TransFusion: Covariate-Shift Robust Transfer Learning for High-Dimensional Regression

Zelin He
Dept. of Statistics
Penn State Univ.

Ying Sun
School of EECS,
Penn State Univ.

Jingyuan Liu
Dept. of Statistics
and Data Science,
SOE, Xiamen Univ.

Runze Li
Dept. of Statistics
Penn State Univ.

Abstract

The main challenge that sets transfer learning apart from traditional supervised learning is the distribution shift, reflected as the shift between the source and target models and that between the marginal covariate distributions. In this work, we tackle model shifts in the presence of covariate shifts in the high-dimensional regression setting. Specifically, we propose a two-step method with a novel fused-regularizer that effectively leverages samples from source tasks to improve the learning performance on a target task with limited samples. Nonasymptotic bound is provided for the estimation error of the target model, showing the robustness of the proposed method to covariate shifts. We further establish conditions under which the estimator is minimax-optimal. Additionally, we extend the method to a distributed setting, allowing for a pretraining-finetuning strategy, requiring just one round of communication while retaining the estimation rate of the centralized version. Numerical tests validate our theory, highlighting the method's robustness to covariate shifts.

Transfer learning is a technique that leverages knowledge from source tasks to improve learning performance in a related but possibly different target task (Torrey and Shavlik, 2010). In this paper, we consider the high-dimensional setting where the target sample size is much smaller than the number of features. In this context, applying transfer learning techniques to extract information from a larger pool of source samples can be particularly beneficial in identifying the model parameters. For example, in genetic studies of rare diseases, transferring information from larger, related set of studies could uncover highly disease-relevant genetic patterns (Taroni et al., 2019).

In contrast to learning from i.i.d. samples, a fundamental challenge in transfer learning is handling the distribution shifts between the source sample \((X_S, y_S)\) and the target sample \((X_T, y_T)\) (Pan and Yang, 2009). The discrepancy between the distributions typically shows in two ways: 1) model shift: \(P(y_S|X) \neq P(y_T|X)\), indicating a shift in the learning models, and 2) covariate shift: \(P(X_S) \neq P(X_T)\), indicating a shift in the marginal covariate distributions. In either case, models achieving small training errors on the source tasks may experience high risks on the target task (Lu et al., 2020). Therefore, to improve the learning performance of the target model \(P(y_T|X)\) using knowledge from the source samples, one should not only capture and correct the model shift but also be robust to the covariate shifts. In the high-dimensional setting, handling such differences becomes even more difficult due to accumulated noise and limited samples (Fan et al., 2020). This leads to the following question:

How to tackle model shifts in high-dimensional transfer learning while being robust to covariate shifts?

Apart from the challenge brought by the distribution shift, modern learning problems often involve datasets distributed across multiple computing nodes. In such a scenario, performing centralized training by pooling all the raw data in a single machine can be undesirable due to storage, communication, and privacy issues. This situation prompts another question:

How to transfer knowledge from distributed source datasets in a communication-efficient manner?

This paper proposes a solution for the above two questions for high-dimensional linear regression with \(K\)
source tasks, where the target model is \( p \)-dimensional with sparsity level \( s \). Our contributions are:

- **Covariate-Shift Robust Regularizer.** We propose a novel fused-regularizer achieving two purposes: it promotes sparse solutions for the high-dimensional model parameter while simultaneously capturing model shifts between source and target datasets. Our theoretical results further show that this regularizer can separate model shifts from shared patterns in a robust manner under covariate shifts.

- **Optimal Estimation Procedure.** Leveraging the proposed regularizer, we introduce a two-step procedure termed TransFusion. When the source tasks are sufficiently diverse, we show applying the first step on the source and target tasks jointly suffices to yield a fast rate of \( O(\frac{s \log p}{n_T^{1/3} + k_{n_S}} + \bar{h}\sqrt{\frac{\log p}{n_S}}) \), where \( n_T \) is the target sample size, \( n_S \) is the source sample size, and \( \bar{h} \) measures task similarity. The rate significantly improves over the one achieved on target task without transfer learning, i.e., the rate of \( O(\frac{s \log p}{n_T}) \), when \( n_S \gg \bar{h}n_T^2 \). For cases that do not meet the diversity criteria, TransFusion incorporates a second step refining the estimate on the target task, ensuring a rate of \( O(\frac{s \log p}{n_T^{1/3} + k_{n_S}} + \bar{h}\sqrt{\frac{\log p}{n_T}} \wedge \bar{h}^2) \), which is minimax-optimal when \( p \gg s \) and \( \bar{h} \) is relatively small.

- **Efficient Distributed Learning.** We develop a distributed variant of our method, termed \( D \)-TransFusion, requiring only one-shot communication of the pre-trained local models from source tasks nodes to target task node, significantly reducing communication overhead. More importantly, it offers the flexibility to quickly adapt the models to different downstream tasks while avoiding training from scratch. We further show that when the source sample size \( n_S \) is sufficiently large, \( D \)-TransFusion achieves the optimal statistical rate, matching its centralized counterpart.

**Related Works:** This paper develops transfer learning methods for high-dimensional regression problems under both model and covariate shifts. Related works can be broadly divided into the following categories.

**Domain Adaptation** methods primarily focus on handling covariate shifts, usually assuming the underlying models remain the same (Quinonero-Candela et al. (2008), Redko et al. (2020)). One prevalent approach in this category focuses on aligning the source and target covariate distributions by learning domain-invariant representations (Redko et al. (2020), Mansour et al. (2009), Cortes and Mohri (2011), Cortes and Mohri (2014)). Another line of research involves correcting estimators to address covariate shifts, often using the importance weighting (Quinonero-Candela et al. (2008), Sugiyama and Kawanabe (2012), Chen et al. (2016)). In contrast, we explicitly address model shifts and aim for robustness to covariate shifts.

**Multitask learning** aims to handle model shifts across multiple tasks and learn shared features to improve the performance of each task (Pan and Yang, 2009). In regression settings, regularization techniques are often employed to promote information transfer. Examples include the Frobenius and spectral norm (Argyriou et al. (2007), Tian et al. (2023)), mixed \( \ell_{2,1} \) norm (Lounici et al., 2009), hard-thresholding (Huang et al., 2023), and the total variation norm (Li and Sang (2019), Zhang et al. (2022), Tung and Song (2016)). These works typically require all 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, often with far fewer samples, is the primary focus.

**Transfer Learning** has been intensively studied under regression settings (Du et al. (2017), Lei et al. (2021), Lin and Reinherr (2022:2024)). However, most works are restricted to low-dimensional problems. Recently, transfer learning in the high-dimensional regression settings has been studied in Takada and Fujisawa (2020), Bastani (2021), Li et al. (2022) and Tian and Feng (2022). These works, however, deal with scenarios with only a single source or are sensitive to covariate shifts across multiple sources, and their learning accuracy degrades quickly if such shifts are severe. More recent works such as Li et al. (2023) and Liu (2023) attempt to mitigate the impact of covariate shifts. However, these methods either rely on strong assumptions or are computationally demanding. Specifically, Li et al. (2023) established the convergence rate of the proposed estimator assuming the empirical loss function in the high-dimensional setting is smooth, and computing the estimator requires solving a nonsmooth optimization problem with multiple constraints, while Liu (2023) assumes the target sample has a comparable size as the source sample.

In contrast, theoretical guarantees of our method are established under weaker, more practical conditions, and numerically it can be computed efficiently using algorithms such as iterative soft thresholding.

**Notation:** We use bold upper- and lowercase letters for matrices and vectors, respectively. For a matrix \( \mathbf{A} \in \mathbb{R}^{m \times n} \), we denote its \((i, j)\)-th element by \( A_{ij} \), maximum eigenvalue by \( \Lambda_{\text{max}}(\mathbf{A}) \), and minimum eigenvalue by \( \Lambda_{\text{min}}(\mathbf{A}) \). We let \( a \vee b \) denote \( \max\{a, b\} \) and \( a \wedge b \) denote \( \min\{a, b\} \). We use \( c, c_0, c_1, \ldots \) to denote generic constants independent of \( n, p \) and \( K \). Let \( a_n = O(b_n) \) and \( a_n \lesssim b_n \) denote \(|a_n/b_n| \leq c \) for some constant \( c \) when \( n \) is large enough: \( a_n = o(b_n) \) and \( b_n \gg a_n \) if \( a_n = O(c_n b_n) \) for some \( c_n \to 0 \); \( a \asymp b \) if \( a = O(b) \) and \( b = O(a) \).
1 Preliminaries

We consider a transfer learning problem involving one target task and $K$ source tasks. For the target task, we observe a sample $(X_i^{(0)}, y_i^{(0)})$ generated from the target model

$$y_i^{(0)} = (X_i^{(0)})^T \beta^{(0)} + \epsilon_i^{(0)}, \quad i = 1, \ldots, n_T,$$

where $\beta^{(0)} \in \mathbb{R}^p$ is the parameter of primary interest and $\epsilon_i^{(0)}$ is the observation noise. We focus on a high-dimensional scenario where the dimension $p$ is much larger than the target sample size $n_T$, yet the ground truth $\beta^{(0)}$ is a sparse vector with $s := \|\beta^{(0)}\|_0$ nonzero elements, which is much smaller than $p$, i.e., $p \gg s$.

In addition to the target sample, we also have access to $K$ source samples $\{(X_i^{(k)}, y_i^{(k)})\}_{k=1}^K$, generated from the source model

$$y_i^{(k)} = (X_i^{(k)})^T \beta^{(k)} + \epsilon_i^{(k)}, \quad i = 1, \ldots, n_S, \quad k = 1, \ldots, K.$$

For the $k$-th source model, $\beta^{(k)} \in \mathbb{R}^p$ is the unknown source task-specific parameter, and $\epsilon_i^{(k)}$ accounts for the observation noise. For simplicity, we assume the source samples have the same size $n_S$.

Our goal is to estimate $\beta^{(0)}$ using both the target and source samples under the challenging scenario where distributions of the samples are heterogeneous, as characterized by both model and covariate shift described next.

Model Shift. In our context, the model shift is the situation where each source model differs from the target model, and is measured by $\delta^{(k)} := \beta^{(k)} - \beta^{(0)}$ for $1 \leq k \leq K$. Throughout the paper, we refer $\delta^{(k)}$ as the “parameter contrast” or “task-specific signal”. A source task is considered informative for transfer learning if $\delta^{(k)}$ is relatively small. Formally, let $\beta := ((\beta^{(0)})^T, (\beta^{(1)})^T, \ldots, (\beta^{(K)})^T)^T \in \mathbb{R}^{(K+1)p}$, we assume $\beta$ belongs to the following parameter space

$$\Theta(s, h) := \left\{ \beta : \|\beta^{(0)}\|_0 \leq s, \|\beta^{(k)} - \beta^{(0)}\|_1 \leq h_k \right\},$$

with $h := (h_1, \ldots, h_K)^T$. In (1), the informative level of the $k$-th source task is quantified by the $l_1$-sparsity of $\delta^{(k)}$, and is upper bounded by a factor $h_k \geq 0$.

Remark 1. We choose an $l_1$-sparse constraint for the high-dimensional contrast $\beta^{(k)} - \beta^{(0)}$, as it aligns well with practical applications where model shifts typically spread over multiple dimensions but their overall magnitude does not grow too fast. The results in the paper can be naturally extended to a general $l_q$-sparse case for $q \in [0, 1]$.

Covariate Shift. In addition to the shift in the model parameters, we also consider the covariate shift, defined as the difference in the distributions of $X_i^{(k)}$s across the tasks. In this work, we only impose the following mild tail condition on the distribution of $X_i^{(k)}$s but allow other distribution characteristics, such as the covariance structures, to vary across different tasks.

Assumption 1 (Sub-Gaussian designs). For any $0 \leq k \leq K$, $X_i^{(k)}$s are independent sub-Gaussian random vectors with mean zero and covariance $\Sigma^{(k)}$. Furthermore, there exists some universal constant $c$ such that $1/c \leq \min_{0 \leq k \leq K} \Lambda_{\min}(\Sigma^{(k)}) \leq \max_{0 \leq k \leq K} \Lambda_{\max}(\Sigma^{(k)}) \leq c$. Finally, we assume that the random noises follow independent Gaussian distributions, a typical assumption for high-dimensional regression analysis.

Assumption 2 (Gaussian random errors). For all $0 \leq k \leq K$, the $\epsilon_i^{(k)}$s are independent Gaussian random variables with zero mean and uniformly upper bounded variance, and are independent of $X_i^{(k)}$s.

2 Covariate-Shift Robust Transfer Learning

We now introduce a method called TransFusion (Transfer Learning with a Fused-Regularization), designed to address high-dimensional model shifts in the presence of covariate shifts, thereby transferring knowledge from source tasks to the target task. The method consists of two steps. First, we perform a co-training step using both source and target samples, leveraging the $l_0$-sparsity of $\beta^{(0)}$ and the $l_1$-sparsity of the contrast $\delta^{(k)}$. We show that when the source tasks are sufficiently diverse, see Definition 1, performing the first step of the TransFusion method suffices to ensure a fast rate. When such a condition is not met, we further perform a second step by fine-tuning the model on the target dataset. The method is shown to be rate-optimal and robust to covariate shifts.

2.1 Step 1: Co-Training

We start with the first step, a co-training step involving both target and source samples. The challenge is tackling the distribution shifts while extracting the shared pattern between source and target samples to estimate $\beta^{(0)}$. Pooling all data as i.i.d. samples leverages larger sample size and reduces noise, but suffers from large bias if the source and target distributions differ significantly. In contrast, training exclusively on the target sample prevents any information transfer from the source samples. It is therefore critical to strike a balance between these two extremes to improve the estimation of $\beta^{(0)}$. To this end, we propose
TransFusion: Covariate-Shift Robust Transfer Learning for High-Dimensional Regression

a co-training step that estimates $\beta^{(k)}_s$ by solving the following problem:

$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^{(K+1)p}} \left\{ \frac{1}{2N} \sum_{k=0}^{K} \| y^{(k)} - X^{(k)} \beta^{(k)} \|^2_2 + \lambda_0 \left( \| \beta^{(0)} \|_1 + \sum_{k=1}^{K} a_k \| \beta^{(k)} - \beta^{(0)} \|_1 \right) \right\}, \quad (2)$$

where $N = Kn_S + n_T$ is the total sample size, $\lambda_0$ is the tuning parameter and $\{a_k\}_{k=1}^{K}$ are weights that will be specified later. In (2), the first term measures the average fitness of the models with parameter $\{\beta^{(k)}\}_{k=0}^{K}$, while the fused-regularization term simultaneously promotes the sparsity of $\beta^{(0)}$ and captures the $\ell_1$-sparse contrast between $\beta^{(0)}$ and $\beta^{(k)}$ by penalizing their difference.

We construct the first-step estimator as $\hat{w} = \frac{n_S}{N} \sum_{k=1}^{K} \hat{\beta}^{(k)} + \frac{n_T}{N} \hat{\beta}^{(0)}$. The motivation for this estimator is twofold: first, averaging across both target and source estimators utilizes the full sample, yielding an estimator with low variance. In addition, when the source datasets are sufficiently diverse, the bias of $\hat{w}$ is small. When the reduction in variance dominates the increase in bias, the one-step estimator $\hat{w}$ serves as a promising estimator of $\beta^{(0)}$ than using the target sample alone.

We now formally characterize the source task diversity.

