p \approx 1{,}600$ covariates. We treat the brain tissues as source tasks and each non-brain tissue as a target task, resulting in heterogeneous sample sizes across targets. We evaluate prediction performance using five-fold cross-validation on each target tissue, with mean squared error (MSE) computed on held-out samples and averaged across folds.

As reported in Table 4, *TransMission* achieves the best average MSE among all competing methods, improving upon the second-best method by a substantial margin. Moreover, *TransMission* attains the best performance in 32 out of 36 target tissues and ranks second in the remaining 4 tissues, demonstrating consistently strong performance across a wide range of tissue-specific tasks. These results confirm the effectiveness of *TransMission* in high-dimensional biological settings.

Handwritten-digit classification. We next evaluate the proposed TransMission method on a handwritten-digit classification task using the MNIST dataset and a corrupted variant from MNIST-C, where images are affected by impulse noise (Mu and Gilmer, 2019). This type of corruption randomly perturbs pixel intensities, introducing structured distribution shifts that test the robustness of different transfer learning methods under a logistic regression model.

To construct the multi-source binary classification task, we define four different source datasets ($K = 4$) and one target dataset. The target dataset consists of clean images from MNIST, while the source datasets are drawn from impulse noise-corrupted MNIST-C images. Each source dataset consists of 100 training samples ($n_S = 100$), while the target dataset contains 50 training samples ($n_T = 50$). All images are represented as flattened pixel vectors, resulting in a 784-dimensional feature space ($p = 784$). For each digit, we create a binary classification problem where that digit serves as the positive class, and the negative class is sampled from other digits. To introduce the parameter contrast $\boldsymbol{\delta}^{(k)} = \boldsymbol{\beta}^{(k)} - \boldsymbol{\beta}^{(0)}$ across sources, each source dataset is assigned a different negative-class digit. For example, if the positive class is digit "3", one source dataset may use digit "5" as the negative class, while another source may use digit "7". In addition, the impulse noise applied to source datasets introduces further distribution shifts, allowing us to evaluate how each method captures transferable parameter information under heterogeneous conditions.

We evaluate performance using classification accuracy on an independent set of 2000 target samples. The results, averaged over 100 independent replications, are reported in Table 5, which presents test classification accuracy for the 10 binary classification tasks. TransMission achieves the highest accuracy in 8 out of 10 cases, demonstrating its effec-

Table 4: Performance comparison for different target tissues. The best performance for each setting is highlighted in bold. The second best performance is underlined.

Target Tissue	Agg-Lasso	Single-Lasso	Trans-GLM	TransHDGLM	TransMission
adipose subcutaneous	158.82	181.86	105.12	185.44	**70.81**
adipose visceral omentum	138.60	173.07	88.30	178.34	**76.24**
adrenal gland	54.45	68.79	57.60	57.04	**46.29**
artery aorta	19.85	24.95	**17.33**	25.20	18.78
artery coronary	82.81	74.45	**66.63**	72.52	70.87
artery tibial	20.96	32.78	18.71	31.95	**11.16**
bladder	63.28	68.29	58.47	70.13	**46.61**
breast mammary tissue	194.30	249.33	212.79	250.59	**67.86**
cells cultured fibroblasts	17.80	14.67	20.23	16.33	**4.02**
cells ebv-transformed lymphocytes	7.52	11.96	12.10	12.56	**4.33**
colon sigmoid	107.23	157.50	82.94	163.35	**40.24**
colon transverse	79.76	98.50	83.36	100.55	**29.66**
esophagus gastroesophageal junction	98.29	131.02	72.41	135.69	**53.46**
esophagus mucosa	29.51	29.53	27.56	29.66	**6.34**
esophagus muscularis	197.70	211.08	177.15	208.13	**78.73**
heart atrial appendage	11.77	11.40	11.22	11.05	**9.49**
heart left ventricle	10.85	14.99	11.28	12.04	**8.21**
kidney cortex	39.70	69.01	38.85	54.62	**41.63**
liver	22.08	21.90	22.28	22.78	**1.96**
lung	278.73	281.14	**156.71**	285.86	247.45
minor salivary gland	22.73	29.19	27.31	32.99	**15.90**
muscle skeletal	3.05	8.73	2.97	8.80	**1.58**
nerve tibial	68.51	76.58	34.13	76.99	**29.91**
ovary	72.68	76.15	73.69	77.18	**64.16**
pancreas	9.90	9.95	10.07	9.99	**1.74**
pituitary	121.04	179.89	110.82	206.03	**62.01**
prostate	75.00	78.77	72.15	86.13	**43.24**
skin not sun exposed suprapubic	12.67	20.88	13.25	21.04	**5.86**
skin sun exposed lower leg	11.99	20.00	11.74	19.74	**6.29**
small intestine terminal ileum	63.02	88.94	63.49	85.56	**55.84**
spleen	2.78	8.42	2.21	8.27	**1.34**
stomach	36.18	53.47	37.45	55.98	**16.61**
testis	95.37	95.18	57.98	97.10	**12.39**
thyroid	53.40	64.49	53.84	66.30	**36.70**
uterus	205.43	296.47	**128.50**	293.40	138.60
vagina	88.14	127.75	89.96	139.63	**82.97**
Average	71.55	87.81	59.18	89.14	**41.92**

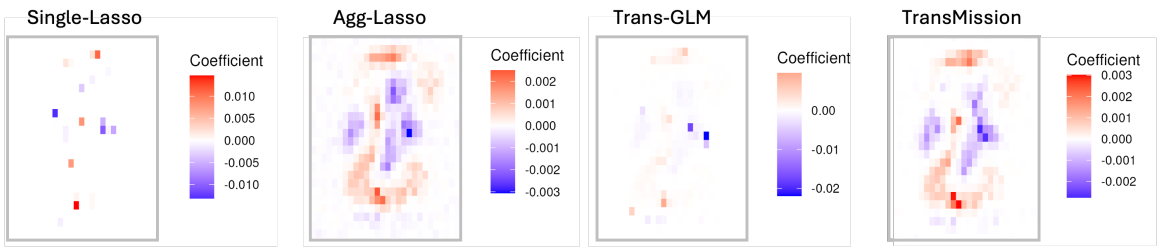

Figure 1: Heatmaps of learned coefficients for classifying digit 8 (positive class) vs. digit 9 (negative class) across different methods. Red shows pixels that strongly contribute to predicting an 8, while blue shows pixels that discourage classification as an 8.

Table 5: Classification accuracy (%) for different digits using various methods. The best performance for each digit is highlighted in bold.

Method	Digit									
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Single-lasso	97.72	95.89	91.09	92.63	89.82	86.46	96.97	89.82	86.93	86.80
Agg-Lasso	97.88	97.20	93.99	90.84	82.47	94.65	98.49	88.50	92.72	86.14
Trans-GLM	97.79	**97.57**	94.45	92.76	**90.08**	91.61	98.47	89.88	90.28	89.75
TransHDGLM	96.37	94.95	90.85	87.39	86.06	84.28	93.96	84.00	85.01	85.40
TransMission	**97.94**	97.38	**95.24**	**93.35**	87.87	**94.85**	**98.73**	**90.51**	**94.81**	**90.12**

tiveness in leveraging transferable information while mitigating distribution shifts. Notably, its improvement is most noticeable in classifying digit 8. To further investigate this, we analyze the heatmaps of learned parameters for each method in Figure 1, where red indicates positive pixel importance and blue represents negative pixel importance. *Single-Lasso*, constrained by the limited target sample size, focuses on a sparse set of pixels. *Agg-Lasso* captures more informative patterns but fails to adapt to distribution shifts, as shown by the background blue pixels indicating overfitting to injected noise. *Trans-GLM* primarily emphasizes pixels from the target dataset, neglecting transferable patterns from sources. In contrast, *TransMission* effectively extracts useful features from source data while filtering out irrelevant background noise, resulting in a more robust and generalizable model.

6. Conclusion and Future Work

We propose *TransMission*, a parameter-transfer framework for high-dimensional M-estimation. The joint regularized training process ensures effective knowledge transfer while remaining resilient to distribution shifts. A novel target-data-oriented constraint is introduced to fully exploit the target data to identify the target parameter, as well as prevent negative transfer issues. Our theoretical analysis establishes minimax-optimal statistical rates in the absence of shifts and demonstrates robustness when shifts are present. In addition, we introduce *D-TransMission*, a communication-efficient variant using only pre-estimated source parameters. Extensive experiments validate our theoretical findings.

A number of research questions can be further studied from the framework and methods developed here. First, our analysis is built upon an M-estimation setting, an important future direction is to extend the proposed framework to more general nonlinear structures, such as nonparametric models. Second, while the current theoretical results rely on standard regularity assumptions such as Restricted Strong Convexity/Smoothness (cf. Assumption 1) and the sparsity of the inverse Fisher information matrix (cf. Assumption 13), it is interesting to explore ways to relax these conditions.

Acknowledgments

The authors would like to thank the Editor and the two reviewers for their constructive comments, which led to a significant improvement of this work. The research of Zelin He and Runze Li was partially supported by an NSF grant DMS 2514400 and NIH grants

R01GM163244 and R01 AI192205. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NSF or NIH.

References

Alekh Agarwal, Sahand Negahban, and Martin J Wainwright. Fast global convergence rates of gradient methods for high-dimensional statistical recovery. *Advances in Neural Information Processing Systems*, 23, 2010.

Ruiqi Bai, Yijiao Zhang, Hanbo Yang, and Zhongyi Zhu. Transfer learning for high-dimensional quantile regression with distribution shift. *arXiv preprint arXiv:2411.19933*, 2024.

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

Peter Bühlmann and Sara Van De Geer. *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media, 2011.

Tianxi Cai, Mengyan Li, and Molei Liu. Semi-supervised triply robust inductive transfer learning. *Journal of the American Statistical Association*, 120(550):1037–1047, 2025. doi: 10.1080/01621459.2024.2393463. URL `https://doi.org/10.1080/01621459.2024.2393463`.

AW Charney, DM Ruderfer, EA Stahl, JL Moran, K Chambert, RA Belliveau, L Forty, Katherine Gordon-Smith, Arianna Di Florio, PH Lee, et al. Evidence for genetic heterogeneity between clinical subtypes of bipolar disorder. *Translational Psychiatry*, 7(1): e993–e993, 2017.

Shuxiao Chen, Qinqing Zheng, Qi Long, and Weijie J Su. Minimax estimation for personalized federated learning: An alternative between fedavg and local training? *Journal of Machine Learning Research*, 24(262):1–59, 2023.

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

Theodoros Evgeniou, Charles A Micchelli, Massimiliano Pontil, and John Shawe-Taylor. Learning multiple tasks with kernel methods. *Journal of Machine Learning Research*, 6: 615–637, 2005.

Jianqing Fan. Features of big data and sparsest solution in high confidence set. *Past, present, and future of statistical science*, pages 531–548, 2013.

Jianqing Fan and Runze Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

Jianqing Fan, Runze Li, Cun-Hui Zhang, and Hui Zou. *Statistical foundations of data science*. CRC press, 2020.

Jianqing Fan, Zhuoran Yang, and Mengxin Yu. Understanding implicit regularization in over-parameterized single index model. *Journal of the American Statistical Association*, 118(544):2315–2328, 2023.

Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010.

Yimiao Gao and Yuehan Yang. Transfer learning on stratified data: joint estimation transferred from strata. *Pattern Recognition*, 140:109535, 2023.

Trevor Hastie, Robert Tibshirani, and Martin J Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*, volume 143 of *Monographs on Statistics and Applied Probability*. CRC Press, 2015.

Zelin He, Ying Sun, and Runze Li. Transfusion: Covariate-shift robust transfer learning for high-dimensional regression. In *International Conference on Artificial Intelligence and Statistics*, pages 703–711. PMLR, 2024.

Adel Javanmard and Andrea Montanari. Confidence intervals and hypothesis testing for high-dimensional regression. *Journal of Machine Learning Research*, 15(1):2869–2909, 2014.

Clete A Kushida, Deborah A Nichols, Rik Jadrnicek, Ric Miller, James K Walsh, and Kara Griffin. Strategies for de-identification and anonymization of electronic health record data for use in multicenter research studies. *Medical Care*, 50:S82–S101, 2012.

Jason D Lee, Qiang Liu, Yuekai Sun, and Jonathan E Taylor. Communication-efficient sparse regression. *Journal of Machine Learning Research*, 18(1):115–144, 2017.

Furong Li and Huiyan Sang. Spatial homogeneity pursuit of regression coefficients for large datasets. *Journal of the American Statistical Association*, 114(527):1050–1062, 2019.

Sai Li, T Tony Cai, and Hongzhe Li. 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, 2022.

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

Molei Liu, Yi Zhang, Katherine P Liao, and Tianxi Cai. Augmented transfer regression learning with semi-non-parametric nuisance models. *Journal of Machine Learning Research*, 24(293):1–50, 2023.

Shuo Shuo Liu. Unified transfer learning in high-dimensional linear regression. In *International Conference on Artificial Intelligence and Statistics*, pages 1036–1044. PMLR, 2024.

Po-Ling Loh and Martin J Wainwright. High-dimensional regression with noisy and missing data: Provable guarantees with non-convexity. *Advances in Neural Information Processing Systems*, 24, 2011.

Po-Ling Loh and Martin J Wainwright. Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. *Advances in Neural Information Processing Systems*, 26, 2013.

Mingsheng Long, Jianmin Wang, Guiguang Ding, Sinno Jialin Pan, and Philip S Yu. Adaptation regularization: A general framework for transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 26(5):1076–1089, 2013.

Ryan Mcdonald, Mehryar Mohri, Nathan Silberman, Dan Walker, and Gideon Mann. Efficient large-scale distributed training of conditional maximum entropy models. *Advances in Neural Information Processing Systems*, 22, 2009.

Norman Mu and Justin Gilmer. Mnist-c: A robustness benchmark for computer vision. *arXiv preprint arXiv:1906.02337*, 2019.

Mats Nagel, Kyoko Watanabe, Sven Stringer, Danielle Posthuma, and Sophie Van Der Sluis. Item-level analyses reveal genetic heterogeneity in neuroticism. *Nature Communications*, 9(1):905, 2018.

Sahand N Negahban, Pradeep Ravikumar, Martin J Wainwright, and Bin Yu. A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Statistical Science*, 27(4):538–557, 2012.

Yang Ning and Han Liu. A general theory of hypothesis tests and confidence regions for sparse high dimensional models. *The Annals of Statistics*, 45(1):158–195, 2017.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2009.

Garvesh Raskutti, Martin J Wainwright, and Bin Yu. Minimax rates of estimation for high-dimensional linear regression over l_q-balls. *IEEE Transactions on Information Theory*, 57(10):6976–6994, 2011.

Mark Rudelson and Shuheng Zhou. Reconstruction from anisotropic random measurements. In *Proceedings of the 25th Annual Conference on Learning Theory*, volume 23, pages 10.1–10.24. PMLR, 2012.

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

Ye Tian, Yuqi Gu, and Yang Feng. Learning from similar linear representations: Adaptivity, minimaxity, and robustness. *Journal of Machine Learning Research*, 26(187):1–125, 2025.

Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(1):91–108, 2005.

Lisa Torrey and Jude Shavlik. Transfer learning. In *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, pages 242–264. IGI global, 2010.

Sara van de Geer, Peter Bühlmann, Ya'acov Ritov, and Ruben Dezeure. On asymptotically optimal confidence regions and tests for high-dimensional models. *The Annals of Statistics*, 42(3):1166–1202, 2014.

Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. In Yonina C Eldar and Gitta Kutyniok, editors, *Compressed Sensing: Theory and Applications*, pages 210–268. Cambridge University Press, 2012.

Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

Jingshu Wang, Divyansh Agarwal, Mo Huang, Gang Hu, Zilu Zhou, Chengzhong Ye, and Nancy R Zhang. Data denoising with transfer learning in single-cell transcriptomics. *Nature Methods*, 16(9):875–878, 2019.

Cun-Hui Zhang and Stephanie S Zhang. Confidence intervals for low dimensional parameters in high dimensional linear models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 76(1):217–242, 2014.

Wen Zhang, Lingfei Deng, Lei Zhang, and Dongrui Wu. A survey on negative transfer. *IEEE/CAA Journal of Automatica Sinica*, 10(2):305–329, 2023.

Xin Zhang, Jia Liu, and Zhengyuan Zhu. Learning coefficient heterogeneity over networks: a distributed spanning-tree-based fused-lasso regression. *Journal of the American Statistical Association*, 119(545):485–497, 2024.

Doudou Zhou, Molei Liu, Mengyan Li, and Tianxi Cai. Doubly robust augmented model accuracy transfer inference with high dimensional features. *Journal of the American Statistical Association*, 120(549):524–534, 2025.

Appendix A. Additional Methodology Details

A.1 Notations

Table A.1: Summary of notation used throughout the paper.

