

---

# Transfer Learning via $\ell_1$ Regularization

---

**Masaaki Takada**  
Toshiba Corporation  
Tokyo 105-0023, Japan  
masaaki1.takada@toshiba.co.jp

**Hironori Fujisawa**  
The Institute of Statistical Mathematics  
Tokyo 190-8562, Japan  
fujisawa@ism.ac.jp

## Abstract

Machine learning algorithms typically require abundant data under a stationary environment. However, environments are nonstationary in many real-world applications. Critical issues lie in how to effectively adapt models under an ever-changing environment. We propose a method for transferring knowledge from a source domain to a target domain via  $\ell_1$  regularization in high dimension. We incorporate  $\ell_1$  regularization of differences between source parameters and target parameters, in addition to an ordinary  $\ell_1$  regularization. Hence, our method yields sparsity for both the estimates themselves and changes of the estimates. The proposed method has a tight estimation error bound under a stationary environment, and the estimate remains unchanged from the source estimate under small residuals. Moreover, the estimate is consistent with the underlying function, even when the source estimate is mistaken due to nonstationarity. Empirical results demonstrate that the proposed method effectively balances stability and plasticity.

## 1 Introduction

Machine learning algorithms typically require abundant data under a stationary environment. However, real-world environments are often nonstationary due to, for example, changes in the users' preferences, hardware or software faults affecting a cyber-physical system, or aging effects in sensors [39]. Concept drift, which means the underlying functions change over time, is recognized as a root cause of decreased effectiveness in data-driven information systems [25].

Under an ever-changing environment, critical issues lie in how to effectively adapt models to a new environment. Traditional approaches tried to detect concept drift based on hypothesis test [11, 26, 19, 5], but they are hard to capture continuously ever-changing environments. Continuously updating approaches, in contrast, are effective for complex concept drift by avoiding misdetection. These include tree-based methods [7, 16, 24] and ensemble-based methods [31, 20, 10]. Additionally, parameter-based transfer learning for transferring knowledge from past (source domains) to present (target domains) has been studied empirically and theoretically [27, 35, 21, 22]. They employed an empirical risk minimization with  $\ell_2$  regularization, and the regularization was extended to strongly convex functions. However, these methods do not yield sparsity of parameter changes, so that even slight changes of data incur update of all parameters.

Specifically, we consider the problem of sparse regression [32, 14]. Sparse models are widely used in decision making since they have few active features and thus easy to obtain some insight. However, existing sparse regression methods are not necessarily effective for routine decision making, because they can significantly change parameters even when the data changes only slightly.

In this paper, we provide a method for transferring knowledge in high dimension via  $\ell_1$  regularization that allows sparse estimates and changes. We incorporate the  $\ell_1$  regularization of the difference between source parameters and target parameters into the ordinary Lasso regularization. The ordinary Lasso regularization has a role of restricting the model complexity in high dimension. The additional

$\ell_1$  regularization of difference plays a key role of sparse update, that is, only a small number of parameters are changed. Because of these two kinds of sparsity, it is easy to interpret and manage models and their changes. The proposed method has a single additional hyper-parameter compared to the ordinary Lasso. It controls the regularization strengths of estimates themselves and their changes, thereby balances stability and plasticity to mitigate so-called *stability-plasticity dilemma* [13, 6]. Therefore, our method transfers knowledge from past to present when the environment is stationary; while it discards the outdated knowledge when concept drift occurs.

Our main contribution is to give a method with clear theoretical justifications. We demonstrate three favorable characteristics for our method. First, our method presents a smaller estimation error than Lasso when the underlying functions do not change and the source estimate is the same as a target parameter. This indicates that our method effectively transfers knowledge under a stationary environment. Second, our method gives a consistent estimate even when the source estimate is mistaken, albeit with a weak convergence rate due to the phenomenon of so-called *negative transfer* [42]. This implies that our method can effectively discard the outdated knowledge and obtain new knowledge under nonstationary environment. Third, our method does not update estimates when the residuals of the predictions are small and the regularization is large. Hence, our method has an implicit stationarity detection mechanism.

The remainder of this paper is organized as follows. We begin with the description of the proposed method in Section 2. We also give some reviews on related work, including concept drift, transfer learning, and online learning. We next show some theoretical properties in Section 3. We finally illustrate empirical results in Section 4 and conclude in Section 5. All the proofs, as well as additional theoretical properties and empirical results, are given in the supplementary material.

## 2 Methods

### 2.1 Transfer Lasso

Let  $X_i \in \mathcal{X}$  and  $Y_i \in \mathbb{R}$  be the feature and response, respectively, for  $i = 1, \dots, n$ . Consider a linear function

$$f_\beta(\cdot) = \sum_{j=1}^p \beta_j \psi_j(\cdot),$$

where  $\beta = (\beta_j) \in \mathbb{R}^p$  and  $\psi_j(\cdot)$  is a dictionary function from  $\mathcal{X}$  to  $\mathbb{R}$ . Let the target function and noise be denoted by

$$f^*(\cdot) = f_{\beta^*}(\cdot) := \sum_{j=1}^p \beta_j^* \psi_j(\cdot) \text{ and } \varepsilon_i := Y_i - f^*(X_i),$$

and in matrix notion,  $\mathbf{f}^* := \mathbf{X}\beta^*$  and  $\varepsilon := \mathbf{y} - \mathbf{f}^*$ , where  $\mathbf{f}^* = (f^*(X_i)) \in \mathbb{R}^n$ ,  $\mathbf{X} = (\psi_j(X_i)) \in \mathbb{R}^{n \times p}$ ,  $\beta^* = (\beta_j^*) \in \mathbb{R}^p$ , and  $\mathbf{y} = (Y_i) \in \mathbb{R}^n$ .

In high-dimensional settings, a reasonable approach to estimating  $\beta^*$  is to assume sparsity of  $\beta^*$ , in which the cardinality  $s = |S|$  of its support  $S := \{j \in \{1, \dots, p\} : \beta_j^* \neq 0\}$  satisfies  $s \ll p$ , and to solve the Lasso problem [32], given by

$$\min_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{2n} \sum_{i=1}^n (Y_i - f_\beta(X_i))^2 + \lambda \|\beta\|_1 \right\}.$$

Lasso shrinks the estimate to zero and yields a sparse solution. We focus on the squared loss function, but it is applicable to other loss function, as seen in Section 4.3.

Suppose that we have an initial estimate of  $\beta^*$  as  $\tilde{\beta} \in \mathbb{R}^p$  and that the initial estimate is associated with the present estimate. Then, a natural assumption is that the difference between initial and present estimates is sparse. Thus, we employ the  $\ell_1$  regularization of the estimate difference and incorporate it into the ordinary Lasso regularization as

$$\hat{\beta} = \operatorname{argmin}_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{2n} \sum_{i=1}^n (Y_i - f_\beta(X_i))^2 + \lambda \left( \alpha \|\beta\|_1 + (1 - \alpha) \|\beta - \tilde{\beta}\|_1 \right) \right\} =: \mathcal{L}(\beta; \tilde{\beta}), \quad (1)$$

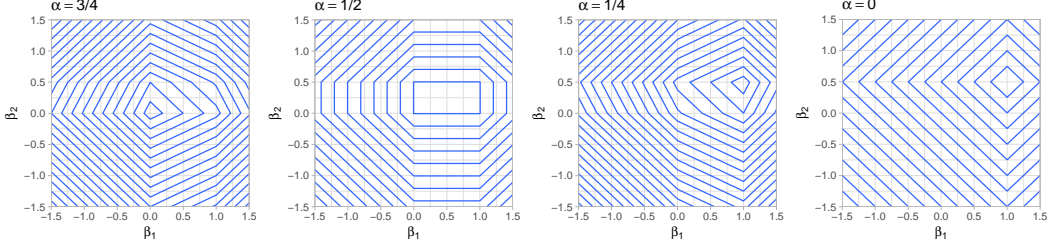


Figure 1: Contours of our regularizer for  $\alpha = 3/4, 1/2, 1/4, 0$  with  $\tilde{\beta} = (1, 1/2)^\top$ .

where  $\lambda = \lambda_n > 0$  and  $0 \leq \alpha \leq 1$  are regularization parameters. We call this method ‘‘Transfer Lasso’’. There are two anchor points, zero and the initial estimate. The first regularization term in (1) shrinks the estimate to zero and induces the sparsity of the estimate. The second regularization term in (1) shrinks the estimate to the initial estimate and induces the sparsity of changes from the initial estimates. The parameter  $\alpha$  controls the balance between transferring and discarding knowledge. It is preferable to transfer knowledge of the initial estimate when the underlying functions remain unchanged, while not preferable to transfer when a concept drift occurred. As a particular case, if  $\alpha = 1$ , Transfer Lasso reduces to ordinary Lasso and discards knowledge of the initial estimate. On the other hand, if  $\alpha = 0$ , Transfer Lasso reduces to Lasso predicting the residuals of the initial estimate,  $\mathbf{y} - \mathbf{X}\tilde{\beta}$ , and the initial estimate is utilized as a base learner. The regularization parameters,  $\lambda$  and  $\alpha$ , are typically determined by cross validation.

Figure 1 shows the contours of our regularizer for  $p = 2$ . Contours are polygons pointed at  $\beta_j = 0$  and  $\beta_j = \tilde{\beta}_j$  so that our estimate can shrink to zero and the initial estimate. The regularization parameter  $\alpha$  controls the shrinkage strengths to zero and the initial estimate. We also see that Transfer Lasso mitigates feature selection instability in the presence of highly correlated features. This is because the loss function tends to be parallel to  $\beta_1 + \beta_2 = c$  for highly correlated features but the contours are not necessarily parallel to  $\beta_1 + \beta_2 = c$  for a quadrant of  $\tilde{\beta}$ . For  $\alpha = 1/2$ , the sum of the two regularization terms equals  $\lambda\tilde{\beta}$  in the rectangle of  $\beta_j \in [\min\{\tilde{\beta}_j, 0\}, \max\{\tilde{\beta}_j, 0\}]$ . If the least square estimate lies in this region, it becomes the solution of Transfer Lasso, that is, it does not have any estimation bias.

## 2.2 Algorithm and Soft-Threshold Function

We provide a coordinate descent algorithm for Transfer Lasso. It is guaranteed to converge to a global optimal solution [36], because the problem is convex and the penalty is separable. Let  $\beta$  be the current value. Consider a new value  $\beta_j$  as a minimizer of  $\mathcal{L}(\beta; \tilde{\beta})$  when other elements of  $\beta$  except for  $\beta_j$  are fixed. We have

$$\partial_{\beta_j} \mathcal{L}(\beta; \tilde{\beta}) = -\frac{1}{n} \mathbf{X}_j^\top (\mathbf{y} - \mathbf{X}_{-j} \beta_{-j}) + \beta_j + \lambda \alpha \operatorname{sgn}(\beta_j) + \lambda(1 - \alpha) \operatorname{sgn}(\beta_j - \tilde{\beta}_j) = 0,$$

where  $\mathbf{X}_j$  and  $\mathbf{X}_{-j}$  denote the  $j$ -th column of  $X$  and  $X$  without  $j$ -th column, respectively, and  $\operatorname{sgn}(\cdot)$  denotes the sign function, hence we obtain the update rule as

$$\beta_j \leftarrow \mathcal{T} \left( \frac{1}{n} \mathbf{X}_j^\top (\mathbf{y} - \mathbf{X}_{-j} \beta_{-j}), \lambda, \lambda(2\alpha - 1), \tilde{\beta}_j \right),$$

where

$$\mathcal{T}(z, \gamma_1, \gamma_2, b) := \begin{cases} 0 & \text{for } -\gamma_1 \leq z \leq \gamma_2 \\ b & \text{for } \gamma_2 + b \leq z \leq \gamma_1 + b \\ z - \gamma_2 \operatorname{sgn}(b) & \text{for } \gamma_2 \leq z \leq \gamma_2 + b \\ z - \gamma_1 \operatorname{sgn}(z) & \text{for } \text{otherwise} \end{cases} \quad \left| \quad \begin{cases} b \geq 0 & b \leq 0 \\ -\gamma_2 \leq z \leq \gamma_1 & -\gamma_1 + b \leq z \leq -\gamma_2 + b \\ -\gamma_2 + b \leq z \leq -\gamma_2 & \text{otherwise.} \end{cases}$$

The computational complexity of Transfer Lasso is the same as ordinary Lasso.