**Definition 1** (Source task diversity). Given $\beta \in \Theta(s, h)$, we quantify the diversity across source tasks with the metric $\| \frac{n_T}{N} \sum_{k=1}^{K} \delta^{(k)} \|_1 \leq \varepsilon_D$, where $\delta^{(k)} := \beta^{(k)} - \beta^{(0)}$ is the task-specific signal.

A small $\varepsilon_D$ implies that $\{\beta^{(k)}\}_{k=1}^{K}$ are centered around $\beta^{(0)}$ and cover all the directions, such that the average parameter $w := \frac{n_T}{N} \sum_{k=1}^{K} \beta^{(k)} + \frac{n_T}{N} \beta^{(0)}$ does not align with any direction significantly more than $\beta^{(0)}$. This kind of assumption is commonly imposed in transfer learning settings (Du et al. (2020), Tripuraneni et al. (2020)).

We proceed to establish the statistical estimation rate for $\hat{w}$ under the setting of diverse source tasks.

**Theorem 1.** Under Assumption 1 and 2, if $n_S \gg s \log p$, then by choosing $\lambda_0 = c_0 \sqrt{\log p / N}$ for some universal constant $c_0$ and $a_k = 8\sqrt{n_S / N}$, we have

$$\| \hat{w} - w \|^2_2 \lesssim \frac{s \log p}{N} + (1 + v_n)\hat{h} \sqrt{\frac{\log p}{n_S}}, \quad (3)$$

and

$$\| \hat{w} - \beta^{(0)} \|^2_2 \lesssim \frac{s \log p}{N} + (1 + v_n)\hat{h} \sqrt{\frac{\log p}{n_S}} + \varepsilon_D^2, \quad (4)$$

with probability at least $1 - c_1 \exp(-c_2 n_T) - c_3 \exp(-c_4 \log p)$, where $v_n := \sqrt{K^2 \log p / n_S}$ and $\hat{h} := \frac{n_S}{N} \sum_{k=1}^{K} h_k$.

Let us break down the upper bound provided by equation (4). The first term, $s \log p / N$, represents the rate from estimating an $s$-sparse coefficient $\beta^{(0)}$ based on $N = Kn_S + n_T$ i.i.d. samples. This term reveals the benefit of using both the source and target datasets for estimating the target parameter $\beta^{(0)}$. The second term, $\hat{h} \sqrt{\log p / n_S}$, accounts for the estimation error of $\delta^{(k)}$ unique to each source task and thus is limited by the source sample size $n_S$. The factor $v_n$ is sample dependent and is negligible when $n_S \gg \hat{h}^2 K^2 \log p$.

The first two terms together quantify the estimation error $\| w - \hat{w} \|^2_2$. The third term, $\varepsilon_D^2$, measures the difference between $w$ and $\beta^{(0)}$ and contributes to the bias introduced by averaging. Notably, to obtain the bound (4), we do not require a homogeneous distribution of the covariates $X^{(k)}$s but only impose the mild tail assumption as outlined in Assumption 1, and the bound does not depend on the target sample size $n_T$.

As a comparison, if we apply the LASSO regression on the target data, the estimation error is of order $O(s \log p / n_T)$. Therefore, if $N \gg n_T$, $n_S \gg \hat{h}^2 (n_T^2 \vee K^2 \log p)$ and $\varepsilon_D \ll \sqrt{\log p / n_T}$, that is, the source tasks are sufficiently diverse with adequately large sample size, then one-step TransFusion method achieves a sharper estimation rate. This corroborates our design intuition and quantitatively shows the benefit of transferring information from diverse source tasks even under covariate shifts.

**Remark 2** (Adaptive version of TransFusion). In Theorem 1, we choose a weight $a_k$ that does not depend on $h_k$, as we treat $h_k$ as an unknown priori. As a compromise, the estimation rate depends on $\hat{h}$, the averaged magnitude of model shifts. In fact, for a general choice of $\lambda_0$ and $a_k$, TransFusion could yield a bound

$$\| \hat{w} - \beta^{(0)} \|^2_2 \lesssim \frac{s \log p}{N} + \sum_{k=1}^{K} a_k \lambda_0 h_k + \varepsilon_D^2$$

under certain conditions (cf. Lemma 5). So if we have some information on $h_k$, we may adjust $a_k$ accordingly, focusing more on informative datasets with small $h_k$ and less or not at all on those with large $h_k$. In such cases, this adaptive version of TransFusion could potentially yield a fast estimation rate that is less sensitive to the magnitude of model shifts.

**Remark 3** (Scalability with task number $K$). TransFusion incorporates a novel fused regularizer capturing the task-specific signals in the joint learning step. This technique robustifies the method against covariate-shift
and introduces a dependency of the convergence rate on $K$ as a tradeoff. Specifically, the convergence rate of the first-step estimator is given by (3) with $v_n := \sqrt{K^2 \log p/nS \hat{h}}$ due to the non-strong convexity of the local empirical loss (cf. Lemma 4). If we increase $K$ while fixing $n_S$, for large $K$, the sum will be dominated by the second term, which grows with $K$. Otherwise, if we increase $n_S$ with $K$, TransFusion would have a consistent error improvement. This is supported by the simulation results and discussions in Appendix E.

2.2 Step 2: Local Debias

Despite its merits, the one-step TransFusion method may experience large bias when $\varepsilon_D$ is large. This is especially the case when the source tasks exhibit a skewed model shift towards one specific direction than $\beta^{(0)}$. In such cases, we employ an additional debias step that refines the initial estimator $\hat{w}$ and mitigates the impact of $\varepsilon_D$. Specifically, we correct $\hat{w}$ using the target sample as:

$$\hat{\delta} \in \arg\min_{\delta \in \mathbb{R}^p} \left\{ \frac{1}{2nT} \| y^{(0)} - X^{(0)} \hat{w} - X^{(0)} \delta \|_2^2 + \lambda \| \delta \|_1 \right\},$$

$$\hat{\beta}_{\text{TransFusion}}^{(0)} = \hat{w} + \hat{\delta}. \quad (5)$$

Next, we demonstrate that with an appropriate choice of estimation strategy and tuning parameters, we can attain an optimal estimation rate of $\beta^{(0)}$ without requiring a small $\varepsilon_D$. Define the event

$$A = \{ s \log p/n_S \geq \bar{h} \sqrt{\log p/n_T} \}, \quad (6)$$

and $A^c$ as its complement. The following theorem establishes an upper bound on the estimation error for the two-step TransFusion algorithm.

**Theorem 2.** Under the assumptions of Theorem 1, if $n_T \geq s \log p$, $n_S \geq K^2 s \log p$ and $\bar{h} \sqrt{\log p/n_T} + K s \log p/n_S \geq o(1)$, then by choosing the parameters

$$\lambda_0 = c_0 \left( \frac{\log p}{N} \mathbb{1}_A + \frac{\log p}{n_S} \mathbb{1}_{A^c} \right),$$

$$a_k = 8 \left( \frac{n_S}{N} \mathbb{1}_A + \frac{n_S}{N} \mathbb{1}_{A^c} \right),$$

and $\bar{\lambda} = c_1 \sqrt{\log p/n_T}$ for some universal constants $c_0$ and $c_1$, the solution of the two-step TransFusion method satisfies

$$\| \hat{\beta}_{\text{TransFusion}}^{(0)} - \beta^{(0)} \|_2^2 \leq \frac{s \log p}{N} + \bar{h} \sqrt{\frac{\log p}{n_T}}, \quad (7)$$

with probability at least $1 - c_2 \exp(-c_3 \log p)$.

By comparing the results of Theorem 1 and 2 [cf. (4) and (7)], we see when $\sqrt{\log p/n_T} \lesssim \varepsilon_D/\bar{h}$, performing the second step improves the estimation precision. The ratio $\varepsilon_D/\bar{h}$ quantifies the normalized (by the magnitude of $\delta^{(k)}$) source task diversity, and thus our result shows applying the second step is beneficial for non-diverse source tasks. Note that the condition on the sample size $n_T$ and $n_S$ in Theorem 2 is stronger than Theorem 1. Such a condition is required to ensure the target-specific signals $\delta^{(k)}$ being accurately captured to perform the correction in (5).

On the other hand, if $\sqrt{\log p/n_T} \gtrsim \varepsilon_D/\bar{h}$ applying the second step may even harm the model performance. Therefore, choosing between the one-step and two-step TransFusion methods carefully is key to getting the optimal estimation results. The following corollary provides guidelines for making this choice.

**Corollary 1.** Under the assumptions of Theorem 2, if we apply the one-step TransFusion method when $n_T \lesssim \log p/\bar{h}^2$ and apply the two-step TransFusion method otherwise, then obtained estimator $\hat{\beta}_{\text{TransFusion}}^{(0)}$ satisfies

$$\| \hat{\beta}_{\text{TransFusion}}^{(0)} - \beta^{(0)} \|_2^2 \lesssim \frac{s \log p}{N} + \bar{h} \sqrt{\frac{\log p}{n_T}} \land \bar{h}^2, \quad (8)$$

with probability at least $1 - c_2 \exp(-c_3 \log p)$.

Next, we establish the minimax optimality of the above strategy under certain conditions. The following result follows from minor modifications of Theorem 2 in Li et al. (2022).

**Proposition 1.** Under Assumption 1 and Assumption 2, if $N \gg s \log p, h_k \sim \bar{h}$ and $\bar{h} \sqrt{\log p/n_T} = o(1)$, then any estimator $\hat{\beta}'$ that is a measurable function of the sample $\{(X^{(k)}, y^{(k)})\}_{0 \leq k \leq K}$ satisfies

$$\inf_{\hat{\beta}'} \sup_{\beta'^{\star} \in \Omega(s, h)} \| \hat{\beta}' - \beta'^{\star} \|_2^2 \gtrsim \frac{s \log p}{N} + \frac{s \log p}{n_T} \land \bar{h} \sqrt{\frac{\log p}{n_T}} \land \bar{h}^2, \quad (9)$$

with probability at least $1/2$.

Comparing with the upper bound (8), we can conclude that, given the conditions outlined in Theorem 2, if source datasets are sufficient informative such that $h \lesssim s \sqrt{\log p/n_T}$, then the proposed procedure is minimax optimal, even under covariate shifts.

**Remark 4 (Implementation of TransFusion).** Notice that although the two-step TransFusion method only involves one tuning parameter in each step, as discussed in Theorem 2 and Corollary 1, it relies on a dichotomous strategy that depends on the value of $\bar{h}$. However, it can still be practically applied without knowing $\bar{h}$ in advance by implementing both choices and selecting the one with smaller validation error. On the computation front, the global minimizer of each
TransFusion step can be efficiently found by numerical algorithms such as iterative soft-thresholding. The implementation details are provided in Appendix D.

2.3 Understanding the Robustness of the TransFusion Method to Covariate Shift

In this section, we discuss the underlying mechanisms that make the TransFusion method robust to covariate shifts via comparing with the two-step method proposed in Li et al. (2022) and Tian and Feng (2022). While both methods aim to address the high-dimensional transfer learning problem, we take a significantly different approach in the first co-training step.

In their approach, the first step pools samples from both target and source tasks, and performs a sparse regression to obtain the initial estimator. In the linear regression setting, this estimator comes with an asymptotic bias expressed as

$$\delta_{\text{Pooling}} := \left( \sum_{k=1}^{K} \Sigma^{(k)} \right)^{-1} \sum_{k=1}^{K} \Sigma^{(k)} \delta^{(k)}.$$  \hfill (10)

Due to the weights introduced by the covariance matrices, the contrast $\delta^{(k)}$’s can be amplified by a factor of $C_\Sigma$, defined as

$$C_\Sigma := 1 + \max_{j \leq p} \max_{k} \left\{ e_j^T (\Sigma^{(k)} - \Sigma^{(0)}) \left( \sum_{1 \leq k \leq K} \frac{1}{K} \Sigma^{(k)} \right)^{-1} \right\}.$$  \hfill (11)

Consequently, in the linear regression setting, their estimator yields the following estimation rate (Li et al., 2022, Theorem 4):

$$s \log p \over N + \left( C_\Sigma \sqrt{\frac{\log p}{n_T}} \right) \land \left( C_\Sigma^2 \hat{\delta}^2 \right).$$

When the $\Sigma^{(k)}$’s are dissimilar, the factor $C_\Sigma$ can diverge with dimension $p$ even if Assumption 1 holds, considerably deteriorating the estimation accuracy. See Appendix C for a detailed discussion.

In contrast, as shown in (8), our method is robust to such covariate heterogeneity and thus doesn’t involve the $C_\Sigma$ factor. This is achieved by incorporating a fused-regularizer, allowing us to accurately capture task-specific signals under covariate shifts. Solving the objective leads to an initial estimator with asymptotic bias

$$\delta_{\text{TransFusion}} := \frac{1}{K} \sum_{k=1}^{K} \delta^{(k)},$$

which is free from the impact of the covariance matrices. This bias is much smaller than $\delta_{\text{Pooling}}$ under covariate shift settings with a large $C_\Sigma$.

3 D-TransFusion: Distributed Transfer Learning in One-Shot

In this section, we consider the distributed transfer learning problem where the target and $K$ source samples are stored by different computing nodes. Such a setting is of primary interest in learning problems involving a massive amount of training data, where brute-force pooling the raw data is not admissible due to practical constraints such as storage limitation, communication cost, and privacy concerns.

This motivates us to consider developing a communication-efficient distributed TransFusion method, termed D-TransFusion. Our method is based on TransFusion and leverages the idea of divide-and-conquer to facilitate communication efficiency, aiming to achieve a comparable estimation error as TransFusion but using only one-shot communication. Specifically, D-TransFusion consists of the following two steps.

**Step 1.** Each node $k$ computes an estimator $\hat{\beta}^{(k)}$ (to be specified later) locally based on source sample $(X^{(k)}, y^{(k)})$ and transmits it to the target node. The target node then aggregates them with its own sample $(X^{(0)}, y^{(0)})$ via solving the following problem:

$$\hat{\beta}_C \in \arg\min_{\beta \in \mathbb{R}^{(K+1)p}} \left\{\frac{1}{2N} \sum_{k=1}^{K} \| \sqrt{n_S} (\hat{\beta}^{(k)} - \beta^{(k)}) \|^2 + \frac{1}{2} \| y^{(0)} - X^{(0)} \hat{\beta}^{(0)} \|^2 + \lambda_0 R(\beta) \right\},$$  \hfill (11)

where $R(\beta) := \| \beta^{(0)} \|_1 + \sum_{k=1}^{K} a_k \| \beta^{(k)} - \beta^{(0)} \|_1$. With the solution $\hat{\beta}_C$, the target node computes $\hat{\omega}_C = \frac{n_0}{K} \sum_{k=1}^{K} \hat{\beta}^{(k)} + \frac{n_0}{K} \hat{\beta}^{(0)}$.

**Step 2.** The target node corrects $\hat{\omega}_C$ on its local sample $(X^{(0)}, y^{(0)})$ by solving

$$\hat{\delta}_C \in \arg\min_{\delta \in \mathbb{R}^p} \left\{\frac{1}{2n_T} \| y^{(0)} - X^{(0)} \hat{\omega}_C - X^{(0)} \delta \|^2 + \lambda \| \delta \|_1 \right\},$$

and outputs the estimator $\hat{\beta}^{(0)}_{\text{D-TransFusion}} = \hat{\omega}_C + \hat{\delta}_C$.