Notation	Description
$\mathbf{x}, \mathbf{x}_i$	Vector in $\mathbb{R}^p$ and its ith entry.
$\boldsymbol{A}, \boldsymbol{A}_i, \boldsymbol{A}_{ij}$	Matrix in $\in \mathbb{R}^{m\times n}$, its ith row, and its (i,j)th entry.
$\langle \cdot, \cdot \rangle$	Euclidean inner product.
$\|\mathbf{x}\|_q$	ℓ_q-norm: $\|\mathbf{x}\|_q = (\sum_{i=1}^p \|\mathbf{x}_i\|^q)^{1/q}$, for $q > 0$.
$\|\mathbf{x}\|_0$	ℓ_0-norm: $\|\mathbf{x}\|_0 = \#\{j : \mathbf{x}_j \neq 0\}$.
$\|\mathbf{x}\|_\infty$	ℓ_∞-norm: $\|\mathbf{x}\|_\infty = \max_{1\le j\le p} \|\mathbf{x}_j\|$.
$a \vee b$, $a \wedge b$	$\max\{a,b\}$ and $\min\{a,b\}$, respectively.
$[K]$	Index set $\{1, \ldots, K\}$.
$c, c_0, c_1, \ldots$	Generic positive constants.
$a_n = O(b_n)$, $a_n \lesssim b_n$	$\|a_n/b_n\| \le c$ for some constant c (for large n).
$a_n \asymp b_n$	$a_n = O(b_n)$ and $b_n = O(a_n)$.
$a_n = o(b_n)$, $b_n \gg a_n$	$a_n/(b_n) \to 0$ (equivalently, b_n dominates a_n).
p	Feature dimension.
n_T, n_S	Target and per-source sample sizes.
K	Number of source tasks.
$z_i^{(k)}$	A single data point from the k-th task (e.g., $z_i^{(k)} = (\mathbf{X}_i^{(k)}, \mathbf{y}_i^{(k)})$).
$\mathcal{S}^{(0)}$	Target sample, i.i.d. from the target distribution: $\mathcal{S}^{(0)} := \{z_i^{(0)}\}_{i\in[n_T]}$.
$\mathcal{S}^{(k)}$	k-th source sample, i.i.d. from the source distribution: $\mathcal{S}^{(k)} := \{z_i^{(k)}\}_{i\in[n_S]}$
$\ell^{(0)}(z^{(0)}; \boldsymbol{\beta})$	Target loss evaluated at parameter $\boldsymbol{\beta} \in \mathbb{R}^p$.
$\ell^{(k)}(z^{(k)}; \boldsymbol{\beta})$	k-th source loss evaluated at parameter $\boldsymbol{\beta} \in \mathbb{R}^p$.
$\mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)})$	Empirical loss for the target task: $\mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)}) = (2n_T)^{-1} \sum_{i=1}^{n_T} \ell^{(0)}(z_i^{(0)}; \boldsymbol{\beta}^{(0)})$.
$\mathcal{L}^{(k)}(\boldsymbol{\beta}^{(k)})$	Empirical loss for the k-th source task: $\mathcal{L}^{(k)}(\boldsymbol{\beta}^{(k)}) = (2n_S)^{-1} \sum_{i=1}^{n_S} \ell^{(k)}(z_i^{(k)}; \boldsymbol{\beta}^{(k)})$.
$\boldsymbol{\beta}_*^{(0)}$	Target parameter, $\boldsymbol{\beta}_*^{(0)} \in \arg\min_{\boldsymbol{\beta}} \ \mathbb{E}_{z^{(0)}\sim\mathbb{P}^{(0)}}[\ell^{(0)}(z^{(0)}; \boldsymbol{\beta})]$.
$\boldsymbol{\beta}_*^{(k)}$	k-th source parameter, $\boldsymbol{\beta}_*^{(k)} \in \arg\min_{\boldsymbol{\beta}} \ \mathbb{E}_{z^{(k)}\sim\mathbb{P}^{(k)}}[\ell^{(k)}(z^{(k)}; \boldsymbol{\beta})]$.
s	Sparsity level of the target parameter: $s := \|\boldsymbol{\beta}_*^{(0)}\|_0$
$\boldsymbol{\delta}^{(k)}$	Parameter contrast (source-specific signal) for task k: $\boldsymbol{\delta}^{(k)} := \boldsymbol{\beta}_*^{(k)} - \boldsymbol{\beta}_*^{(0)}$.
h	Upper bound on informativeness level: $\|\boldsymbol{\delta}^{(k)}\|_1 \le h$.
$\mathbf{X}_i^{(k)}$, $\mathbf{y}_i^{(k)}$	Covariate/response for the i-th sample in source task k.
$\boldsymbol{\Sigma}^{(k)}$	Covariance matrix of $\mathbf{X}_i^{(k)}$ in source task k.
$\boldsymbol{\epsilon}_i^{(k)}$	Noise term in linear models for task k.
$\mathbb{P}_X^{(k)}$, $\mathbb{P}_{y\|X}^{(k)}$	Marginal covariate and conditional response distributions for task k.
$\Psi_k(\cdot)$, ϕ_k	Cumulant generating function function and dispersion parameter for task k.
$\nabla^l g(\boldsymbol{\beta})$	l-th order derivative of a scalar function $g(\cdot)$ at $\boldsymbol{\beta}$.
$\nabla_j^l g(\boldsymbol{\beta})$	Univariate l-th order partial derivative w.r.t. coordinate j.
$\boldsymbol{e}_j$	The j-th canonical basis vector, whose j-th entry is 1 and all others are 0.

A.2 Practical Implementation and Tuning Parameter Selection

This subsection is dedicated to the practical implementation of *TransMission*, with a detailed guide for tuning parameter selection. For illustration purposes, we take the generalized linear model with a known and shared dispersion parameter ϕ and a log-partition function $\Phi(\cdot)$. Extensions to settings with task-specific ϕ_k and $\Phi_k(\cdot)$ follow analogously by replacing the shared quantities with their task-specific counterparts. The optimization problem consists of two components: the objective function (6a) and the constraint (6b).

For solving objective (6a), based on the one-to-one transformation $\boldsymbol{\theta} := ((\boldsymbol{\beta}^{(1)}-\boldsymbol{\beta}^{(0)})^\top, (\boldsymbol{\beta}^{(2)}-\boldsymbol{\beta}^{(0)})^\top, \ldots, (\boldsymbol{\beta}^{(K)}-\boldsymbol{\beta}^{(0)})^\top, \boldsymbol{\beta}^{(0)})$, we can reformulate the problem into the following standard regularized regression problem:

$$\hat{\boldsymbol{\theta}} \in \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \left\{ \frac{1}{N\phi} \left(-\mathbf{y}^T \mathbf{X}\boldsymbol{\theta} + \sum_{i=1}^{N} \Phi\left(\mathbf{X}_i^\top \boldsymbol{\theta}\right) \right) + \lambda_0 \sum_{k=0}^{K} a_k \|\boldsymbol{\theta}^{(k)}\|_1 \right\}, \tag{A.1}$$

where $a_0 = 1$ and $a_1 = \frac{\lambda_1}{\lambda_0}$ for $k \in [K]$, and $\mathbf{y}$ and $\mathbf{X}$ are defined as

$$\mathbf{y} := \begin{pmatrix} \mathbf{y}^{(1)} \\ \mathbf{y}^{(2)} \\ \vdots \\ \mathbf{y}^{(K)} \\ \mathbf{y}^{(0)} \end{pmatrix} \quad \mathbf{X} := \begin{pmatrix} \mathbf{X}^{(1)} & 0 & \cdots & 0 & \mathbf{X}^{(1)} \\ 0 & \mathbf{X}^{(2)} & \cdots & 0 & \mathbf{X}^{(2)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \mathbf{X}^{(K)} & \mathbf{X}^{(K)} \\ 0 & 0 & \cdots & 0 & \mathbf{X}^{(0)} \end{pmatrix}.$$

Such an objective falls into the class of ℓ_1-penalized regression problems and can therefore be efficiently solved via the standard `glmnet` library (Friedman et al., 2010). In this formulation, the tuning parameter λ_0 is selected through a three-fold cross-validation of the target training dataset, while a_k is fixed according to the theoretical ratio established in Corollary 1 with $h = 0$, namely $a_k = \frac{n_S}{\sqrt{N n_T}}$, where n_S and n_T denote the sample sizes of the source and target data, respectively. This theoretically informed procedure has shown consistently strong performance across a wide range of settings.

We next discuss the practical handling of constraint (6b). Rather than enforcing the constraint during optimization, which would make the problem substantially harder to solve, we adopt a verification-based approach.

We consider a geometric grid of λ_T values. For each λ_T, we compute a single-task reference estimator

$$\hat{\boldsymbol{\beta}}_T^{(0)}(\lambda_T) \in \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \mathcal{L}^{(0)}(\boldsymbol{\beta}) + \lambda_T \|\boldsymbol{\beta}\|_1 \right\}, \tag{A.2}$$

which yields the reference ℓ_1 norm $\|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1$ and sparsity estimate $\hat{s} = \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_0$. Following Theorem 3, which prescribes $\kappa_T \propto s\lambda_T$, we set $\kappa_T = \hat{s}\lambda_T$ as a practical plug-in, replacing the unknown true sparsity s with the estimated $\hat{s}$ and absorbing the proportionality constant. Together, λ_T and κ_T define the constraint set $\mathcal{C}_T(\lambda_T)$.

For each λ_T, we solve the unconstrained weighted-Lasso problem (A.1) over a grid of λ_0 values via three-fold cross-validation on the target data, obtaining candidate estimators

$\{\hat{\boldsymbol{\beta}}(\lambda_0)\}$ together with their validation losses. For each pair (λ_0, λ_T), we verify whether $\hat{\boldsymbol{\beta}}(\lambda_0) \in \mathcal{C}_T(\lambda_T)$, i.e., whether

$$\|\nabla \mathcal{L}_0(\hat{\boldsymbol{\beta}}(\lambda_0))\|_\infty \le \lambda_T \quad \text{and} \quad \|\hat{\boldsymbol{\beta}}(\lambda_0)\|_1 \le \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1 + \hat{s}\lambda_T.$$

Among all feasible pairs, we select the one with the lowest target validation loss. If no feasible pair exists, we fall back to the single-task estimator.

In summary, *TransMission* involves two tuning parameters, λ_0 and λ_T, jointly selected by TransMission validation performance subject to the solution lying in $\mathcal{C}_T(\lambda_T)$. This procedure consistently delivers strong performance across a wide range of settings. An open-source implementation is available at `https://github.com/ZLHe0/TransMission`.

A.3 Choice of Tuning Parameters in Corollary 9

Recall that we define a pooling estimator $\hat{\boldsymbol{\beta}}_{\text{Pooling}}$ in (4) and its population counterpart as $\boldsymbol{\beta}_{\text{Pooling}}$ in (9). We also define a factor C_Σ in (10) that measures the distribution shifts across sources. To establish a faster convergence rate compared with (8), we adopt the following choice of tuning parameters:

$$\lambda_0 \asymp \left[\left(\frac{h^2 \log p}{s^2 n_T}\right)^{1/4} + \left(\frac{\log p}{N}\right)^{1/2}\right], \quad \lambda_1 \asymp \frac{n_S}{N}\left(\frac{\log p}{n_T}\right)^{1/2}, \quad \text{and} \quad \lambda_T \asymp \left(\frac{\log p}{n_T}\right)^{1/2}, \tag{A.3}$$

if $h \le (C_\Sigma^{-2}\sqrt{\log p/n_T}) \wedge (C_\Sigma^{-1}\sqrt{s \log p/n_T})$ and

$$\lambda_0 \asymp \sqrt{\frac{\log p}{N}}, \lambda_1 \gtrsim \left\|\frac{n_S}{N}\nabla \mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{\text{Pooling}})\right\|_\infty, \tag{A.4}$$

otherwise. To understand the rationale behind this parameter selection, note that under (A.4), λ_1 is chosen sufficiently large to shrink all $\hat{\boldsymbol{\beta}}^{(k)}$ toward $\hat{\boldsymbol{\beta}}^{(0)}$. As a result, the TransMission estimator $\hat{\boldsymbol{\beta}}^{(0)}$ effectively reduces to the pooled estimator $\hat{\boldsymbol{\beta}}_{\text{Pooling}}$, which is optimal in homogeneous settings where source and target distributions are nearly identical. In other cases, the parameters follow the original choices in Corollary 5, as specified in (A.3).

As the condition (A.4) depends on the estimator $\hat{\boldsymbol{\beta}}_{\text{Pooling}}$, in the following, we further establish a sufficient condition for (A.4) to hold. The result requires an additional ℓ_∞ curvature assumption, which is similar to the RSM condition but is measured in terms of ℓ_∞ norm. Such an assumption also holds for common regression models (Wainwright, 2019).

Lemma 12 *Under the assumption of Corollary 9, if we in addition assume that for any $k \in [K]$, $\boldsymbol{\beta}, \boldsymbol{\Delta} \in \mathbb{R}^p$, there exists constants $\eta_k, \zeta_k \ge 0$ such that*

$$\|\nabla \mathcal{L}^{(k)}(\boldsymbol{\beta} + \boldsymbol{\Delta}) - \nabla \mathcal{L}^{(k)}(\boldsymbol{\beta})\|_\infty \le \eta_k \|\boldsymbol{\Delta}\|_\infty + \zeta_k \sqrt{\frac{\log p}{n_S}} \|\boldsymbol{\Delta}\|_1,$$

with probability larger than $1 - c_1 \exp(-c_2 \log p)$. Then a sufficient condition for $\lambda_1 \gtrsim \|\frac{n_S}{N}\nabla \mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{Pooling})\|_\infty$ is

$$\lambda_1 \gtrsim \frac{n_S}{N}\left[K_P + \sqrt{\frac{\log p}{N}} + \sqrt{\frac{\log p}{n_S}}\left(s\sqrt{\frac{\log p}{N}} + C_\Sigma h\right)\right],$$

where $K_P = \max_{k\in[K]} \|\boldsymbol{\beta}_{Pooling} - \boldsymbol{\beta}_*^{(k)}\|_\infty$ *is the maximum elementwise difference between* $\boldsymbol{\beta}_{Pooling}$ *and the true source parameter* $\boldsymbol{\beta}_*^{(k)}$.

A.4 Additional Details On D-TransMission Method

In this paper, we construct $\hat{\boldsymbol{\Theta}}^{(k)}$ to be the de-sparsified lasso proposed in van de Geer et al. (2014). Following their framework, we limit our discussion to a regression framework with covariables $\mathbf{X}_i^{(k)} \in \mathcal{X}^{(k)} \subseteq \mathbb{R}^p$ and univariate responses $\mathbf{y}_i^{(k)} \in \mathcal{Y}^{(k)} \subseteq \mathbb{R}$ for $i = 1, \ldots, n_S$, and the source task loss function has the form

$$\ell^{(k)}(\boldsymbol{\beta}; z_i^{(k)}) = \rho^{(k)}((\mathbf{X}_i^{(k)})^\top \boldsymbol{\beta}, \mathbf{y}_i^{(k)}).$$

We assume ρ is locally Lipschitz with well-defined second-order derivative $\ddot{\rho}^{(k)}(a,b) = \frac{d^2}{da^2}\rho(a,b)$. Many commonly used regression models, such as the generalized linear regression discussed in Example 2, fit within this framework. Within this setup, the problem could be formulated into a linear regression problem with the weighted design matrix $\mathbf{X}_{\hat{\boldsymbol{\beta}}_L^{(k)}} := \boldsymbol{W}_{\hat{\boldsymbol{\beta}}_L^{(k)}} \mathbf{X}^{(k)}$, where $\boldsymbol{W}_{\hat{\boldsymbol{\beta}}_L^{(k)}}$ is diagonal and its diagonal entries defined as $(\boldsymbol{W}_{\hat{\boldsymbol{\beta}}_L^{(k)}})_{i,i} := \ddot{\rho}^{(k)}((\mathbf{X}_i^{(k)})^\top \hat{\boldsymbol{\beta}}_L^{(k)}, \mathbf{y}_i^{(k)})^{\frac{1}{2}}$. van de Geer et al. (2014) proposes to estimate $\hat{\boldsymbol{\Theta}}^{(k)}$ by constructing its j-th row as the solution to the following nodewise-regression problem:

$$\hat{\boldsymbol{\Theta}}_{jk}^{(k)} = -\hat{\tau}_j^{-2}\hat{\boldsymbol{\gamma}}_{jk}, \hat{\boldsymbol{\Theta}}_{jj}^{(k)} = 1 \text{ with } \hat{\boldsymbol{\gamma}}_j^{(k)} := \underset{\boldsymbol{\gamma}\in\mathbb{R}^{p-1}}{\operatorname{argmin}} \frac{1}{2n_S} \left\| \mathbf{X}^{(k)}_{\hat{\boldsymbol{\beta}}_L^{(k)},j} - \mathbf{X}^{(k)}_{\hat{\boldsymbol{\beta}}_L^{(k)},-j}\boldsymbol{\gamma} \right\|_2^2 + \mu_{jk}\|\boldsymbol{\gamma}\|_1,$$

$$\text{and } \hat{\tau}_j := \left(\frac{1}{2n_S} \left\| \mathbf{X}^{(k)}_{\hat{\boldsymbol{\beta}}_L^{(k)},j} - \mathbf{X}^{(k)}_{\hat{\boldsymbol{\beta}}_L^{(k)},-j}\hat{\boldsymbol{\gamma}}_j^{(k)} \right\|_2^2 + \mu_{jk}\|\hat{\boldsymbol{\gamma}}_j^{(k)}\|_1 \right)^{\frac{1}{2}}. \tag{A.5}$$

with μ_{jk} being the tuning parameters for $j \in [p]$.

To establish the theoretical results, we introduce the following regularity conditions.

Assumption 13 *(D1) The pairs of random variables* $\left\{ \left(\mathbf{y}_i^{(k)}, \mathbf{X}_i^{(k)} \right) \right\}_{i\in[n_S]}$ *are i.i.d. and* $\|\mathbf{X}^{(k)}\|_\infty = \max_{i,j} \left| \mathbf{X}_{i,j}^{(k)} \right| = \mathcal{O}(1)$ *and* $\left\| \mathbf{X}^{(k)}_{\boldsymbol{\beta}_*^{(k)},-j} \boldsymbol{\gamma}^{(k)}_{\boldsymbol{\beta}_*^{(k)},j} \right\|_\infty = \mathcal{O}(1)$, *where*

$$\boldsymbol{\gamma}_{\boldsymbol{\beta}_*^{(k)},j} := \underset{\boldsymbol{\gamma}\in\mathbb{R}^{p-1}}{\arg\min}\ \mathbb{E}\left[\left\| \mathbf{X}_{\boldsymbol{\beta}_*^{(k)},j} - \mathbf{X}_{\boldsymbol{\beta}_*^{(k)},-j}\boldsymbol{\gamma} \right\|_2^2 \right].$$

(D2) Denote $s_j^{(k)} = \|\mathbb{E}[(\nabla^2 \mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)}))_j^{-1}]\|_0$. *It holds that* $s_j^{(k)}\sqrt{\log p / n_S} = o(1)$.

(D3) Define $\boldsymbol{\Sigma}^{(k)}_{\boldsymbol{\beta}_*^{(k)}} := \frac{1}{n_S}\mathbb{E}[(\mathbf{X}^{(k)}_{\boldsymbol{\beta}_*^{(k)}})^\top \mathbf{X}^{(k)}_{\boldsymbol{\beta}_*^{(k)}}]$. *The smallest and largest eigenvalue of* $\boldsymbol{\Sigma}^{(k)}_{\boldsymbol{\beta}_*^{(k)}}$ *is bounded uniformly over* $k \in [K]$.