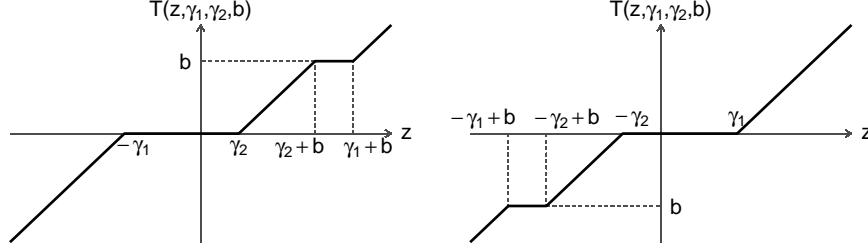


Figure 2: The soft-thresholding function for  $b \geq 0$  (left) and  $b \leq 0$  (right) with  $\gamma_1 > 0$  and  $|\gamma_2| \leq \gamma_1$ .

Figure 2 shows the soft-threshold function  $\mathcal{T}(z, \gamma_1, \gamma_2, b)$  with  $|\gamma_2| = |\lambda(2\alpha - 1)| \leq \lambda = \gamma_1$ . There are two steps at 0 and  $b = \tilde{\beta}$ . This implies that each parameter  $\tilde{\beta}_j$  is likely to be zero or the initial estimate  $\tilde{\beta}_j$ . As  $\alpha$  approaches 1, the step of the initial estimate disappears and reduces to the standard soft-thresholding function. As  $\alpha$  instead approaches 0, the step of zero disappears and the parameter only shrinks to the initial estimate.

### 2.3 Related Work

Transfer Lasso relates to concept drift, transfer learning, and online learning, as reviewed below.

Concept drift is a scenario where underlying functions change over time [12, 6]. There are two strategies for learning concept drift, active and passive approaches. The active approach explicitly detects concept drift and updates the model [11, 26, 19, 5]. Although they work well for abrupt concept drift, it is hard to capture gradual concept drift. The passive approach, on the other hand, continuously updates the model every time. There are some ensemble learner methods [31, 20, 10] and single learner methods including tree-based models [7, 16, 24] and neural network-based models [41, 3, 4]. They are effective for gradual and complex concept drift empirically. However, most methods always update models even if an environment is stationary, and these ad-hoc algorithms are hard to support their effectiveness theoretically. In contrast, Transfer Lasso can remain the estimate unchanged when the underlying functions do not change and also has some theoretical justifications.

Transfer learning is a framework that improves the performance of learners on target domains by transferring the knowledge of source domains [28, 38, 42]. This paper considers a homogeneous inductive transfer learning setting, which means that feature spaces and label spaces are the same between the source and target domains, and the label information of both domains is available. Hypothesis transfer learning is a typical approach for this problem [27, 35, 21, 22]. It transfer knowledge of source estimate  $\tilde{\beta}$  by solving

$$\operatorname{argmin}_{\beta \in \mathbb{R}^p} \frac{1}{2n} \sum_{i=1}^n (Y_i - f_\beta(X_i))^2 + \lambda \|\beta - \tilde{\beta}\|_2^2.$$

Similarly, single-model knowledge transfer [34] and multi-model knowledge transfer [33] employed another regularization  $\|\beta - \alpha\tilde{\beta}\|_2^2 = (1 - \alpha)\|\beta\|_2^2 + \alpha\|\beta - \tilde{\beta}\|_2^2 + \text{const}$ , where  $\alpha$  is a hyper-parameter. The  $\ell_2$  regularization and its extension of strongly convex regularization are easy to analyze the generalization ability theoretically. In contrast, Transfer Lasso employs  $\ell_1$  regularization, so that it yields sparsity of the changes of the estimates and requires different techniques for theoretical analysis. Sparsity is beneficial in practice because we can interpret and manage models by handling only a small number of estimates and their changes.

Online learning is a method where a learner attempts to learn from a sequence of instances one-by-one at each time [15]. The algorithms consist of the minimization of a cost function including  $\|\beta - \beta^t\|_2^2$ , where  $\beta^t$  is a previous estimate, to stabilize the optimization [23, 9, 40, 8]. This is related to Transfer Lasso by regarding  $\beta^t$  as an initial estimate, although these online algorithms work under a stationary environment.

### 3 Theoretical Properties

We analyze the statistical properties of Transfer Lasso. First, we construct estimation error bound and demonstrate the effectiveness under the correct and incorrect initial estimate. Second, we explicitly derive the condition that the model remains unchanged. Third, we investigate the behavior of Transfer Lasso when an initial estimate is a Lasso solution using another dataset.

#### 3.1 Estimation Error

We prepare the following assumption and definition for our analysis.

**Assumption 1** (Sub-Gaussian). *The noise sequence  $\{\varepsilon_i\}_{i=1}^n$  is i.i.d. sub-Gaussian with  $\sigma$ , i.e.,*

$$\mathbb{E}[\exp(t\varepsilon)] \leq \exp\left(\frac{\sigma^2 t^2}{2}\right), \quad \forall t \in \mathbb{R}.$$

**Definition 1** (Generalized Restricted Eigenvalue Condition (GRE)). *We say that the generalized restricted eigenvalue condition holds for a set  $\mathcal{B} \subset \mathbb{R}^p$  if we have*

$$\phi = \phi(\mathcal{B}) := \inf_{v \in \mathcal{B}} \frac{v^\top \frac{1}{n} \mathbf{X}^\top \mathbf{X} v}{\|v\|_2^2} > 0.$$

The GRE condition is a generalized notion of the restricted eigenvalue condition [1, 2, 14]. From the above assumption and definition, we have the following theorems and corollary. Let  $\Delta := \tilde{\beta} - \beta^*$  and  $v_S$  be the vector  $v$  restricted to the index set  $S$ .

**Theorem 1** (Estimation Error). *Suppose that Assumption 1 is satisfied. Suppose that the generalized restricted eigenvalue condition (Definition 1) holds for  $\mathcal{B} = \mathcal{B}(\alpha, c, \Delta)$ , where*

$$\mathcal{B}(\alpha, c, \Delta) := \{v \in \mathbb{R}^p : (\alpha - c)\|v_{S^c}\|_1 + (1 - \alpha)\|v - \Delta\|_1 \leq (\alpha + c)\|v_S\|_1 + (1 - \alpha)\|\Delta\|_1\},$$

with some constant  $c > 0$ . Then, we have

$$\|\hat{\beta} - \beta^*\|_2^2 \leq \frac{(\alpha + c)^2 \lambda_n^2 s}{\phi^2} \left(1 + \sqrt{1 + \frac{2(1 - \alpha)\phi\|\Delta\|_1}{(\alpha + c)^2 \lambda_n s}}\right)^2 \quad (2)$$

with probability at least  $1 - \nu_{n,c}$ , where  $\nu_{n,c} := \exp(-nc^2 \lambda_n^2 / 2\sigma^2 + \log(2p))$ .

The estimation error bound for Lasso is obtained from (2) with  $\alpha = 1$  and  $\Delta = 0$  as  $4(1 + c)^2 \lambda_n^2 s / \phi(\mathcal{B}_0)^2$ , where  $\mathcal{B}_0 = \mathcal{B}(1, c, 0) = \{v \in \mathbb{R}^p : (1 - c)\|v_{S^c}\|_1 \leq (1 + c)\|v_S\|_1\}$ . Consider the case  $\tilde{\beta} = \beta^*$ , that is,  $\Delta = 0$ . Then, the estimation error bound of Transfer Lasso reduces to  $4(\alpha + c)^2 \lambda_n^2 s / \phi(\mathcal{B}(\alpha, c, 0))^2$ , where  $\mathcal{B}(\alpha, c, 0) = \{v \in \mathbb{R}^p : (1 - c)\|v_{S^c}\|_1 \leq (2\alpha - 1 + c)\|v_S\|_1\}$ . Because  $\mathcal{B}(\alpha, c, 0) \subset \mathcal{B}_0$  and so  $\phi(\mathcal{B}(\alpha, c, 0)) \geq \phi(\mathcal{B}_0)$ , the bound of Transfer Lasso ( $\alpha < 1$ ) is smaller than that of Lasso.

The GRE condition for  $\mathcal{B}(\alpha, c, \Delta)$  is not so restricted. The ordinary restricted eigenvalue condition holds for quite general classes of Gaussian matrices with high probability [29]. The GRE condition for  $\mathcal{B}(\alpha, c, \Delta)$  with  $\Delta = 0$  holds under a milder condition because  $\mathcal{B}(\alpha, c, \Delta) \subset \mathcal{B}_0$ . The GRE condition for  $\mathcal{B}(\alpha, c, \Delta)$  with  $2\alpha - c - 1 > 0$  holds as well except for constant factors because  $\mathcal{B}(\alpha, c, \Delta) \subset \{v \in \mathbb{R}^p : (2\alpha - c - 1)\|v_{S^c}\|_1 \leq (1 + c)\|v_S\|_1\}$ .

**Theorem 2** (Convergence Rate). *Assume the same conditions as in Theorem 1. Then, with probability at least  $1 - \nu_{n,c}$ , we have*

$$\|\hat{\beta} - \beta^*\|_2^2 = O\left((\alpha + c)^2 \lambda_n^2 s + (1 - \alpha)\lambda_n \|\Delta\|_1\right), \quad \text{as } \lambda_n \rightarrow 0.$$

Let  $\lambda_n = O(\sqrt{\log p/n})$  and  $\|\Delta\|_1 = O(s\sqrt{\log p/n})$ . The order of  $\lambda_n$  comes from the constant value of  $\nu_{n,c}$ , and the order of  $\Delta$  is as in the ordinary Lasso rate. Then, the convergence rate is evaluated as  $\|\hat{\beta} - \beta^*\|_2^2 = O(s \log p/n)$ , which is an almost minimax optimal [30].

Let us consider a misspecified initial estimate,  $\|\Delta\|_1 \rightarrow 0$ . For example, the case  $\|\Delta\|_1 = O(s)$  is obtained when the initial estimate  $\tilde{\beta}$  fails to detect the true value  $\beta^*$ , but most of the zeros are truly identified. Transfer Lasso estimates retain consistency even in this situation when  $\|\Delta\|_1 \lambda_n \rightarrow 0$ , although the convergence rate becomes worse as  $\|\hat{\beta} - \beta^*\|_2^2 = O(\|\Delta\|_1 \sqrt{\log p/n})$  if  $\alpha < 1$ . This implies that negative transfer can happen but not severely, and is avoidable by setting  $\alpha = 1$ .

**Theorem 3** (Feature Screening). *Assume the same conditions as in Theorem 1. Suppose that the beta-min condition*

$$|\beta_S^*| > \frac{(\alpha + c)^2 \lambda_n^2 s}{\phi^2} \left( 1 + \sqrt{1 + \frac{2(1 - \alpha)\phi \|\Delta\|_1}{(\alpha + c)^2 \lambda_n s}} \right)^2$$

*is satisfied. Then, we have  $S \subset \text{supp}(\hat{\beta})$  with probability at least  $1 - \nu_{n,c}$ .*

This theorem implies that Transfer Lasso succeeds in feature screening if the true parameters are not so small. The minimum value of true parameters for Transfer Lasso can be smaller than that for Lasso when  $\|\Delta\|_1$  is small.