Comparing with (2) and (5), we can see that D-TransFusion differs from the centralized TransFusion method only in the first step. To avoid the involvement of source samples, D-TransFusion replaces the least square loss $\| y^{(k)} - X^{(k)} \hat{\beta}^{(k)} \|^2$ by the squared loss $\| \sqrt{n_S} (\hat{\beta}^{(k)} - \beta^{(k)}) \|^2$, wherein $\hat{\beta}^{(k)}$ serves as a “pseudo sample” summarizing the information of $\beta^{(k)}$ that the $k$-th source sample contains. By doing so, only one-shot communication is required to transmit the summary statistics $\hat{\beta}^{(k)}$ from the source to the target node, significantly reducing the communication overhead. Here, $\hat{\beta}^{(k)}$ is carefully selected as a de-biased
LASSO estimator (Javanmard and Montanari, 2014) that minimizes estimator variance while also controlling the bias under a given threshold. See Appendix G for a detailed discussion. This choice ensures that D-TransFusion can significantly reduce communication overhead while achieving the minimum loss of sample efficiency compared to centralized TransFusion.

More importantly, D-TransFusion allows for pre-training on each source data nodes before transfer learning. The decoupling of training on the source and target samples eliminates the need for training from scratch when the target samples change, and thus enhances the model’s adaptability to downstream tasks.

We now establish the statistical precision of the one-step D-TransFusion method. Define \( \delta_k = s \log p + n_S \sqrt{\frac{\log p}{n_S} h_k} \) and \( \delta_0 = K \log p + \sqrt{\frac{\log p}{n_S} h}. \) The following theorem provides an upper bound for the estimation error of the one-step D-TransFusion estimator \( \hat{\mathbf{w}}_C. \)

**Theorem 3.** Under Assumption 1 and 2 and the assumptions \( n_S \gg K^2 \log p, n_S \gg (\hat{h}^2 \vee K^2) s \log p, h_k \leq h, \) if we construct \( \{\hat{\beta}^{(k)}\}_{k=1}^K \) through (57) and (58) with parameters \( \lambda_k = \mu_k = c_1 \sqrt{\log p n_S}, \) and solve problem (11) with parameters \( \lambda_0 \) and \( \{a_k\}_{k=1,...,K} \) chosen such that

\[
\lambda_0 = c_0 \left( \sqrt{\frac{\log p}{N}} + \delta_0 \right),
\]

\[
a_k \lambda_0 = c_0 \left( 8 \sqrt{\frac{h}{h_k}} \right) \left( \sqrt{\frac{n_S \log p}{N}} + \delta_k \right),
\]

for some universal constant \( c_0 \) and \( c_1, \) then with probability at least \( 1 - c_2 \exp(-c_3 \log p), \)

\[
\| \hat{\mathbf{w}}_C - \mathbf{\beta}^{(0)} \|^2 \lesssim s \frac{\log p}{N} + \sqrt{\frac{\log p}{n_S}} h + \varepsilon_D^2 + s \delta_0^2 + \sum_{k=1}^K \delta_k h_k,
\]

if we further assume \( h = O(1), \) then we have

\[
\| \hat{\mathbf{w}}_C - \mathbf{\beta}^{(0)} \|^2 \lesssim s \frac{\log p}{N} + \sqrt{\frac{\log p}{n_S}} h + \varepsilon_D^2.
\]

The second step of D-TransFusion can be designed in analogue to that of TransFusion. Recall the event \( A = \{s \log p / n_S \geq \bar{h} / \sqrt{\log p / n_T} \} \) defined in (6). The following theorem establishes the statistical estimation rate of the final estimator obtained from the two-step D-TransFusion method.

**Theorem 4.** Under the assumptions of Theorem 3, if further assume \( n_T \gtrsim s \log p, \bar{h} / \sqrt{\log p / n_T} = o(1), \) then by choosing \( \lambda_0 \) and \( \{a_k\}_{k=1,...,K} \) such that

\[
\lambda_0 = c_0 \left( \sqrt{\frac{\log p}{N}} + \sqrt{\frac{\log p}{n_S}} + \delta_0 \right),
\]

\[
a_k \lambda_0 = c_0 \left( 8 \sqrt{\bar{h} / h_k} \right) \left( \sqrt{\frac{n_S \log p}{N}} + \delta_k \right),
\]

and \( \lambda = c_1 \sqrt{\log p / n_T} \) for some universal constants \( c_0 \) and \( c_1, \) we have

\[
\| \hat{\mathbf{\beta}}^{(0)}_{\text{D-TransFusion}} - \mathbf{\beta}^{(0)} \|^2 \lesssim \frac{\log p}{N} + \sqrt{\frac{\log p}{n_T}} \bar{h},
\]

with probability at least \( 1 - c_2 \exp(-c_3 \log p). \)

Theorem 4 ensures that the two-step D-TransFusion method achieves a statistical rate of the same order as the centralized two-step TransFusion method under the previously discussed additional conditions. By employing similar reasoning as in Corollary 1 and Proposition 1, we can further establish conditions under which D-TransFusion is minimax optimal. These results demonstrate that D-TransFusion is an efficient and robust solution when dealing with large-scale distributed datasets with covariate shifts.

**Remark 5** (The efficacy of D-TransFusion). D-TransFusion aims to address the scenario where the source datasets are not co-located and cannot to be merged. This differs from the traditional distributed computing paradigm, where one splits the whole data into parts and parallelizes the cost due to the large data size. As for the implementation cost, since \( K \) debiased lasso estimators are computed in step 1 and transmitted in step 2, it requires per source node storing and transmitting a \( p \)-dimensional vector. The computation cost readily follows that of the debiased and standard lasso, provided in Lee et al. (2017). Although concerns may arise about the computation cost of the debiased Lasso, it is noteworthy that under mild conditions, the de-biased lasso estimator \( \hat{\beta}^{(k)} \) can be replaced by other asymptotically unbiased estimator such as the SCAD estimator (Fan and Li (2001)), which enables the D-TransFusion to enjoy both comparable computational complexity, but distributed to \( K \) parallel processors, and statistical precision to its non-distributed counterpart given a moderate task number \( K \).
4 Simulation

We evaluate the empirical performance of our proposed methods, TransFusion and D-TransFusion, and compare with existing methods including Trans-Lasso (Li et al., 2022) and TransHDGLM (Li et al., 2023). For two-step approaches, we report and compare the performance of both steps to better understand scenarios where the second step is necessary. As a baseline, we include the estimation error obtained by the LASSO regression on the target task, which we call Lasso (baseline). Each simulation setting is replicated with 100 independent trials, and we report the average performance. All methods are implemented based on R package glmnet with standard configuration, and parameters are chosen via 10-fold cross-validation.

We follow a similar experimental setup as in Li et al. (2022) and Li et al. (2023) by considering a high-dimensional linear regression problem with $p = 500$ and sparsity level $s = 10$. The target model is set as $\beta_j^{(0)} = 0.3$ for $1 \leq j \leq s$ and $\beta_j^{(0)} = 0$ otherwise. We generate $n_T = 150$ independent target samples $(X^{(0)}, y^{(0)})$ by $y^{(0)} = X^{(0)} \beta^{(0)} + \epsilon^{(0)}$ with $X_i^{(0)} \sim N(0, I)$ and $\epsilon_i^{(0)} \sim N(0,1)$.

The source sample size is set to be $n_S = 200$, and the source task number $K$ varies in the range $\{1, 3, 5, 7, 9\}$. We set $h_k = 12$ for $1 \leq k \leq K$. To simulate model and covariate shift we consider the following parameter configurations for the source tasks.

**Model Shift.** To investigate the impact of task diversity, we simulate two types of model shifts.

(i) **Diverse source tasks.** For $k = 1, \ldots, K - 1$ we set $\beta_j^{(k)} = \beta_j^{(0)} + \delta_j^{(k)}$ with $\delta_j^{(k)} \sim N(0, (h_k/50)^2)$ for $1 \leq j \leq 50$ and $\delta_j^{(k)} = 0$ otherwise. The last source model is generated with $\delta_j^{(K)} = -\sum_{k=1}^{K-1} \delta_j^{(k)}$ so that the task diversity measure $\varepsilon_D = 0$.

(ii) **Non-diverse source tasks.** Each $k$-th task-specific signal is generated as $\delta_j^{(k)} \sim N(0.1, (h_k/50)^2)$ for $1 \leq j \leq 50$ and $\delta_j^{(k)} = 0$ otherwise.

**Covariate Shift.** To demonstrate the robustness of TransFusion to covariate shifts, we consider two settings with different covariate distributions. In each setting, we generate $n_S$ independent samples for each source task.

(a) **Homogeneous design.** Each $X_i^{(k)} \sim N(0, I)$.

(b) **Heterogeneous design.** Each $X_i^{(k)} \sim N(0, \Sigma^{(k)})$ with $\Sigma^{(k)} = (A^{(k)})^\top (A^{(k)}) + I$. Here $A^{(k)}$ is a random matrix with each entry equals 0.3 with probability 0.3 and equals 0 with probability 0.7.

We consider four experimental settings based on combinations of model design (i) and (ii) with covariate design (a) and (b) to generate the source samples. Fig. 1 reports the $\ell_2$ estimation error of $\beta^{(0)}$ versus the source task number $K$ in these four settings. Fig. 2 reports a focused comparison between the performance of D-TransFusion and TransFusion in heterogeneous design (b). More simulation results are reported in the Appendix E. The following comments are in order.

- Robustness to covariate shift. Fig. 1 (i-a) shows in the diverse source tasks setting (i) when there is no covariate shift (a setting in favor of Trans-Lasso), TransFusion achieves comparable estimation error with the state-of-the-art methods. However, when covariate shift exists, a comparison of Fig. 1 (i-b) with (i-a) shows the estimator errors obtained by both Trans-Lasso and TransHDGLM increase significantly, and are even larger than Lasso (baseline) for small values of $K$.

On the contrary, the performance of TransFusion re-
mains the same, showing its robustness to covariate shift. When the source tasks are non-diverse, Fig.1 (ii-a) and (ii-b) reveal similar advantages achieved by two-step TransFusion. Consequently, TransFusion offers better reliability when the degree of covariate shift and source task diversity are unknown a priori.

- Impact of source task diversity. In Theorem 1 we have proved that when the source tasks are sufficiently diverse, applying one-step TransFusion suffices to obtain a small estimation error. Fig.1 (i-a) and (i-b) corroborate this statement: under both homogeneous and heterogeneous covariate designs, one- and two-step TransFusion yields comparable estimation errors. In the more challenging setting with non-diverse source tasks, Fig.1 (ii-a) and (ii-b) indicate applying the second de-bias step reduces the estimation error, which is also consistent with the implications of Theorem 2.

- D-TransFusion matches the performance of TransFusion. Fig. 2 demonstrates for small $K$, D-TransFusion attains comparable or even smaller estimation error than TransFusion, but uses only one-shot communication and has the ability to quickly adapt to downstream tasks. As $K$ increases, the gap reduces, and TransFusion gradually outperforms D-TransFusion. This is because D-TransFusion has a more restrictive growth condition on the source sample size $n_S$ with $K$, a requirement common to divide-and-conquer type methods. It also reveals the tradeoff between sample efficiency and communication cost.

In addition to the simulation study using synthetic data, the covariate-shift robustness of TransFusion method is further validated on real-world application through the MNIST handwritten digit classification task. The results are provided in Appendix F.

5 Conclusion

In this paper, we introduce a novel solution to tackle model shifts in high-dimensional transfer learning problems, ensuring robustness to covariate shifts and efficiency in knowledge transfer across distributed datasets. We provide a theoretical guarantee, showing its capacity to fully utilize the source samples to achieve an optimal estimation rate with one-shot communication and under covariate shifts. Simulation results validate our theory and showcase the state-of-the-art performance of the proposed method under various settings.

References


A Proof of Theorems

In this section, we provide proof of all the theorems. Throughout the section, we adopt the following notations to analyze the solution of the problem (2):

\[
\begin{pmatrix}
y^{(1)} \\
y^{(2)} \\
\vdots \\
y^{(K)} \\
y^{(0)}
\end{pmatrix}
= \begin{pmatrix}
X^{(1)} & 0 & \cdots & 0 & X^{(1)} \\
0 & X^{(2)} & \cdots & 0 & X^{(2)} \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & X^{(K)} & X^{(K)} \\
0 & 0 & \cdots & 0 & X^{(0)}
\end{pmatrix}
\begin{pmatrix}
\theta^{(1)} \\
\theta^{(2)} \\
\vdots \\
\theta^{(K)} \\
\theta^{(0)}
\end{pmatrix}
= \begin{pmatrix}
(\theta^{(1)})^{(1)} \\
(\theta^{(2)})^{(2)} \\
\vdots \\
(\theta^{(K)})^{(K)} \\
(\theta^{(0)})^{(0)}
\end{pmatrix}
= \begin{pmatrix}
\beta^{(1)} - \beta^{(0)} \\
\beta^{(2)} - \beta^{(0)} \\
\vdots \\
\beta^{(K)} - \beta^{(0)} \\
\beta^{(0)}
\end{pmatrix}.
\]

where recall that \(\beta^{(0)}\) is the target model parameter and \(\beta^{(k)}\)'s are source model parameters.

Under this transformation, solving problem (2) is equivalent as solving

\[
\hat{\theta} = \arg\min_\theta \{L(\theta) + \lambda_0 R(\theta)\},
\]

where we define \(L(\theta) := \frac{1}{n_k} \|y - X\theta\|^2_2\) and \(\lambda_0 R(\theta) := \lambda_0 \|\theta^{(0)}\|_1 + \lambda_0 \sum_{k=1}^K a_k \|\theta^{(k)}\|_1 = \sum_{k=0}^K \lambda_0 a_k \|\theta^{(k)}\|_1\), for any \(\theta \in \mathbb{R}^{(K+1)p}\). Since there exists a one-to-one transformation between \(\theta^*\) and \(\beta\), we can quantify the estimation error \(\hat{\beta} - \beta\) by analyzing \(\hat{\theta} - \theta^*\).

We first establish an essential property of sub-Gaussian design matrices, which serves as fundamental building blocks for our subsequent analysis. The following lemma 1 shows that the least square objective function has a restricted strongly convex (RSC) and restricted smooth (RSM) property. For a detailed discussion and proof of this lemma, interested readers can refer to Lemma 13 of Loh and Wainwright (2011) and Lemma 6 of Agarwal et al. (2010).

**Lemma 1** (RSC and RSM property). Under Assumption 1, for any \(\Delta \in \mathbb{R}^p\), with probability at least \(1 - c_1 \exp(-c_2 n_k)\),

\[
\frac{1}{n_k} \|X^{(k)} \Delta\|^2_2 = \Delta^\top \Sigma^{(k)} \Delta \geq \alpha_k \|\Delta\|^2_2 - \beta_k \frac{\log p}{n_k} \|\Delta\|^2_1, \\
\frac{1}{n_k} \|X^{(k)} \Delta\|^2_2 = \Delta^\top \Sigma^{(k)} \Delta \leq \gamma_k \|\Delta\|^2_2 + \tau_k \frac{\log p}{n_k} \|\Delta\|^2_1,
\]

where \(\alpha_k = \frac{1}{\gamma} \Lambda_{\min}(\Sigma^{(k)}) \geq 1/c, \gamma_k = 2 \Lambda_{\max}(\Sigma^{(k)}) \leq c\) and \(\beta_k, \tau_k \leq c, n_S = n_S\) for \(k = 1, \ldots, K\) and \(n_T\) for \(k = 0\).