(D4) For some $\boldsymbol{\delta} > 0$ *and all* $\left\| \boldsymbol{\beta} - \boldsymbol{\beta}_*^{(k)} \right\|_1 \le \boldsymbol{\delta}$, *it holds that* $\boldsymbol{W}_{\boldsymbol{\beta}}$ *stays away from zero and that* $\|\boldsymbol{W}_{\boldsymbol{\beta}}\|_\infty = \mathcal{O}(1)$. *We further require that for all such* $\boldsymbol{\beta}$ *and all* $\mathbf{x}$ *and* $\mathbf{y}$,

$$\left| \nabla^2 \rho^{(k)}(\mathbf{x}\boldsymbol{\beta}, \mathbf{y}) - \nabla^2 \rho^{(k)}(\mathbf{x}\boldsymbol{\beta}_*^{(k)}, \mathbf{y}) \right| \le \left| \mathbf{x}^\top \left(\boldsymbol{\beta}_*^{(k)} - \boldsymbol{\beta} \right) \right|.$$

A.5 Heterogeneous Binomial Regression: Overdispersion Setup

To introduce overdispersion in the binomial regression tasks, we model each source task using a Beta-Binomial framework. Instead of using a fixed probability parameter p_k determined directly by the logistic function, we assume that p_k follows a Beta distribution with task-specific shape parameters, introducing additional variability across tasks. For the k-th source task, we first compute a baseline linear predictor η_{ki} using the standard logistic regression model with $\eta_{ki} = (\mathbf{X}_i^{(k)})^\top \boldsymbol{\beta}^{(k)}$. In the following for notational simplicity we drop the subscript i. Then, instead of defining the probability parameter as $p_k = (1 + e^{-\eta_k})^{-1}$, we generate p_k from a Beta distribution with parameters α_k and β_k, ensuring that the mean of p_k aligns with the logistic function while its variance varies across tasks. Specifically, we define $\alpha_k = \alpha_0$ and $\beta_k \sim U(\beta_{\min}, \beta_{\max})$. The shape parameters of the Beta distribution are then computed as:

$$\alpha_{\text{source},k} = \alpha_k \cdot \frac{e^{\eta_k}}{1 + e^{\eta_k}}, \quad \beta_{\text{source},k} = \beta_k \cdot \left(1 - \frac{e^{\eta_k}}{1 + e^{\eta_k}}\right).$$

Given the Beta distribution parameters, the probability p_k for each sample in the k-th source task is drawn as:

$$p_k \sim \text{Beta}(\alpha_{\text{source},k}, \beta_{\text{source},k}).$$

Finally, the binary response y_k is sampled from a Bernoulli distribution with success probability p_k:

$$y_k \sim \text{Bernoulli}(p_k).$$

This formulation allows the mean of p_k to remain close to the target probability, while the variance of p_k changes across tasks.

Appendix B. Proof of Theorems and Propositions

In this section, we provide proof of all the theorems and propositions. Throughout the section, we adopt the following notations to analyze the solution of the problem (6):

$$\boldsymbol{\theta}_* = \begin{pmatrix} \boldsymbol{\theta}_*^{(1)} \\ \boldsymbol{\theta}_*^{(2)} \\ \vdots \\ \boldsymbol{\theta}_*^{(K)} \\ \boldsymbol{\theta}_*^{(0)} \end{pmatrix} := \begin{pmatrix} \boldsymbol{\beta}_*^{(1)} - \boldsymbol{\beta}_*^{(0)} \\ \boldsymbol{\beta}_*^{(2)} - \boldsymbol{\beta}_*^{(0)} \\ \vdots \\ \boldsymbol{\beta}_*^{(K)} - \boldsymbol{\beta}_*^{(0)} \\ \boldsymbol{\beta}_*^{(0)} \end{pmatrix}, \tag{A.6}$$

where $\boldsymbol{\beta}_*^{(k)}$s are the true source parameters defined in (2) and $\boldsymbol{\beta}_*^{(0)}$ is the true target parameter defined in (1). Under this transformation, solving problem (6) is equivalent as solving

$$\hat{\boldsymbol{\theta}} \in \underset{\boldsymbol{\theta}}{\text{argmin}}\{\mathcal{L}(\boldsymbol{\theta}) + \lambda_0 \mathcal{R}(\boldsymbol{\theta})\}, \quad \text{subject to} \quad \boldsymbol{\theta}^{(0)} \in \mathcal{C}_T. \tag{A.7}$$

where we define the objective function as $\mathcal{L}(\boldsymbol{\theta}) = \frac{n_T}{N}\mathcal{L}^{(0)}(\boldsymbol{\theta}^{(0)}) + \sum_{k=1}^K \frac{n_S}{N}\mathcal{L}^{(k)}(\boldsymbol{\theta}^{(0)} + \boldsymbol{\theta}^{(k)})$, with the corresponding regularization term $\mathcal{R}(\boldsymbol{\theta}) = \|\boldsymbol{\theta}^{(0)}\|_1 + \sum_{k=1}^K \frac{\lambda_1}{\lambda_0}\|\boldsymbol{\theta}^{(k)}\|_1$. Since there

exists a one-to-one mapping between $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$, the solution to the regularized M-estimation problem in (A.7), $\hat{\boldsymbol{\theta}}$, can be transformed into the solution for the original problem (6) without ambiguity. In the subsequent analysis, we first establish the theoretical properties of $\hat{\boldsymbol{\theta}}$ and then use this mapping to derive corresponding results for $\hat{\boldsymbol{\beta}}$.

B.1 Proof of Theorem 3

We begin by introducing some notations. Let S denote the support set of the target coefficient $\boldsymbol{\beta}_*^{(0)}$, and let S^c be its complement. With a slight abuse of notation, we define $\boldsymbol{\theta}_S^{(k)}$ as the sub-vector of $\boldsymbol{\theta}_*^{(k)}$ indexed by S. Throughout the proof, we assume $n_k = n_S$ for $k = 1, \ldots, K$ and $n_k = n_T$ for $k = 0$. Let $\hat{\boldsymbol{\Delta}} := \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_*$ be the estimation error vector, with its k-th block denoted as $\hat{\boldsymbol{\Delta}}^{(k)} := \hat{\boldsymbol{\theta}}^{(k)} - \boldsymbol{\theta}_*^{(k)}$. Our goal is to establish an upper bound for $\|\hat{\boldsymbol{\Delta}}^{(0)}\|_2^2$, which corresponds to the ℓ_2 estimation error of the target parameter $\boldsymbol{\theta}_*^{(0)} = \boldsymbol{\beta}_*^{(0)}$.

The proof of Theorem 3 relies on three key technical lemmas, whose proofs are provided in Appendix B.5. The first lemma establishes an upper bound for the first-order term of the Taylor series expansion of $\mathcal{L}(\boldsymbol{\theta}_*)$.

Lemma 14 *Under Assumption 1 we choose λ_0 and λ_1 such that $\lambda_0 \geq 2 \left\| \sum_{k=0}^{K} \frac{n_k}{N} \nabla \mathcal{L}^{(k)}(\boldsymbol{\theta}_*^{(k)}) \right\|_\infty$ and $\lambda_1 \geq 2 \max_{k \in [K]} \left\| \frac{n_S}{N} \nabla \mathcal{L}^{(k)}(\boldsymbol{\theta}_*^{(k)}) \right\|_\infty$, we then have for any*

$$\boldsymbol{\Delta} = \left(\left(\boldsymbol{\Delta}^{(1)} \right)^\top, \ldots, \left(\boldsymbol{\Delta}^{(K)} \right)^\top, \left(\boldsymbol{\Delta}^{(0)} \right)^\top \right)^\top \in \mathbb{R}^{(K+1)p},$$

we have

$$\left| \left\langle \nabla \mathcal{L}\left(\boldsymbol{\theta}_*\right), \boldsymbol{\Delta} \right\rangle \right| \leq \sum_{k=1}^{K} \frac{\lambda_1}{2} \left\| \boldsymbol{\Delta}^{(k)} \right\|_1 + \frac{\lambda_0}{2} \left\| \boldsymbol{\Delta}^{(0)} \right\|_1.$$

The next lemma establishes a restricted set of directions in which $\hat{\boldsymbol{\Delta}}$ lies and the feasibility of the solution $\boldsymbol{\theta}_*$.

Lemma 15 *Under the conditions of Lemma 14 and assume $n_T > 64 \left(\tau_0/\alpha_0\right)^2 s \log p$. If we choose $\lambda_T \geq 2\|\nabla \mathcal{L}^{(0)}(\boldsymbol{\theta}_*^{(0)})\|_\infty$, $\kappa_T = 24 s \lambda_T / \alpha_0$, then $\boldsymbol{\theta}_*$ is feasible to problem (6) and*

$$\sum_{k=1}^{K} \lambda_1 \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1 + \lambda_0 \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1 \leq 4\lambda_0 \left\| \hat{\boldsymbol{\Delta}}_S^{(0)} \right\|_1 + 4 \sum_{k=1}^{K} \lambda_1 h. \tag{A.8}$$

The following lemma ensures a property analogous to restricted strong convexity for $\hat{\boldsymbol{\Delta}}$.

Lemma 16 *Under the conditions of Lemma 15, if $n_S > n_T$, the estimation error $\hat{\boldsymbol{\Delta}}$ satisfies*

$$\begin{aligned} &\mathcal{L}\left(\boldsymbol{\theta}_* + \hat{\boldsymbol{\Delta}}\right) - \mathcal{L}\left(\boldsymbol{\theta}_*\right) - \left\langle \nabla \mathcal{L}\left(\boldsymbol{\theta}_*\right), \hat{\boldsymbol{\Delta}} \right\rangle \\ &\geq \left(\frac{\alpha_{\min}^2}{\gamma_0} - u_n \right) \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_2^2 + \frac{\alpha_{\min}^2}{\gamma_0} \sum_{k=1}^{K} \frac{n_k}{N} \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_2^2 - \frac{2\alpha_{\max}}{\gamma_0} \sum_{k=1}^{K} \frac{n_k}{N} \lambda_T \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1 - v_n \sum_{k=1}^{K} \lambda_1 h, \end{aligned} \tag{A.9}$$

where

$$u_n = \frac{64(\alpha_{\max}\tau_0 + \beta_{\max}\gamma_0)}{\gamma_0}\frac{n_S}{n_T}\frac{\log p}{N}\cdot\frac{\lambda_0^2 s}{\lambda_1^2 \wedge [(n_S/N)\lambda_0^2]},$$
$$v_n = \frac{64(\alpha_{\max}\tau_0 + \beta_{\max}\gamma_0)}{\gamma_0}\frac{n_S}{n_T}\frac{\log p}{N}\cdot\frac{\sum_{k=1}^K \lambda_1 h}{\lambda_1^2 \wedge [(n_S/N)\lambda_0^2]},$$
$$\alpha_{\min} = \min_{0\le k\le K}\alpha_k, \quad \alpha_{\max} = \max_{0\le k\le K}\alpha_k \text{ and } \quad \beta_{\max} = \max_{0\le k\le K}\beta_k,$$

with RSC constants (α_k, τ_k) *and RSM constants* (β_k, γ_k) *defined in Assumption 1.*

We now proceed to the proof of the theorem. For notational convenience, we first define a optimal value gap function function F as follows:

$$F(\boldsymbol{\Delta}) = \mathcal{L}\left(\boldsymbol{\theta}_* + \boldsymbol{\Delta}\right) - \mathcal{L}\left(\boldsymbol{\theta}_*\right) + \lambda_0\mathcal{R}\left(\boldsymbol{\theta}_* + \boldsymbol{\Delta}\right) - \lambda_0\mathcal{R}\left(\boldsymbol{\theta}_*\right),$$

where $\boldsymbol{\theta}_*$ is the transformed model parameter defined in (A.6), and

$$\boldsymbol{\Delta} = \left(\left(\boldsymbol{\Delta}^{(1)}\right)^\top, \ldots, \left(\boldsymbol{\Delta}^{(K)}\right)^\top, \left(\boldsymbol{\Delta}^{(0)}\right)^\top\right)^\top \in \mathbb{R}^{(K+1)p}.$$

By Hölder's inequality and mean value theorem, we have

$$\begin{aligned}
F(\hat{\boldsymbol{\Delta}}) =& \mathcal{L}\left(\boldsymbol{\theta}_* + \hat{\boldsymbol{\Delta}}\right) - \mathcal{L}\left(\boldsymbol{\theta}_*\right) + \lambda_0\mathcal{R}\left(\boldsymbol{\theta}_* + \hat{\boldsymbol{\Delta}}\right) - \lambda_0\mathcal{R}\left(\boldsymbol{\theta}_*\right) \\
\ge& -\left\|\nabla\mathcal{L}\left(\boldsymbol{\theta}_*\right)\right\|_\infty\left\|\hat{\boldsymbol{\Delta}}\right\|_1 + \hat{\boldsymbol{\Delta}}^\top\nabla^2\mathcal{L}\left(\boldsymbol{\theta}_* + \phi\hat{\boldsymbol{\Delta}}\right)\hat{\boldsymbol{\Delta}} \quad (\phi\in(0,1)) \\
&+ \sum_{k=1}^K \lambda_1\left(\left\|\boldsymbol{\theta}^{(k)} + \hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - \left\|\boldsymbol{\theta}^{(k)}\right\|_1\right) + \lambda_0\left(\left\|\boldsymbol{\theta}_*^{(0)} + \hat{\boldsymbol{\Delta}}^{(0)}\right\|_1 - \left\|\boldsymbol{\theta}_*^{(0)}\right\|_1\right).
\end{aligned}$$

Notice that here we implicitly assume the existence of second-order derivatives; however, the proof remains valid without this assumption, albeit requiring lengthier arguments. Applying Lemma 14, the triangle inequality, and the fact that $\|\boldsymbol{\theta}_{S^c}^{(0)}\|_1 = 0$, $\|\boldsymbol{\theta}_*^{(k)}\|_1 = \|\boldsymbol{\beta}_*^{(k)} - \boldsymbol{\beta}_*^{(0)}\|_1 \le h$ for $1\le k\le K$, we have for some $\phi\in[0,1]$,

$$\begin{aligned}
F(\hat{\boldsymbol{\Delta}}) \ge& -\sum_{k=1}^K\frac{\lambda_1}{2}\left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - \frac{\lambda_0}{2}\left\|\hat{\boldsymbol{\Delta}}^{(0)}\right\|_1 + \hat{\boldsymbol{\Delta}}^\top\nabla^2\mathcal{L}\left(\boldsymbol{\theta}_* + \phi\hat{\boldsymbol{\Delta}}\right)\hat{\boldsymbol{\Delta}} \\
&+ \sum_{k=1}^K\lambda_1\left(\left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - 2\left\|\boldsymbol{\theta}_*^{(k)}\right\|_1\right) \\
&+ \lambda_0\left(\left\|\boldsymbol{\theta}_S^{(0)}\right\|_1 - \left\|\hat{\boldsymbol{\Delta}}_S^{(0)}\right\|_1 + \left\|\hat{\boldsymbol{\Delta}}_{S^c}^{(0)}\right\|_1 - \left\|\boldsymbol{\theta}_{S^c}^{(0)}\right\|_1 - \left\|\boldsymbol{\theta}_S^{(0)}\right\|_1 - \left\|\boldsymbol{\theta}_{S^c}^{(0)}\right\|_1\right) \\
\ge& \hat{\boldsymbol{\Delta}}^\top\nabla^2\mathcal{L}\left(\boldsymbol{\theta}_* + \phi\hat{\boldsymbol{\Delta}}\right)\hat{\boldsymbol{\Delta}} + \frac{\lambda_0}{2}\left(\left\|\hat{\boldsymbol{\Delta}}_{S^c}^{(0)}\right\|_1 - 3\left\|\hat{\boldsymbol{\Delta}}_S^{(0)}\right\|_1\right) + \sum_{k=1}^K\frac{\lambda_1}{2}\left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - 2\sum_{k=1}^K\lambda_1 h.
\end{aligned}$$

With Lemma 16, we can obtain a lower bound of the quadratic term $\hat{\boldsymbol{\Delta}}^\top \nabla^2 \mathcal{L}\left(\boldsymbol{\theta}_* + \phi\hat{\boldsymbol{\Delta}}\right)\hat{\boldsymbol{\Delta}}$. Plug in the lower bound, we then obtain

$$\begin{aligned}
F(\hat{\boldsymbol{\Delta}}) &\geq \left(\frac{\alpha_{\min}^2}{\gamma_0} - u_n\right)\left\|\hat{\boldsymbol{\Delta}}^{(0)}\right\|_2^2 + \frac{\alpha_{\min}^2}{\gamma_0}\sum_{k=1}^K \frac{n_S}{N}\left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_2^2 - \frac{2\alpha_{\max}}{\gamma_0}\sum_{k=1}^K \frac{n_S}{N}\lambda_T\left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - v_n\sum_{k=1}^K \lambda_1 h \\
&\quad + \frac{\lambda_0}{2}\left(\left\|\hat{\boldsymbol{\Delta}}_{S^c}^{(0)}\right\|_1 - 3\left\|\hat{\boldsymbol{\Delta}}_S^{(0)}\right\|_1\right) + \sum_{k=1}^K \frac{\lambda_1}{2}\left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - 2\sum_{k=1}^K \lambda_1 h \\
&\geq \left(\frac{\alpha_{\min}^2}{\gamma_0} - u_n\right)\left\|\hat{\boldsymbol{\Delta}}^{(0)}\right\|_2^2 - \frac{3\sqrt{s}\lambda_0}{2}\left\|\hat{\boldsymbol{\Delta}}^{(0)}\right\|_2 \\
&\quad + \left(\sum_{k=1}^K \frac{\lambda_1}{2} - \frac{2\alpha_{\max}}{\gamma_0}\sum_{k=1}^K \frac{n_S}{N}\lambda_T\right)\left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - (2+v_n)\sum_{k=1}^K \lambda_1 h. \qquad \text{(A.10)}
\end{aligned}$$

Recall that we choose $\lambda_1 \geq 4\frac{\alpha_{\max}}{K\gamma_0}\lambda_T \geq 4\frac{\alpha_{\max}}{\gamma_0}\frac{n_S}{N}\lambda_T$. From Lemma 15, we have $\boldsymbol{\theta}_*$ is feasible to problem (6a). Since $\hat{\boldsymbol{\theta}}$ is the solution to the problem (A.7), then we have $F(\hat{\boldsymbol{\Delta}}) \leq F(\mathbf{0}) = 0$. These results, combining with (A.10), lead to

$$0 \geq \left(\frac{\alpha_{\min}^2}{\gamma_0} - u_n\right)\left\|\hat{\boldsymbol{\Delta}}^{(0)}\right\|_2^2 - \frac{3\sqrt{s}\lambda_0}{2}\left\|\hat{\boldsymbol{\Delta}}^{(0)}\right\|_2 - (2+v_n)\sum_{k=1}^K \lambda_1 h, \qquad \text{(A.11)}$$

which is an inequality quadratic in $\|\hat{\boldsymbol{\Delta}}^{(0)}\|_2$, we can subsequently establish the convergence rate as

$$\|\hat{\boldsymbol{\Delta}}^{(0)}\|_2^2 \leq \frac{1}{(\alpha_{\min}^2/\gamma_0 - u_n)^2}\left(\frac{9}{4}s\lambda_0^2 + 2(\alpha_{\min}^2/\gamma_0 - u_n)(2+v_n)\sum_{k=1}^K \lambda_1 h\right). \qquad \text{(A.12)}$$

Recall the definition of u_n and v_n in Lemma 16. Under the condition

$$n_S > n_T > \left[\frac{C_0(2\lambda_0^2 s + K\lambda_1 h/\gamma_0)}{(K\lambda_1^2)\wedge\lambda_0^2} + \frac{64\tau_0^2 s}{\alpha_0^2}\right]\log p,$$

imposed in Theorem 3, we have $u_n \leq \alpha_{\min}^2/(2\gamma_0)$ and $v_n \leq 1$. Therefore bound (A.12) leads to the first term of the bound in (7).