### 3.2 Unchanging Condition

The next theorem shows that the estimate remains unchanged under a certain condition.

**Theorem 4** (Unchanging Condition). *Let  $r(\beta) := \mathbf{y} - \mathbf{X}\beta$ . There exists an unchanging solution  $\hat{\beta} = \tilde{\beta}$  if and only if*

$$\begin{aligned} \left| \frac{1}{n} \mathbf{X}_j^\top r(\tilde{\beta}) \right| &\leq \lambda \text{ for } \forall j \text{ s.t. } \tilde{\beta}_j = 0, \text{ and} \\ -\lambda \left( (1 - \alpha) - \alpha \text{sgn}(\tilde{\beta}_j) \right) &\leq \frac{1}{n} \mathbf{X}_j^\top r(\tilde{\beta}) \leq \lambda \left( (1 - \alpha) + \alpha \text{sgn}(\tilde{\beta}_j) \right) \text{ for } \forall j \text{ s.t. } \tilde{\beta}_j \neq 0. \end{aligned}$$

*In addition, there exists a zero solution  $\hat{\beta} = 0$  if and only if*

$$\begin{aligned} \left| \frac{1}{n} \mathbf{X}_j^\top r(0) \right| &\leq \lambda \text{ for } \forall j \text{ s.t. } \tilde{\beta}_j = 0, \text{ and} \\ -\lambda \left( \alpha + (1 - \alpha) \text{sgn}(\tilde{\beta}_j) \right) &\leq \frac{1}{n} \mathbf{X}_j^\top r(0) \leq \lambda \left( \alpha - (1 - \alpha) \text{sgn}(\tilde{\beta}_j) \right) \text{ for } \forall j \text{ s.t. } \tilde{\beta}_j \neq 0. \end{aligned}$$

This theorem shows that the estimate remains unchanged if and only if correlations between residuals and features are small and  $\lambda$  is large. This is useful for constructing a search space for  $\lambda$ . We determine a sequence of  $\lambda$  from  $\lambda_{\max}$  to  $\lambda_{\min}$ , where  $\lambda_{\max}$  is the smallest value for which all coefficients are zero or initial estimates by Theorem 4, and  $\lambda_{\min}$  is defined by a user specified fraction of  $\lambda_{\min}/\lambda_{\max}$ .

### 3.3 Transfer Lasso as a Two-Stage Estimation

The initial estimate  $\tilde{\beta}$  is arbitrary. We investigate the behavior of Transfer Lasso as a two-stage estimation. We suppose that the initial estimate is a Lasso solution using another dataset  $\mathbf{X}' \in \mathbb{R}^{m \times p}$ ,  $\mathbf{y}' \in \mathbb{R}^m$ , and the true parameter  $\tilde{\beta}^*$ . Define  $S' := \text{supp}(\tilde{\beta}^*)$ ,  $s' := |S'|$ , and  $\Delta^* := \tilde{\beta}^* - \beta^*$ . Then, we have the following corollary.

**Corollary 5.** *Suppose that Assumption 1 is satisfied and the generalized restricted eigenvalue condition (Definition 1) holds with  $\phi' = \phi'(\mathcal{B}')$  and  $\mathcal{B}' = \mathcal{B}'(1, c', 0)$  on  $\mathbf{X}'$ ,  $\mathbf{y}'$ , and  $\tilde{\beta}^*$ . Assume the same conditions as in Theorem 1. Then, with probability at least  $1 - \nu_{n,c} - \nu_{m,c'}$ , we have*

$$\|\hat{\beta} - \beta^*\|_2^2 \leq \frac{(\alpha + c)^2 \lambda_n^2 s}{\phi^2} \left( 1 + \sqrt{1 + \frac{4(1 - \alpha)(1 + c')\phi \lambda_m s'}{(\alpha + c)^2 \phi' \lambda_n s} + \frac{2(1 - \alpha)\phi \|\Delta^*\|_1}{(\alpha + c)^2 \lambda_n s}} \right)^2.$$

If there are abundant source data but few target data ( $m \gg n$  and  $\lambda_m \ll \lambda_n$ ), and the same true parameters ( $\Delta^* = 0$ ), then we have  $\|\hat{\beta} - \beta^*\|_2^2 \lesssim 4(\alpha + c)^2 \lambda_n^2 s / \phi^2$ . This implies that Transfer Lasso with a small  $\alpha$  is beneficial in terms of the estimation error. Additionally, we can see a similar weak convergence rate as in Theorem 2 even when  $\|\Delta^*\|_1 \rightarrow 0$ .

## 4 Empirical Results

We first present two numerical simulations in concept drift and transfer learning scenarios. We then show real-data analysis results.

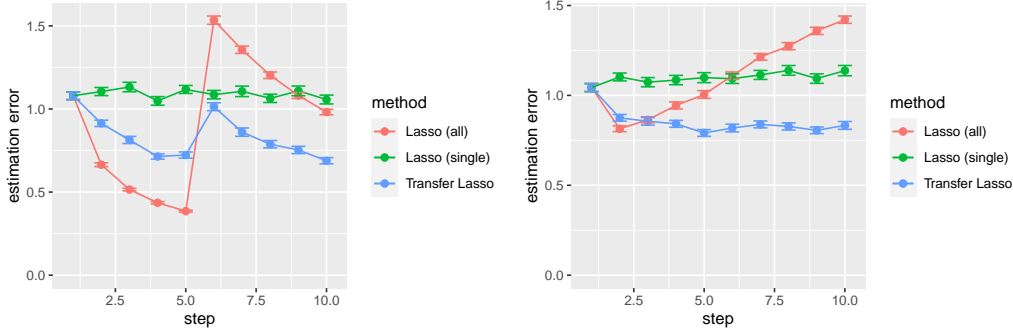


Figure 3: Estimation errors under the scenarios I (left; abrupt concept drift) and II (right; gradual concept drift).

#### 4.1 Concept Drift Simulation

We first simulated concept drift scenarios. We used ten datasets  $\{(\mathbf{X}^{(k)}; \mathbf{y}^{(k)})\}_{k=1}^{10}$ , and each dataset is generated by a linear regression  $\mathbf{y}^{(k)} = \mathbf{X}^{(k)}\beta^{(k)} + \varepsilon^{(k)}$ , where  $\mathbf{y}^{(k)} \in \mathbb{R}^n$ ,  $\mathbf{X}^{(k)} \in \mathbb{R}^{n \times p}$ ,  $\beta^{(k)} \in \mathbb{R}^p$ ,  $\varepsilon^{(k)} \in \mathbb{R}^n$ ,  $n = 50$ ,  $p = 100$ , and  $s = 10$ . Elements of  $\mathbf{X}^{(k)}$  and  $\varepsilon^{(k)}$  were randomly generated from a standard Gaussian distribution. We examined two nonstationary scenarios, abrupt concept drift and gradual concept drift. Following these scenarios, we arranged different parameter sequences  $\{\beta^{(k)}\}_{k=1}^{10}$ .

**Scenario I (Abrupt concept drift scenario).** The underlying model suddenly changes drastically. At step  $k = 1$ , ten active features are randomly selected, and their coefficients are randomly generated from a uniform distribution of  $[-1, 1]$ . The former steps ( $k = 1, \dots, 5$ ) use the same  $\beta$ . At step  $k = 6$ , five active features are abruptly switched to other features, and their coefficients are also assigned in the same way. The remaining steps ( $k = 6, \dots, 10$ ) use the same values as  $k = 6$ .

**Scenario II (Gradual concept drift scenario).** The underlying model gradually changes. The first step is the same as in Scenario I. Then, at every step, one active feature switches to another, with its coefficient assigned from a uniform distribution.

We compared three methods, including our proposed method. (i) Lasso (all): We built the  $k$ -th model by Lasso using the first through  $k$ -th datasets. (ii) Lasso (single): We built the  $k$ -th model by Lasso using only a single  $k$ -th dataset. (iii) Transfer Lasso: We sequentially built each model by Transfer Lasso. For the  $k$ -th model, we applied Transfer Lasso to the  $k$ -th dataset, along with an initial estimate using Transfer Lasso applied to the  $(k - 1)$ -th dataset. We used Lasso for the first model.

The regularization parameters  $\lambda$  and  $\alpha$  were determined by ten-fold cross validation. The parameter  $\lambda$  was selected by a decreasing sequence from  $\lambda_{\max}$  to  $\lambda_{\max} * 10^{-4}$  in log-scale, where  $\lambda_{\max}$  was calculated as in Section 3.2. The parameter  $\alpha$  was selected among  $\{0, 0.25, 0.5, 0.75, 1\}$ . Each dataset was centered and standardized such that  $\bar{\mathbf{y}} = 0$ ,  $\bar{\mathbf{X}}_j = 0$  and  $\text{sd}(\mathbf{X}_j) = 1$  in preprocessing.

Figure 3 shows the  $\ell_2$ -error for estimated parameters at each step. Averages and standard errors for the  $\ell_2$ -errors were evaluated in 100 experiments. In Scenario I, although Lasso (all) outperformed the others when the environment was stationary, it incurred significant errors after the abrupt concept drift. In contrast, Transfer Lasso gradually reduced estimation errors as the steps proceeded, and was not so worse when abrupt the concept drift occurred. Transfer Lasso always outperformed Lasso (single). In Scenario II, Transfer Lasso outperformed the others at most steps so that it balanced transferring and discarding knowledge. Lasso (all) used enough instances but induced a large estimation bias because various concepts (true models) exist in the datasets. Lasso (single) might not induce estimation bias, but incurred a lack of instances due to using only a single dataset.

#### 4.2 Transfer Learning Simulation

We simulated a transfer learning scenario in which there were abundant source data but few target data. We used  $\mathbf{y}^s = \mathbf{X}^s\beta^s + \varepsilon$  and  $\mathbf{y}^t = \mathbf{X}^t\beta^t + \varepsilon$  for a source and target domain, respectively,

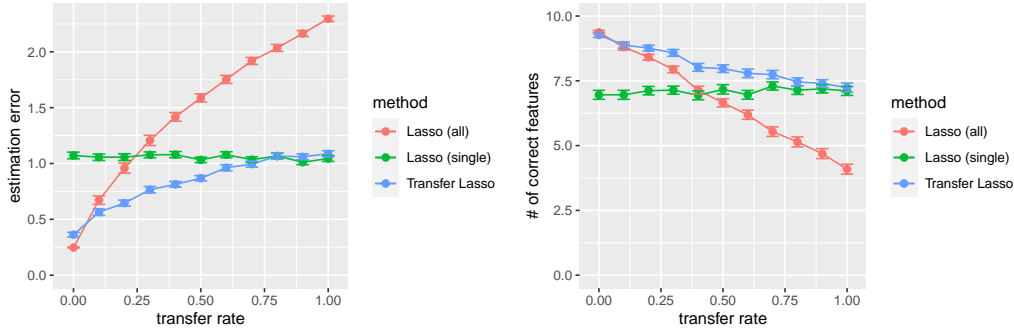


Figure 4: Estimation errors (left) and number of correct selected features (right) for transfer learning simulations.

where  $\mathbf{X}^s \in \mathbb{R}^{n_s \times p}$ ,  $\mathbf{X}^t \in \mathbb{R}^{n_t \times p}$ ,  $n_s = 500$ ,  $n_t = 50$ , and  $p = 100$ . In the source domain, we generated  $\beta^s$  in the same manner as in the concept drift simulation. For  $\beta^t$  in the target domain, we switched each active features in  $\beta^s$  to another feature at a “transfer rate” probability of 0 to 1. We compared three methods: Lasso (all), Lasso (single), and Transfer Lasso. Regularization parameters were determined in the same manner as above.