**A.1 Proof of Theorem 1**

Define \(\hat{\Delta} := \hat{\theta} - \theta^*\) as the estimation error of \(\theta\) and the corresponding k-th block \(\hat{\Delta}^{(k)} := \hat{\theta}^{(k)} - (\theta^*)^{(k)}\). For brevity, we will omit the superscript and write \((\theta^*)^{(k)}\) as \(\theta^{(k)}\) for \(0 \leq k \leq K\) when there is no ambiguity.

Further define \(\hat{\Delta}^w := \sum_{k=1}^K \frac{n_S}{N} \hat{\Delta}^{(k)} + \hat{\Delta}^{(0)} = \hat{w} - w\) as the estimation error of the parameter average \(w\). Our goal is to establish an upper bound for \(\|\hat{\Delta}^w\|^2_2\).

The proof of Theorem 1 relies on three key technical lemmas. The proof of these lemmas is in Appendix B. The first lemma establishes an upper bound for the first-order term of the Taylor series expansion of \(L(\theta)\).

**Lemma 2.** Under Assumption 1 and 2, if \(n_S \geq \log p\), then by choosing \(\lambda_0 = c_0 \sqrt{\frac{\log p}{n_S}}\) and \(\lambda_k = a_k \lambda_0 = c_0 \sqrt{\frac{n_S \log p}{N}}\) for some appropriate constant \(c_0\), we have for any \(\Delta = \left(\begin{pmatrix} \Delta^{(1)} \end{pmatrix}^\top, \ldots, \left(\begin{pmatrix} \Delta^{(K)} \end{pmatrix}^\top, \left(\begin{pmatrix} \Delta^{(0)} \end{pmatrix}^\top \end{pmatrix}^\top \end{pmatrix}^\top \right)^\top \right) \in \mathbb{R}^{(K+1)p}\),

\[
\|\nabla L(\theta^*) \cdot \Delta\| \leq \sum_{k=1}^K \frac{\lambda_k}{2} \|\Delta^{(k)}\|^2_1 + \frac{\lambda_0}{2} \|\Delta^{(0)}\|^2_1.
\]
with probability larger than $1 - c_1 \exp (-c_2 \log p)$.

Recall that we define $\lambda_k = a_k \lambda_0$. The next lemma establishes a restricted set of directions in which $\Delta^w = \hat{w} - w$ lies.

**Lemma 3.** Under Assumption 1 and 2, and the conditions of Lemma 2, if further assume $\lambda_k \geq 8 \lambda_0 \frac{n_S}{n} \Delta^w$ and $n_S > n_T$, then the estimation error $\Delta^w$ satisfies the inequality

$$
\sum_{k=1}^{K} \lambda_k \left\| \Delta^{(k)} \right\|_1 + 2 \lambda_0 \left\| \Delta^{(0)} \right\|_1 \leq 8 \lambda_0 \left\| \Delta^w \right\|_1 + 8 \sum_{k=1}^{K} \lambda_k h_k,
$$

with probability larger than $1 - c_1 \exp(-c_2 \log p)$, where $S$ is the support set of $\beta^{(0)}$.

The following lemma ensures a property analogous to restricted strong convexity for $\Delta^w$.

**Lemma 4.** Under Assumption 1 and 2 and the conditions of Lemma 3, the estimation error $\Delta^w$ satisfies

$$
\Delta^T \Sigma \Delta = \mathcal{L} \left( \theta^* + \Delta \right) - \mathcal{L} (\theta^*) - \left\langle \nabla \mathcal{L} (\theta^*), \Delta \right\rangle \geq (1 - u_n) \alpha_{\min} \left\| \Delta^w \right\|_2 - v_n \sum_{k=1}^{K} \lambda_k h_k
$$

with probability larger than $1 - c_1 \exp(c_2 n_T) - c_3 \exp(c_4 \log p)$, where $u_n := \frac{256 \beta_{\max} \lambda_0^2}{\alpha_{\min}^2 \lambda_k^2 \wedge \frac{\lambda_0}{(K+1)}} \frac{s \log p}{N}$, $v_n := \frac{256 \beta_{\max} \lambda_0^2}{\alpha_{\min}^2 \lambda_k^2 \wedge \frac{\lambda_0}{(K+1)}} \frac{s \log p}{N}$.

with RSC constants $(\alpha_k, \beta_k)$ defined in Lemma 1.

We now turn to the proof of the theorem. In the following proof, we make use of the function $F : \mathbb{R}^{(K+1)p} \to \mathbb{R}$, given by

$$
F(\Delta) = \mathcal{L} (\theta^* + \Delta) - \mathcal{L} (\theta^*) - \mathcal{L} (\theta^* + \Delta) - \lambda_0 \mathcal{R} (\theta^* + \Delta) - \lambda_0 \mathcal{R} (\theta^*)
$$

where $\theta^*$ is the transformed model parameter defined in (16), and $\Delta = \left( (\Delta^{(1)})^T, \ldots, (\Delta^{(K)})^T, (\Delta^{(0)})^T \right)^T \in \mathbb{R}^{(K+1)p}$.

By Lemma 2, triangle inequality, and the fact that $\left\| \theta^{(k)}_{S^c} \right\|_1 = 0$, $\left\| \theta^{(k)}_S \right\|_1 \leq h_k$ for $1 \leq k \leq K$, we have

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

$$
\geq - \left\| \mathcal{L} (\theta^*) \right\|_\infty \left\| \Delta \right\|_1 + \left\| \nabla^2 \mathcal{L} (\theta^* + \gamma \Delta) \Delta \right\| (\gamma \in (0, 1))
$$

$$
+ \sum_{k=1}^{K} \lambda_k \left( \left\| \theta^{(k)} + \Delta^{(k)} \right\|_1 - \left\| \theta^{(k)} \right\|_1 \right) + \lambda_0 \left\| \theta^{(0)} + \Delta^{(0)} \right\|_1 - \lambda_0 \left\| \theta^{(0)} \right\|_1
$$

$$
\geq - \sum_{k=1}^{K} \lambda_k \left( \left\| \Delta^{(k)} \right\|_1 - \frac{\lambda_0}{2} \left\| \Delta^{(0)} \right\|_1 + \Delta^T \Sigma \Delta \right)
$$

$$
+ \sum_{k=1}^{K} \lambda_k \left( \left\| \Delta^{(k)} \right\|_1 - 2 \left\| \theta^{(k)} \right\|_1 \right) + \lambda_0 \left( \left\| \theta^{(0)} \right\|_1 - \left\| \theta^{(0)} \right\|_1 \right) + \left\| \Delta^{(0)} \right\|_1 - \left\| \theta^{(0)} \right\|_1
$$

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

By the definition of \( \hat{\Delta}^w \), we have \( \hat{\Delta}^{(0)} = \hat{\Delta}^w - \sum_{k=1}^K \frac{n_S}{N} \hat{\Delta}^{(k)} \). Therefore, applying triangle inequality yields

\[
F(\hat{\Delta}) \geq \hat{\Delta}^\top \Sigma \hat{\Delta} + \frac{1}{2} \lambda_0 \left\| \hat{\Delta}_S^w \right\|_1 - \frac{1}{2} \lambda_0 \sum_{k=1}^K \frac{n_S}{N} \left\| \hat{\Delta}_S^{(k)} \right\|_1 - \frac{3}{2} \lambda_0 \left\| \hat{\Delta}_S^w \right\|_1 - \frac{3}{2} \lambda_0 \sum_{k=1}^K \frac{n_S}{N} \left\| \hat{\Delta}_S^{(k)} \right\|_1 + \frac{K}{2} \lambda_k \left\| \hat{\Delta}^{(k)} \right\|_1 - 2 \sum_{k=1}^K \lambda_k h_k.
\]

(19)

Recall that we select \( \lambda_0, \ldots, \lambda_k \) such that \( \frac{\lambda_k}{2} \geq \frac{3 n_0}{2 N} \lambda_0 \), so we have

\[
\sum_{k=1}^K \frac{\lambda_k}{2} \left\| \hat{\Delta}^{(k)} \right\|_1 - \frac{3}{2} \lambda_0 \sum_{k=1}^K \frac{n_S}{N} \left\| \hat{\Delta}_S^{(k)} \right\|_1 \geq 0.
\]

(20)

Notice that \( \hat{\theta} \) is the solution to the problem (17). We then have \( \hat{\Delta} = \hat{\theta} - \theta^* = \text{argmin}_\Delta F(\Delta) \). Since \( F(\theta) = 0 \), it follows that \( F(\hat{\Delta}) \leq 0 \). This summing with (19) and (20) leads to

\[
0 \geq \hat{\Delta}^\top \Sigma \hat{\Delta} - \frac{3}{2} \lambda_0 \left\| \hat{\Delta}_S^w \right\|_1 + \frac{1}{2} \lambda_0 \left\| \hat{\Delta}_S^w \right\|_1 - 2 \sum_{k=1}^K \lambda_k h_k
\]

(21)

\[
= \hat{\Delta}^\top \Sigma \hat{\Delta} - \frac{3}{2} \lambda_0 \left\| \hat{\Delta}_S^w \right\|_1 + 2 \lambda_0 \left\| \hat{\Delta}_S^w \right\|_1 - 2 \sum_{k=1}^K \lambda_k h_k.
\]

(22)

We now establish the upper bound for error measured in \( \ell_2 \) norm, \( \| \hat{\Delta}^w \|_2 \). An application of Lemma 4 on (21) yields that with probability larger than \( 1 - c_1 \exp(c_2 nT) - c_3 \exp(c_4 \log p) \),

\[
0 \geq (1 - u_n) \alpha_{\min} \left\| \hat{\Delta}^w \right\|_2^2 - \frac{3}{2} \lambda_0 \sqrt{s} \left\| \hat{\Delta}^w \right\|_2 - (2 + v_n) \sum_{k=1}^K \lambda_k h_k
\]

where we use the fact that \( \| \hat{\Delta}_S^w \|_1 \geq 0 \). If \( u_n = o(1) \), we can show that for a sufficiently large \( n_S \),

\[
\left\| \hat{\Delta}^w \right\|_2 \leq \frac{3 \lambda_0 \sqrt{s} + \sqrt{\frac{9}{4} \lambda_0^2 s + 4 (1 - u_n) (2 + v_n) \alpha_{\min} \sum_{k=1}^K \lambda_k h_k}}{2 (1 - u_n) \alpha_{\min}} \leq \sqrt{\frac{s \log p}{N}} + \sqrt{(1 + v_n) \sum_{k=1}^K \frac{n_S}{N} \sqrt{\log \frac{p}{n_S} h_k}}
\]

(23)

by plugging in the choice of \( \lambda_0 \) and \( \lambda_k s \), which is the desired result.

It remains to show the order of \( v_n \) and prove that \( u_n = o(1) \) under the conditions of Theorem 1. By the assumptions in Theorem 1 and the choice of \( \lambda_0, \ldots, \lambda_K \), we have

\[
u_n = \frac{256 \beta_{\max} \lambda_0^2}{\alpha_{\min} \lambda_k^2 \wedge (\lambda_0^2/(K + 1))} \frac{s \log p}{N} \leq \frac{s \log p}{n_S} = o(1),
\]

and

\[
v_n = \frac{256 \beta_{\max}}{\lambda_k^2 \wedge (\lambda_0^2/(K + 1))} \frac{\log p}{N} \left( \sum_{k=1}^K \lambda_k h_k \right) \leq \sqrt{K^2 \log \frac{p}{h_k}}.
\]

The proof is then finished.
A.2 Proof of Theorem 2

Similar to the arguments in the proof of Theorem 1, we first define \( \hat{\mathcal{L}}(\delta) = \frac{1}{n_T} \| y^{(0)} - X^{(0)} \tilde{w} - X^{(0)} \delta \|_2^2 \) and \( \hat{F}(\Delta) = \hat{\mathcal{L}}(\delta^* + \Delta) - \hat{\mathcal{L}}(\delta^*) + \hat{\lambda} \| \delta^* + \Delta \|_1 - \hat{\lambda} \| \delta^* \|_1 \), where \( \delta^* = \beta^{(0)} - w \) is the contrast between the target parameter and the averaged parameter. Denoting \( \Delta^\delta = \delta - \delta^* \). Recall that \( \Delta^w = \tilde{w} - w \), by Hölder inequality and triangle inequality,

\[
\langle \nabla \hat{\mathcal{L}}(\delta^*), \Delta^\delta \rangle = \frac{1}{n_T} \left\langle (X^{(0)})^T [y^{(0)} - X^{(0)} \beta^{(0)} - X^{(0)} (\tilde{w} + \delta^* - \beta^{(0)})], \Delta^\delta \right\rangle \\
= \frac{1}{n_T} \left\langle (X^{(0)})^T [\epsilon^{(0)} - X^{(0)} (\tilde{w} - w)], \Delta^\delta \right\rangle \\
\leq \frac{1}{n_T} \left\| (X^{(0)})^T \epsilon^{(0)} \right\|_\infty \| \Delta^\delta \|_1 + \frac{1}{2} \left( \Delta^\delta \right)^T \hat{\Sigma}^{(0)} \Delta^\delta + \frac{1}{2} \left( \Delta^w \right)^T \Sigma^{(0)} \Delta^w.
\]

By Lemma 2, if \( n_T \gtrsim \log p \), we can choose \( \hat{\lambda} = c \sqrt{\frac{\log p}{n_T}} \) for some constant \( c \) so that \( \frac{1}{n_T} \left\| (X^{(0)})^T \epsilon^{(0)} \right\|_\infty \leq \frac{\hat{\lambda}}{2} \) with probability larger than \( 1 - c_1 \exp(c_2 \log p) \). Therefore, it holds that

\[
\hat{\mathcal{L}} \left( \hat{\Delta}^\delta + \delta^* \right) - \hat{\mathcal{L}}(\hat{\Delta}) = \left\langle \nabla \hat{\mathcal{L}}(\delta^*), \Delta^\delta \right\rangle + \left( \Delta^\delta \right)^T \hat{\Sigma}^{(0)} \Delta^\delta \\
\geq -\frac{\hat{\lambda}}{2} \| \Delta^\delta \|_1 + \frac{1}{2} \left( \Delta^\delta \right)^T \hat{\Sigma}^{(0)} \Delta^\delta - \frac{1}{2} \left( \Delta^w \right)^T \Sigma^{(0)} \Delta^w \tag{24}
\]

So by the optimality condition,

\[
0 \geq \hat{F}(\hat{\Delta}) \\
\geq \hat{\mathcal{L}} \left( \hat{\Delta}^\delta + \delta^* \right) - \hat{\mathcal{L}}(\hat{\Delta}) + \hat{\lambda} \| \Delta^\delta + \delta^* \|_1 - \hat{\lambda} \| \delta^* \|_1 \\
\geq \hat{\mathcal{L}} \left( \hat{\Delta}^\delta + \delta^* \right) - \hat{\mathcal{L}}(\hat{\Delta}) + \hat{\lambda} \| \Delta^\delta \|_1 - 2\hat{\lambda} \| \delta^* \|_1 \\
\geq \frac{\hat{\lambda}}{2} \| \Delta^\delta \|_1 + \frac{1}{2} \left( \Delta^\delta \right)^T \hat{\Sigma}^{(0)} \Delta^\delta - \frac{1}{2} \left( \Delta^w \right)^T \Sigma^{(0)} \Delta^w - 2\hat{\lambda} \| \delta^* \|_1
\]

Since the result involves \( \Delta^w \), we first establish the following auxiliary lemma:

**Lemma 5.** Under Assumption 1 and 2, \( n_S \gtrsim \log p, n_S > n_T \), if we choose \( \lambda_0 \gtrsim \sqrt{\frac{\log p}{N}} \), \( \lambda_k = a_k \lambda_0 \gtrsim \sqrt{\frac{p}{N}} \sqrt{\frac{\log p}{N}} \)

such that

\[
\lambda_k \geq 8 \lambda_0 \frac{n_S}{N}, \quad u_n = \frac{256 \beta_{\max}^2 \lambda_0^2}{a_{\min}^2 \lambda_k^2 \wedge (\lambda_0^2/(K + 1))} s \log p = o(1) \text{ and } v_n = \frac{256 \beta_{\max}^2}{\lambda_k^2 \wedge (\lambda_0^2/(K + 1))} \log p \left( \sum_{k=1}^{K} \lambda_k h_k \right) = O(1),
\]

then we have

\[
\| \Delta^w \|_2 \lesssim \sqrt{s} \lambda_0 + \sqrt{\sum_{k=1}^{K} \lambda_k h_k} \\
\| \Delta^w \|_1 \lesssim s \lambda_0 + \sqrt{s} \sum_{k=1}^{K} \lambda_k h_k + \frac{\sum_{k=1}^{K} \lambda_k h_k}{\lambda_0}
\]

with probability larger than \( 1 - c_1 \exp(c_2 n_T) - c_3 \exp(c_4 \log p) \).