To finish the proof of Theorem 3, it remains to show the second term in the bound (7). Recall that we impose the constraint

$$\mathcal{C}_T = \{\boldsymbol{\beta}^{(0)} \in \mathbb{R}^p : \|\nabla\mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)})\|_\infty \leq \lambda_T, \|\boldsymbol{\beta}^{(0)}\|_1 \leq \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1 + \kappa_T\}, \qquad \text{(A.13)}$$

on the solution of problem (6) with

$$\hat{\boldsymbol{\beta}}_T^{(0)} = \operatorname*{argmin}_{\boldsymbol{\beta}\in\mathbb{R}^p}\left\{\mathcal{L}^{(0)}(\boldsymbol{\beta}) + \lambda_T\|\boldsymbol{\beta}\|_1\right\}.$$

Notice that $\boldsymbol{\beta}_{S^c}^{(0)} = \mathbf{0}$ and $\|\boldsymbol{\beta}_*^{(0)}\|_1 = \|\boldsymbol{\beta}_S^{(0)}\|_1$ (for simplicity, we omit the subscript $*$ in $\boldsymbol{\beta}_S^{(0)}$). From the optimality condition of (6), we have $\hat{\boldsymbol{\beta}}^{(0)} \in \mathcal{C}_T$ and thus $\|\hat{\boldsymbol{\beta}}^{(0)}\|_1 \leq \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1 + \kappa_T$.

Then with triangle inequality we can show that

$$\begin{aligned}\|\boldsymbol{\beta}_*^{(0)}\|_1 &\geq \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1 - \|\boldsymbol{\beta}_*^{(0)} - \hat{\boldsymbol{\beta}}_T^{(0)}\|_1 \geq \|\hat{\boldsymbol{\beta}}^{(0)}\|_1 - \|\boldsymbol{\beta}_*^{(0)} - \hat{\boldsymbol{\beta}}_T^{(0)}\|_1 - \kappa_T \\ &\geq \left(\|\hat{\boldsymbol{\beta}}_S^{(0)}\|_1 + \|\hat{\boldsymbol{\beta}}_{S^c}^{(0)} - \boldsymbol{\beta}_{S^c}^{(0)}\|_1\right) - \|\boldsymbol{\beta}_*^{(k)} - \hat{\boldsymbol{\beta}}_T^{(0)}\|_1 - \kappa_T \\ &\geq \left[\left(\|\boldsymbol{\beta}_*^{(0)}\|_1 - \|\hat{\boldsymbol{\beta}}_S^{(0)} - \boldsymbol{\beta}_S^{(0)}\|_1\right) + \|\hat{\boldsymbol{\beta}}_{S^c}^{(0)} - \boldsymbol{\beta}_{S^c}^{(0)}\|_1\right] - \|\boldsymbol{\beta}_*^{(0)} - \hat{\boldsymbol{\beta}}_T^{(0)}\|_1 - \kappa_T \\ &= \left[\left(\|\boldsymbol{\beta}_*^{(0)}\|_1 - \|\hat{\boldsymbol{\Delta}}_S^{(0)}\|_1\right) + \|\hat{\boldsymbol{\Delta}}_{S^c}^{(0)}\|_1\right] - \|\boldsymbol{\beta}_*^{(0)} - \hat{\boldsymbol{\beta}}_T^{(0)}\|_1 - \kappa_T,\end{aligned}$$

where recall that we denote $\hat{\boldsymbol{\Delta}}^{(0)} = \hat{\boldsymbol{\theta}}^{(0)} - \boldsymbol{\theta}_*^{(0)} = \hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}_*^{(0)}$.

According to (A.27) in the proof of Lemma 15, we have $\|\boldsymbol{\beta}_*^{(0)} - \hat{\boldsymbol{\beta}}_T^{(0)}\|_1 \leq \kappa_T$. Reorganizing the terms and applying Cauchy–Schwarz inequality yields

$$\begin{aligned}\|\hat{\boldsymbol{\Delta}}^{(0)}\|_1 &= \|\hat{\boldsymbol{\Delta}}_S^{(0)}\|_1 + \|\hat{\boldsymbol{\Delta}}_{S^c}^{(0)}\|_1 \\ &\leq 2\|\hat{\boldsymbol{\Delta}}_S^{(0)}\|_1 + \|\boldsymbol{\beta}_*^{(0)} - \hat{\boldsymbol{\beta}}_T^{(0)}\|_1 + \kappa_T \\ &\leq 2\sqrt{s}\|\hat{\boldsymbol{\Delta}}_S^{(0)}\|_2 + 2\kappa_T \\ &\leq 2\sqrt{s}\|\hat{\boldsymbol{\Delta}}^{(0)}\|_2 + 2\kappa_T, \end{aligned} \tag{A.14}$$

where recall $s := |S|$ is the cardinality of set S.

Recall that we choose $\lambda_T \geq 2\|\nabla\mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)})\|_\infty$ and have shown that $\boldsymbol{\beta}_*^{(0)} \in \mathcal{C}_T$. Hence, we have

$$\|\nabla\mathcal{L}^{(0)}(\hat{\boldsymbol{\beta}}^{(0)}) - \nabla\mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)})\|_\infty \leq \|\langle\nabla\mathcal{L}^{(0)}(\hat{\boldsymbol{\beta}}^{(0)})\|_\infty + \|\langle\nabla\mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)})\|_\infty \leq 2\lambda_T,$$

which together with Holder's inequality implies

$$\langle\nabla\mathcal{L}^{(0)}(\hat{\boldsymbol{\beta}}^{(0)}) - \nabla\mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)}), \hat{\boldsymbol{\Delta}}^{(0)}\rangle \leq 2\lambda_T\|\hat{\boldsymbol{\Delta}}^{(0)}\|_1.$$

Applying the RSC condition in Assumption 1 to the left-hand side of the inequality leads to

$$\alpha_0\|\hat{\boldsymbol{\Delta}}^{(0)}\|_2^2 - \tau_0\frac{\log p}{n_T}\|\hat{\boldsymbol{\Delta}}^{(0)}\|_1^2 \leq 2\lambda_T\|\hat{\boldsymbol{\Delta}}^{(0)}\|_1.$$

Now plugging in the results in (A.14), the choice of κ_T and applying the ℓ_1 error bound in (A.26), we can show that

$$\left(\alpha_0 - u_n'\right)\|\hat{\boldsymbol{\Delta}}^{(0)}\|_2^2 - 4\sqrt{s}\lambda_T\|\hat{\boldsymbol{\Delta}}^{(0)}\|_2 - 4\left(\frac{24}{\alpha_0} + v_n'\right)s\lambda_T^2, \tag{A.15}$$

with

$$u_n' = 8\tau_0 s\frac{\log p}{n_T} \text{ and } v_n' = \frac{24^2\tau_0}{\alpha_0^2}\frac{s\log p}{n_T}. \tag{A.16}$$

Under the choice of n_T specified in Theorem 3, we have $u_n' \leq \alpha_0/2$ and $v_n' \leq 24/\alpha_0$. Therefore, (A.15) as a quadratic inequality in $\|\hat{\boldsymbol{\Delta}}^{(0)}\|_2$ and solving for its upper bound yields the second term in (7). The proof is then finished.

B.2 Proof of Corollary 5

Applying the Hoeffding-type inequality (cf. Proposition 5.1 in Vershynin (2012)) with Assumption 4 and $n_S > n_T \gtrsim \log p$, we have $\|\nabla \mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)})\|_\infty \lesssim \sqrt{\log p / n_T}$,

$$\Big\| \frac{n_T}{N} \nabla \mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)}) + \sum_{k=1}^{K} \frac{n_k}{N} \nabla \mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)}) \Big\|_\infty \lesssim \sqrt{\frac{\log p}{N}}, \text{ and } \Big\| \nabla \mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)}) \Big\|_\infty \lesssim \sqrt{\frac{\log p}{n_S}},$$

with probability larger than $1 - c_1 \exp(-c_2 \log p)$. Therefore, the choice of tuning parameters in Corollary 5 satisfies the requirement stated in Theorem 3, thereby we can directly plug the choice of tuning parameters into the bound (7) to obtain the probabilistic bound. It remains to verify that the conditions on n_T and n_S satisfy the requirements of Theorem 3. To establish the first term of the bound in (7), it suffices to show that u_n and v_n from (A.12) and u_n' and v_n' from (A.16) are all of order $o(1)$ when $h\sqrt{\frac{\log p}{n_T}} \lesssim \frac{s \log p}{n_T}$.

Under the assumption $n_T \gg s \log p$, we immediately obtain $u_n' = o(1)$ and $v_n' = o(1)$. It remains to establish the same for u_n and v_n. We proceed by discussing different cases based on the order of λ_0. Recall the tuning parameter choices given in the theorem:

$$\lambda_0 = c_0 \left[\Big(\frac{h^2 \log p}{s^2 n_T} \Big)^{1/4} + \Big(\frac{\log p}{N} \Big)^{1/2} \right], \quad \lambda_1 = c_0 \frac{n_S}{N} \Big(\frac{\log p}{n_T} \Big)^{1/2}.$$

Case 1: If $\frac{s \log p}{N} \lesssim h\sqrt{\frac{\log p}{n_T}} \lesssim \frac{s \log p}{n_T}$, then we have $\lambda_0 \asymp \Big(\frac{h^2 \log p}{s^2 n_T} \Big)^{1/4}$ and $\lambda_1 \asymp \frac{n_S}{N} \sqrt{\frac{\log p}{n_T}}$. Recalling that we set $n_k = n_S$ for $k \in [K]$, we consider the following two cases:

Case 1.1: If $\lambda_1^2 \lesssim \frac{n_S}{N} \lambda_0^2$, by the assumption that $Ks \frac{\log p}{n_T} = o(1)$, we have

$$u_n \lesssim \frac{1}{K} \frac{\log p}{n_T} \frac{\lambda_0^2 s}{\lambda_1^2} \lesssim Kh \sqrt{\frac{\log p}{n_T}} \lesssim Ks \frac{\log p}{n_T} = o(1),$$

where in the first inequality we use the fact that $n_S / N \geq 1/K$ and for the last inequality we use the fact that $h \lesssim s\sqrt{\frac{\log p}{n_T}}$. Similarly, we can establish that

$$v_n \lesssim \frac{1}{K} \frac{\log p}{n_T} \frac{\Big(\sum_{k=1}^K \lambda_1 h \Big)}{\lambda_1^2} \lesssim \sqrt{\frac{\log p}{n_T}} \sum_{k=1}^K h \lesssim Ks \frac{\log p}{n_T} = o(1).$$

Case 1.2: If $\lambda_1^2 \gtrsim \frac{n_S}{N} \lambda_0^2$, we can similarly establish that $u_n \lesssim \frac{s \log p}{n_T} = o(1)$, which follows from the assumptions that $\frac{s \log p}{n_T} = o(1)$. In this case, noting that $\sum_{k=1}^K \lambda_1 h \lesssim Kh\sqrt{\frac{\log p}{n_T}}$, thereby we have

$$v_n \lesssim \frac{1}{K} \frac{\log p}{n_T} \frac{\sum_{k=1}^K \lambda_1 h}{\frac{n_S}{N} \lambda_0^2} \lesssim \frac{s \log p}{n_T} = o(1).$$

Case 2: If $\frac{s \log p}{N} \gtrsim h\sqrt{\frac{\log p}{n_T}}$, then we have $\lambda_0 \asymp \sqrt{\frac{\log p}{N}}$ and $\lambda_1 \asymp \frac{n_S}{N} \sqrt{\frac{\log p}{n_T}}$. In this case, we have $\lambda_1^2 \gtrsim (n_S / N) \lambda_0^2$ as $n_S > n_T$. So following the discussion in the first case, we have

$$u_n \lesssim \frac{1}{K} \frac{\log p}{n_T} sK = \frac{s \log p}{n_T} = o(1).$$

Further notice that in this case, it holds that $\sum_{k=1}^{K}\lambda_1 h \lesssim h\sqrt{\frac{\log p}{n_T}} \lesssim \frac{s\log p}{N}$. Therefore, we further obtain

$$v_n \lesssim \frac{1}{K}\frac{\log p}{n_T}\frac{\sum_{k=1}^{K}\lambda_1 h}{(n_S/N)\lambda_0^2} \lesssim \frac{1}{K}\frac{\log p}{n_T} sK \lesssim \frac{s\log p}{n_T} = o(1).$$

The proof is then finished.

B.3 Proof of Corollary 9

When the tuning parameters are chosen as in (A.3), Corollary 5 already yields the desired rate. Thus, it suffices to consider the remaining case (A.4), where we show that $\hat{\boldsymbol{\beta}}^{(0)} = \hat{\boldsymbol{\beta}}_{\text{Pooling}}$. Then an application of Lemma 1 in Li et al. (2022) and Theorem 1 in Negahban et al. (2012) yield

$$\|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}_*^{(0)}\|_2^2 = \|\hat{\boldsymbol{\beta}}_{\text{Pooling}} - \boldsymbol{\beta}_*^{(0)}\|_2^2 \lesssim \frac{s\log p}{N} + C_\Sigma^2 h^2.$$

Notice that for problem (6), the solution for source parameter, $\hat{\boldsymbol{\beta}}^{(k)}$, satisfies

$$\hat{\boldsymbol{\beta}}^{(k)} \in \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ \frac{n_S}{N}\mathcal{L}^{(k)}(\boldsymbol{\beta}) + \lambda_1\|\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^{(0)}\|_1 \right\}. \tag{A.17}$$

For any $\boldsymbol{\beta} \neq \hat{\boldsymbol{\beta}}^{(0)}$, by the convexity of $\mathcal{L}^{(k)}(\cdot)$ and Hölder's inequality, we have

$$\begin{aligned}
\frac{n_S}{N}\mathcal{L}^{(k)}(\boldsymbol{\beta}) + \lambda_1\left\|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}\right\|_1 \geq& \frac{n_S}{N}\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}^{(0)}) + \left\langle \frac{n_S}{N}\nabla\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}^{(0)}), \boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^{(0)} \right\rangle + \lambda_1\left\|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}\right\|_1 \\
\geq& \frac{n_S}{N}\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}^{(0)}) - \left\|\frac{n_S}{N}\nabla\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}^{(0)})\right\|_\infty \left\|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}\right\|_1 + \lambda_1\left\|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}\right\|_1 \\
=& \frac{n_S}{N}\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}^{(0)}) + \left(\lambda_1 - \left\|\frac{n_S}{N}\nabla\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}^{(0)})\right\|_\infty\right)\left\|\hat{\boldsymbol{\beta}}^{(0)} - \boldsymbol{\beta}\right\|_1.
\end{aligned}$$

Therefore, if $\|\frac{n_S}{N}\nabla\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}^{(0)})\|_\infty < \lambda_1$, then $\hat{\boldsymbol{\beta}}^{(0)}$ is the unique minimizer of problem (A.17), which further implies that $\hat{\boldsymbol{\beta}}^{(k)} = \hat{\boldsymbol{\beta}}^{(0)} = \hat{\boldsymbol{\beta}}_{\text{Pooling}}$ for all $k \in [K]$. The proof is then finished.

B.4 Proof of Theorem 11

Recall that we have the decomposition

$$\tilde{\boldsymbol{\beta}}^{(k)} - \boldsymbol{\beta}_*^{(k)} = -\hat{\boldsymbol{\Theta}}^{(k)}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)}) + \underbrace{\left(\boldsymbol{I} - \hat{\boldsymbol{\Theta}}^{(k)}\nabla^2\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{\text{Int}}^{(k)})\right)\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)}_{\boldsymbol{b}^{(k)}}.$$

We first show a Lemma discussing the bias and variance components in (14), whose proof is based on van de Geer et al. (2014) but taking into account that $\boldsymbol{\beta}_*^{(k)}$ is not exactly s-sparse (due to the difference term $\boldsymbol{\delta}^{(k)}$).

Lemma 17 *Under Assumption 1 and 4 and* $\frac{s\log p}{n_S}+h\sqrt{\frac{\log p}{n_S}}=o(1)$*, if we construct* $\{\hat{\boldsymbol{\beta}}_L^{(k)}\}_{k\in[K]}$ *as the local lasso estimators defined in Section 4 and* $\{\hat{\boldsymbol{\Theta}}^{(k)}\}_{k\in[K]}$ *using (A.5), with parameters* $\tilde{\lambda}_k=\mu_{jk}\asymp\sqrt{\frac{\log p}{n_S}}$*, then we have that for* $k=1,\ldots,K$*,*

$$\left\|\hat{\boldsymbol{\Theta}}^{(k)}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\right\|_\infty\lesssim\sqrt{\frac{\log p}{n_S}}+h\sqrt{\frac{\log p}{n_S}},\tag{A.18}$$

and

$$\left\|\boldsymbol{b}^{(k)}\right\|_\infty\lesssim\frac{s\log p}{n_S}+h\sqrt{\frac{\log p}{n_S}},\tag{A.19}$$

with probability larger than $1-c_1\exp(-c_2\log p)$*.*

Now we proceed to the proof of the theorem. We first show that by reparametrization, problem (11) is essentially a special case of problem (6). Then we apply techniques similar to those used in Theorem 3 to prove the results.