Figure 4 shows the results of the transfer learning simulations. Averages and standard errors for the  $\ell_2$ -errors were evaluated in 100 experiments. Transfer Lasso outperformed others in terms of  $\ell_2$ -error at almost all transfer rates, although Lasso (all) dominated when the transfer rate was zero, and Lasso (single) slightly dominated when the transfer rates were high. Transfer Lasso also showed the best accuracy in terms of feature screening.

### 4.3 Newsgroup Message Data

The newsgroup message data<sup>1</sup> comprises messages from Usenet posts on different topics. We basically followed the concept drift experiments in [18] and used preprocessed data<sup>2</sup>. The problem is to predict either the user is interested in email messages or not. There are 1500 examples and 913 attributes of boolean bag-of-words features. We suppose that the user is interested in the topics of space and baseball in the first 600 examples, while the user’s interest changes to the topic of medicine in the remaining 900 examples. Thus, there is a abrupt concept drift of user’s interests. The examples were divided into 30 batches without changing the order of the samples, each containing 50 examples. We trained models using each batch and evaluated them using the next batch. We compared three methods: Lasso (all), Lasso (single), and Transfer Lasso. Since this is a classification problem, we changed the squared loss function in (1) to the logistic loss. We used the coordinate descent algorithms as well. Regularization parameters were determined by ten-fold cross validation in the same manner as above except for  $\alpha = 0.501$  instead of  $\alpha = 0.5$  because of computational instability for binary features.

Figure 5 shows the results. Transfer Lasso outperformed Lasso (single) at almost all steps in terms of AUC (area under the curve). Lasso (all) performed well before concept drift (until 12-th batch), but significantly worsened after drift (from 13-th batch). Transfer Lasso showed stable behaviors of the estimates, and some coefficients remained unchanged. These results indicate that Transfer Lasso can follow data tendencies with minimal changes in the model.

## 5 Conclusion

We proposed and analyzed the  $\ell_1$  regularization-based transfer learning framework. This approach is applicable to any parametric models, including GLM, GAM, and neural networks.

<sup>1</sup><https://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups.html>

<sup>2</sup>[http://lpls.csd.auth.gr/mlkd/concept\\_drift.html](http://lpls.csd.auth.gr/mlkd/concept_drift.html)



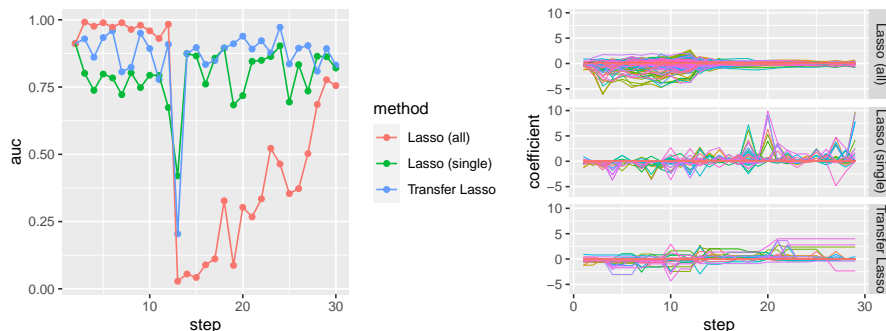


Figure 5: AUC (left) and coefficients (right) for newsgroup message data. Each coefficient is colored for legibility.

## Broader Impact

In this paper, sparsity meets transfer learning.

Sparsity has a key role in model transparency because a sparse model explains a phenomenon with few parameters. This is why Lasso is widely used in science such as genomics and economics, and in industries such as advanced electronics/semiconductors, chemicals, and health-care systems.

Our motivating examples of sparse estimation includes quality management in manufacturing. Production yield is one of the primary interests in factories and plants. Using Lasso, quality managers can screen and identify important factors for the yield from thousands or millions of candidates.

Here, we describe five types of impact of our approach along with possible applications, although our approach might have many potential impact/applications.

First, our approach enhances the efficiency of routine decision making. In manufacturing applications, quality managers need to analyze the factors behind yield fluctuations on a daily, weekly, or monthly basis. Since our approach can highlight the changes of parameters, they have only to check the difference from the past, which greatly streamlines the analysis and decision making.

Second, our approach also enhances the manageability of many models. In manufacturing applications, there are many kinds of products, so that many models are necessary. By transferring models from base products (source parameters) to the derivative products (target parameters), the total amount of active parameters is reduced, making it easier to manage a large number of models.

Third, our approach improves the model accuracy and robustness for high-dimensional small-sample data. Data scientists would easily build models from insufficient data, and furthermore, these modeling could be (semi-)automated.

Fourth, transferring knowledge among different companies is another possible application. Our approach only shares model parameters instead of data itself, hence secure and privacy-preserving transfer learning is possible.

Finally, one negative perspective could be transferring wrong knowledge, resulting in inaccurate models and hence incorrect/biased knowledge. However, we can decide whether the prior knowledge is transferred or not by controlling hyper-parameters. Additionally, we can incorporate our domain knowledge into the initial estimate. For example, a non-zero value of a certain initial estimate can be set to zero, or it can be replaced by another highly correlated feature. Therefore, we believe that this kind of concern could be overcome using appropriate domain knowledge. Such collaboration between human knowledge and real-world data is a key to model-based decision making, and it leads to a new paradigm of theory-guided data science [17] and informed machine learning [37].

## Acknowledgments

The authors received no third party funding for this work.

## References

- [1] Peter J Bickel, Yaacov Ritov, Alexandre B Tsybakov, et al. Simultaneous analysis of lasso and dantzig selector. *The Annals of Statistics*, 37(4):1705–1732, 2009.
- [2] Peter Bühlmann and Sara Van De Geer. *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media, 2011.
- [3] Lior Cohen, Gil Avrahami, Mark Last, and Abraham Kandel. Info-fuzzy algorithms for mining dynamic data streams. *Applied Soft Computing*, 8(4):1283–1294, 2008.
- [4] Lior Cohen, Gil Avrahami-Bakish, Mark Last, Abraham Kandel, and Oscar Kipersztok. Real-time data mining of non-stationary data streams from sensor networks. *Information Fusion*, 9(3):344–353, 2008.
- [5] Tamraparni Dasu, Shankar Krishnan, Suresh Venkatasubramanian, and Ke Yi. An information-theoretic approach to detecting changes in multi-dimensional data streams. In *In Proc. Symp. on the Interface of Statistics, Computing Science, and Applications*. Citeseer, 2006.
- [6] Gregory Ditzler, Manuel Roveri, Cesare Alippi, and Robi Polikar. Learning in nonstationary environments: A survey. *IEEE Computational Intelligence Magazine*, 10(4):12–25, 2015.
- [7] Pedro Domingos and Geoff Hulten. Mining high-speed data streams. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 71–80, 2000.
- [8] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(Jul):2121–2159, 2011.
- [9] John Duchi and Yoram Singer. Efficient online and batch learning using forward backward splitting. *Journal of Machine Learning Research*, 10(Dec):2899–2934, 2009.
- [10] Ryan Elwell and Robi Polikar. Incremental learning of concept drift in nonstationary environments. *IEEE Transactions on Neural Networks*, 22(10):1517–1531, 2011.
- [11] Joao Gama, Pedro Medas, Gladys Castillo, and Pedro Rodrigues. Learning with drift detection. In *Brazilian symposium on artificial intelligence*, pages 286–295. Springer, 2004.
- [12] João Gama, Indrè Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, 46(4):1–37, 2014.
- [13] Stephen Grossberg. Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural networks*, 1(1):17–61, 1988.
- [14] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical learning with sparsity: the lasso and generalization*. CRC press, 2015.
- [15] Steven CH Hoi, Doyen Sahoo, Jing Lu, and Peilin Zhao. Online learning: A comprehensive survey. *arXiv preprint arXiv:1802.02871*, 2018.
- [16] Geoff Hulten, Laurie Spencer, and Pedro Domingos. Mining time-changing data streams. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 97–106, 2001.
- [17] Anuj Karpatne, Gowtham Atluri, James H Faghmous, Michael Steinbach, Arindam Banerjee, Auroop Ganguly, Shashi Shekhar, Nagiza Samatova, and Vipin Kumar. Theory-guided data science: A new paradigm for scientific discovery from data. *IEEE Transactions on knowledge and data engineering*, 29(10):2318–2331, 2017.
- [18] Ioannis Katakis, Grigorios Tsoumakas, and Ioannis Vlahavas. Tracking recurring contexts using ensemble classifiers: an application to email filtering. *Knowledge and Information Systems*, 22(3):371–391, 2010.
- [19] Daniel Kifer, Shai Ben-David, and Johannes Gehrke. Detecting change in data streams. In *VLDB*, volume 4, pages 180–191. Toronto, Canada, 2004.

- [20] J Zico Kolter and Marcus A Maloof. Dynamic weighted majority: An ensemble method for drifting concepts. *Journal of Machine Learning Research*, 8(Dec):2755–2790, 2007.
- [21] Ilja Kuzborskij and Francesco Orabona. Stability and hypothesis transfer learning. In *International Conference on Machine Learning*, pages 942–950, 2013.
- [22] Ilja Kuzborskij and Francesco Orabona. Fast rates by transferring from auxiliary hypotheses. *Machine Learning*, 106(2):171–195, 2017.
- [23] John Langford, Lihong Li, and Tong Zhang. Sparse online learning via truncated gradient. *Journal of Machine Learning Research*, 10(Mar):777–801, 2009.
- [24] Jing Liu, Xue Li, and Weicai Zhong. Ambiguous decision trees for mining concept-drifting data streams. *Pattern Recognition Letters*, 30(15):1347–1355, 2009.
- [25] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang. Learning under concept drift: A review. *IEEE Transactions on Knowledge and Data Engineering*, 31(12):2346–2363, 2019.
- [26] Kyosuke Nishida and Koichiro Yamauchi. Detecting concept drift using statistical testing. In *International conference on discovery science*, pages 264–269. Springer, 2007.
- [27] Francesco Orabona, Claudio Castellini, Barbara Caputo, Angelo Emanuele Fiorilla, and Giulio Sandini. Model adaptation with least-squares svm for adaptive hand prosthetics. In *2009 IEEE International Conference on Robotics and Automation*, pages 2897–2903. IEEE, 2009.
- [28] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.
- [29] Garvesh Raskutti, Martin J Wainwright, and Bin Yu. Restricted eigenvalue properties for correlated gaussian designs. *The Journal of Machine Learning Research*, 11:2241–2259, 2010.
- [30] Garvesh Raskutti, Martin J Wainwright, and Bin Yu. Minimax rates of estimation for high-dimensional linear regression over  $\ell_q$ -balls. *IEEE transactions on information theory*, 57(10):6976–6994, 2011.
- [31] W Nick Street and YongSeog Kim. A streaming ensemble algorithm (sea) for large-scale classification. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 377–382, 2001.
- [32] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.
- [33] T. Tommasi, F. Orabona, and B. Caputo. Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3081–3088, June 2010.
- [34] Tatiana Tommasi and Barbara Caputo. The more you know, the less you learn: from knowledge transfer to one-shot learning of object categories. In *BMVC*, number CONF, 2009.
- [35] Tatiana Tommasi, Francesco Orabona, Claudio Castellini, and Barbara Caputo. Improving control of dexterous hand prostheses using adaptive learning. *IEEE Transactions on Robotics*, 29(1):207–219, 2012.
- [36] Paul Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494, 2001.
- [37] Laura von Rueden, Sebastian Mayer, Katharina Beckh, Bogdan Georgiev, Sven Giesselbach, Raoul Heese, Birgit Kirsch, Julius Pfrommer, Annika Pick, Rajkumar Ramamurthy, et al. Informed machine learning—a taxonomy and survey of integrating knowledge into learning systems. *arXiv preprint arXiv:1903.12394*, 2019.
- [38] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):9, 2016.