Notice that according to the theorem statement, the choice of \( \lambda_0 \) and \( \lambda_k \)s depends on the event \( A \). It can be verified that either selection fulfills the conditions outlined in Lemma 5. We now discuss by cases and apply Lemma 5 to prove the result.
Case 1: We start with the case when the event $A$ holds. That is, we have
\[ s \log p/n_S \geq \hat{h} \sqrt{\log p/n_T} \]
Under this condition, we choose
\[ \lambda_0 = c_1 \sqrt{\frac{\log p}{N}}, \quad \text{and} \quad a_k = 8 \sqrt{\frac{n_S}{N}}. \]
Applying Lemma 5 yields
\begin{align*}
\|\Delta^w\|_2 & \lesssim \sqrt{\frac{s \log p}{N}} + \sqrt{\frac{\log p}{n_S}} \hat{h} \\
\|\Delta^w\|_1 & \lesssim s \sqrt{\frac{\log p}{N}} + \sqrt{\frac{\log p}{n_S}} \hat{h} + \sqrt{\frac{N}{n_S}} \tilde{h}
\end{align*}
with probability larger than $1 - c_1 \exp(c_2 n_T) - c_3 \exp(c_4 \log p)$.

(i). If $\frac{1}{2} (\Delta^w) \Sigma^{(0)} \Delta^w \geq 2\lambda \|\delta^*\|_1$, according to (24), we have
\[ 0 \geq \frac{\lambda}{2} \|\Delta^\delta\|_1 + \frac{1}{2} (\Delta^\delta) \Sigma^{(0)} \Delta^\delta - (\Delta^w) \Sigma^{(0)} \Delta^w. \]
By Lemma 1 and the condition $n_T \gtrsim \log p$, we have $(\Delta^w) \Sigma^{(0)} \Delta^w \leq \gamma_0 \|\hat{\Delta^w}\|_2^2 + \gamma_0 \log p/n_T \|\hat{\Delta^w}\|_1^2$ for some constants $\gamma_0$ and $\gamma_0$, this together with (28) indicates
\begin{align*}
\frac{\hat{\lambda}}{2} \|\Delta^\delta\|_1 & \leq (\Delta^w) \Sigma^{(0)} \Delta^w \leq \gamma_0 \|\hat{\Delta^w}\|_2^2 + \gamma_0 \log p/n_T \|\hat{\Delta^w}\|_1^2 \\
& \lesssim \frac{s \log p}{N} + (1 + \frac{s \log p}{n_T}) \sqrt{\frac{\log p}{n_S} \hat{h} + \frac{(s \log p)^2}{n_T N} + \frac{N}{n_S} \tilde{h}^2 \log p/n_T}.
\end{align*}
where the last inequality is based on the results in (26) and (27). In Theorem 2 we assume $\frac{s \log p}{n_T} = O(1)$ and $\hat{h} \sqrt{\log p/n_T} = o(1)$. According to (25), we have $(N/n_S) \tilde{h} \sqrt{\log p/n_T} \leq (K + 1) \log p/n_S = O(1)$. Applying these results to (29) yields $\frac{\hat{\lambda}}{2} \|\Delta^\delta\|_1 = o_p(1)$.

On the other hand, if we apply Lemma 1 to (28), we then have
\[ 0 \geq \left(1 - \frac{\beta_0 \log p}{n_T} \|\Delta^\delta\|_1\right) \frac{\hat{\lambda}}{2} \|\Delta^\delta\|_1 + \frac{1}{2} \alpha_0 \|\Delta^\delta\|_2^2 - \gamma_0 \|\hat{\Delta^w}\|_2^2 - \frac{\log p}{n_T} \|\hat{\Delta^w}\|_1^2.
\]
Notice that we choose $\hat{\lambda} = c\sqrt{\frac{\log p}{n_T}}$ for some universal constant $c$. Therefore, we may choose $c > \sqrt{2/\mu_0}$ so we have $\beta_0 \log p/n_T < \frac{\hat{\lambda}^2}{2}$. This together with (29) and the fact that $\frac{\hat{\lambda}}{2} \|\Delta^\delta\|_1 = o_p(1)$ leads to
\[ \frac{1}{2} \alpha_0 \|\Delta^\delta\|_2^2 \leq \gamma_0 \|\hat{\Delta^w}\|_2^2 + \frac{\log p}{n_T} \|\hat{\Delta^w}\|_1^2.
\]
Based on similar arguments to those in (29), we have
\[ \|\Delta^\delta\|_2 \lesssim \sqrt{\frac{s \log p}{N}} + \sqrt{\frac{\log p}{n_S} \hat{h}} + \sqrt{\frac{N}{n_S} \tilde{h}}, \]
with probability larger than $1 - c_1 \exp(c_2 n_T) - c_3 \exp(c_4 \log p)$.

(ii). If $\frac{1}{2} (\Delta^w) \Sigma^{(0)} \Delta^w \leq 2\lambda \|\delta^*\|_1$, we have
\[ 0 \geq \frac{\hat{\lambda}}{2} \|\Delta^\delta\|_1 + \frac{1}{2} (\Delta^\delta)^\top \Sigma^{(0)} \Delta^\delta - 4\lambda \|\delta^*\|_1
\]
which implies that $\|\Delta^\delta\|_1 \leq 8\|\delta^*\|_1 \leq 8\hat{h}$.

By applying Lemma 1 again, we have

$$0 \geq \frac{\lambda}{2} \|\Delta^\delta\|_1 + \frac{1}{2} a_0 \|\Delta^\delta\|_2^2 - \frac{1}{2} \beta_0 \frac{\log p}{n_T} \|\Delta^\delta\|_1^2 - 4\lambda \|\delta^*\|_1 \geq \frac{1}{2} a_0 \|\Delta^\delta\|_2^2 - 32\beta_0 \frac{\log p}{n_T} \|\delta^*\|_1^2 - 4\lambda \|\delta^*\|_1$$

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

So in this case, we have

$$\|\Delta^\delta\|_2 \leq \sqrt{\frac{64\beta_0 \log p}{a_0 n_T}} \hat{h}^2 + 8\frac{\hat{h}}{a_0} \leq \sqrt{\frac{\log p}{n_T}} \hat{h} + \sqrt{\frac{\log p}{n_T}} \hat{h}$$

and

$$\|\hat{\Delta}^\delta\|_2 \leq \|\Delta^\delta\|_1 \leq \hat{8} \hat{h}$$

Under the assumption that $\hat{h} \sqrt{\frac{\log p}{n_T}} = o(1)$, we have $\|\hat{\Delta}^\delta\|_2 \leq \sqrt{\frac{\log p}{n_T}} \hat{h} \wedge \hat{h}$.

Therefore, by combining the results from the two cases discussed above, we have

$$\left\|\hat{w} + \hat{\delta} - \beta^{(0)}\right\|_2 \leq \left\|\Delta^\delta\right\|_2 + \left\|\Delta^w\right\|_2 \lesssim \sqrt{\frac{s \log p}{N}} + \sqrt{\frac{\log p}{n_S}} \hat{h} + \sqrt{K + 1} \sqrt{\frac{\log p}{n_T}} \hat{h} + \sqrt{\frac{\log p}{n_T}} \hat{h} \wedge \hat{h}$$

with probability larger than $1 - c_1 \exp(c_2 n_T) - c_2 \exp(c_4 \log p)$.

Since $A$ holds, together with the condition $K^2 s \log p/n_S = O(1)$, we have

$$\sqrt{K + 1} \sqrt{\frac{\log p}{n_T}} \hat{h} \leq \sqrt{K + 1} \frac{s \log p}{n_S} \leq \sqrt{(K + 1)^2 s \log p} \sqrt{\frac{s \log p}{N}} \lesssim \sqrt{\frac{s \log p}{N}}$$

which implies

$$\left\|\hat{w} + \hat{\delta} - \beta^{(0)}\right\|_2 \leq \left\|\Delta^\delta\right\|_2 + \left\|\Delta^w\right\|_2 \lesssim \sqrt{\frac{s \log p}{N}} + \sqrt{\frac{\log p}{n_S}} \hat{h} + \sqrt{\frac{\log p}{n_T}} \hat{h} \wedge \hat{h}.$$  

**Case 2:** Next we discuss the case when the event $A^c$ holds, i.e.,

$$s \log p/n_S \leq \hat{h} \sqrt{\log p/n_T}$$

In this case, we choose

$$\lambda_0 = c_0 \sqrt{\frac{\log p}{n_S}}, \text{ and } a_k = \frac{8n_S}{N}$$

Applying Lemma 5 again, we have

$$\|\hat{\Delta}^w\|_2 \lesssim \sqrt{\frac{s \log p}{n_S}} + \sqrt{\frac{\log p}{n_S}} \hat{h} \quad \text{(30)}$$

and

$$\|\hat{\Delta}^w\|_1 \lesssim s \sqrt{\frac{\log p}{n_S}} + \sqrt{\frac{\log p}{n_S}} \hat{h} + \sqrt{\frac{\log p}{n_T}} \hat{h} \wedge \hat{h} \quad \text{(31)}$$

with probability larger than $1 - c_1 \exp(c_2 n_T) - c_2 \exp(c_4 \log p)$.

Plugging the new bound (31) and (30) into the arguments in Case 1 leads to

$$\left\|\hat{w} + \hat{\delta} - \beta^{(0)}\right\|_2 \lesssim \sqrt{\frac{s \log p}{n_S}} + \sqrt{\frac{\log p}{n_S}} \hat{h} + \sqrt{\frac{\log p}{n_T}} \hat{h} + \sqrt{\frac{\log p}{n_T}} \hat{h} \wedge \hat{h}.$$
Recall that in Theorem 2 we assume $\sqrt{\log p/n_T} = o(1)$ and $\log p/n_T = O(1)$. Therefore, in the above bound, the third term has a smaller order comparing to the fourth term. Hence, we have

$$
\left\| \hat{w} + \delta - \beta^{(0)} \right\|_2 \lesssim \sqrt{\frac{s \log p}{n_S}} + \sqrt{\frac{\log p}{n_S} h} + \sqrt{\frac{\log p}{n_T} h \wedge \tilde{h}}.
$$

As we assume $A^c$ holds in this case, we have $s \log p/n_S \leq \tilde{h} \sqrt{\log p/n_T}$, which further implies

$$
\left\| \hat{w} + \delta - \beta^{(0)} \right\|_2 \lesssim \sqrt{\frac{\log p}{n_T} \tilde{h}}.
$$

Combining the results from the two cases discussed above, we have

$$
\left\| \hat{w} + \delta - \beta^{(0)} \right\|_2 \lesssim \sqrt{\frac{s \log p}{N}} + \sqrt{\frac{s \log p}{n_T} \tilde{h}}
$$

with probability larger than $1 - c_1 \exp(c_2 n_T) - c_3 \exp(c_4 \log p)$. The proof is then completed.

### A.3 Proof of Theorem 3

We first show a lemma discussing the bias and variance components in (59), whose proof is based on the results in Javanmard and Montanari (2014) but taking into account that $\beta^{(k)}$ is not exactly $s$-sparse (due to the contrast term $\delta^{(k)}$).

**Lemma 6.** Under Assumption 1 and 2 and $s \log p/n_S = o(1)$, if we construct $\{\hat{\beta}^{(k)}_{\text{LASSO}}\}_{k=1, \ldots, K}$ through (57) and $\{\hat{\Theta}^{(k)}\}_{k=1, \ldots, K}$ using (58), with parameters $\tilde{\lambda}_k = \mu_k = c_0 \sqrt{\log p/n_S}$ for some universal constant $c_0$, then we have that for $k = 1, \ldots, K$,

$$
\frac{1}{n_S} \left\| \hat{\Theta}^{(k)} (X^{(k)})^\top e^{(k)} \right\|_2 \lesssim \sqrt{\frac{\log p}{n_S}}
$$

and

$$
\left\| b^{(k)} \right\|_\infty := \left\| \left( \hat{\Theta}^{(k)} \hat{\Sigma}^{(k)} - I \right) (\hat{\beta}^{(k)}_{\text{LASSO}} - \beta^{(k)}) \right\|_\infty \lesssim \frac{s \log p}{n_S} + h_k \sqrt{\frac{\log p}{n_S}},
$$

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 (57) is essentially a special case of problem (2). Then we apply techniques similar to those used in Theorem 1 to prove the results. Following the arguments in (35), we may reformulate problem (57) into a generalized LASSO problem:

$$
\hat{\theta} = \arg \min_{\theta} \left\{ \hat{L}(\theta) + \lambda_0 R(\theta) \right\} := \arg \min_{\theta} \left\{ \frac{1}{2N} \left\| \hat{y} - \hat{X} \theta \right\|^2_2 + \lambda_0 R(\theta) \right\}
$$

where

$$
\hat{y} = \begin{pmatrix}
\sqrt{n_S} \beta^{(1)} \\
\sqrt{n_S} \beta^{(2)} \\
\vdots \\
\sqrt{n_S} \beta^{(K)} \\
y^{(0)}
\end{pmatrix}, \quad
\hat{X} = \begin{pmatrix}
\sqrt{n_S} I_p & 0 & \cdots & 0 & \sqrt{n_S} I_p \\
0 & \sqrt{n_S} I_p & \cdots & 0 & \sqrt{n_S} I_p \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & \sqrt{n_S} I_p & 0 \\
0 & 0 & \cdots & 0 & \sqrt{n_S} I_p
\end{pmatrix}.
$$
Similarly, we can define the random noise $\tilde{e} = \hat{y} - \hat{X}\theta^*$. The k-th block of $\tilde{e}$ is given by $\sqrt{n_S} \left( \Theta^{(k)} (X^{(k)})^\top e^{(k)}/n_S + b^{(k)} \right)$ for $k = 1, \ldots, K$, whereas the last block is $e^{(0)}$, the observation noise for the target model.