Following the arguments in the proof of Theorem 3, we again adopt the reparametrization in (A.6) and reformulate problem (11) as

$$\hat{\boldsymbol{\theta}}_D\in\operatorname*{argmin}_{\boldsymbol{\theta}}\{\tilde{\mathcal{L}}(\boldsymbol{\theta})+\lambda_0\mathcal{R}(\boldsymbol{\theta})\},\quad\text{subject to}\quad\boldsymbol{\theta}^{(0)}\in\mathcal{C}_T.\tag{A.20}$$

where we define $\tilde{\mathcal{L}}(\boldsymbol{\theta}):=\frac{n_T}{N}\mathcal{L}^{(0)}(\boldsymbol{\theta}^{(0)})+\sum_{k=1}^K\frac{n_S}{N}\tilde{\mathcal{L}}^{(k)}(\boldsymbol{\theta}^{(0)}+\boldsymbol{\theta}^{(k)})$ and $\mathcal{R}(\boldsymbol{\theta}):=\left\|\boldsymbol{\theta}^{(0)}\right\|_1+\sum_{k=1}^K\frac{\lambda_1}{\lambda_0}\left\|\boldsymbol{\theta}^{(k)}\right\|_1$, with $\tilde{\mathcal{L}}^{(k)}(\boldsymbol{\beta}):=\frac{n_S}{2}\|\tilde{\boldsymbol{\beta}}^{(k)}-\boldsymbol{\beta}\|_2^2$.

Following the aforementioned reformulation, we can employ an approach similar to that used in Lemma 14 to establish the result about $\left\langle\nabla\tilde{\mathcal{L}}\left(\boldsymbol{\theta}_*\right),\boldsymbol{\Delta}\right\rangle$. Define $\delta_k=\frac{s\log p}{N}+\frac{n_S}{N}\sqrt{\frac{\log p}{n_S}}h$ and $\delta_0=\frac{Ks\log p}{N}+\sum_{k=1}^K\frac{n_S}{N}\sqrt{\frac{\log p}{n_S}}h$, then the result is stated as follows:

Lemma 18 *Under the conditions in Lemma 17, if we choose* $\lambda_1=c_k\left(\sqrt{\frac{n_S}{N}\frac{\log p}{N}}+\delta_k\right)$ *and* $\lambda_0=c_0\left(\sqrt{\frac{\log p}{N}}+\delta_0\right)$ *for some appropriate constants* $c_1,\ldots,c_K$*, then we have for any* $\boldsymbol{\Delta}=\left(\left(\boldsymbol{\Delta}^{(0)}\right)^\top,\left(\boldsymbol{\Delta}^{(1)}\right)^\top,\ldots,\left(\boldsymbol{\Delta}^{(K)}\right)^\top\right)^\top\in\mathbb{R}^{(K+1)p}$*,*

$$\left|\left\langle\nabla\tilde{\mathcal{L}}\left(\boldsymbol{\theta}_*\right),\boldsymbol{\Delta}\right\rangle\right|\leq\sum_{k=1}^K\frac{\lambda_1}{2}\left\|\boldsymbol{\Delta}^{(k)}\right\|_1+\frac{\lambda_0}{2}\left\|\boldsymbol{\Delta}^{(0)}\right\|_1.$$

with probability larger than $1-c_1\exp\left(-c_2\log p\right)$*.*

One may notice that the only difference between Lemma 18 and Lemma 14 is the choice of λ_0 and λ_1. With this new choice of parameters, it is easy to verify that $\lambda_1\gtrsim\frac{n_S}{N}\lambda_T$ still holds and the arguments used in the proof of Lemma 15, Lemma 16 and Theorem 3 remain applicable for the new problem (11).

Hence, following similar procedures, we obtain

$$\|\hat{\mathbf{\Delta}}^{(0)}\|_2^2 \leq \frac{1}{(\alpha_{\min}^2/\gamma_0 - u_n)^2}\left(\frac{9}{4}s\lambda_0^2 + 2(\alpha_{\min}^2/\gamma_0 - u_n)(2+v_n)\sum_{k=1}^{K}\lambda_1 h\right)$$

To prove the theorem, it remains to show that $u_n = o(1)$ and $v_n = o(1)$ under the new set of choice of tuning parameters. Notice that, similar to the proof of Theorem 3, we only need to prove the result under the regime when $h\sqrt{\frac{\log p}{n_T}} \lesssim \frac{s\log p}{n_T}$. Recall that the tuning parameter choices

$$\lambda_0 \asymp \Big(\frac{h^2\log p}{s^2 n_T}\Big)^{1/4} + \Big(\frac{\log p}{N}\Big)^{1/2} + \frac{Ks\log p}{N} + \sum_{k=1}^{K}\frac{n_S}{N}\sqrt{\frac{\log p}{n_S}}h,$$
$$\lambda_1 \asymp \frac{n_S}{N}\Big(\frac{\log p}{n_T}\Big)^{1/2} + \frac{s\log p}{N} + \frac{n_S}{N}\sqrt{\frac{\log p}{n_S}}h.$$

We consider the following two cases:

Case 1: If $\lambda_1^2 \lesssim \frac{n_S}{N}\lambda_0^2$, then we have

$$\begin{aligned} u_n &\lesssim \frac{1}{K}\frac{\log p}{n_T}\frac{\lambda_0^2 s}{\lambda_1^2} \lesssim \frac{1}{K}\frac{s\log p}{n_T}\frac{\frac{h\sqrt{\log p}}{s\sqrt{n_T}} + \frac{\log p}{N} + \Big(\frac{Ks\log p}{N}\Big)^2 + \frac{\log p}{n_S}h^2}{\frac{1}{K^2}\frac{\log p}{n_T} + \Big(\frac{s\log p}{N}\Big)^2 + \frac{1}{K^2}\frac{\log p}{n_S}h^2} \\ &\lesssim \frac{1}{K}\frac{s\log p}{n_T}\frac{\frac{h\sqrt{\log p}}{s\sqrt{n_T}} + \frac{\log p}{N} + \Big(\frac{Ks\log p}{N}\Big)^2 + \frac{\log p}{n_S}h^2}{\frac{1}{K^2}\frac{\log p}{n_T}} \\ &\lesssim Kh\sqrt{\frac{\log p}{n_T}} + \frac{s\log p}{n_S} + \frac{Ks^2\log^2 p}{n_S^2} + \frac{Ks\log p}{n_S}h^2 = o(1) \end{aligned} \tag{A.21}$$

Here, for the last inequality we use the assumption that $n_T \gg \big((K^2h^2)\vee s\big)\log p$ and $n_S \gg (K^2h^4 + K^2h^2 + Kh^2s + Khs + Ks + sh^2 + s^{3/2})\log p$, which are obtained from the assumption $n_T \gg K^2 s\log p$ and $n_S \gg K^2 s^2\log p$ in Theorem 11 and the fact that $h \lesssim s\sqrt{\frac{\log p}{n_T}}$ under this regime. Similarly, we can establish that

$$v_n \lesssim \frac{1}{K}\frac{\log p}{n_T}\frac{\sqrt{\frac{\log p}{n_T}}h + \frac{s\log p}{n_S}h + \sqrt{\frac{\log p}{n_S}}h^2}{\frac{1}{K^2}\frac{\log p}{n_T}} \lesssim K\sqrt{\frac{\log p}{n_T}}h + \frac{Ks\log p}{n_S}h + K\sqrt{\frac{\log p}{n_S}}h^2 = o(1).$$

Case 2: If $\lambda_1^2 \gtrsim \frac{n_S}{N}\lambda_0^2$, we can similarly establish that $u_n \lesssim \frac{s\log p}{n_T} = o(1)$. For v_n, one can show that

$$\begin{aligned} v_n &\lesssim \frac{1}{K}\frac{\log p}{n_T}\frac{\sum_{k=1}^{K}\lambda_1 h}{\frac{n_S}{N}\lambda_0^2} \lesssim \frac{Kh\log p}{n_T}\frac{\lambda_1}{\lambda_0^2} \lesssim \frac{Kh\log p}{n_T}\frac{\frac{n_S}{N}\Big(\frac{\log p}{n_T}\Big)^{1/2} + \frac{s\log p}{N} + \frac{n_S}{N}\sqrt{\frac{\log p}{n_S}}h}{\frac{h\sqrt{\log p}}{s\sqrt{n_T}}} \\ &\lesssim \frac{s\log p}{n_T} + \frac{s^{3/2}\log p}{n_S}\sqrt{\frac{s\log p}{n_T}} + \sqrt{\frac{s\log p}{n_T}}\sqrt{\frac{s\log p}{n_S}}h = o(1). \end{aligned} \tag{A.22}$$

The proof is then finished.

B.5 Proof of Lemmas

Proof of Lemma 12 Notice that by triangle inequality, we obtain

$$\left\|\frac{n_S}{N}\nabla\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{\text{Pooling}})\right\|_\infty \leq \left\|\frac{n_S}{N}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\right\|_\infty + \left\|\frac{n_S}{N}\left(\nabla\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{\text{Pooling}}) - \nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\right)\right\|_\infty.$$

Since we assume $n_S \gtrsim \log p$, by Assumption 4 and the Hoeffding-type inequality (cf. Proposition 5.1 in Vershynin (2012)) we have $\|\frac{n_S}{N}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\|_\infty \lesssim \frac{n_S}{N}\sqrt{\frac{\log p}{n_S}}$ with probability larger than $1 - c_1\exp(-c_2\log p)$. By the ℓ_∞ curvature assumption imposed in the Lemma and triangle inequality, we have

$$\left\|\nabla\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{\text{Pooling}}) - \nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\right\|_\infty \leq \eta_k\left\|\hat{\boldsymbol{\beta}}_{\text{Pooling}} - \boldsymbol{\beta}_*^{(k)}\right\|_\infty + \zeta_k\sqrt{\frac{\log p}{n_S}}\left\|\hat{\boldsymbol{\beta}}_{\text{Pooling}} - \boldsymbol{\beta}_*^{(k)}\right\|_1, \tag{A.23}$$

with probability larger than $1 - c_1\exp(-c_2\log p)$.

Another application of Lemma 1 in Li et al. (2022) and Theorem 1 in Negahban et al. (2012) leads to

$$\|\hat{\boldsymbol{\beta}}_{\text{Pooling}} - \boldsymbol{\beta}_*^{(k)}\|_\infty \lesssim \sqrt{\frac{\log p}{N}} + \|\boldsymbol{\beta}_{\text{Pooling}} - \boldsymbol{\beta}_*^{(k)}\|_\infty, \text{ and } \|\hat{\boldsymbol{\beta}}_{\text{Pooling}} - \boldsymbol{\beta}_*^{(k)}\|_1 \lesssim s\sqrt{\frac{\log p}{N}} + C_\Sigma h. \tag{A.24}$$

Recall that we define $K_P = \max_{k\in[K]}\|\boldsymbol{\beta}_{\text{Pooling}} - \boldsymbol{\beta}_*^{(k)}\|_\infty$. Plug the results in (A.24) back into (A.23) finishes the proof.

Proof of Lemma 14 By definition, we have

$$\nabla\mathcal{L}(\boldsymbol{\theta}_*) = \left(\frac{n_S}{N}\nabla\mathcal{L}^{(1)}(\boldsymbol{\beta}_*^{(1)}), \ldots, \frac{n_S}{N}\nabla\mathcal{L}^{(K)}(\boldsymbol{\beta}_*^{(K)}), \frac{n_T}{N}\nabla\mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)}) + \sum_{k=1}^K\frac{n_S}{N}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\right)^\top$$

Therefore, by Hölder's inequality, we have

$$\begin{aligned}|\langle\nabla\mathcal{L}(\boldsymbol{\theta}_*), \boldsymbol{\Delta}\rangle| &= \sum_{k=1}^K\left|\left\langle\frac{n_S}{N}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)}), \boldsymbol{\Delta}^{(k)}\right\rangle\right| + \left|\left\langle\sum_{k=0}^K\frac{n_k}{N}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)}), \boldsymbol{\Delta}^{(0)}\right\rangle\right| \\ &\leq \sum_{k=1}^K\left\|\frac{n_S}{N}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\right\|_\infty\left\|\boldsymbol{\Delta}^{(k)}\right\|_1 + \left\|\sum_{k=0}^K\frac{n_k}{N}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\right\|_\infty\left\|\boldsymbol{\Delta}^{(0)}\right\|_1.\end{aligned}$$

Therefore, as long as we choose λ_0 and λ_1 such that $\lambda_0 \geq 2\left\|\sum_{k=0}^K\frac{n_k}{N}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\right\|_\infty$ and $\lambda_1 \geq 2\max_{k\in[K]}\left\|\frac{n_S}{N}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\right\|_\infty$, we then have

$$|\langle\nabla\mathcal{L}(\boldsymbol{\theta}_*), \boldsymbol{\Delta}\rangle| \leq \sum_{k=1}^K\frac{\lambda_1}{2}\left\|\boldsymbol{\Delta}^{(k)}\right\|_1 + \frac{\lambda_0}{2}\left\|\boldsymbol{\Delta}^{(0)}\right\|_1,$$

as claimed.

B.6 Proof of Lemma 15

We first establish the feasibility of $\boldsymbol{\theta}_*$ for the problem (A.7). Recall that we impose the constraint

$$\mathcal{C}_T = \{\boldsymbol{\beta}^{(0)} \in \mathbb{R}^p : \|\nabla \mathcal{L}^{(0)}(\boldsymbol{\beta}^{(0)})\|_\infty \leq \lambda_T, \|\boldsymbol{\beta}^{(0)}\|_1 \leq \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1 + \kappa_T\}, \tag{A.25}$$

with

$$\hat{\boldsymbol{\beta}}_T^{(0)} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \mathcal{L}^{(0)}(\boldsymbol{\beta}) + \lambda_T \|\boldsymbol{\beta}\|_1 \right\}.$$

Since we choose $\lambda_T \geq 2\|\nabla \mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)})\|_\infty$, the first constraint in (A.25) holds. Furthermore, recall that we denote S as the support of $\boldsymbol{\beta}_*^{(0)}$. By Corollary 1 in Negahban et al. (2012) with $\mathcal{M}(S) := \left\{\boldsymbol{\beta}^{(0)} \in \mathbb{R}^p \mid \boldsymbol{\beta}_j^{(0)} = 0 \text{ for all } j \notin S\right\}$, under Assumption 1 and the condition $n_T > 64\,(\tau_0/\alpha_0)^2\, s \log p$, we have

$$\|\hat{\boldsymbol{\beta}}_T^{(0)} - \boldsymbol{\beta}_*^{(0)}\|_2^2 \leq 18 s \lambda_T^2/\alpha_0^2 \text{ and } \|\hat{\boldsymbol{\beta}}_T^{(0)} - \boldsymbol{\beta}_*^{(0)}\|_1 \leq 24 s \lambda_T/\alpha_0, \tag{A.26}$$

which together with the choice that $\kappa_T = 24 s \lambda_T/\alpha_0$ implies that

$$\|\boldsymbol{\beta}_*^{(0)}\|_1 - \|\hat{\boldsymbol{\beta}}_T^{(0)}\|_1 \leq \|\hat{\boldsymbol{\beta}}_T^{(0)} - \boldsymbol{\beta}_*^{(0)}\|_1 \leq \kappa_T. \tag{A.27}$$

therefore the second constraint in (A.25) is also satisfied. Hence, we have $\boldsymbol{\theta}^{(0)} = \boldsymbol{\beta}_*^{(0)} \in \mathcal{C}_T$, which implies $\boldsymbol{\theta}_*$ is feasible.

To prove the lemma, it remains to show the result in (A.8). We first define the optimal value gap function $F : \mathbb{R}^{(K+1)p} \to \mathbb{R}$ as

$$F(\boldsymbol{\Delta}) = \mathcal{L}\left(\boldsymbol{\theta}_* + \boldsymbol{\Delta}\right) - \mathcal{L}\left(\boldsymbol{\theta}_*\right) + \lambda_0 \mathcal{R}\left(\boldsymbol{\theta}_* + \boldsymbol{\Delta}\right) - \lambda_0 \mathcal{R}\left(\boldsymbol{\theta}_*\right), \tag{A.28}$$

and $\hat{\boldsymbol{\theta}}$ as the solution to the problem (A.7). We then have $\hat{\boldsymbol{\Delta}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_* = \underset{\boldsymbol{\Delta}}{\operatorname{argmin}} F(\boldsymbol{\Delta})$ and $F(\mathbf{0}) = 0$. Since both $\hat{\boldsymbol{\theta}}$ and $\boldsymbol{\theta}_*$ are feasible solutions to problem (A.7), we conclude that $F(\hat{\boldsymbol{\Delta}}) \leq 0$.