- [39] Gerhard Widmer and Miroslav Kubat. Learning in the presence of concept drift and hidden contexts. *Machine learning*, 23(1):69–101, 1996.
- [40] Lin Xiao. Dual averaging method for regularized stochastic learning and online optimization. In *Advances in Neural Information Processing Systems 22*, pages 2116–2124. 2009.
- [41] Yibin Ye, Stefano Squartini, and Francesco Piazza. Online sequential extreme learning machine in nonstationary environments. *Neurocomputing*, 116:94–101, 2013.
- [42] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, pages 1–34, 2020.

## A Additional Theoretical Properties

We provide additional theoretical properties.

### A.1 Sign Recovery of Transfer Lasso

**Theorem 6** (Sign Recovery). *Assume that  $\mathbf{X}_S^\top \mathbf{X}_S/n = I$ . Then, we have  $\text{sgn}(\hat{\beta}) = \text{sgn}(\beta^*)$  if and only if*

$$\text{sgn}(\beta_S^* - w_S) = \text{sgn}(\beta_S^*), \quad (3)$$

$$\left| \frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_S w_S + \frac{1}{n} \mathbf{X}_{S^c}^\top \varepsilon + (1 - \alpha) \lambda \text{sgn}(\tilde{\beta}_{S^c}) \right| \leq \lambda \left( \alpha + (1 - \alpha) 1_{\{\tilde{\beta}_{S^c} = 0\}} \right), \quad (4)$$

where

$$w_j := \begin{cases} (2\alpha - 1) \lambda \text{sgn}(\beta_j^*) - \frac{1}{n} \mathbf{X}_j^\top \varepsilon, & \text{for } \beta_j^* > 0 \text{ and } \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j < \lambda(2\alpha - 1), \\ & \text{for } \beta_j^* < 0 \text{ and } \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j > -(2\alpha - 1)\lambda, \\ -\Delta_j, & \text{for } \beta_j^* > 0 \text{ and } (2\alpha - 1)\lambda \leq \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j \leq \lambda, \\ & \text{for } \beta_j^* < 0 \text{ and } -\lambda \leq \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j \leq -(2\alpha - 1)\lambda, \\ \lambda \text{sgn}(\beta_j^*) - \frac{1}{n} \mathbf{X}_j^\top \varepsilon, & \text{for } \beta_j^* > 0 \text{ and } \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j > \lambda, \\ & \text{for } \beta_j^* < 0 \text{ and } \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j < -\lambda. \end{cases}$$

**Remark 1.** *If  $\varepsilon = 0$  and  $\tilde{\beta}_{S^c} = 0$ , the condition reduces to*

$$\text{sgn}(\beta_S^* - w_S) = \text{sgn}(\beta_S^*),$$

$$\left| \frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_S w_S \right| \leq \lambda,$$

where

$$w_j := \begin{cases} 0, & \text{for } (\Delta_j \geq 0 \text{ if } \beta_j^* > 0) \text{ or } (\Delta_j \leq 0 \text{ if } \beta_j^* < 0) \\ -\Delta_j, & \text{for } (-\lambda \leq \Delta_j \leq 0 \text{ if } \beta_j^* > 0) \text{ or } (0 \leq \Delta_j \leq \lambda \text{ if } \beta_j^* < 0) \\ \lambda \text{sgn}(\beta_j^*), & \text{for } (\Delta_j \leq -\lambda \text{ if } \beta_j^* > 0) \text{ or } (\Delta_j \geq \lambda \text{ if } \beta_j^* < 0). \end{cases}$$

*For the ordinary Lasso with  $\varepsilon = 0$ , the condition reduces to*

$$\text{sgn}(\beta_S^* - \lambda \text{sgn}(\beta_S^*)) = \text{sgn}(\beta_S^*),$$

$$\left| \frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_S \text{sgn}(\beta_S^*) \right| \leq 1.$$

*Since it holds that  $|w_S| \leq \lambda$ , the condition of the Transfer Lasso is milder than that of the ordinary Lasso.*

**Theorem 7** (Sign Recovery under Sub-Gaussian Noise). *Suppose that Assumption 1 is satisfied. Suppose that the generalized restricted eigenvalue condition (Definition 1) holds for  $\mathcal{B} = \mathcal{B}(\alpha, c, \Delta)$ . Assume that  $\mathbf{X}_S^\top \mathbf{X}_S/n = I$ . Then, we have  $\text{sgn}(\hat{\beta}) = \text{sgn}(\beta^*)$  if*

$$\begin{aligned} |\Delta_S| &\leq \frac{1}{2} \lambda_n, \\ \beta_{\min}^* &> \lambda_n \max \left\{ \frac{3}{2} - 2\alpha, 2\alpha - \frac{1}{2} \right\}, \\ \tilde{\beta}_{S^c} &= 0, \\ \left\| \frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_{S^c} \right\|_\infty &\leq 1. \end{aligned}$$

*with probability at least  $1 - \exp(-n\lambda_n^2/8\sigma^2 + \log(2p))$ .*

## A.2 Sign Unchanging Condition

**Theorem 8** (Sign Unchanging Condition). *Assume that  $\mathbf{X}_S^\top \mathbf{X}_S/n = I$ . Then, we have  $\text{sgn}(\hat{\beta}) = \text{sgn}(\tilde{\beta})$  if and only if*

$$\text{sgn}\left(\tilde{\beta}_{\tilde{S}} - w_{\tilde{S}}\right) = \text{sgn}(\tilde{\beta}_{\tilde{S}}), \quad (5)$$

$$\left| \frac{1}{n} \mathbf{X}_{\tilde{S}^c}^\top \mathbf{X}_{\tilde{S}} (\Delta_{\tilde{S}} - w_{\tilde{S}}) - \frac{1}{n} \mathbf{X}_{\tilde{S}^c}^\top \varepsilon \right| \leq \lambda, \quad (6)$$

where  $\tilde{S} = \{j : \tilde{\beta}_j \neq 0\}$  and

$$w_j := \mathcal{S}\left(\lambda\alpha \text{sgn}(\tilde{\beta}_j) + \Delta_j - \frac{1}{n} \mathbf{X}_j^\top \varepsilon, \lambda(1 - \alpha)\right).$$

**Remark 2.** *If  $\varepsilon = 0$  and  $\Delta_{\tilde{S}} = 0$ , the condition reduces to*

$$|\beta_{\tilde{S}}^*| > \lambda(2\alpha - 1),$$

$$\left| \frac{1}{n} \mathbf{X}_{\tilde{S}^c}^\top \mathbf{X}_{\tilde{S}} \mathcal{S}\left(\lambda\alpha \text{sgn}(\tilde{\beta}_{\tilde{S}}), \lambda(1 - \alpha)\right) \right| \leq \lambda.$$

*This condition is always satisfied if  $\alpha \leq 1/2$ .*

**Theorem 9** (Sign Unchanging Condition under Sub-Gaussian Noise). *Suppose that Assumption 1 is satisfied. Suppose that the generalized restricted eigenvalue condition (Definition 1) holds for  $\mathcal{B} = \mathcal{B}(\alpha, c, \Delta)$ . Assume that  $\mathbf{X}_S^\top \mathbf{X}_S/n = I$ . Then, we have  $\text{sgn}(\hat{\beta}) = \text{sgn}(\tilde{\beta})$  if*

$$\begin{aligned} |\Delta_{\tilde{S}}| &\leq \frac{1}{2} \lambda_n, \\ \beta_{\min}^* &> 2\lambda_n \alpha, \\ \left\| \frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_{S^c} \right\|_\infty &\leq \frac{1}{(4\alpha - 1)_+}. \end{aligned}$$

*with probability at least  $1 - \exp(-n\lambda_n^2/8\sigma^2 + \log(2p))$ .*

**Remark 3.** *This implies that the estimated sign does not change from the initial estimate if  $\alpha$  and  $\Delta_{\tilde{S}}$  are small enough.*

## A.3 Unchanging Condition

**Corollary 10.** *There exists a unchanging solution  $\hat{\beta} = \tilde{\beta}$  if*

$$\alpha \leq \frac{1}{2} \text{ and } \max_j \left| \frac{1}{n} \mathbf{X}_j^\top r \right| \leq \lambda(1 - 2\alpha).$$

*There exists a zero solution  $\hat{\beta} = 0$  if*

$$\alpha \geq \frac{1}{2} \text{ and } \max_j \left| \frac{1}{n} \mathbf{X}_j^\top \mathbf{y} \right| \leq \lambda(2\alpha - 1).$$

**Remark 4.** *This is useful for constructing a search space for  $\lambda$ .*

## B Proofs

We give proofs as below.

## B.1 Proof of Theorem 1

*Proof.*

$$\begin{aligned}
& \mathcal{L}(\hat{\beta}; \tilde{\beta}) \leq \mathcal{L}(\beta^*; \tilde{\beta}) \\
& \Leftrightarrow \frac{1}{2n} \|\mathbf{X}\beta^* + \varepsilon - \mathbf{X}\hat{\beta}\|_2^2 + \lambda_n \alpha \|\hat{\beta}\|_1 + \lambda_n (1 - \alpha) \|\hat{\beta} - \tilde{\beta}\|_1 \\
& \leq \frac{1}{2n} \|\varepsilon\|_2^2 + \lambda_n \alpha \|\beta^*\|_1 + \lambda_n (1 - \alpha) \|\tilde{\beta} - \beta^*\|_1 \\
& \Leftrightarrow \frac{1}{2n} \|\mathbf{X}(\hat{\beta} - \beta^*)\|_2^2 + \lambda_n \alpha \|\hat{\beta}\|_1 + \lambda_n (1 - \alpha) \|\hat{\beta} - \tilde{\beta}\|_1 \\
& \leq \frac{1}{n} \varepsilon^\top \mathbf{X}(\hat{\beta} - \beta^*) + \lambda_n \alpha \|\beta^*\|_1 + \lambda_n (1 - \alpha) \|\tilde{\beta} - \beta^*\|_1 \\
& \Rightarrow \frac{1}{2n} \|\mathbf{X}(\hat{\beta} - \beta^*)\|_2^2 + \lambda_n \alpha \|\hat{\beta}\|_1 + \lambda_n (1 - \alpha) \|\hat{\beta} - \tilde{\beta}\|_1 \\
& \leq \left\| \frac{1}{n} \mathbf{X}^\top \varepsilon \right\|_\infty \|\hat{\beta} - \beta^*\|_1 + \lambda_n \alpha \|\beta^*\|_1 + \lambda_n (1 - \alpha) \|\tilde{\beta} - \beta^*\|_1
\end{aligned}$$

Because we assume that  $\varepsilon$  is sub-Gaussian with  $\sigma$ , we have

$$\mathbb{P} \left( \left\| \frac{1}{n} \mathbf{X}^\top \varepsilon \right\|_\infty \leq \gamma_n \right) \geq 1 - \exp \left( -\frac{n\gamma_n^2}{2\sigma^2} + \log(2p) \right), \quad \forall \gamma_n > 0.$$