Following the aforementioned reformulation, we can employ an approach similar to that used in Lemma 2 to establish the result about $\langle \nabla \hat{L} (\theta), \Delta \rangle$. Define $\delta_k = \frac{s \log p}{N} + \frac{n_S}{N} \sqrt{\log p/n_S} h_k$ and $\delta_0 = \frac{Ks \log p}{N} + \sqrt{\log p/n_S} h$, the result is stated as follows:

**Lemma 7.** Under assumptions 1 and 2, if $n_S \geq \log p$, $\lambda_k = c_k \left( \frac{n_S}{N} \log p + \delta_k \right)$ and $\lambda_0 = c_0 \left( \frac{\log p}{N} + \delta_0 \right)$ for some appropriate constants $c_0, \ldots, c_K$, then we have for any $\Delta = \left( (\Delta^{(1)})^\top, \ldots, (\Delta^{(K)})^\top, (\Delta^{(0)})^\top \right)^\top \in \mathbb{R}^{(K+1)p}$,

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

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

Notice that the only difference between Lemma 7 and Lemma 2 is the choice of $\{\lambda_k\}_{k=0}^{K}$. With this new choice of parameters, we can verify that if further $n_S \gg Ks \log p$, $n_S \gg K^2 \log p$, and $h_k \approx \bar{h}$ for any $1 \leq k \leq K$, the conditions of Lemma 5 hold. Therefore, we can apply Lemma 5 and obtain

$$\| \hat{w}_C - w \|_2 \lesssim \sqrt{s} \lambda_0 + \sqrt{\sum_{k=1}^K \lambda_k h_k} \lesssim \sqrt{s} \left( \sqrt{\frac{\log p}{N}} + \delta_0 \right) + \sqrt{\sum_{k=1}^K \left( \frac{n_S}{N} \log p/n_S + \delta_k \right) h_k}.$$

with probability larger than $1 - c_1 \exp(c_2 n_T) - c_3 \exp(c_4 \log p)$. This together with the fact that $\|w - \beta^{(0)}\|_2^2 \leq c_0^2$ implies the bound (13).

Furthermore, if we assume $h_k \approx \bar{h} = O(1)$, then we have

$$s^2 \delta_0 = \frac{K^2 s^3 \log^2 p}{N^2} + \frac{s \log p}{n_S} \bar{h}^2 \lesssim \frac{s \log p}{N} + \sqrt{\frac{\log p}{n_S} \bar{h}}$$

(37)

and

$$\sum_{k=1}^K \delta_k h_k = s \frac{\log p}{n_S} \bar{h} + \sqrt{\frac{\log p}{n_S} \sum_{k=1}^K \frac{n_S}{N} h_k^2} \lesssim \sqrt{\frac{\log p}{n_S} \bar{h}}$$

(38)

based on the condition that $n_S \gg Ks^2 \log p$. Therefore, we have the bound (14).

**A.4 Proof of Theorem 4**

We use a similar line of arguments as the proof of Theorem 2. Recall that we choose

$$\lambda_k = c_0 \left( 8 \sqrt{\frac{\bar{h}}{h_k}} \right) \left( \frac{n_S \log p}{N} \frac{n_S}{N} + \delta_k \right), \quad \delta_k = \frac{s \log p}{n_S} + \frac{n_S \log p}{n_S} \bar{h} h_k, \quad \delta_0 = \frac{Ks \log p}{N} + \sqrt{\frac{\log p}{n_S} h_k}.$$

(39)

$$\lambda_0 = c_0 \left( \frac{\log p}{N} \bar{A} + \sqrt{\frac{\log p}{n_S} \bar{A}^2} + \delta_0 \right), \quad \delta_0 = \frac{Ks \log p}{N} + \sqrt{\frac{\log p}{n_S} \sum_{k=1}^K n_S h_k}.$$

(40)
We can verify that if \( n_S \gg Ks \log p, n_S \gtrsim K^2 \log p \) and \( h_k \asymp \bar{h} \) for any \( 1 \leq k \leq K \), this choice of parameters satisfies the conditions in Lemma 5. Therefore we can apply Lemma 5 and obtain

\[
\| \Delta \|_2 \lesssim \sqrt{s} \lambda_0 + \sqrt{\sum_{k=1}^K \lambda_k h_k} \lesssim \sqrt{\frac{s \log p}{N} + \frac{s \log p}{n_S} \| A \|_1 + \frac{\log p}{n_S} \| \bar{h} \| + \sqrt{s} \delta_0 + \sqrt{\sum_{k=1}^K \delta_k h_k},
\]  

\[
\| \Delta \|_1 \lesssim s \lambda_0 + s \sum_{k=1}^K \frac{\lambda_k h_k}{\lambda_0} \lesssim \sqrt{\frac{s^2 \log p}{N} + \frac{s^2 \log p}{n_S} \| A \|_1 + \frac{s \log p}{n_S} \bar{h} + s \delta_0 + \frac{s \delta_k h_k}{\lambda_0} + K \bar{h} \| A \|_1 + \bar{h} \| A \|_1}
\]

by plugging in the choice of \( \lambda_0 \) and \( \lambda_k \)'s and using the fact that \( \sum_{k=1}^K \delta_k = \delta_0 \).

To prove the theorem, it suffices to show that in the bounds (41) and (42), the terms involving \( \delta_0 \) and \( \delta_k \) are of orders that align with some other terms, then we can follow exactly the same proof in Theorem 2 to prove the result. To show this, we can use the results in (37) and (38), which gives us

\[
\sqrt{s} \delta_0 + \sqrt{\sum_{k=1}^K \delta_k h_k} \lesssim \sqrt{\frac{s \log p}{N} + \sqrt{\frac{s \log p}{n_S} \bar{h}}},
\]

with this, the proof is completed.

B Proof of Technical Lemmas

B.1 Proof of Lemma 2

By definition, \(- \nabla L (\theta) = \frac{1}{N} X^T (y - X \theta) = \left( \frac{1}{N} (X^{(1)})^T \epsilon^{(1)}, \ldots, \frac{1}{N} (X^{(K)})^T \epsilon^{(K)}, \frac{1}{N} \sum_{k=0}^K (X^{(k)})^T \epsilon^{(k)} \right)^T \).

Therefore, by Hölder’s inequality, we have

\[
\| (\nabla L (\theta), \Delta) \| = \sum_{k=1}^K \left\| \frac{1}{N} (X^{(k)})^T \epsilon^{(k)} , \Delta^{(k)} \right\| + \left\| \frac{1}{N} \sum_{k=0}^K (X^{(k)})^T \epsilon^{(k)} , \Delta^{(0)} \right\| 
\]

\[
\leq \sum_{k=1}^K \left\| \frac{1}{N} (X^{(k)})^T \epsilon^{(k)} \right\|_\infty \| \Delta^{(k)} \|_1 + \left\| \frac{1}{N} \sum_{k=0}^K (X^{(k)})^T \epsilon^{(k)} \right\|_\infty \| \Delta^{(0)} \|_1.
\]

Recall that we define \( n_k = n_S \) for \( 1 \leq k \leq K \) and \( n_k = n_T \) for \( k = 0 \). Define

\[
A^{(k)} = \left\{ \max_{1 \leq j \leq p} \left\{ \frac{1}{n_k} \sum_{i=1}^{n_k} (x^{(k)}_{ij})^2 \right\} \leq 2 \max_{1 \leq j \leq p} E (x^{(k)}_{ij})^2 \right\},
\]

and

\[
A = \left\{ \max_{1 \leq j \leq p} \left\{ \frac{1}{n_k} \sum_{k=0}^K \sum_{i=1}^{n_k} (x^{(k)}_{ij})^2 \right\} \leq 2 \max_{1 \leq k \leq K, 1 \leq j \leq p} E (x^{(k)}_{ij})^2 \right\}.
\]

Since \( X^{(k)} \) is sub-Gaussian with uniformly bounded second moment and \( n_S \gtrsim \log p \), we have \( P \left( A^{(k)} \right) \leq c_1 \exp(-c_2 n_S) \) for \( 1 \leq k \leq K \) and \( P \left( A \right) \leq c_1 \exp(-c_2 N) \) for some universal constants \( c_1 \) and \( c_2 \).

In addition, as \( \epsilon^{(k)} \sim N \left( 0, \sigma_k^2 I \right) \) for some finite \( \sigma_k \), by Proposition 5.10 in Vershynin (2010), we can establish
that with some universal constant $c_3$, for $1 \leq k \leq K$,

$$P \left( \max_{1 \leq j \leq p} \left| \frac{1}{n_S} \sum_{i=1}^{n_S} \epsilon_i^{(k)} x_{ij} \right| \geq t \right) \leq P \left( \max_{1 \leq j \leq p} \left| \frac{1}{n_S} \sum_{i=1}^{n_S} \epsilon_i^{(k)} x_{ij} \right| \geq t \right) + P(A^{(k)})$$

$$\leq p \cdot e \cdot \exp \left( -\frac{c_3 n S t^2}{4 \sigma_k^2 \max_{1 \leq j \leq p} E \left( x_{ij}^{(k)} \right)^2} \right) + c_1 \exp (-c_2 n S).$$

Since by Assumption 1, there exist a constant $c$ such that $\max_{1 \leq k \leq K} \Lambda_{\max}(\Sigma^{(k)}) \leq c$, so $\max_{1 \leq j \leq p} E \left( x_{ij}^{(k)} \right)^2$ is uniformly bounded above. Therefore for $1 \leq k \leq K$, by choosing $t = \sqrt{c_4 \log p / n_S}$ for some constant $c_4$, with probability larger than $1 - c_1 \exp (-c_2 \log p)$, we have

$$\left\| \frac{1}{N} \left( X^{(k)} \right)^\top \epsilon^{(k)} \right\|_\infty \lesssim \frac{n_S \sqrt{\log p}}{n_S} = \frac{n_S}{N} \sqrt{\frac{\log p}{N}}.$$

Similarly, we have

$$P \left( \max_{1 \leq j \leq p} \left| \frac{1}{N} \sum_{k=0}^K \sum_{i=1}^{n_k} \epsilon_i^{(k)} x_{ij} \right| \geq t \right) \leq P \left( \max_{1 \leq j \leq p} \left| \frac{1}{N} \sum_{k=0}^K \sum_{i=1}^{n_k} \epsilon_i^{(k)} x_{ij} \right| \geq t \right) + P(\overline{A})$$

$$\leq p \cdot e \cdot \exp \left( -\frac{c_4 N t^2}{4 \max_{0 \leq k \leq K, 1 \leq j \leq p} \sigma_k^2 E \left( x_{ij}^{(k)} \right)^2} \right) + c_1 \exp (-c_2 N)$$

So we have with probability larger than $1 - c_1 \exp (-c_2 \log p)$,

$$\left\| \sum_{k=0}^K \frac{1}{N} \left( X^{(k)} \right)^\top \epsilon^{(k)} \right\|_\infty \lesssim \sqrt{\frac{\log p}{N}}.$$

Therefore, by choosing $\lambda_k = a_k \lambda_k = c_0 \sqrt{\frac{\log p}{N}} \sqrt{\frac{\log p}{N}}$ and $\lambda_0 = c_0 \sqrt{\frac{\log p}{N}}$ for some sufficiently large constant $c_0$, we have the desired result.

### B.2 Proof of Lemma 3

Define $S$ is the support set of $\theta^{(0)} = \beta^{(0)}$, and $S^c$ as its complement. Then we have $\|\theta_S^*\|_0 = s$ and $\|\theta_{S^c}^*\|_1 \leq \sum_{k=1}^K h_k \leq (K + 1) \hat{h}$ as $n_S \geq nT$.

We define $F : \mathbb{R}^{(K+1)p} \to \mathbb{R}$ as

$$F(\Delta) = \mathcal{L}(\theta^* + \Delta) - \mathcal{L}(\theta^*) + \lambda_0 \mathcal{R}(\theta^* + \Delta) - \lambda_0 \mathcal{R}(\theta^*),$$

and $\hat{\theta}$ as the solution to the problem (17). We then have $\hat{\Delta} = \hat{\theta} - \theta^* = \mathop{\text{argmin}}_{\Delta} F(\Delta)$ and $F(0) = 0$. Consequently, it follows that $F(\hat{\Delta}) \leq 0$.

Since $\mathcal{L}$ is a convex function, by Lemma 2, we can choose $\lambda_k = c_0 \sqrt{\frac{\log p}{N}} \sqrt{\frac{\log p}{N}}$ and $\lambda_0 = c_0 \sqrt{\frac{\log p}{N}}$ so that

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

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

Since the $\ell_1$-norm function is decomposable and $\|\theta_{S^c}^{(0)}\|_1 = 0$, by triangle inequality we have
\[
\lambda_0 \mathcal{R}(\theta^* + \Delta) - \lambda_0 \mathcal{R}(\theta^*) = \sum_{k=1}^{K} \lambda_k \left( \|\theta^{(k)} + \hat{\Delta}^{(k)}\|_1 - \|\theta^{(k)}\|_1 \right) + \lambda_0 \left( \|\theta^{(0)} + \hat{\Delta}^{(0)}\|_1 - \|\theta^{(0)}\|_1 \right)
\]
\[
\geq \sum_{k=1}^{K} \lambda_k \left( \|\Delta^{(k)}\|_1 - 2 \|\theta^{(k)}\|_1 \right) + \lambda_0 \left( \|\hat{\Delta}^{(0)}_S\|_1 - \|\hat{\Delta}^{(0)}_S\|_1 - 2 \|\theta^{(0)}\|_1 \right)
\]
\[
\geq \sum_{k=1}^{K} \lambda_k \|\Delta^{(k)}\|_1 - 2 \sum_{k=1}^{K} \lambda_k h_k + \lambda_0 \left( \|\hat{\Delta}^{(0)}_S\|_1 - \|\hat{\Delta}^{(0)}_S\|_1 \right)
\]  \hspace{1cm} (45)

Combining (44) and (45) yields

\[
0 \geq F(\hat{\Delta}) \geq \sum_{k=1}^{K} \frac{\lambda_k}{2} \|\Delta^{(k)}\|_1 - 2 \sum_{k=1}^{K} \lambda_k h_k + \frac{\lambda_0}{2} \left( \|\hat{\Delta}^{(0)}_S\|_1 - 3 \|\hat{\Delta}^{(0)}_S\|_1 \right)
\]  \hspace{1cm} (46)

which leads to the following inequality:

\[
\sum_{k=0}^{K} \lambda_k \|\Delta^{(k)}\|_1 + \lambda_0 \|\hat{\Delta}^{(0)}\|_1 \leq 4\lambda_0 \|\hat{\Delta}^{(0)}_S\|_1 + 4 \sum_{k=1}^{K} \lambda_k h_k
\]

Recalling that we define \( \hat{\Delta}^w = \sum_{k=1}^{K} \frac{n_S}{N} \hat{\Delta}^{(k)} + \hat{\Delta}^{(0)} \), it follows that

\[
\sum_{k=1}^{K} \lambda_k \|\Delta^{(k)}\|_1 + \lambda_0 \|\hat{\Delta}^{(0)}\|_1 \leq 4\lambda_0 \|\hat{\Delta}^{(0)}_S\|_1 + 4 \sum_{k=1}^{K} \lambda_k h_k
\]
\[
\leq 4\lambda_0 \|\hat{\Delta}^w\|_1 + 4 \sum_{k=1}^{K} \lambda_0 \frac{n_S}{N} \|\hat{\Delta}^{(k)}_S\|_1 + 4 \sum_{k=1}^{K} \lambda_k h_k
\]  \hspace{1cm} (47)