Since $\mathcal{L}(\cdot)$ is a convex function, by Lemma 14, we can choose λ_0 and λ_1 such that $\lambda_0 \geq 2\left\|\sum_{k=0}^K \frac{n_k}{N} \nabla \mathcal{L}^{(k)}(\boldsymbol{\theta}_*^{(k)})\right\|_\infty$ and $\lambda_1 \geq 2 \max_{k \in [K]} \left\|\frac{n_S}{N} \nabla \mathcal{L}^{(k)}(\boldsymbol{\theta}_*^{(k)})\right\|_\infty$ so that

$$\mathcal{L}\left(\boldsymbol{\theta}_* + \hat{\boldsymbol{\Delta}}\right) - \mathcal{L}\left(\boldsymbol{\theta}_*\right) \geq \left\langle \nabla \mathcal{L}\left(\boldsymbol{\theta}_*\right), \hat{\boldsymbol{\Delta}} \right\rangle \geq -\sum_{k=1}^K \frac{\lambda_1}{2} \left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - \frac{\lambda_0}{2} \left\|\hat{\boldsymbol{\Delta}}^{(0)}\right\|_1. \tag{A.29}$$

Since the ℓ_1-norm function is decomposable and $\|\boldsymbol{\theta}^{(0)}_{S^c}\|_1 = 0$, by triangle inequality we have

$$\begin{aligned}
&\lambda_0 \mathcal{R}\left(\boldsymbol{\theta}_* + \hat{\boldsymbol{\Delta}}\right) - \lambda_0 \mathcal{R}\left(\boldsymbol{\theta}_*\right) \\
=&\sum_{k=1}^{K} \lambda_1 \left(\left\|\boldsymbol{\theta}^{(k)}_* + \hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - \left\|\boldsymbol{\theta}^{(k)}_*\right\|_1\right) + \lambda_0 \left(\left\|\boldsymbol{\theta}^{(0)}_* + \hat{\boldsymbol{\Delta}}^{(0)}\right\|_1 - \left\|\boldsymbol{\theta}^{(0)}_*\right\|_1\right) \\
\geq&\sum_{k=1}^{K} \lambda_1 \left(\left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - 2\left\|\boldsymbol{\theta}^{(k)}_*\right\|_1\right) + \lambda_0 \left(\left\|\hat{\boldsymbol{\Delta}}^{(0)}_{S^c}\right\|_1 - \left\|\hat{\boldsymbol{\Delta}}^{(0)}_{S}\right\|_1 - 2\left\|\boldsymbol{\theta}^{(0)}_{S^c}\right\|_1\right) \\
\geq&\sum_{k=1}^{K} \lambda_1 \left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - 2\sum_{k=1}^{K} \lambda_1 h + \lambda_0 \left(\left\|\hat{\boldsymbol{\Delta}}^{(0)}_{S^c}\right\|_1 - \left\|\hat{\boldsymbol{\Delta}}^{(0)}_{S}\right\|_1\right). \qquad \text{(A.30)}
\end{aligned}$$

where we use the property that $\left\|\boldsymbol{\theta}^{(k)}_*\right\|_1 \leq h$ from the parameter space $\Theta(s, h)$.

Combining (A.29) with (A.30) yields

$$0 \geq F(\hat{\boldsymbol{\Delta}}) \geq \sum_{k=1}^{K} \frac{\lambda_1}{2} \left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - 2\sum_{k=1}^{K} \lambda_1 h + \frac{\lambda_0}{2}\left(\left\|\hat{\boldsymbol{\Delta}}^{(0)}_{S^c}\right\|_1 - 3\left\|\hat{\boldsymbol{\Delta}}^{(0)}_{S}\right\|_1\right), \qquad \text{(A.31)}$$

which leads to the following inequality:

$$\sum_{k=1}^{K} \lambda_1 \left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 + \lambda_0 \left\|\hat{\boldsymbol{\Delta}}^{(0)}\right\|_1 \leq 4\lambda_0 \left\|\hat{\boldsymbol{\Delta}}^{(0)}_{S}\right\|_1 + 4\sum_{k=1}^{K} \lambda_1 h,$$

as claimed.

B.7 Proof of Lemma 16

Applying the mean value theorem for the loss function $\mathcal{L}$ gives

$$\begin{aligned}
&\mathcal{L}\left(\boldsymbol{\theta}_* + \hat{\boldsymbol{\Delta}}\right) - \mathcal{L}\left(\boldsymbol{\theta}_*\right) - \left\langle \nabla \mathcal{L}\left(\boldsymbol{\theta}_*\right), \hat{\boldsymbol{\Delta}} \right\rangle \\
=&\hat{\boldsymbol{\Delta}}^{\top} \nabla^2 \mathcal{L}\left(\boldsymbol{\theta}_* + \phi\hat{\boldsymbol{\Delta}}\right) \hat{\boldsymbol{\Delta}} \, (\phi \in (0,1)) \\
=&\sum_{k=1}^{K} \frac{n_S}{N} \left(\hat{\boldsymbol{\Delta}}^{(k)} + \hat{\boldsymbol{\Delta}}^{(0)}\right)^{\top} \nabla^2 \mathcal{L}^{(k)}\left(\boldsymbol{\theta}^{(k)} + \phi\hat{\boldsymbol{\Delta}}^{(k)}\right)\left(\hat{\boldsymbol{\Delta}}^{(k)} + \hat{\boldsymbol{\Delta}}^{(0)}\right) \\
&+ \frac{n_T}{N}\left(\hat{\boldsymbol{\Delta}}^{(0)}\right)^{\top} \nabla^2 \mathcal{L}^{(0)}(\boldsymbol{\theta}^{(0)}_* + \phi\hat{\boldsymbol{\Delta}}^{(0)})\hat{\boldsymbol{\Delta}}^{(0)}.
\end{aligned}$$

An application of both RSC and RSM property stated in Assumption 1 leads to

$$\begin{aligned}
&\mathcal{L}\left(\boldsymbol{\theta}_* + \hat{\boldsymbol{\Delta}}\right) - \mathcal{L}\left(\boldsymbol{\theta}_*\right) - \left\langle \nabla \mathcal{L}\left(\boldsymbol{\theta}_*\right), \hat{\boldsymbol{\Delta}} \right\rangle \\
\geq & \sum_{k=1}^{K} \frac{n_S \alpha_k}{N} \left\| \hat{\boldsymbol{\Delta}}^{(k)} + \hat{\boldsymbol{\Delta}}^{(0)} \right\|_2^2 + \frac{n_T \alpha_0}{N} \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_2^2 - R_1(\hat{\boldsymbol{\Delta}}) \\
\geq & \sum_{k=1}^{K} \frac{n_S \alpha_k}{N} \cdot \frac{1}{\gamma_0} \left(\hat{\boldsymbol{\Delta}}^{(k)} + \hat{\boldsymbol{\Delta}}^{(0)} \right)^\top \nabla^2 \mathcal{L}^{(0)}(\boldsymbol{\theta}_*^{(0)} + \phi \hat{\boldsymbol{\Delta}}^{(0)}) \left(\hat{\boldsymbol{\Delta}}^{(k)} + \hat{\boldsymbol{\Delta}}^{(0)} \right) - R_1(\hat{\boldsymbol{\Delta}}) \\
& + \frac{n_T \alpha_0}{N} \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_2^2 - R_2(\hat{\boldsymbol{\Delta}}),
\end{aligned} \tag{A.32}$$

where

$$\begin{aligned}
R_1(\hat{\boldsymbol{\Delta}}) &:= \sum_{k=1}^{K} \frac{\beta_k \log p}{N} \left\| \hat{\boldsymbol{\Delta}}^{(k)} + \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 + \frac{\beta_0 \log p}{N} \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 \\
R_2(\hat{\boldsymbol{\Delta}}) &:= \sum_{k=1}^{K} \frac{n_S \alpha_k}{N} \frac{\tau_0}{\gamma_0} \frac{\log p}{n_T} \left\| \hat{\boldsymbol{\Delta}}^{(k)} + \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 .
\end{aligned} \tag{A.33}$$

In addition, noting that $\hat{\boldsymbol{\theta}}^{(0)}$ satisfies the constraint outlined in problem (A.7). Again by the mean value theorem and the assumption that $\lambda_T \geq 2\|\mathcal{L}^{(0)}(\boldsymbol{\theta}_*^{(0)})\|_\infty$, we have

$$\begin{aligned}
\left\| \nabla^2 \mathcal{L}^{(0)}(\boldsymbol{\theta}_*^{(0)} + \phi \hat{\boldsymbol{\Delta}}^{(0)}) \hat{\boldsymbol{\Delta}}^{(0)} \right\|_\infty &= \left\| \nabla \mathcal{L}^{(0)}(\hat{\boldsymbol{\theta}}^{(0)}) - \nabla \mathcal{L}^{(0)}(\boldsymbol{\theta}_*^{(0)}) \right\|_\infty \\
&\leq \left\| \nabla \mathcal{L}^{(0)}(\hat{\boldsymbol{\theta}}^{(0)}) \right\|_\infty + \left\| \nabla \mathcal{L}^{(0)}(\boldsymbol{\theta}_*^{(0)}) \right\|_\infty \\
&\leq 2\lambda_T.
\end{aligned}$$

Hence, a direct application of the Hölder's inequality leads to

$$\left(\hat{\boldsymbol{\Delta}}^{(k)} \right)^\top \nabla^2 \mathcal{L}^{(0)}(\boldsymbol{\theta}_*^{(0)} + \phi \hat{\boldsymbol{\Delta}}^{(0)}) \hat{\boldsymbol{\Delta}}^{(0)} \leq 2\lambda_T \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1 . \tag{A.34}$$

Recall that we define $n_k = n_S$ for $k = 1, \ldots, K$ and $n_k = n_T$ for $k = 0$. Combining (A.34) with (A.32) and applying the RSC property again, we have

$$\begin{aligned}
&\mathcal{L}\left(\boldsymbol{\theta}_* + \hat{\boldsymbol{\Delta}}\right) - \mathcal{L}\left(\boldsymbol{\theta}_*\right) - \left\langle \nabla \mathcal{L}\left(\boldsymbol{\theta}_*\right), \hat{\boldsymbol{\Delta}} \right\rangle \\
\geq & \sum_{k=1}^{K} \frac{n_S \alpha_k}{N} \cdot \frac{1}{\gamma_0} \left[\alpha_0 \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_2^2 + \alpha_0 \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_2^2 - 2\lambda_T \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1 \right] \\
& + \frac{n_T \alpha_0}{N} \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_2^2 - R_1(\hat{\boldsymbol{\Delta}}) - R_2(\hat{\boldsymbol{\Delta}}) - R_3(\hat{\boldsymbol{\Delta}}) \\
\geq & \frac{\alpha_{\min}^2}{\gamma_0} \left[\sum_{k=0}^{K} \frac{n_k}{N} \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_2^2 + \sum_{k=1}^{K} \frac{n_S}{N} \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_2^2 \right] - \frac{2\alpha_{\max}}{\gamma_0} \sum_{k=1}^{K} \frac{n_S}{N} \lambda_T \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1 - \sum_{t=1}^{3} R_t(\hat{\boldsymbol{\Delta}}),
\end{aligned}$$

where R_1, R_2 are defined in (A.33) and

$$R_3(\hat{\boldsymbol{\Delta}}) := \sum_{k=1}^{K} \frac{n_S \alpha_k}{N} \frac{\beta_0}{\gamma_0} \frac{\log p}{n_T} \left(\left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1^2 + \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 \right).$$

Notice that

$$\begin{aligned}
\sum_{t=1}^{3} R_t(\hat{\boldsymbol{\Delta}}) =& \sum_{k=1}^{K} \frac{\beta_k \log p}{N} \left\| \hat{\boldsymbol{\Delta}}^{(k)} + \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 + \frac{\beta_0 \log p}{N} \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 + \sum_{k=1}^{K} \frac{n_S \alpha_k}{N} \frac{\tau_0}{\gamma_0} \frac{\log p}{n_T} \left\| \hat{\boldsymbol{\Delta}}^{(k)} + \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 \\
&+ \sum_{k=1}^{K} \frac{n_S \alpha_k}{N} \frac{\beta_0}{\gamma_0} \frac{\log p}{n_T} \left(\left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1^2 + \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 \right) \\
\leq& \left(\frac{\beta_0 \log p}{N} + 2 \sum_{k=1}^{K} \frac{\beta_k \log p}{N} + 2 \sum_{k=1}^{K} \frac{n_S \alpha_k}{N} \frac{\tau_0}{\gamma_0} \frac{\log p}{n_T} \right) \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 \\
&+ \sum_{k=1}^{K} \left(\frac{2\beta_k \log p}{N} + \frac{2 n_S \alpha_k}{N} \frac{\tau_0}{\gamma_0} \frac{\log p}{n_T} \right) \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1^2 \\
\leq& \left(2 \sum_{k=0}^{K} \frac{\beta_k \log p}{N} + 2 \sum_{k=1}^{K} \frac{n_S \alpha_k}{N} \frac{\tau_0}{\gamma_0} \frac{\log p}{n_T} \right) \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 \\
&+ \sum_{k=1}^{K} \left(\frac{2\beta_k \log p}{N} + \frac{2 n_S \alpha_k}{N} \frac{\tau_0}{\gamma_0} \frac{\log p}{n_T} \right) \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1^2 \\
\leq& \frac{2(\alpha_{\max} \tau_0 + \beta_{\max} \gamma_0)}{\gamma_0} \left[\left(\frac{\log p}{N} + \sum_{k=1}^{K} \frac{n_S}{n_T} \frac{\log p}{N} \right) \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 + \sum_{k=1}^{K} \frac{n_S}{n_T} \frac{\log p}{N} \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1^2 \right].
\end{aligned}$$

Therefore, Lemma 15 together with the triangle inequality yields

$$\begin{aligned}
&\sum_{t=1}^{3} R_t(\widehat{\boldsymbol{\Delta}}) \cdot \left(\frac{2(\alpha_{\max} \tau_0 + \beta_{\max} \gamma_0)}{\gamma_0} \right)^{-1} \\
\leq& \sum_{k=1}^{K} \frac{n_S}{n_T} \frac{\log p}{N} \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1^2 + \left(1 + \sum_{k=1}^{K} \frac{n_S}{n_T} \right) \frac{\log p}{N} \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 \\
=& \frac{n_S}{n_T} \frac{\log p}{N} \sum_{k=1}^{K} \frac{\lambda_1^2}{\lambda_1^2} \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1^2 + \frac{\log p}{N} \frac{1}{n_T} \left(n_T + \sum_{k=1}^{K} n_S \right) \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 \\
=& \frac{n_S}{n_T} \frac{\log p}{N} \left(\sum_{k=1}^{K} \frac{\lambda_1^2}{\lambda_1^2} \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1^2 + \frac{\lambda_0^2}{\lambda_0^2/(N/n_S)} \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1^2 \right) \\
\leq& \frac{n_S}{n_T} \frac{\log p}{N} \cdot \frac{1}{\lambda_1^2 \wedge [(n_S/N)\lambda_0^2]} \left(\sum_{k=1}^{K} \lambda_1 \left\| \hat{\boldsymbol{\Delta}}^{(k)} \right\|_1 + \lambda_0 \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_1 \right)^2 \\
\leq& \frac{n_S}{n_T} \frac{\log p}{N} \cdot \frac{1}{\lambda_1^2 \wedge [(n_S/N)\lambda_0^2]} \left[32 \lambda_0^2 s \left\| \hat{\boldsymbol{\Delta}}^{(0)} \right\|_2^2 + 32 \left(\sum_{k=1}^{K} \lambda_1 h \right)^2 \right],
\end{aligned}$$

Recall that we introduce the shorthand

$$u_n = \frac{64(\alpha_{\max}\tau_0 + \beta_{\max}\gamma_0)}{\gamma_0}\frac{n_S}{n_T}\frac{\log p}{N}\cdot\frac{\lambda_0^2 s}{\lambda_1^2 \wedge [(n_S/N)\lambda_0^2]}$$
$$v_n = \frac{64(\alpha_{\max}\tau_0 + \beta_{\max}\gamma_0)}{\gamma_0}\frac{n_S}{n_T}\frac{\log p}{N}\cdot\frac{\sum_{k=1}^K \lambda_1 h}{\lambda_1^2 \wedge [(n_S/N)\lambda_0^2]}.$$

Therefore, by collecting all the pieces together, we have

$$\begin{aligned}
&\mathcal{L}\left(\boldsymbol{\theta}_* + \hat{\boldsymbol{\Delta}}\right) - \mathcal{L}\left(\boldsymbol{\theta}_*\right) - \left\langle \nabla\mathcal{L}\left(\boldsymbol{\theta}_*\right), \hat{\boldsymbol{\Delta}}\right\rangle \\
&\geq \left(\frac{\alpha_{\min}^2}{\gamma_0} - u_n\right)\left\|\hat{\boldsymbol{\Delta}}^{(0)}\right\|_2^2 + \frac{\alpha_{\min}^2}{\gamma_0}\sum_{k=1}^K \frac{n_k}{N}\left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_2^2 - \frac{2\alpha_{\max}}{\gamma_0}\sum_{k=1}^K \frac{n_k}{N}\lambda_T\left\|\hat{\boldsymbol{\Delta}}^{(k)}\right\|_1 - v_n\sum_{k=1}^K \lambda_1 h,
\end{aligned}$$

which finishes the proof.

B.8 Proof of Lemma 17

Recall that we have the decomposition

$$\tilde{\boldsymbol{\beta}}^{(k)} - \boldsymbol{\beta}_*^{(k)} = -\hat{\boldsymbol{\Theta}}^{(k)}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)}) + \underbrace{\left(\boldsymbol{I} - \hat{\boldsymbol{\Theta}}^{(k)}\nabla^2\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{\text{Int}}^{(k)})\right)\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)}_{\boldsymbol{b}^{(k)}}.$$

where $\hat{\boldsymbol{\beta}}_{\text{Int}}^{(k)}$ is a vector intermediating $\hat{\boldsymbol{\beta}}_L^{(k)}$ and $\boldsymbol{\beta}_*^{(k)}$. In the following we define $\boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{(k)} := \mathbb{E}\left[(\mathbf{X}_{\boldsymbol{\beta}}^{(k)})^\top(\mathbf{X}_{\boldsymbol{\beta}}^{(k)})/n_S\right]$, and $\boldsymbol{\Theta}^{(k)} = (\boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{(k)})^{-1}$.

The proof of the lemma relies on two additional lemmas. Recall that $\left\|\boldsymbol{\beta}_*^{(0)}\right\|_0 = s$ and $\left\|\boldsymbol{\beta}_*^{(0)} - \boldsymbol{\beta}_*^{(k)}\right\|_1 \leq h$. The following lemma is a direct consequence of Theorem 1 and Corollary 1 in Negahban et al. (2012).

Lemma 19 *Under the condition of Lemma 17, we have*

$$\left\|\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right\|_1 \lesssim s\sqrt{\frac{\log p}{n_S}} + h,\ \left\|\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right\|_2^2 \lesssim s\frac{\log p}{n_S} + h\sqrt{\frac{\log p}{n_S}}.$$

with probability at least $1 - c_1\exp(-c_2\log p)$.

The next lemma comes from an application of Theorem 3.2 in van de Geer et al. (2014), with condition (D5) in van de Geer et al. (2014) replaced with the result in Lemma 19.