By taking  $\gamma_n = c\lambda_n$ , with probability at least  $1 - \exp(-nc^2\lambda_n^2/2\sigma^2 + \log(2p))$ , we have

$$\begin{aligned}
& \frac{1}{2n} \|\mathbf{X}(\hat{\beta} - \beta^*)\|_2^2 + \lambda_n \alpha \|\hat{\beta}\|_1 + \lambda_n (1 - \alpha) \|\hat{\beta} - \tilde{\beta}\|_1 \\
& \leq c\lambda_n \|\hat{\beta} - \beta^*\|_1 + \lambda_n \alpha \|\beta^*\|_1 + \lambda_n (1 - \alpha) \|\tilde{\beta} - \beta^*\|_1 \\
& \Leftrightarrow \frac{1}{2n} \|\mathbf{X}(\hat{\beta} - \beta^*)\|_2^2 + \lambda_n \alpha \|\hat{\beta}_S\|_1 + \lambda_n \alpha \|\hat{\beta}_{S^c}\|_1 + \lambda_n (1 - \alpha) \|\hat{\beta}_S - \tilde{\beta}_S\|_1 + \lambda_n (1 - \alpha) \|\hat{\beta}_{S^c} - \tilde{\beta}_{S^c}\|_1 \\
& \leq c\lambda_n \|\hat{\beta}_S - \beta_S^*\|_1 + c\lambda_n \|\hat{\beta}_{S^c}\|_1 + \lambda_n \alpha \|\beta_S^*\|_1 + \lambda_n (1 - \alpha) \|\tilde{\beta}_S - \beta_S^*\|_1 + \lambda_n (1 - \alpha) \|\tilde{\beta}_{S^c}\|_1 \\
& \Rightarrow \frac{1}{2n} \|\mathbf{X}(\hat{\beta} - \beta^*)\|_2^2 + \lambda_n (\alpha - c) \|\hat{\beta}_{S^c}\|_1 + \lambda_n (1 - \alpha) \|\hat{\beta}_S - \tilde{\beta}_S\|_1 + \lambda_n (1 - \alpha) \|\hat{\beta}_{S^c} - \tilde{\beta}_{S^c}\|_1 \\
& \leq \lambda_n (\alpha + c) \|\hat{\beta}_S - \beta_S^*\|_1 + \lambda_n (1 - \alpha) \|\tilde{\beta}_S - \beta_S^*\|_1 + \lambda_n (1 - \alpha) \|\tilde{\beta}_{S^c}\|_1
\end{aligned}$$

where we used a triangular inequality  $\|\beta_S^*\|_1 \leq \|\hat{\beta}_S - \beta_S^*\|_1 + \|\hat{\beta}_S\|_1$ . This indicates that  $\hat{\beta} - \beta^* \in \mathcal{B} := \{v \in \mathbb{R}^p : (\alpha - c)\|v_{S^c}\|_1 + (1 - \alpha)\|v - \Delta\|_1 \leq (\alpha + c)\|v_S\|_1 + (1 - \alpha)\|\Delta\|_1\}$

where  $\Delta := \tilde{\beta} - \beta^*$ . On the other hand, we have

$$\frac{1}{2n} \|\mathbf{X}(\hat{\beta} - \beta^*)\|_2^2 \leq \lambda_n (\alpha + c) \|\hat{\beta}_S - \beta_S^*\|_1 + \lambda_n (1 - \alpha) \|\Delta\|_1.$$

From the GRE condition, we have

$$\frac{1}{2n} \|\mathbf{X}(\hat{\beta} - \beta^*)\|_2^2 = \frac{1}{2} (\hat{\beta} - \beta^*)^\top \left( \frac{1}{n} \mathbf{X}^\top \mathbf{X} \right) (\hat{\beta} - \beta^*) \geq \frac{\phi}{2} \|\hat{\beta} - \beta^*\|_2^2.$$

Since

$$\|\hat{\beta}_S - \beta_S^*\|_1 \leq \sqrt{s} \|\hat{\beta}_S - \beta_S^*\|_2 \leq \sqrt{s} \|\hat{\beta} - \beta^*\|_2,$$

we have

$$\begin{aligned}
& \frac{\phi}{2} \|\hat{\beta} - \beta^*\|_2^2 \leq (\alpha + c) \lambda_n \sqrt{s} \|\hat{\beta} - \beta^*\|_2 + \lambda_n (1 - \alpha) \|\Delta\|_1 \\
& \Rightarrow \|\hat{\beta} - \beta^*\|_2 \leq \frac{(\alpha + c) \lambda_n \sqrt{s} + \sqrt{(\alpha + c)^2 \lambda_n^2 s + 2\phi \lambda_n (1 - \alpha) \|\Delta\|_1}}{\phi} \\
& \Rightarrow \|\hat{\beta} - \beta^*\|_2^2 \leq \frac{\left( (\alpha + c) \lambda_n \sqrt{s} + \sqrt{(\alpha + c)^2 \lambda_n^2 s + 2\phi \lambda_n (1 - \alpha) \|\Delta\|_1} \right)^2}{\phi^2}
\end{aligned}$$

□

## B.2 Proof of Theorem 2

*Proof.* From Theorem 1, we have

$$\begin{aligned}
\|\hat{\beta} - \beta^*\|_2^2 &\leq \frac{(\alpha + c)^2 \lambda_n^2 s}{\phi^2} \left( 1 + \sqrt{1 + \frac{2(1-\alpha)\phi\|\Delta\|_1}{(\alpha + c)^2 \lambda_n s}} \right)^2 \\
&\leq \frac{(\alpha + c)^2 \lambda_n^2 s}{\phi^2} \left( 2\sqrt{1 + \frac{2(1-\alpha)\phi\|\Delta\|_1}{(\alpha + c)^2 \lambda_n s}} \right)^2 \\
&= \frac{4(\alpha + c)^2 \lambda_n^2 s}{\phi^2} + \frac{8(1-\alpha)\lambda_n\|\Delta\|_1}{\phi} \\
&= O\left((\alpha + c)^2 \lambda_n^2 s + (1-\alpha)\lambda_n\|\Delta\|_1\right).
\end{aligned}$$

□

## B.3 Proof of Theorem 3

Combining Theorem 1 and the beta-min condition conclude the assertion.

## B.4 Proof of Theorem 4 and 10

*Proof.* By KKT condition, there exists a null solution  $\hat{\beta} = 0$  if and only if

$$\left| \frac{1}{n} \mathbf{X}_j^\top \mathbf{y} \right| \leq \lambda \text{ for } \forall j \text{ s.t. } \tilde{\beta}_j = 0,$$

and

$$\left| \frac{1}{n} \mathbf{X}_j^\top \mathbf{y} + \lambda(1-\alpha) \operatorname{sgn}(\tilde{\beta}_j) \right| \leq \lambda\alpha \text{ for } \forall j \text{ s.t. } \tilde{\beta}_j \neq 0. \quad (7)$$

(7) is equivalent to

$$-\lambda\alpha - \lambda(1-\alpha) \operatorname{sgn}(\tilde{\beta}_j) \leq \frac{1}{n} \mathbf{X}_j^\top \mathbf{y} \leq \lambda\alpha - \lambda(1-\alpha) \operatorname{sgn}(\tilde{\beta}_j) \text{ for } \forall j \text{ s.t. } \tilde{\beta}_j \neq 0.$$

Let  $r := \mathbf{y} - \mathbf{X}\tilde{\beta}$ . By KKT condition, there exists an invariant solution  $\hat{\beta} = \tilde{\beta}$  if and only if

$$\left| \frac{1}{n} \mathbf{X}_j^\top r \right| \leq \lambda \text{ for } \forall j \text{ s.t. } \tilde{\beta}_j = 0,$$

and

$$\left| \frac{1}{n} \mathbf{X}_j^\top r - \lambda\alpha \operatorname{sgn}(\tilde{\beta}_j) \right| \leq \lambda(1-\alpha) \text{ for } \forall j \text{ s.t. } \tilde{\beta}_j \neq 0. \quad (8)$$

(8) is equivalent to

$$-\lambda(1-\alpha) + \lambda\alpha \operatorname{sgn}(\tilde{\beta}_j) \leq \frac{1}{n} \mathbf{X}_j^\top r \leq \lambda(1-\alpha) + \lambda\alpha \operatorname{sgn}(\tilde{\beta}_j) \text{ for } \forall j \text{ s.t. } \tilde{\beta}_j \neq 0.$$

□

## B.5 Proof of Corollary 5

By standard theoretical analyses (or by Theorem 1 with  $\alpha = 1$ ), we have, for some constant  $c' > 0$ ,

$$\|\tilde{\beta} - \tilde{\beta}^*\|_2^2 \leq \frac{4(1+c')^2 \lambda_m^2 s'}{\phi'^2} \text{ and } \|\tilde{\beta} - \tilde{\beta}^*\|_1 \leq \frac{2(1+c')\lambda_m s'}{\phi'},$$



where

$$\phi' = \phi'(\mathcal{B}') := \inf_{v \in \tilde{\mathcal{B}}} \frac{v^\top \frac{1}{n} \mathbf{X}'^\top \mathbf{X}' v}{\|v\|_2^2} > 0, \quad \mathcal{B}' = \mathcal{B}'(c') := \{v \in \mathbb{R}^p : (1 - c')\|v_{S^c}\|_1 \leq (1 + c')\|v_{S'}\|_1\},$$

with probability at least  $1 - \nu_{m,c'}$ . Hence, we have

$$\|\Delta\|_1 = \|\tilde{\beta} - \beta^*\|_1 \leq \|\tilde{\beta} - \tilde{\beta}^*\|_1 + \|\tilde{\beta}^* - \beta^*\|_1 \leq \frac{2(1 + c')\lambda_m \tilde{s}}{\phi'} + \|\Delta^*\|_1,$$

where we define  $\Delta^* := \tilde{\beta}^* - \beta^*$ . Combining this and Theorem 1, we obtain the corollary.

## B.6 Proof of Theorem 6

*Proof.* By KKT condition,  $\hat{\beta}$  is the estimate if and only if  $\partial_\beta \mathcal{L}(\beta; \tilde{\beta}) = 0$  where  $\partial$  denotes sub-gradient. Now, we have

$$\partial_\beta \mathcal{L}(\beta; \tilde{\beta}) = \frac{1}{n} \mathbf{X}^\top \mathbf{X}(\beta - \beta^*) - \frac{1}{n} \mathbf{X}^\top \varepsilon + \alpha \lambda \partial_\beta \|\beta\|_1 + (1 - \alpha) \lambda \partial_\beta \|\beta - \tilde{\beta}\|_1.$$

where

$$\partial_{\beta_j} \|\beta\|_1 = \begin{cases} \text{sgn}(\beta_j) & \text{if } \beta_j \neq 0, \\ [-1, 1] & \text{if } \beta_j = 0 \end{cases} \quad \text{and} \quad \partial_{\beta_j} \|\beta - \tilde{\beta}\|_1 = \begin{cases} \text{sgn}(\beta_j - \tilde{\beta}_j) & \text{if } \beta_j \neq \tilde{\beta}_j, \\ [-1, 1] & \text{if } \beta_j = \tilde{\beta}_j \end{cases}$$

Deviding  $\beta = [\beta_S, \beta_{S^c}]$ , we have

$$\frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) + \frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_{S^c} \hat{\beta}_{S^c} - \frac{1}{n} \mathbf{X}_S^\top \varepsilon + \lambda \alpha \partial_{\hat{\beta}_S} \|\hat{\beta}_S\|_1 + \lambda (1 - \alpha) \partial_{\hat{\beta}_S} \|\hat{\beta}_S - \tilde{\beta}_S\|_1 = 0$$

$$\frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) + \frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_{S^c} \hat{\beta}_{S^c} - \frac{1}{n} \mathbf{X}_{S^c}^\top \varepsilon + \lambda \alpha \partial_{\hat{\beta}_{S^c}} \|\hat{\beta}_{S^c}\|_1 + \lambda (1 - \alpha) \partial_{\hat{\beta}_{S^c}} \|\hat{\beta}_{S^c} - \tilde{\beta}_{S^c}\|_1 = 0$$

On the other hand, the sign consistency condition is

$$\text{sgn}(\hat{\beta}_S) = \text{sgn}(\beta_S^*) \quad \text{and} \quad \hat{\beta}_{S^c} = 0.$$

Hence, the estimate  $\hat{\beta}$  is sign consistent if and only if

$$\frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) - \frac{1}{n} \mathbf{X}_S^\top \varepsilon + \lambda \alpha \text{sgn}(\beta_S^*) + \lambda (1 - \alpha) \hat{w}_S = 0 \quad (9)$$

$$\frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) - \frac{1}{n} \mathbf{X}_{S^c}^\top \varepsilon + \lambda \alpha \hat{z}_{S^c} + \lambda (1 - \alpha) \hat{w}_{S^c} = 0 \quad (10)$$

$$\hat{w}_j = \begin{cases} \text{sgn}(\hat{\beta}_j - \tilde{\beta}_j) & \text{if } \hat{\beta}_j \neq \tilde{\beta}_j, \\ [-1, 1] & \text{if } \hat{\beta}_j = \tilde{\beta}_j \end{cases} \quad (11)$$

$$|\hat{z}_{S^c}| \leq 1 \quad (12)$$

$$\text{sgn}(\hat{\beta}_S) = \text{sgn}(\beta_S^*) \quad (13)$$

$$\hat{\beta}_{S^c} = 0 \quad (14)$$

Assume  $\mathbf{X}_S$  is orthogonal, i.e.,  $\mathbf{X}_S^\top \mathbf{X}_S / n = I$ . Then, from (9),

$$\hat{\beta}_S - \tilde{\beta}_S + \Delta_S - \frac{1}{n} \mathbf{X}_S^\top \varepsilon + \lambda \alpha \text{sgn}(\beta_S^*) + \lambda (1 - \alpha) \hat{w}_S = 0 \quad (15)$$

where  $\Delta_S := \hat{\beta}_S - \tilde{\beta}_S$ .