Since by the choice of parameters, we have \( \frac{\lambda_k}{2} \geq 4\lambda_0 \frac{n_S}{N} \), so reorganizing (47) yields

\[
\sum_{k=1}^{K} \lambda_k \|\Delta^{(k)}\|_1 + 2\lambda_0 \|\hat{\Delta}^{(0)}\|_1 \leq 8\lambda_0 \|\hat{\Delta}^w\|_1 + 8 \sum_{k=1}^{K} \lambda_k h_k
\]  \hspace{1cm} (48)

which is as desired.
B.3 Proof of Lemma 4

Applying Lemma 1 and Jensen’s inequality, along with some algebra, we have

\[
\begin{align*}
&\mathcal{L}\left(\theta^* + \hat{\Delta}\right) - \mathcal{L}(\theta^*) - \left\langle \nabla \mathcal{L}(\theta^*), \hat{\Delta}\right\rangle \\
&= \Delta^\top \nabla^2 \mathcal{L}(\theta^* + \gamma \hat{\Delta}) \Delta \quad (\gamma \in (0, 1)) \\
&= \sum_{k=1}^{K} \frac{n_S}{N} (\hat{\Delta}^{(k)})^\top \Sigma^{(k)} \hat{\Delta}^{(k)} + 2 \sum_{k=1}^{K} \frac{n_S}{N} (\hat{\Delta}^{(k)})^\top \Sigma^{(k)} (\hat{\Delta}^{(0)})^\top \left( \sum_{k=1}^{K} \frac{n_S}{N} \Sigma^{(k)} + \frac{n_T}{N} \Sigma^{(0)} \right) \hat{\Delta}^{(0)} \\
&= \sum_{k=1}^{K} \frac{n_S}{N} (\hat{\Delta}^{(k)} + \hat{\Delta}^{(0)})^\top \Sigma^{(k)} (\hat{\Delta}^{(k)} + \hat{\Delta}^{(0)}) + \frac{n_T}{N} (\hat{\Delta}^{(0)})^\top \Sigma^{(0)} \hat{\Delta}^{(0)} \\
&\geq \sum_{k=1}^{K} \frac{n_S \beta_k}{N} \left\| \hat{\Delta}^{(k)} + \hat{\Delta}^{(0)} \right\|_2^2 + \frac{n_T \alpha_0}{N} \left\| \hat{\Delta}^{(0)} \right\|_2^2 - \mathcal{R}(\hat{\Delta}) \\
&\geq \alpha_{\min} \left\| \sum_{k=1}^{K} \frac{n_S}{N} \hat{\Delta}^{(k)} + \hat{\Delta}^{(0)} \right\|_2^2 - \mathcal{R}(\hat{\Delta}) \\
&= \alpha_{\min} \left\| \hat{\Delta}^w \right\|_2^2 - \mathcal{R}(\hat{\Delta})
\end{align*}
\]

(49)

with probability larger than \(1 - c_1 \exp(-c_2 n_T)\), where we define \(\alpha_{\min} := \min_{0 \leq k \leq K} \alpha_k\) and

\[
\mathcal{R}(\hat{\Delta}) := \sum_{k=1}^{K} \frac{n_S \beta_k}{N} \log p \left\| \hat{\Delta}^{(k)} + \hat{\Delta}^{(0)} \right\|_1^2 + \frac{n_T \alpha_0}{N} \log p \left\| \hat{\Delta}^{(0)} \right\|_1^2.
\]

Notice that by triangle inequality,

\[
\mathcal{R}(\hat{\Delta}) = \sum_{k=1}^{K} \frac{\beta_k \log p}{N} \left\| \hat{\Delta}^{(k)} + \hat{\Delta}^{(0)} \right\|_1^2 + \frac{\alpha_0 \log p}{N} \left\| \hat{\Delta}^{(0)} \right\|_1^2 \\
\leq \sum_{k=1}^{K} \frac{2 \beta_k \log p}{N} \left\| \hat{\Delta}^{(k)} \right\|_1^2 + \sum_{k=0}^{K} \frac{2 \beta_k \log p}{N} \left\| \hat{\Delta}^{(0)} \right\|_1^2
\]

According to the restricted set of directions outlined in (48), it holds that

\[
\sum_{k=1}^{K} \lambda_k \left\| \hat{\Delta}^{(k)} \right\|_1 + \lambda_0 \left\| \hat{\Delta}^{(0)} \right\|_1 \leq \sum_{k=1}^{K} \lambda_k \left\| \hat{\Delta}^{(k)} \right\|_1 + 2 \lambda_0 \left\| \hat{\Delta}^{(0)} \right\|_1 \leq 8 \lambda_0 \left\| \hat{\Delta}^w \right\|_1 + 8 \sum_{k=1}^{K} \lambda_k h_k.
\]

with probability larger than \(1 - c_1 \exp(-c_2 \log p)\).
So if we define $\beta_{\text{max}} = \max_{0 \leq k \leq K} \beta_k$, by triangle inequality and the fact that $|S| = s$, we then have

$$R(\Delta) \leq \frac{2 \beta_{\text{max}} \log p}{N} \left( \sum_{k=1}^{K} \frac{\lambda_k^2}{\lambda_k^2} \left\| \Delta^{(k)} \right\|_1^2 + (K + 1) \frac{\lambda_0^2}{\lambda_0^2} \left\| \Delta^{(0)} \right\|_1^2 \right) \leq \frac{2 \beta_{\text{max}}}{\lambda_k^2 \wedge (\lambda_0^2/(K + 1))} \frac{\log p}{N} \left( \sum_{k=1}^{K} \lambda_k \left\| \Delta^{(k)} \right\|_1^2 + \lambda_0 \left\| \Delta^{(0)} \right\|_1^2 \right) \leq \frac{2 \beta_{\text{max}}}{\lambda_k^2 \wedge (\lambda_0^2/(K + 1))} \frac{\log p}{N} \left( 8 \lambda_0 \left\| \Delta^w \right\|_1^2 + 8 \sum_{k=1}^{K} \lambda_k h_k \right) \leq \frac{2 \beta_{\text{max}}}{\lambda_k^2 \wedge (\lambda_0^2/(K + 1))} \frac{\log p}{N} \left( 128 \lambda_0^2 \left\| \Delta^w \right\|_1^2 + 128 \left( \sum_{k=1}^{K} \lambda_k h_k \right)^2 \right) \leq \frac{2 \beta_{\text{max}}}{\lambda_k^2 \wedge (\lambda_0^2/(K + 1))} \frac{\log p}{N} \left( 128 \lambda_0^2 \left\| \Delta^w \right\|_2^2 + 128 \left( \sum_{k=1}^{K} \lambda_k h_k \right)^2 \right) \leq \frac{2 \beta_{\text{max}}}{\lambda_k^2 \wedge (\lambda_0^2/(K + 1))} \frac{\log p}{N} \left( \sum_{k=1}^{K} \lambda_k h_k \right)$$

Recall that we introduce the shorthand $u_n = \frac{256 \beta_{\text{max}} \lambda_0^2}{s_{\min} \lambda_0^2 (\lambda_0^2/(K + 1))} \frac{\log p}{N}$, and $v_n = \frac{256 \beta_{\text{max}}}{s_{\min} \lambda_0^2 (\lambda_0^2/(K + 1))} \frac{\log p}{N} \left( \sum_{k=1}^{K} \lambda_k h_k \right)$, combining the above argument with (49) leads to

$$\mathcal{L}(\theta^* + \Delta) - \mathcal{L}(\theta^*) - \left\langle \nabla \mathcal{L}(\theta^*), \Delta \right\rangle \geq (1 - u_n) \alpha_{\min} \left\| \Delta^w \right\|_2^2 - v_n \sum_{k=1}^{K} \lambda_k h_k \quad (50)$$

with probability larger than $1 - c_1 \exp(-c_2 n_T) - c_3 \exp(-c_4 \log p)$, which finishes the proof.

### B.4 Proof of Lemma 5

The result in $\ell_2$-norm follows directly from the proof of Theorem 1. Here we investigate the upper bound for the estimation error in $\ell_1$-norm, i.e., $\left\| \Delta^w \right\|_1$. We categorize the problem into two cases based on the relationship between $\frac{1}{2} \lambda_0 \left\| \Delta^w \right\|_1$ and $2 \sum_{k=1}^{K} \lambda_k h_k$, and discuss by cases.

Starting with the first scenario where $\frac{1}{2} \lambda_0 \left\| \Delta^w \right\|_1 > 2 \sum_{k=1}^{K} \lambda_k h_k$, in such case (22) implies

$$0 \geq \Delta^T \Sigma \Delta - \frac{7}{4} \lambda_0 \left\| \Delta^w \right\|_1 + 2 \lambda_0 \left\| \Delta^w \right\|_1 \geq - \frac{7}{4} \lambda_0 \left\| \Delta^w \right\|_1 + 2 \lambda_0 \left\| \Delta^w \right\|_1\left\| \Delta^w \right\|_1,$$

which implies $\frac{1}{2} \left\| \Delta^w \right\|_1 \leq \frac{7}{4} \left\| \Delta^w \right\|_1$, and $\left\| \Delta^w \right\|_1 \leq 8 \left\| \Delta^w \right\|_1 \leq 8 \sqrt{s} \left\| \Delta^w \right\|_2 \leq 8 \sqrt{s} \left\| \Delta^w \right\|_2$. Therefore, in this case, we obtain

$$\left\| \Delta^w \right\|_1 \leq s \lambda_0 + \sqrt{s} \sqrt{\sum_{k=1}^{K} \lambda_k h_k}$$

with probability larger than $1 - c_1 \exp(c_2 n_T) - c_3 \exp(c_4 \log p)$.

We now transit to the second scenario when $\frac{1}{2} \lambda_0 \left\| \Delta^w \right\|_1 \leq 2 \sum_{k=1}^{K} \lambda_k h_k$. In this instance, we directly obtain

$$\left\| \Delta^w \right\|_1 \leq \frac{8 \sum_{k=1}^{K} \lambda_k h_k}{\lambda_0}$$

Taking into account both the discussed scenarios, we can conclude that

$$\left\| \Delta^w \right\|_1 \leq s \lambda_0 + \sqrt{s} \left( \sqrt{\sum_{k=1}^{K} \lambda_k h_k} + \frac{\sum_{k=1}^{K} \lambda_k h_k}{\lambda_0} \right) \quad (51)$$
with probability larger than \( 1 - c_1 \exp(c_2 n_T) - c_3 \exp(c_4 \log p) \), which is as desired.

### B.5 Proof of Lemma 6

We start with the first term, \( \frac{1}{n_S} \Theta^{(k)} (X^{(k)})^\top \epsilon^{(k)} \). Let \( (a_j^{(k)})^\top := e_j^\top \Theta^{(k)} (X^{(k)})^\top \). Similar to the proof of Lemma 2, by Vershynin (2010), Proposition 5.10 and a union bound,

\[
P \left( \max_{1 \leq j \leq p} \left| \frac{1}{n_S} (a_j^{(k)})^\top \epsilon^{(k)} \right| > t \right) \leq p \exp \left( - \frac{c n_S^2 t^2}{\sigma_k^2 \max_{1 \leq j \leq p} \| a_j^{(k)} \|_2^2} \right)
\]

for some universal constant \( c > 0 \). Therefore, to prove (32), it suffices to bound

\[
c n_S : = \max_{1 \leq j \leq p} \| a_j^{(k)} \|_2 = \max_{1 \leq j \leq p} \left( \Theta^{(k)} \Sigma^{(k)} \Theta^{(k)} \right)^{\top}_{j,j}.
\]

In order to accomplish this, we employ Lemma 23 from Lee et al. (2017), which is formulated as follows.

**Lemma 8.** Under the conditions of Lemma 6,

\[
P \left( \max_{1 \leq j \leq p} \left( \Sigma^{(k)} \right)_j^{-1} \left( \Sigma^{(k)} \right)_j^{-1} > 2 \max_{1 \leq j \leq p} \left( \Sigma^{(k)} \right)_j^{-1} \right) \leq 2pe^{-c_1 n_S}
\]

for some universal constant \( c_1 > 0 \).

Since \( \Theta^{(k)} \) is the solution of problem (58), combining the optimally condition with Lemma 8 implies

\[
\max_{1 \leq j \leq p} \left( \Theta^{(k)} \Sigma^{(k)} \Theta^{(k)} \right)^{\top}_{j,j} \leq \max_{1 \leq j \leq p} \left( \Sigma^{(k)} \right)_j^{-1} \Sigma^{(k)} \left( \Sigma^{(k)} \right)_j^{-1} \leq 2 \max_{1 \leq j \leq p} \left( \Sigma^{(k)} \right)_j^{-1}
\]

with probability at least \( 1 - 2pe^{-c_1 n_S} \). By assumption 1, \( (\Sigma^{(k)})_{j,j}^{-1} \) is bounded above. Therefore (32) holds.

Next, we aim at the second term, \( b^{(k)} \). Applying Hölder’s inequality to each component we obtain

\[
\| b^{(k)} \|_\infty = \left\| \left( \Theta^{(k)} \Sigma^{(k)} - I \right) (\hat{\beta}_\text{LASSO}^{(k)} - \beta^{(k)}) \right\|_\infty
\leq \max_{1 \leq j \leq p} \| \Theta_j^{(k)} \Sigma^{(k)} - e_j^\top \|_\infty \| \hat{\beta}_\text{LASSO}^{(k)} - \beta^{(k)} \|_1
\]

where recall that \( \Theta_j^{(k)} \) denotes the \( j \)-th row of \( \Theta^{(k)} \). By the optimality condition of (2) and the choice of \( \mu_k \), we have

\[
\max_{1 \leq j \leq p} \| \Theta_j^{(k)} \Sigma^{(k)} - e_j^\top \|_\infty \leq \sqrt{\frac{\log p}{n_S}}.
\]

In addition, recall that \( \| \beta^{(0)} \|_0 = s \) and \( \| \beta^{(0)} - \beta^{(k)} \|_1 \leq h_k \), so we can show the following Lemma:

**Lemma 9.** Under the condition of Lemma 6, we have

\[
\| \hat{\beta}_\text{LASSO}^{(k)} - \beta^{(k)} \|_1 \leq s \sqrt{\frac{\log p}{n_S}} + h_k
\]

with probability at least \( 1 - c_1 p^{-c_2} \).