Lemma 20 *Under Assumption 13 and the condition of Lemma 17. Define* $\tau_j^2 := \boldsymbol{\Theta}_{jj}^{(k)}$. *For* $\hat{\tau}$ *and* $\hat{\boldsymbol{\Theta}}^{(k)}$ *obtained from solving problem (A.5), we have*

$$\left|\hat{\tau}_j^2 - \tau_j^2\right| \lesssim \left(\frac{s_j\log p}{n_S}\right)^{1/2} \vee \left(\frac{s\log p}{n_S} + h\left(\frac{\log p}{n_S}\right)^{1/2}\right)^{1/2}$$

and $\left|\hat{\boldsymbol{\Theta}}_j^{(k)}\boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{(k)}\left(\hat{\boldsymbol{\Theta}}_j^{(k)}\right)^\top - \tau_j^2\right| \lesssim P_{1n} \wedge P_{2n} + \left|\hat{\tau}_j^2 - \tau_j^2\right|$, *where*

$$P_{1n} = \left\|\boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{(k)}\right\|_\infty \left(\frac{(s_j^2 \vee s^2)\log p}{n_S} + h^2\right), P_{2n} = \Lambda_{\max}^2 \left(\frac{(s_j \vee s)\log p}{n_S} + h\left(\frac{\log p}{n_S}\right)^{1/2}\right),$$

with $\Lambda_{\max}^2$ *being the largest eigenvalue of* $\boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{(k)}$.

We start with the first term, $\hat{\boldsymbol{\Theta}}^{(k)}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})$. Let $\left(\boldsymbol{a}_j^{(k)}\right)^\top := \boldsymbol{e}_j^\top\hat{\boldsymbol{\Theta}}^{(k)}$. Plugging in the definition of $\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})$ and applying Assumption 4, we have for some universal constant $c > 0$,

$$P\left(\max_{1\le j\le p}\left|\frac{1}{2n_S}\sum_{i=1}^{n_S}\left(\boldsymbol{a}_j^{(k)}\right)^\top\nabla\ell^{(k)}(\boldsymbol{\beta}_*^{(k)}; z_i^{(k)})\right| > t \,\middle|\, \hat{\boldsymbol{\Theta}}^{(k)}\right) \le p\exp\left(-\frac{cn_S^2t^2}{\sigma^2 c_\Omega}\right)$$

where

$$c_\Omega := \max_{1\le j\le p}\left(\hat{\boldsymbol{\Theta}}^{(k)}\mathbb{E}\left(\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})[\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})]^\top\right)\left(\hat{\boldsymbol{\Theta}}^{(k)}\right)^\top\right)_{j,j} = \max_{1\le j\le p}\left(\hat{\boldsymbol{\Theta}}^{(k)}\boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{(k)}\left(\hat{\boldsymbol{\Theta}}^{(k)}\right)^\top\right)_{j,j}.$$

where recall we define $\boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{(k)} := \mathbb{E}\left[(\mathbf{X}_{\boldsymbol{\beta}}^{(k)})^\top(\mathbf{X}_{\boldsymbol{\beta}}^{(k)})/n_S\right]$. Therefore, to prove (A.18), it suffices to bound $c_\Omega/(n_S^2t^2)$ with $t = \sqrt{\frac{\log p}{n_S}} + h\sqrt{\frac{\log p}{n_S}}$. According to Lemma 20 and the conditions outlined in Theorem 11, we have $P_{1n} = O(h^2) + o(1)$ and $P_{2n} = o(1)$ and thus $c_\Omega = \max_{1\le j\le p}\tau_j^2 + O(h^2) + o(1)$. By Assumption 13, $\max_{1\le j\le p}\tau_j^2 = \max_{1\le j\le p}\boldsymbol{\Theta}_{jj}^{(k)}$ is bounded above. Therefore (A.18) holds.

Next, we aim at the second term, $\boldsymbol{\beta}^{(k)}$. Notice that with a slight abuse of notation, we have $\nabla^2\ell(z_i^{(k)};\boldsymbol{\beta}) = \mathbf{X}_i^{(k)}\nabla^2\rho(\boldsymbol{\beta})(\mathbf{X}_i^{(k)})^\top$. Applying Hölder's inequality and triangle inequality, we obtain

$$\begin{aligned}
\left\|\boldsymbol{\beta}^{(k)}\right\|_\infty &= \left\|\left(\boldsymbol{I} - \hat{\boldsymbol{\Theta}}^{(k)}\nabla^2\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{\mathrm{Int}}^{(k)})\right)\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)\right\|_\infty \\
&\le \left\|\left(\boldsymbol{I} - \hat{\boldsymbol{\Theta}}^{(k)}\nabla^2\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_L^{(k)})\right)\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)\right\|_\infty \\
&\quad + \left\|\hat{\boldsymbol{\Theta}}^{(k)}\left(\nabla^2\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_L^{(k)}) - \nabla^2\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{\mathrm{Int}}^{(k)})\right)\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)\right\|_\infty \\
&\le \underbrace{\left\|\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right\|_1}_{E_1}\underbrace{\max_{1\le j\le p}\left\|\hat{\boldsymbol{\Theta}}_j^{(k)}\nabla^2\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_L^{(k)}) - \boldsymbol{e}_j^\top\right\|_\infty}_{E_2} \\
&\quad + \underbrace{\frac{1}{n_S}\sum_{i=1}^{n_S}\left|\left(\nabla^2\rho(\hat{\boldsymbol{\beta}}_L^{(k)}) - \nabla^2\rho(\hat{\boldsymbol{\beta}}_{\mathrm{Int}}^{(k)})\right)(\mathbf{X}_i^{(k)})^\top\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)\right|}_{E_3}\underbrace{\left\|\hat{\boldsymbol{\Theta}}^{(k)}\mathbf{X}_i^{(k)}\right\|_\infty}_{E_4}.
\end{aligned}$$

where recall that $\hat{\boldsymbol{\Theta}}_j^{(k)}$ denotes the jth row of $\hat{\boldsymbol{\Theta}}^{(k)}$. We then proceed term by term.

For E_1, another application of Lemma 19 yields $E_1 \lesssim s\sqrt{\log p/n_S} + h$. As for E_2, from the optimality condition of nodewise regression problem (A.5), we have $E_2 \leq \tilde{\lambda}_{kj}/\hat{\tau}_j{}^2$. For details, one may check the derivation of (10) in van de Geer et al. (2014). From Lemma 20, we have $\hat{\tau}_j = \tau_j + o(1)$. By Assumption 13, $\tau_j = \mathbf{\Theta}^{(k)}_{jj}$ is bounded below. Therefore as we choose $\tilde{\lambda}_{kj} \asymp \sqrt{\log p/n_S}$, we have $E_2 \asymp \sqrt{\log p/n_S}$. As for E_3, according to the Lipschitz condition on $\nabla^2\rho(\cdot)$, we have

$$\begin{aligned}
E_3 =& \frac{1}{n_S}\sum_{i=1}^{n_S}\left|\left(\nabla^2\rho(\hat{\boldsymbol{\beta}}_L^{(k)}) - \nabla^2\rho(\hat{\boldsymbol{\beta}}_{\text{Int}}^{(k)})\right)(\mathbf{X}_i^{(k)})^\top\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)\right| \\
\leq& \frac{1}{n_S}\sum_{i=1}^{n_S}\left|(\mathbf{X}_i^{(k)})^\top\hat{\boldsymbol{\beta}}_L^{(k)} - (\mathbf{X}_i^{(k)})^\top\hat{\boldsymbol{\beta}}_{\text{Int}}^{(k)}\right|\left|(\mathbf{X}_i^{(k)})^\top\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)\right| \\
\leq& \frac{1}{n_S}\sum_{i=1}^{n_S}\left|(\mathbf{X}_i^{(k)})^\top\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)\right|^2 = \left\|\mathbf{X}^{(k)}\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)\right\|_2^2
\end{aligned}$$

With condition (D1) in Assumption 13, an combination of Lemma 13 in Loh and Wainwright (2011) and Lemma 19 yields

$$\left\|\mathbf{X}^{(k)}\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)\right\|_2^2 \lesssim \left\|\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right\|_2^2 + \frac{\log p}{n_S}\left\|\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right\|_1^2 \lesssim s\frac{\log p}{n_S} + h\sqrt{\frac{\log p}{n_S}}.$$

where the last inequality is based on the assumption that $s\log p/n_S = o(1)$ and $h\sqrt{\log p/n_S} = o(1)$. Therefore, we have $E_3 \lesssim s\log p/n_S + h\sqrt{\log p/n_S}$.

The analysis of E_4 follows similar arguments to those presented in the proof of Theorem 21 in Lee et al. (2017); for completeness, we restate the key steps here. By applying Lemma 20, we have:

$$\begin{aligned}
\left\|\hat{\mathbf{\Theta}}^{(k)}\mathbf{X}_i^{(k)}\right\|_\infty \leq& \max_{1\leq j\leq p}\left\|\hat{\mathbf{\Theta}}_j^{(k)}(\mathbf{X}^{(k)})^\top\right\|_\infty \lesssim \max_{1\leq j\leq p}\left\|\hat{\mathbf{\Theta}}_j^{(k)}(\mathbf{X}_{\boldsymbol{\beta}_*^{(k)}})^\top\right\|_\infty \\
\leq& \max_{1\leq j\leq p}\frac{1}{\hat{\tau}_j^2}\left\|\mathbf{X}_{\boldsymbol{\beta}_*^{(k)},j} - \mathbf{X}_{\boldsymbol{\beta}_*^{(k)},-j}\hat{\boldsymbol{\gamma}}_j\right\|_\infty \\
\lesssim& \max_{1\leq j\leq p}\frac{1}{\tau_j^2}\left\|\mathbf{X}_{\boldsymbol{\beta}_*^{(k)},j} - \mathbf{X}_{\boldsymbol{\beta}_*^{(k)},-j}\hat{\boldsymbol{\gamma}}_j\right\|_\infty \\
\lesssim& \max_{1\leq j\leq p}\left[\frac{1}{\tau_j^2}\left\|\mathbf{X}_{\boldsymbol{\beta}_*^{(k)},j} - \mathbf{X}_{\boldsymbol{\beta}_*^{(k)},-j}\boldsymbol{\gamma}_j\right\|_\infty + \frac{1}{\tau_j^2}\left\|\mathbf{X}_{\boldsymbol{\beta}_*^{(k)},-j}\right\|_\infty\left\|\hat{\boldsymbol{\gamma}}_j - \boldsymbol{\gamma}_j\right\|_1\right] \\
\lesssim& 1 + \frac{\max_{1\leq j\leq p} s_j\log p}{n_S}.
\end{aligned}$$

where for the last inequality we use condition (D1) in Assumption 13. Therefore, since $\max_{1\leq j\leq p} s_j\log p/n_S = o(1)$, we have E_4 bounded above. Integrating the above arguments leads to $\|\boldsymbol{\beta}^{(k)}\|_\infty \lesssim \frac{s\log p}{n_S} + h\sqrt{\frac{\log p}{n_S}}$ with probability larger than $1 - c_1\exp(-c_2\log p)$. This finishes the proof.

Proof of Lemma 18 Notice that $\nabla\tilde{\mathcal{L}}^{(k)}(\boldsymbol{\beta}) = \boldsymbol{\beta} - \tilde{\boldsymbol{\beta}}^{(k)}$. Therefore, following the proof of Lemma 14, by applying Hölder's inequality, we obtain

$$\begin{aligned}|\langle\nabla\mathcal{L}\left(\boldsymbol{\theta}_*\right), \boldsymbol{\Delta}\rangle| &= \sum_{k=1}^{K}\left|\left\langle\frac{n_S}{N}(\boldsymbol{\beta}_*^{(k)} - \tilde{\boldsymbol{\beta}}^{(k)}), \boldsymbol{\Delta}^{(k)}\right\rangle\right| + \left|\left\langle\sum_{k=1}^{K}\frac{n_S}{N}(\boldsymbol{\beta}_*^{(k)} - \tilde{\boldsymbol{\beta}}^{(k)}) + \frac{n_T}{N}\nabla\mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)}), \boldsymbol{\Delta}^{(0)}\right\rangle\right| \\ &\leq \sum_{k=1}^{K}\left\|\frac{n_S}{N}(\boldsymbol{\beta}_*^{(k)} - \tilde{\boldsymbol{\beta}}^{(k)})\right\|_\infty\left\|\boldsymbol{\Delta}^{(k)}\right\|_1 + \left\|\sum_{k=1}^{K}\frac{n_S}{N}(\boldsymbol{\beta}_*^{(k)} - \tilde{\boldsymbol{\beta}}^{(k)}) + \frac{n_T}{N}\nabla\mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)})\right\|_\infty\left\|\boldsymbol{\Delta}^{(0)}\right\|_1.\end{aligned}$$

Notice that we have

$$\tilde{\boldsymbol{\beta}}^{(k)} - \boldsymbol{\beta}_*^{(k)} = -\hat{\boldsymbol{\Theta}}^{(k)}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)}) + \underbrace{\left(\boldsymbol{I} - \hat{\boldsymbol{\Theta}}^{(k)}\nabla^2\mathcal{L}^{(k)}(\hat{\boldsymbol{\beta}}_{\mathrm{Int}}^{(k)})\right)\left(\hat{\boldsymbol{\beta}}_L^{(k)} - \boldsymbol{\beta}_*^{(k)}\right)}_{\boldsymbol{b}^{(k)}}. \quad \text{(A.35)}$$

In Lemma 17 we have shown that with probability larger than $1 - c_1\exp(-c_2\log p)$,

$$\left\|\hat{\boldsymbol{\Theta}}^{(k)}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)})\right\|_\infty \lesssim \sqrt{\frac{\log p}{n_S}} + h\sqrt{\frac{\log p}{n_S}} \quad \text{and} \quad \left\|\boldsymbol{\beta}^{(k)}\right\|_\infty \lesssim \frac{s\log p}{n_S} + h\sqrt{\frac{\log p}{n_S}}. \quad \text{(A.36)}$$

Thus in order to guarantee

$$\lambda_1 \geq \left\|-\frac{n_S}{N}\hat{\boldsymbol{\Theta}}^{(k)}\nabla\mathcal{L}^{(k)}(\boldsymbol{\beta}_*^{(k)}) + \frac{n_S}{N}\boldsymbol{\beta}^{(k)}\right\|_\infty,$$

it suffices to choose $\lambda_1 = c_k\left(\sqrt{\frac{n_S}{N}\frac{\log p}{N}} + \frac{s\log p}{N} + \frac{n_S}{N}\sqrt{\frac{\log p}{n_S}}h\right)$ for some sufficiently large constant c_k.

Next, we shift our focus to the second term. Similar to the arguments in Lemma 14 and Lemma 17 with $\left(\boldsymbol{a}_j^{(k)}\right)^\top := \boldsymbol{e}_j^\top\hat{\boldsymbol{\Theta}}^{(k)}$ for $k = 1, \ldots, K$ and $\left(\boldsymbol{a}_j^{(0)}\right)^\top := \boldsymbol{e}_j^\top$, we have

$$P\left(\max_{1\leq j\leq p}\left|\frac{1}{2N}\sum_{k=0}^{K}\sum_{i=1}^{n_k}\left(\boldsymbol{a}_j^{(k)}\right)^\top\nabla\ell^{(k)}(\boldsymbol{\beta}_*^{(k)}; z_i^{(k)})\right| > t \,\middle|\, \{\hat{\boldsymbol{\Theta}}^{(k)}\}_{k=1}^K\right) \leq p\exp\left(-\frac{cN^2t^2}{c_{\Omega'}}\right).$$

With $t = \sqrt{\frac{\log p}{N}} + h\sqrt{\frac{\log p}{N}}$, we similarly have $c_{\Omega'}/(N^2t^2)$ bounded above with probability larger than $1 - c_1\exp(-c_2\log p)$. This result together with (A.36) indicates that

$$\left\|\sum_{k=1}^{K}\frac{n_S}{N}(\boldsymbol{\beta}_*^{(k)} - \tilde{\boldsymbol{\beta}}^{(k)}) + \frac{n_T}{N}\nabla\mathcal{L}^{(0)}(\boldsymbol{\beta}_*^{(0)})\right\|_\infty \lesssim \sqrt{\frac{\log p}{N}} + \frac{Ks\log p}{N} + \sum_{k=1}^{K}\frac{n_S}{N}\sqrt{\frac{\log p}{n_S}}h,$$

so it suffices to choose $\lambda_0 = c_0\left(\sqrt{\frac{\log p}{N}} + \frac{Ks\log p}{N} + \sum_{k=1}^{K}\frac{n_S}{N}\sqrt{\frac{\log p}{n_S}}h\right)$ for some sufficiently large constant c_0. The proof is then finished.

Appendix C. Additional Simulation and Real Data Analysis

C.1 Scalability with Respect to Dimension p and Number of Source Tasks K

We next examine the scalability of *TransMission* and its distributed variant (*D-TransMission*) with respect to the feature dimension p and the number of source tasks K. Table A.3 reports the prediction and estimation performance across different feature dimensions $p \in \{500, 1000, 1500, 2000\}$, under both homogeneous and heterogeneous settings. Consistent with the observations in the main text, *TransMission* maintains comparable accuracy in homogeneous settings and demonstrates superior robustness under distribution shifts. The results show no significant degradation in performance even when p quadruples from 500 to 2000.

The computational complexity of the proposed joint optimization framework depends on the choice of optimization algorithm. With first-order type methods, the iteration complexity is dimension-independent. Within each iteration, the solution involves performing matrix-vector multiplications, leading to a computational complexity of $O(Kp)$, which indeed increases with both p and K. In *D-TransMission*, if node-wise Lasso is used for constructing the de-biased estimator, the computational cost would increase by a factor p, leading to a more expensive computational cost $O(Kp^2)$. However, in practice, the de-biased estimator can be replaced by other asymptotically unbiased estimators such as the SCAD estimator (Fan and Li, 2001), which reduces the computational complexity of *D-TransMission* back to $O(Kp)$.

We would like to note that in practice, one could first cluster the original datasets into K^* homogeneous groups, then perform *(D-)TransMission* on these $K^* \ll K$ clusters to account for inter-cluster distribution shifts. Such a clustering approach can alleviate the growing computational cost with K of the proposed method, but at the same time benefit from the robustness property.

C.2 Robustness under Heavy-Tailed Noise Distributions

To further evaluate the robustness of *TransMission* against conditional shifts arising from heterogeneous noise distributions, we conduct additional experiments where the target samples retain Gaussian noise while the source samples are drawn from heavy-tailed distributions. Specifically, the source task errors are generated from a Student-t distribution with degrees of freedom $t_{\mathrm{df}} \in \{3, 5\}$, while the target task follows the standard normal noise distribution. Table A.2 summarizes the results across these designs.