We consider the condition (15) for (i)  $\hat{\beta}_j \neq \tilde{\beta}_j$  and (ii)  $\hat{\beta}_j = \tilde{\beta}_j$ .

(i) For  $j \in S$  such that  $\hat{\beta}_j \neq \tilde{\beta}_j$ , from (15) and (11), we have

$$\begin{aligned} \hat{\beta}_j - \tilde{\beta}_j + \Delta_j - \frac{1}{n} \mathbf{X}_j^\top \varepsilon + \lambda \alpha \text{sgn}(\beta_j^*) + \lambda (1 - \alpha) \text{sgn}(\hat{\beta}_j - \tilde{\beta}_j) &= 0 \\ \Rightarrow \hat{\beta}_j &= \tilde{\beta}_j + \mathcal{S} \left( \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \lambda \alpha \text{sgn}(\beta_j^*) - \Delta_j, \lambda (1 - \alpha) \right), \end{aligned}$$

where we define the thresholding function  $\mathcal{S}(u, \gamma)$  as

$$\mathcal{S}(u, \gamma) := \text{sgn}(u)(|u| - \gamma)_+ = \begin{cases} u - \gamma & \text{if } u > 0 \text{ and } \gamma < |u|, \\ u + \gamma & \text{if } u < 0 \text{ and } \gamma < |u|, \\ 0 & \text{if } |u| \leq \gamma. \end{cases}$$

We note that  $\hat{\beta}_j \neq \tilde{\beta}_j$  requires

$$\left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \alpha \lambda \text{sgn}(\beta_j^*) - \Delta_j \right| > \lambda(1 - \alpha).$$

(ii) For  $j \in S$  such that  $\hat{\beta}_j = \tilde{\beta}_j$ , from (15), we have

$$\hat{\beta}_j - \beta_j^* + \Delta_j - \frac{1}{n} \mathbf{X}_j^\top \varepsilon + \lambda \alpha \text{sgn}(\beta_j^*) + \lambda(1 - \alpha) \hat{w}_j = 0.$$

From (11), we have

$$\left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \lambda \alpha \text{sgn}(\beta_j^*) - \Delta_j \right| \leq \lambda(1 - \alpha).$$

Combining (i) and (ii), the conditions (15) and (11) reduces to

$$\hat{\beta}_S = \tilde{\beta}_S + \mathcal{S} \left( \frac{1}{n} \mathbf{X}_S^\top \varepsilon - \lambda \alpha \text{sgn}(\beta_S^*) - \Delta_S, \lambda(1 - \alpha) \right). \quad (16)$$

Next, we consider the condition (10) for (i)  $\hat{\beta}_j \neq \tilde{\beta}_j$  and (ii)  $\hat{\beta}_j = \tilde{\beta}_j$ .

(i) For  $j \in S^c$  such that  $\hat{\beta}_j \neq \tilde{\beta}_j$ , from (10) and (14), we have

$$\frac{1}{n} \mathbf{X}_j^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) - \frac{1}{n} \mathbf{X}_j^\top \varepsilon + \lambda \alpha \hat{z}_j - \lambda(1 - \alpha) \text{sgn}(\tilde{\beta}_j) = 0$$

By (12), we have

$$\left| \frac{1}{n} \mathbf{X}_j^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) - \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \lambda(1 - \alpha) \text{sgn}(\tilde{\beta}_j) \right| \leq \lambda \alpha.$$

We note that  $\hat{\beta}_j \neq \tilde{\beta}_j$  requires  $\tilde{\beta}_j \neq 0$ .

(ii) For  $j \in S^c$  such that  $\hat{\beta}_j = \tilde{\beta}_j$ , from (10), (12), and (14), we have

$$\left| \frac{1}{n} \mathbf{X}_j^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) - \frac{1}{n} \mathbf{X}_j^\top \varepsilon \right| \leq \lambda.$$

We note that  $\hat{\beta}_j = \tilde{\beta}_j$  requires  $\tilde{\beta}_j = 0$ .

Combining (i) and (ii), the conditions (10), (12), and (14) reduces to

$$\left| \frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) - \frac{1}{n} \mathbf{X}_{S^c}^\top \varepsilon - (1 - \alpha) \lambda \text{sgn}(\tilde{\beta}_{S^c}) \right| \leq \lambda \left( \alpha + (1 - \alpha) \mathbf{1}\{\tilde{\beta}_{S^c} = 0\} \right). \quad (17)$$

Hence, (13), (16), and (17) concludes

$$\text{sgn}(\beta_S^* - w_S) = \text{sgn}(\beta_S^*),$$

$$\left| \frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_S w_S + \frac{1}{n} \mathbf{X}_{S^c}^\top \varepsilon + \lambda(1 - \alpha) \text{sgn}(\tilde{\beta}_{S^c}) \right| \leq \lambda \left( \alpha + (1 - \alpha) \mathbf{1}\{\tilde{\beta}_{S^c} = 0\} \right),$$

where for  $j \in S$ ,

$$w_j := -\Delta_j + \mathcal{S} \left( \lambda \alpha \text{sgn}(\beta_j^*) + \Delta_j - \frac{1}{n} \mathbf{X}_j^\top \varepsilon, \lambda(1 - \alpha) \right) = \begin{cases} \lambda(2\alpha - 1) \text{sgn}(\beta_j^*) - \frac{1}{n} \mathbf{X}_j^\top \varepsilon, & \text{for } \beta_j^* > 0 \text{ and } \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j < \lambda(2\alpha - 1), \\ \text{for } \beta_j^* < 0 \text{ and } \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j > -\lambda(2\alpha - 1), \\ -\Delta_j, & \text{for } \beta_j^* > 0 \text{ and } \lambda(2\alpha - 1) \leq \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j \leq \lambda, \\ \text{for } \beta_j^* < 0 \text{ and } -\lambda \leq \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j \leq -\lambda(2\alpha - 1) - \varepsilon, \\ \lambda \text{sgn}(\beta_j^*) - \frac{1}{n} \mathbf{X}_j^\top \varepsilon, & \text{for } \beta_j^* > 0 \text{ and } \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j > \lambda, \\ \text{for } \beta_j^* < 0 \text{ and } \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j < -\lambda. \end{cases}$$

□

## B.7 Proof of Theorem 7

*Proof.* We derive a sufficient condition for (3) and (4).

It is sufficient for (3) that

$$\beta_{\min}^* := \min_{j \in S} |\beta_j^*| > \max_{j \in S} |w_j|.$$

Assume  $|\Delta_j| \leq c\lambda_n$ ,  $0 \leq c \leq 1$ . Because we assume that  $\varepsilon$  is sub-Gaussian with  $\sigma$ , we have

$$\mathbb{P} \left( \left\| \frac{1}{n} \mathbf{X}_S^\top \varepsilon \right\|_\infty \leq t \right) \geq 1 - \exp \left( -\frac{nt^2}{2\sigma^2} + \log(2s) \right).$$

Taking  $t = (1-c)\lambda_n$ , we have

$$\max_{j \in S} \left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon \right| \leq (1-c)\lambda_n,$$

with probability at least  $1 - \exp(-n(1-c)^2\lambda_n^2/2\sigma^2 + \log(2s))$ . Now, we have

$$\frac{1}{n} \mathbf{X}_j^\top \varepsilon - \Delta_j \leq \left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon \right| + |\Delta_j| \leq (1-c)\lambda_n + c\lambda_n = \lambda_n,$$

and thus

$$\begin{aligned} \max_{j \in S} |w_j| &\leq \max_{j \in S} \left\{ |2\alpha - 1|\lambda + \left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon \right|, |\Delta_j| \right\} \\ &\leq \max_{j \in S} \{ |2\alpha - 1|\lambda + (1-c)\lambda_n, c\lambda_n \} \\ &= \lambda_n \max \{ 2 - 2\alpha - c, 2\alpha - c, c \} \end{aligned}$$

Hence,  $|\Delta_j| \leq c\lambda_n$ ,  $0 \leq c \leq 1$ , and  $\beta_{\min}^* > \lambda_n \max\{2 - 2\alpha - c, 2\alpha - c, c\}$  imply the condition (3).

On the other hand, it is sufficient for (4) that for  $\forall j \in S^c$ ,

$$\left\| \frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_j \right\|_\infty |w_S| + \left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon \right| + \lambda_n(1-\alpha) \operatorname{sgn}(\tilde{\beta}_j) \leq \lambda_n \left( \alpha + (1-\alpha)1\{\tilde{\beta}_j = 0\} \right).$$

Since  $\varepsilon$  is sub-Gaussian, we have

$$\max_{j \in S^c} \left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon \right| \leq (1-c)\lambda_n,$$

with probability at least  $1 - \exp(-n(1-c)^2\lambda_n^2/2\sigma^2 + \log(2(p-s)))$ , and

$$\max_{j \in S} |w_j| \leq \lambda_n \max \{ 2 - 2\alpha - c, 2\alpha - c, c \}.$$

Hence, the condition (4) requires

$$\left\| \frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_j \right\|_\infty \max \{ 2 - 2\alpha - c, 2\alpha - c, c \} + (1-c) + (1-\alpha) \operatorname{sgn}(\tilde{\beta}_j) \leq \alpha + (1-\alpha)1\{\tilde{\beta}_j = 0\},$$

that is,

$$\left\| \frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_j \right\|_\infty \leq \frac{c - 2(1-\alpha) \left| \operatorname{sgn}(\tilde{\beta}_j) \right|}{\max \{ 2 - 2\alpha - c, 2\alpha - c, c \}}.$$

(a) If  $c = 1/2$ , then we have a sufficient condition:

$$\begin{aligned} |\Delta_j| &\leq \frac{1}{2}\lambda_n, \\ \beta_{\min}^* &> \lambda_n \max \left\{ \frac{3}{2} - 2\alpha, 2\alpha - \frac{1}{2} \right\}, \\ \tilde{\beta}_{S^c} &= 0, \\ \left\| \frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_{S^c} \right\|_\infty &\leq 1. \end{aligned}$$

If  $\alpha = 1/2$ , then  $\beta_{\min}^* > \lambda_n/2$ . Under this condition, the solution has correct sign with probability at least  $1 - \exp(-n\lambda_n^2/8\sigma^2 + \log(2s)) - \exp(-n\lambda_n^2/8\sigma^2 + \log(2(p-s)))$ . On the other hand, if  $\alpha = 1$ , then it requires  $\beta_{\min}^* > 3\lambda_n/2$  with the same probability.