The proof of Lemma 9 comes from a direct application of Lemma 1 in Li et al. (2022). Integrating the above arguments leads to \( \| b^{(k)} \|_\infty \leq p \frac{\log p}{n_S} + h_k \sqrt{\frac{\log p}{n_S}} \) with probability larger than \( 1 - c_1 \exp(c_2 \log p) \). This finishes the proof.
B.6 Proof of Lemma 7

Following the proof of Lemma 2, by applying Hölder’s inequality, we obtain

\[
|\langle \nabla L(\theta), \Delta \rangle| = \sum_{k=1}^{K} \left| \left\langle \frac{\sqrt{n_S}}{N} e^{(k)}, \Delta^{(k)} \right\rangle \right| + \left| \left\langle \sum_{k=1}^{K} \frac{\sqrt{n_S}}{N} e^{(k)} + \frac{1}{N} \left( X^{(0)} \right)^T e^{(0)}, \Delta^{(0)} \right\rangle \right|
\]

\[
\leq \sum_{k=1}^{K} \left\| \frac{\sqrt{n_S}}{N} e^{(k)} \right\|_\infty \left\| \Delta^{(k)} \right\|_1 + \left\| \sum_{k=1}^{K} \frac{\sqrt{n_S}}{N} e^{(k)} + \frac{1}{N} \left( X^{(0)} \right)^T e^{(0)} \right\|_\infty \left\| \Delta^{(0)} \right\|_1
\]

\[
\leq \sum_{k=1}^{K} \left\| \frac{1}{N} \Theta^{(k)} \left( X^{(k)} \right)^T e^{(k)} + \frac{n_S}{N} b^{(k)} \right\|_\infty \left\| \Delta^{(k)} \right\|_1
\]

\[
+ \left\| \frac{1}{N} \left( \sum_{k=1}^{K} \Theta^{(k)} \left( X^{(k)} \right)^T e^{(k)} + \left( X^{(0)} \right)^T e^{(0)} \right) + \sum_{k=1}^{K} \frac{n_S}{N} b^{(k)} \right\|_\infty \left\| \Delta^{(0)} \right\|_1
\]

We start with the first term on the right-hand side of the inequality. In Lemma 6 we have shown that with probability larger than \( 1 - c_1 \exp(-c_2 \log p) \),

\[
\frac{1}{n_S} \left\| \Theta^{(k)} \left( X^{(k)} \right)^T e^{(k)} \right\|_\infty \lesssim \sqrt{\frac{\log p}{n_S}} \quad \text{and} \quad \left\| b^{(k)} \right\|_\infty \lesssim \frac{\log p}{n_S} + h_k \sqrt{\frac{\log p}{n_S}}.
\]

Thus in order to guarantee

\[
\lambda_k \geq \left\| \frac{1}{N} \Theta^{(k)} \left( X^{(k)} \right)^T e^{(k)} + \frac{n_S}{N} b^{(k)} \right\|_\infty,
\]

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

Next, we shift our focus to the second term. Similar to the arguments in Proposition 6, we denote \( \left( a_j^{(k)} \right)^T := e_j^{(k)} \Theta^{(k)} \left( X^{(k)} \right)^T \) for \( k = 1, \ldots, K \) and \( \left( a_j^{(0)} \right)^T := e_j^T \left( X^{(0)} \right)^T \). By Vershynin (2010), Proposition 5.10 and a union bound,

\[
P \left( \max_{1 \leq j \leq p} \left| \frac{1}{N} \sum_{k=0}^{K} a_j^{(k)^T} e^{(k)} \right| > t \mid \left\{ a_j^{(k)} \right\}_{1 \leq k \leq K} \right) \leq p \exp \left( - \frac{c N^2 t^2}{\max_{1 \leq j \leq p} \left| a_j^{(k)} \right|_2^2} \right)
\]

According to Proposition 6 and Assumption 1, \( \max_{0 \leq k \leq K, 1 \leq j \leq p} \left| a_j^{(k)} \right|_2^2 \) is bounded above with probability larger than \( 1 - c_1 \exp(-c_2 \log p) \). This result together with (52) indicates that

\[
\left\| \frac{1}{N} \left( \sum_{k=1}^{K} \Theta^{(k)} \left( X^{(k)} \right)^T e^{(k)} + \left( X^{(0)} \right)^T e^{(0)} \right) + \sum_{k=1}^{K} \frac{n_S}{N} b^{(k)} \right\|_\infty \lesssim \sqrt{\frac{\log p}{N}} + \frac{K s \log p}{N} + \sum_{k=1}^{K} \frac{n_S}{N} \sqrt{\frac{\log p}{n_S} h_k}
\]

so it suffices to choose \( \lambda_0 = c_0 \left( \sqrt{\frac{\log p}{N}} + \frac{K s \log p}{N} + \sum_{k=1}^{K} \frac{n_S}{N} \sqrt{\frac{\log p}{n_S} h_k} \right) \) for some constant \( c_0 \). This finishes the proof.

C Impact of Covariate Shift on \( C_\Sigma \)

Recall constant \( C_\Sigma \) defined as

\[
C_\Sigma := 1 + \max_{j \leq p} \max_k \left\| e_j^T \left( \Sigma^{(k)} - \Sigma^{(0)} \right) \right\|_1 \left( \sum_{1 \leq k \leq K} \frac{1}{K} \Sigma^{(k)} \right)^{-1}.
\]
Clearly, $C_2$ depends on the difference between the source and target covariance matrices. In the following, we provide an example where $C_2$ diverges with $p$.

Let $K = 1$, i.e., there is only one source dataset, and the target data covariance matrix $\Sigma^{(0)} = I$. The source data covariance $\Sigma^{(k)}$ is constructed as $\Sigma^{(k)} = \alpha A + (1 - \alpha) I$, where $A$ is an arrowhead matrix with all nonzero elements equal to one. Next, we provide the choice of $\alpha$ such that the eigenvalues of $\Sigma^{(k)}$ are bounded, thus satisfying Assumption 1. To this end, we first provide an expression of the eigenvalues of $A$. Applying Cauchy’s interlace theorem, we have $\lambda_{\min}(A) \leq 1$, $\lambda_{\max}(A) \geq 1$, and all of the rest eigenvalues equal to one. Further using the fact that $\text{Tr}(A) = p$ and $\det(A) = -(p - 2)$ we conclude $\lambda_{\min}(A) = 1 - \sqrt{p-1}$ and $\lambda_{\max}(A) = 1 + \sqrt{p-1}$.

Based on the eigenvalues we set $\alpha = c/\sqrt{p-1}$ with constant $c \in (0,1)$, which gives $\lambda_{\min}(\Sigma^{(k)}) = 1 - c$ and $\lambda_{\max}(\Sigma^{(k)}) = 1 + c$.

We then compute $C_2$ under the above setting. Using the formula for the inverse of arrowhead matrices provided in Salkuyeh and Beik (2018), we obtain

$$
(\Sigma^{(k)} - \Sigma^{(0)}) \left( \sum_{1 \leq k \leq K} \frac{1}{K} \Sigma^{(k)} \right)^{-1} = \left( \begin{array}{c|c} 1 - \beta^{-1} & \alpha \beta^{-1} \cdots \alpha \beta^{-1} \\ \hline 1 & \alpha \beta^{-1} \\ \vdots & \alpha \beta^{-1} \\ \alpha \beta^{-1} & \\ \hline \alpha \beta^{-1} & \end{array} \right)_{p \times p},
$$

where $\beta := 1 - (p-1)\alpha^2$. Substituting $\alpha = c/\sqrt{p-1}$ reveals $C_2 = O(\sqrt{p})$.

### D Implementation of TransFusion

In this section, we show how the first step of TransFusion method can be implemented using the Proximal Gradient Descent (PGD) algorithm via a change of variables. The second de-bias step (5) is a standard LASSO problem and there is a rich literature on efficient numerical solutions, see Li et al. (2018) and the references therein.

For the first co-training step, notice that under the one-to-one variable transformation $\theta := ((\beta^{(1)} - \beta^{(0)})^\top, (\beta^{(2)} - \beta^{(0)})^\top, \ldots, (\beta^{(K)} - \beta^{(0)})^\top, \beta^{(0)})^\top$, solving problem (2) is equivalent to solving

$$
\hat{\theta} \in \arg \min_{\theta} \left\{ \frac{1}{2N} \| y - X \theta \|_2^2 + \lambda_0 \sum_{k=0}^K a_k \left\| \theta^{(k)} \right\|_1 \right\},
$$

where $a_0 = 1$ and $y$ and $X$ are defined in (16), recalled below for convenience:

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

Problem (54) is a weighted LASSO problem (Zou, 2006), and thus the proximal gradient descent algorithm can be directly applied. Given initialization $\theta_0 \in \mathbb{R}^{(K+1)p}$ and proximal parameter $\gamma > 0$, the PGD iteration reads:

$$
\theta_{t+1} = \arg \min_{\theta \in \mathbb{R}^{(K+1)p}} \left\{ \frac{1}{N} \langle X^\top (X \theta_t - y), \theta - \theta_t \rangle + \frac{1}{2\gamma} \| \theta - \theta_t \|_2^2 + \lambda_0 \sum_{k=0}^K a_k \left\| \theta^{(k)} \right\|_1 \right\}.
$$

Notably, although calculating the gradient $\frac{1}{N} X^\top (X \theta_t - y)$ appears to involve multiplying a $(K+1)p \times (K+1)p$ matrix by a $(K+1)p$-dimensional vector, using the sparse structure of $X$ it is easy to see it can be obtained via computing quantities $X^{(k)^\top} X^{(k)} \beta^{(k)}$ and $X^{(k)^\top} y^{(k)}$ for $k = 0, \ldots, K$. Therefore per iteration it only involves the multiplication of a $p \times p$ matrix by a $p$-dimensional vector, and can be computed efficiently in parallel. The proximal mapping (55) can also be computed in closed form via soft-thresholding.
Figure 3: Comparison of estimation errors under (i) diverse and (ii) non-diverse source task settings with (a) homogeneous design and (b) heterogeneous design with a large choice of $K$.

Figure 4: Comparison of estimation errors under (i) diverse source task settings with (a) homogeneous design and (b) heterogeneous design with different choice of $n_S$ and a fixed $K = 10$. 
E Additional Simulation Results

Fig. 3 shows the results with the number of source tasks $K \in \{1, 5, 9, 13, 17\}$. The results are in general analogous to those in section 4. One interesting observation is that in the diverse source task setting (i), as the source task number $K$ increases and the source task sample size $n_S$ remains a constant, Fig. 3 (i-a) and (i-b) show diminishing improvement of TransFusion estimation error. This is because as our theoretical results suggest, in such settings, TransFusion achieves an estimation error of order $O\left(\frac{s \log p}{n_T + Kn_S} + (1 + v_n)\frac{\sqrt{\log p}}{n_T}\right)$ with $v_n = K^2 \log p \frac{\log n}{n_S}$. If we fix the source sample size $n_S$, increasing $K$ only decreases the first term, and the overall sum will be dominated by the second term. The term $v_n$ contributes to the U-shape error curve in the figures. This is the cost of using only local data to estimate the task-specific signal $\delta^{(k)}$ in order to achieve robustness to covariate shifts.

Meanwhile, the theory implies that TransFusion would have a consistent error improvement if we proportionally increase the source sample size, $n_S$, with $K$. This is verified in Fig. 4, where, in the same diverse source task setting, a fixed $K$ with a growing $n_S$ results in a faster error reduction for TransFusion compared to other methods, especially in the existence of covariate shifts (Fig. 4 (i-b)).

F Case Study: Handwritten-digit Classification

We consider the problem of handwritten-digit classification based on the MNIST-C (Mu and Gilmer, 2019) dataset. MNIST-C is a comprehensive suite of different corruptions applied to the MNIST dataset, for benchmarking out-of-distribution robustness in computer vision. This setup allows us to evaluate the covariate-shift robustness of the proposed TransFusion algorithm.

We choose images corrupted by “brightness”, “fog” and “motion blur” as the source datasets ($K = 3$), drawing $n_S$ source samples from each with $n_S \in \{500, 1000, 1500, 2000\}$. Then we set the original MNIST dataset as the target dataset, from which we collect $n_T = 100$ target samples. We use flattened pixel features of the images as features, amounting to $28 \times 28 = 784$ features per image ($p = 784$). We transform the classification problem into 10 binary classification problems—one for each digit against all others, then evaluate the classification accuracy on 2000 test images from the target dataset.

For this multi-source binary classification task, we employ the TransFusion algorithm and compare its performance with the TransLasso algorithm and Lasso (baseline) algorithm, all based on the logistic regression. Note that although in this paper we mainly focus on the linear regression setting, our theory and methodology can be easily generalized to the logistic regression setting (Friedman et al., 2010; Negahban et al., 2012). The implementation follows a similar manner as discussed in Appendix D.

One of the key challenges in this classification problem is managing the covariate shifts between different tasks. This was evident from the correlation heat maps of flattened pixel features in Figure 5. Compared to the target sample, each source sample exhibits a distinct covariate correlation structure. Such covariate shifts should be carefully handled for effective knowledge transfer.

![Figure 5](image-url)
Figure 6: Test accuracy of different methods for the binary digit classification problem averaged over 10 problems, plotted against varying source sample sizes $n_S$.

Figure 6 shows the average test accuracy across the 10 binary classification problems versus the source sample size $n_S$. From the figure, we can see that TransFusion consistently outperforms other benchmarks. When the source sample size is relatively small, the Trans-Lasso method even performs worse than the baseline Lasso method, suggesting that it fails to utilize the transferable knowledge under the covariate shift. In contrast, TransFusion method still outperforms the baseline, indicating its robustness against the covariate shift.

G Choice of $\tilde{\beta}^{(k)}$

In this section, we specify how to choose $\tilde{\beta}^{(k)}$. An intuitive option is the LASSO estimator computed based on source sample $(X^{(k)}, y^{(k)})$:

$$
\hat{\beta}^{(k)}_{\text{LASSO}} \in \arg\min_{\beta \in \mathbb{R}^{k+1}} \left\{ \frac{1}{2n_S} \left\| y^{(k)} - X^{(k)} \beta \right\|^2_2 + \tilde{\lambda}_k \| \beta \|_1 \right\}.
$$

(56)

However, since the LASSO estimator is biased, computing $\hat{w}_C$ by aggregating the local $\hat{\beta}^{(k)}_C$'s can only reduce the variance and has almost no effects on the bias (Mcdonald et al., 2009). To overcome such a drawback, we propose to first “correct” the bias at the local level before transmitting it to the target node for transfer learning. This is achieved by debiasing $\hat{\beta}^{(k)}_{\text{LASSO}}$ using the method proposed in (Javanmard and Montanari, 2014):

$$
\tilde{\beta}^{(k)} = \hat{\beta}^{(k)}_{\text{LASSO}} + \frac{1}{n_S} \hat{\Theta}^{(k)} (X^{(k)})^\top \left( y^{(k)} - X^{(k)} \hat{\beta}^{(k)}_{\text{LASSO}} \right).
$$

(57)

Here, $\hat{\Theta}^{(k)}$ serves as an approximation of $(\Sigma^{(k)})^{-1}$, whose $j$-th row is defined as the solution of the following optimization problem

$$
\begin{align*}
\text{minimize}_{\theta_j \in \mathbb{R}^p} & \quad \theta_j^\top \hat{\Sigma}^{(k)} \theta_j \\
\text{subject to} & \quad \left\| \hat{\Sigma}^{(k)} \theta_j - e_j \right\|_\infty \leq \mu_k,
\end{align*}
$$

(58)

with parameter $\mu_k > 0$ properly chosen. The source code for solving the problem can be found at https://web.stanford.edu/~montanar/sslasso/code.html.

To understand the choice of $\tilde{\beta}$, we may rewrite (57) by subtracting $\beta^{(k)}$ from both sides to obtain

$$
\tilde{\beta}^{(k)} - \beta^{(k)} = \frac{1}{n_S} \hat{\Theta}^{(k)} (X^{(k)})^\top (y^{(k)} - X^{(k)} \beta^{(k)}) (\hat{\beta}^{(k)}_{\text{LASSO}} - \beta^{(k)}).
$$

(59)

The first term on the right-hand side of the equation is associated with the variance of $\tilde{\beta}^{(k)}$. Through (58), we effectively minimize this variance. The second term, $b^{(k)}$, on the other hand, contributes to the bias. By selecting...
an appropriate constraint parameter $\mu_k$ on $\|\hat{\Sigma}^{(k)} \theta_j - \epsilon_j\|_\infty$ in (58), we can control the bias term to be comparable or even smaller than the variance term, thereby mitigate the impact of bias on the later aggregation step. Hence, this choice of $\tilde{\beta}$ guarantees that the D-TransFusion could achieve much less communication overhead, while at the same time achieving the minimum performance loss compared to centralized TransFusion.