We observe that *TransMission* consistently attains the lowest prediction error and outperforms competing baselines even under substantial departures from normality. Interestingly, its performance advantage becomes apparent even under small h (i.e., mildly heterogeneous settings), indicating that the estimator effectively identifies and leverages transferable information early on, rather than requiring strong cross-task similarity. These findings corroborate the theoretical robustness of the proposed fused-regularization framework to heavy-tailed conditional shifts.

Table A.2: Performance comparison across methods under heavy-tailed or highly correlated designs. The column t_{df} denotes the degrees of freedom of the Student-t distribution used in the experiment design. The column ρ controls the correlation strength in the source covariate covariance matrices, where larger ρ corresponds to stronger correlations.

h	ρ	t_{df}	Agg-Lasso	Single-Lasso	Trans-GLM	TransHDGLM	Trans-Lasso	TransMission
5	0.5	3	0.345	0.891	0.384	0.377	0.360	**0.333**
5	0.5	5	0.261	0.899	0.279	0.261	0.273	**0.237**
5	0.6	3	0.400	0.889	0.449	0.445	0.405	**0.370**
5	0.6	5	0.267	0.896	0.288	0.272	0.276	**0.256**
5	0.7	3	0.443	0.897	0.492	0.512	0.442	**0.424**
5	0.7	5	0.321	0.891	0.337	0.328	0.316	**0.302**
5	0.8	3	0.521	0.896	0.548	0.637	0.518	**0.494**
5	0.8	5	0.415	0.893	0.431	0.436	0.416	**0.359**
10	0.5	3	0.514	0.893	0.519	0.517	0.512	**0.415**
10	0.5	5	0.488	0.895	0.439	0.424	0.491	**0.328**
10	0.6	3	0.595	0.894	0.598	0.556	0.584	**0.449**
10	0.6	5	0.461	0.895	0.439	0.402	0.465	**0.337**
10	0.7	3	0.628	0.898	0.620	0.618	0.618	**0.499**
10	0.7	5	0.508	0.897	0.501	0.447	0.500	**0.385**
10	0.8	3	0.676	0.897	0.678	0.704	0.669	**0.554**
10	0.8	5	0.629	0.894	0.609	0.533	0.621	**0.427**
15	0.5	3	0.805	0.894	0.676	0.654	0.772	**0.529**
15	0.5	5	0.856	0.886	0.624	0.617	0.827	**0.416**
15	0.6	3	0.898	0.894	0.778	0.691	0.862	**0.542**
15	0.6	5	0.793	0.895	0.627	0.573	0.772	**0.451**
15	0.7	3	0.896	0.894	0.782	0.742	0.854	**0.606**
15	0.7	5	0.798	0.894	0.694	0.591	0.770	**0.479**
15	0.8	3	0.912	0.893	0.832	0.795	0.899	**0.637**
15	0.8	5	0.889	0.893	0.782	0.645	0.846	**0.514**
20	0.5	3	1.174	0.896	0.861	0.784	1.104	**0.623**
20	0.5	5	1.290	0.894	0.826	0.817	1.235	**0.504**
20	0.6	3	1.263	0.898	0.927	0.815	1.215	**0.617**
20	0.6	5	1.217	0.894	0.820	0.740	1.177	**0.531**
20	0.7	3	1.218	0.895	0.899	0.833	1.122	**0.684**
20	0.7	5	1.154	0.895	0.871	0.730	1.104	**0.580**
20	0.8	3	1.190	0.894	0.955	0.872	1.162	**0.726**
20	0.8	5	1.164	0.893	0.918	0.762	1.081	**0.604**

Table A.3: Performance comparison across methods under different feature dimension p.

h	ρ	p	Agg-Lasso	Single-Lasso	Trans-GLM	TransHDGLM	Trans-Lasso	TransMission
10	0.100	1000	0.349	1.060	0.373	0.257	0.361	**0.236**
10	0.100	1500	0.343	1.158	0.358	0.267	0.362	**0.262**
10	0.100	2000	0.359	1.246	0.386	**0.300**	0.376	0.301
10	0.200	1000	0.378	1.055	0.383	0.288	0.396	**0.247**
10	0.200	1500	0.367	1.147	0.361	0.284	0.370	**0.266**
10	0.200	2000	0.357	1.241	0.360	0.275	0.387	**0.268**
10	0.300	1000	0.399	1.053	0.389	0.312	0.413	**0.266**
10	0.300	1500	0.407	1.142	0.408	0.321	0.405	**0.295**
10	0.300	2000	0.427	1.243	0.425	0.342	0.431	**0.319**
10	0.400	1000	0.409	1.060	0.406	0.310	0.411	**0.265**
10	0.400	1500	0.411	1.156	0.420	0.305	0.427	**0.285**
10	0.400	2000	0.437	1.239	0.434	0.345	0.460	**0.318**
15	0.100	1000	0.672	1.052	0.609	0.475	0.675	**0.308**
15	0.100	1500	0.658	1.149	0.608	0.507	0.674	**0.361**
15	0.100	2000	0.684	1.240	0.670	0.556	0.669	**0.427**
15	0.200	1000	0.764	1.052	0.633	0.526	0.773	**0.328**
15	0.200	1500	0.690	1.150	0.627	0.512	0.688	**0.375**
15	0.200	2000	0.686	1.238	0.630	0.496	0.707	**0.386**
15	0.300	1000	0.773	1.059	0.657	0.556	0.786	**0.363**
15	0.300	1500	0.747	1.154	0.675	0.561	0.750	**0.396**
15	0.300	2000	0.756	1.249	0.714	0.597	0.746	**0.426**
15	0.400	1000	0.764	1.061	0.677	0.542	0.744	**0.365**
15	0.400	1500	0.776	1.158	0.676	0.520	0.762	**0.409**
15	0.400	2000	0.756	1.239	0.715	0.565	0.777	**0.424**
20	0.100	1000	1.072	1.055	0.969	0.716	1.053	**0.381**
20	0.100	1500	1.050	1.150	0.961	0.755	1.032	**0.456**
20	0.100	2000	1.066	1.233	1.007	0.844	1.066	**0.555**
20	0.200	1000	1.274	1.061	0.997	0.786	1.229	**0.410**
20	0.200	1500	1.116	1.152	0.933	0.770	1.088	**0.480**
20	0.200	2000	1.087	1.255	0.991	0.767	1.083	**0.495**
20	0.300	1000	1.213	1.069	0.945	0.794	1.204	**0.455**
20	0.300	1500	1.151	1.154	0.987	0.808	1.116	**0.504**
20	0.300	2000	1.115	1.236	1.028	0.848	1.098	**0.531**
20	0.400	1000	1.180	1.050	0.967	0.784	1.120	**0.459**
20	0.400	1500	1.156	1.149	0.953	0.762	1.138	**0.497**
20	0.400	2000	1.110	1.233	1.000	0.818	1.131	**0.531**

Table A.4: Performance comparison under log-normal dense shift patterns. The parameter σ controls the concentration of shift energy: $\sigma \in \{0.5, 1.0, 1.5\}$ correspond to approximately 60%, 20%, and 6% effective sparsity, respectively. Bold denotes the best-performing method per row.

h	ρ	σ	Agg-Lasso	Single-Lasso	Trans-GLM	TransHDGLM	TransMission
10	0.1	0.5	0.305	0.902	0.293	0.274	**0.250**
10	0.1	1.0	0.307	0.895	0.306	0.251	**0.211**
10	0.1	1.5	0.301	0.896	0.269	0.181	**0.150**
10	0.2	0.5	0.364	0.886	0.314	0.327	**0.265**
10	0.2	1.0	0.353	0.897	0.300	0.285	**0.229**
10	0.2	1.5	0.318	0.895	0.263	0.205	**0.167**
10	0.3	0.5	0.413	0.887	0.320	0.369	**0.285**
10	0.3	1.0	0.411	0.894	0.319	0.347	**0.252**
10	0.3	1.5	0.372	0.887	0.284	0.230	**0.173**
10	0.4	0.5	0.388	0.895	0.328	0.361	**0.280**
10	0.4	1.0	0.395	0.894	0.330	0.319	**0.249**
10	0.4	1.5	0.365	0.893	0.298	0.237	**0.186**
15	0.1	0.5	0.614	0.894	0.482	0.534	**0.421**
15	0.1	1.0	0.610	0.894	0.486	0.478	**0.329**
15	0.1	1.5	0.602	0.892	0.422	0.329	**0.201**
15	0.2	0.5	0.763	0.896	0.525	0.616	**0.436**
15	0.2	1.0	0.750	0.895	0.503	0.522	**0.355**
15	0.2	1.5	0.658	0.894	0.417	0.360	**0.221**
15	0.3	0.5	0.858	0.893	0.547	0.671	**0.466**
15	0.3	1.0	0.839	0.884	0.543	0.596	**0.380**
15	0.3	1.5	0.745	0.881	0.430	0.404	**0.230**
15	0.4	0.5	0.805	0.897	0.541	0.636	**0.458**
15	0.4	1.0	0.806	0.895	0.536	0.564	**0.380**
15	0.4	1.5	0.729	0.892	0.464	0.392	**0.242**
20	0.1	0.5	1.007	0.895	0.692	0.780	**0.645**
20	0.1	1.0	1.032	0.891	0.727	0.702	**0.482**
20	0.1	1.5	1.008	0.892	0.608	0.518	**0.257**
20	0.2	0.5	1.292	0.892	0.806	0.842	**0.661**
20	0.2	1.0	1.293	0.893	0.770	0.756	**0.511**
20	0.2	1.5	1.155	0.892	0.616	0.544	**0.280**
20	0.3	0.5	1.420	0.895	0.809	0.883	**0.664**
20	0.3	1.0	1.409	0.882	0.823	0.816	**0.523**
20	0.3	1.5	1.239	0.885	0.630	0.598	**0.298**
20	0.4	0.5	1.353	0.894	0.765	0.852	**0.644**
20	0.4	1.0	1.357	0.885	0.812	0.786	**0.514**
20	0.4	1.5	1.220	0.892	0.639	0.579	**0.309**

Table A.5: Performance comparison under hierarchical shift structures across different correlation strengths. The column "Structure" denotes the hierarchical configuration used in the shift design.

h	ρ	Structure	Agg-Lasso	Single-Lasso	Trans-GLM	TransHDGLM	TransMission
10	0.1	10_40	9.460	0.912	4.536	1.219	**0.159**
10	0.1	20_80	4.615	0.918	3.560	1.106	**0.181**
10	0.1	5_100	18.921	0.918	5.052	1.293	**0.114**
10	0.2	10_40	11.017	0.911	7.597	1.223	**0.188**
10	0.2	20_80	6.251	0.918	5.406	1.148	**0.218**
10	0.2	5_100	26.453	0.917	8.880	1.260	**0.127**
10	0.3	10_40	11.377	0.915	7.104	1.238	**0.221**
10	0.3	20_80	6.266	0.896	5.553	1.133	**0.262**
10	0.3	5_100	23.759	0.915	9.646	1.247	**0.132**
10	0.4	10_40	9.496	0.915	6.844	1.223	**0.272**
10	0.4	20_80	6.609	0.913	5.575	1.147	**0.286**
10	0.4	5_100	23.982	0.900	10.707	1.268	**0.154**
15	0.1	10_40	19.650	0.915	6.951	1.277	**0.197**
15	0.1	20_80	9.024	0.912	5.721	1.237	**0.212**
15	0.1	5_100	40.037	0.905	7.851	1.288	**0.143**
15	0.2	10_40	22.254	0.921	13.244	1.261	**0.245**
15	0.2	20_80	12.587	0.923	9.513	1.250	**0.264**
15	0.2	5_100	56.353	0.914	15.694	1.285	**0.168**
15	0.3	10_40	23.992	0.920	13.307	1.262	**0.278**
15	0.3	20_80	13.105	0.902	10.546	1.229	**0.293**
15	0.3	5_100	50.864	0.915	18.544	1.269	**0.157**
15	0.4	10_40	20.128	0.924	12.731	1.245	**0.310**
15	0.4	20_80	13.772	0.918	10.589	1.233	**0.354**
15	0.4	5_100	51.419	0.903	20.651	1.290	**0.164**
20	0.1	10_40	33.582	0.900	9.792	1.278	**0.253**
20	0.1	20_80	15.051	0.908	8.408	1.256	**0.256**
20	0.1	5_100	69.614	0.896	11.426	1.323	**0.172**
20	0.2	10_40	38.054	0.915	20.796	1.276	**0.297**
20	0.2	20_80	21.175	0.917	15.093	1.246	**0.319**
20	0.2	5_100	98.803	0.896	23.902	1.267	**0.194**
20	0.3	10_40	40.681	0.921	21.166	1.269	**0.344**
20	0.3	20_80	22.027	0.895	17.008	1.242	**0.338**
20	0.3	5_100	89.358	0.896	30.822	1.286	**0.182**
20	0.4	10_40	35.034	0.920	21.655	1.291	**0.365**
20	0.4	20_80	23.315	0.914	17.549	1.240	**0.402**
20	0.4	5_100	89.898	0.899	31.777	1.285	**0.205**

Table A.6: Performance comparison under correlated shift settings across different correlation strengths. The column "Correlation" denotes the correlation level used in constructing the shift design.

h	ρ	Correlation	Agg-Lasso	Single-Lasso	Trans-GLM	TransHDGLM	TransMission
10	0.1	0.1	0.363	0.895	0.317	0.296	**0.234**
10	0.1	0.3	0.496	0.908	0.494	0.424	**0.282**
10	0.1	0.5	0.620	0.915	0.588	0.500	**0.353**
10	0.2	0.1	0.423	0.893	0.327	0.333	**0.239**
10	0.2	0.3	0.495	0.915	0.477	0.437	**0.277**
10	0.2	0.5	0.616	0.915	0.593	0.544	**0.360**
10	0.3	0.1	0.458	0.895	0.343	0.356	**0.242**
10	0.3	0.3	0.544	0.916	0.552	0.438	**0.301**
10	0.3	0.5	0.626	0.915	0.600	0.533	**0.345**
10	0.4	0.1	0.438	0.895	0.356	0.360	**0.259**
10	0.4	0.3	0.482	0.908	0.478	0.417	**0.281**
10	0.4	0.5	0.602	0.911	0.609	0.507	**0.342**
15	0.1	0.1	0.776	0.895	0.581	0.553	**0.351**
15	0.1	0.3	1.085	0.915	0.934	0.713	**0.443**
15	0.1	0.5	1.467	0.915	1.191	0.803	**0.632**
15	0.2	0.1	0.890	0.893	0.589	0.569	**0.335**
15	0.2	0.3	1.123	0.915	1.024	0.726	**0.450**
15	0.2	0.5	1.466	0.915	1.243	0.794	**0.601**
15	0.3	0.1	0.978	0.894	0.619	0.632	**0.360**
15	0.3	0.3	1.154	0.915	1.096	0.704	**0.453**
15	0.3	0.5	1.373	0.915	1.208	0.797	**0.609**
15	0.4	0.1	0.949	0.893	0.634	0.601	**0.381**
15	0.4	0.3	1.059	0.915	0.998	0.700	**0.443**
15	0.4	0.5	1.347	0.915	1.229	0.808	**0.607**
20	0.1	0.1	1.330	0.895	0.905	0.787	**0.474**
20	0.1	0.3	1.919	0.917	1.521	0.936	**0.608**
20	0.1	0.5	2.467	0.915	1.700	0.985	**0.789**
20	0.2	0.1	1.535	0.895	0.974	0.782	**0.463**
20	0.2	0.3	1.963	0.916	1.541	0.923	**0.598**
20	0.2	0.5	2.512	0.915	1.883	0.984	**0.772**
20	0.3	0.1	1.622	0.891	0.960	0.834	**0.486**
20	0.3	0.3	1.983	0.915	1.659	0.935	**0.617**
20	0.3	0.5	2.484	0.917	1.870	0.993	**0.785**
20	0.4	0.1	1.559	0.894	0.932	0.798	**0.489**
20	0.4	0.3	1.800	0.917	1.562	0.904	**0.637**
20	0.4	0.5	2.368	0.917	1.913	0.973	**0.804**

C.3 Robustness under More Realistic Shift Patterns

We further expanded our simulation experiments beyond sparse Gaussian contrasts. Specifically, we consider dense shifts, structured shifts, and correlated parameter contrasts $\boldsymbol{\delta}^{(k)}$. We consider the following three patterns for the parameter shift vector $\boldsymbol{\delta}^{(k)}$.

(i) Log-normal dense shift. For each task k, we set $\boldsymbol{\delta}_j^{(k)} = s_j \exp(Z_j)$ for all $j \in [p]$, where $s_j \sim \text{Uniform}\{-1, +1\}$ and $Z_j \sim \text{N}(\mu_\sigma, \sigma^2)$ i.i.d. The location μ_σ is calibrated so that $\mathbb{E}[\|\boldsymbol{\delta}^{(k)}\|^2] = h^2/50$. Larger σ concentrates energy into fewer coordinates despite all p being nonzero; we report effective sparsity as the fraction of coordinates accounting for 90% of total shift energy.

(ii) Hierarchical shift. For each task k, we sample a set of primary features P_k of size 10 and secondary features S_k of size 40 from the remaining coordinates. We set $\boldsymbol{\delta}_j^{(k)} \sim \text{N}(0, (3h/|P_k|)^2)$ for $j \in P_k$, $\boldsymbol{\delta}_j^{(k)} \sim \text{N}(0, (0.5h/|S_k|)^2)$ for $j \in S_k$, and $\boldsymbol{\delta}_j^{(k)} = 0$ otherwise. This induces a few large shifts and many smaller shifts.

(iii) Correlated shift. We select a common support $H \subset [p]$ with $|H| = 50$. For each $j \in H$, the vector $(\boldsymbol{\delta}_j^{(1)}, \ldots, \boldsymbol{\delta}_j^{(K)})^\top$ is sampled from a mean-zero multivariate normal distribution with covariance $(h/50)^2 \boldsymbol{\Sigma}_\delta$, where $\boldsymbol{\Sigma}_\delta$ has unit diagonal and constant off-diagonal correlation ρ_δ. This yields cross-task correlation in the non-transferable components.

As reported in Table A.4, A.5 and A.6, *TransMission* continues to outperform competing methods in most settings, particularly when the heterogeneity level (as measured by h) is high. These additional results demonstrate that the proposed method remains robust under more general and realistic forms of inter-task heterogeneity.