□

## B.8 Proof of Theorem 8

*Proof.* In this proof, we write  $S = \{j : \tilde{\beta}_j \neq 0\}$  instead of  $S = \{j : \beta_j^* \neq 0\}$ . The sign invariant condition is

$$\text{sgn}(\hat{\beta}_S) = \text{sgn}(\tilde{\beta}_S) \text{ and } \hat{\beta}_{S^c} = 0.$$

Hence, the estimate  $\hat{\beta}$  is sign invariant if and only if

$$\frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) - \frac{1}{n} \mathbf{X}_S^\top \varepsilon + \lambda \alpha \text{sgn}(\tilde{\beta}_S) + \lambda(1 - \alpha) \hat{w}_S = 0 \quad (18)$$

$$\frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) - \frac{1}{n} \mathbf{X}_{S^c}^\top \varepsilon + \lambda \alpha \hat{z}_{S^c} + \lambda(1 - \alpha) \hat{w}_{S^c} = 0 \quad (19)$$

$$\hat{w}_j = \begin{cases} \text{sgn}(\hat{\beta}_j - \tilde{\beta}_j) & \text{if } \hat{\beta}_j \neq \tilde{\beta}_j, \\ [-1, 1] & \text{if } \hat{\beta}_j = \tilde{\beta}_j \end{cases} \quad (20)$$

$$|\hat{z}_{S^c}| \leq 1 \quad (21)$$

$$\text{sgn}(\hat{\beta}_S) = \text{sgn}(\tilde{\beta}_S) \quad (22)$$

$$\hat{\beta}_{S^c} = \tilde{\beta}_{S^c} = 0 \quad (23)$$

Assume  $\mathbf{X}_S$  is orthogonal, i.e.,  $\mathbf{X}_S^\top \mathbf{X}_S / n = I$ . Then, from (18),

$$\hat{\beta}_S - \tilde{\beta}_S + \Delta_S - \frac{1}{n} \mathbf{X}_S^\top \varepsilon + \lambda \alpha \text{sgn}(\tilde{\beta}_S) + \lambda(1 - \alpha) \hat{w}_S = 0 \quad (24)$$

where  $\Delta_S := \hat{\beta}_S - \tilde{\beta}_S$ .

We consider the condition (24) for (i)  $\hat{\beta}_j \neq \tilde{\beta}_j$  and (ii)  $\hat{\beta}_j = \tilde{\beta}_j$ .

(i) For  $j \in S$  such that  $\hat{\beta}_j \neq \tilde{\beta}_j$ , from (24) and (20), we have

$$\begin{aligned} \hat{\beta}_j - \tilde{\beta}_j + \Delta_j - \frac{1}{n} \mathbf{X}_j^\top \varepsilon + \lambda \alpha \text{sgn}(\tilde{\beta}_j) + \lambda(1 - \alpha) \text{sgn}(\hat{\beta}_j - \tilde{\beta}_j) &= 0 \\ \Rightarrow \hat{\beta}_j &= \tilde{\beta}_j + \mathcal{S} \left( \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \lambda \alpha \text{sgn}(\tilde{\beta}_j) - \Delta_j, \lambda(1 - \alpha) \right). \end{aligned}$$

We note that  $\hat{\beta}_j \neq \tilde{\beta}_j$  requires

$$\left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \lambda \alpha \text{sgn}(\tilde{\beta}_j) - \Delta_j \right| > \lambda(1 - \alpha).$$

(ii) For  $j \in S$  such that  $\hat{\beta}_j = \tilde{\beta}_j$ , from (24), we have

$$\hat{\beta}_j - \beta_j^* + \Delta_j - \frac{1}{n} \mathbf{X}_j^\top \varepsilon + \lambda \alpha \text{sgn}(\tilde{\beta}_j) + \lambda(1 - \alpha) \hat{w}_j = 0.$$

From (20), we have

$$\left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \lambda \alpha \text{sgn}(\tilde{\beta}_j) - \Delta_j \right| \leq \lambda(1 - \alpha).$$

Combining (i) and (ii), the conditions (24) and (20) reduces to

$$\hat{\beta}_S = \tilde{\beta}_S + \mathcal{S} \left( \frac{1}{n} \mathbf{X}_S^\top \varepsilon - \lambda \alpha \text{sgn}(\tilde{\beta}_S) - \Delta_S, \lambda(1 - \alpha) \right). \quad (25)$$

Next, we consider the condition (19). For  $j \in S^c$ , from (19), (21), and (23), we have

$$\left| \frac{1}{n} \mathbf{X}_j^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) - \frac{1}{n} \mathbf{X}_j^\top \varepsilon \right| \leq \lambda,$$

which is equivalent with

$$\left| \frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_S (\hat{\beta}_S - \beta_S^*) - \frac{1}{n} \mathbf{X}_{S^c}^\top \varepsilon \right| \leq \lambda. \quad (26)$$

Hence, (22), (25), and (26) concludes

$$\begin{aligned} \operatorname{sgn} \left( \tilde{\beta}_S + \mathcal{S} \left( \frac{1}{n} \mathbf{X}_S^\top \varepsilon - \lambda \alpha \operatorname{sgn}(\tilde{\beta}_S) - \Delta_S, \lambda(1 - \alpha) \right) \right) &= \operatorname{sgn}(\tilde{\beta}_S), \\ \left| \frac{1}{n} \mathbf{X}_{S^c}^\top \mathbf{X}_S \left( \Delta_S + \mathcal{S} \left( \frac{1}{n} \mathbf{X}_S^\top \varepsilon - \lambda \alpha \operatorname{sgn}(\tilde{\beta}_S) - \Delta_S, \lambda(1 - \alpha) \right) \right) - \frac{1}{n} \mathbf{X}_{S^c}^\top \varepsilon \right| &\leq \lambda. \end{aligned}$$

□

## B.9 Proof of Theorem 9

*Proof.* We derive a sufficient condition for (5) and (6).

It is sufficient for (5) that

$$\beta_{\min}^* := \min_{j \in S} |\beta_j^*| > \max_{j \in S} |w_j|.$$

Assume  $|\Delta_j| \leq c_1 \lambda_n$ . Because we assume that  $\varepsilon$  is sub-Gaussian with  $\sigma$ , we have

$$\mathbb{P} \left( \left\| \frac{1}{n} \mathbf{X}_S^\top \varepsilon \right\|_\infty \leq t \right) \geq 1 - \exp \left( -\frac{nt^2}{2\sigma^2} + \log(2s) \right).$$

Taking  $t = c_2 \lambda_n$ , we have

$$\max_{j \in S} \left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon \right| \leq c_2 \lambda_n,$$

with probability at least  $1 - \exp(-nc_2^2 \lambda_n^2 / 2\sigma^2 + \log(2s))$ . Now, we have

$$\begin{aligned} |w_j| &= \left( \lambda \alpha \operatorname{sgn}(\tilde{\beta}_j) + \Delta_j - \frac{1}{n} \mathbf{X}_j^\top \varepsilon - \lambda(1 - \alpha) \right)_+ \\ &\leq \lambda(2\alpha - 1) + |\Delta_j| + \left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon \right| \\ &\leq (2\alpha + c_1 + c_2 - 1) \lambda_n. \end{aligned}$$

Hence,  $|\Delta_j| \leq c_1 \lambda_n$ ,  $0 \leq c \leq 1$ , and  $\beta_{\min}^* > (2\alpha + c_1 + c_2 - 1) \lambda_n$  imply the condition (5).

On the other hand, it is sufficient for (6) that for  $\forall j \in S^c$ ,

$$\left\| \frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_j \right\|_\infty \|\Delta_S - w_S\|_\infty + \left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon \right| \leq \lambda_n.$$

Since  $\varepsilon$  is sub-Gaussian, we have

$$\max_{j \in S^c} \left| \frac{1}{n} \mathbf{X}_j^\top \varepsilon \right| \leq c_2 \lambda_n,$$

with probability at least  $1 - \exp(-nc_2^2 \lambda_n^2 / 2\sigma^2 + \log(2(p - s)))$ , and

$$\begin{aligned} \|\Delta_S - w_S\|_\infty &\leq \max_{j \in S^c} \left| \Delta_j - \mathcal{S} \left( \lambda \alpha \operatorname{sgn}(\tilde{\beta}_j) + \Delta_j - \frac{1}{n} \mathbf{X}_j^\top \varepsilon, \lambda(1 - \alpha) \right) \right| \\ &\leq ((2\alpha + c_2 - 1) \lambda_n)_+ \end{aligned}$$

Hence, the condition (4) requires

$$\left\| \frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_j \right\|_\infty \leq \frac{(1 - c_2)}{(2\alpha + c_2 - 1)_+}.$$

If  $c_1 = c_2 = 1/2$ , we have a sufficient condition:

$$\begin{aligned} |\Delta_S| &\leq \frac{1}{2} \lambda_n, \\ \beta_{\min}^* &> 2\lambda_n \alpha, \\ \left\| \frac{1}{n} \mathbf{X}_S^\top \mathbf{X}_{S^c} \right\|_\infty &\leq \frac{1}{(4\alpha - 1)_+}. \end{aligned}$$

□

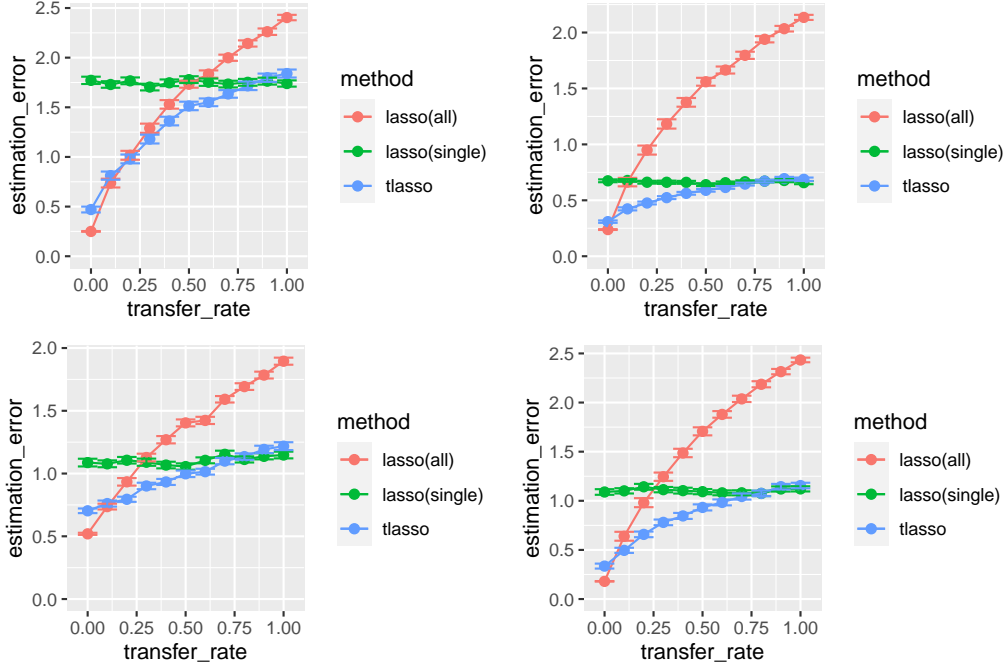


Figure 6: Estimation errors of transfer learning simulations under various source and target sample sizes  $((n_s, n_t) = (500, 20), (500, 100), (100, 50), \text{ and } (1000, 50))$  from left to right.

## C Additional Empirical Results

### C.1 Transfer Learning Simulation under various sample sizes

We assessed the influence of sample size of source and target domains on simulations in Section 4.2. Figure 6 shows that Transfer Lasso was more effective than others under various sample sizes.

### C.2 Newsgroup Message Data with Gradual Concept Drift

We present gradual concept drift experiments in this supplementary material, in addition to the abrupt concept drift experiments in our paper.

We used the newsgroup data<sup>3</sup>. In this experiment, we did not use preprocessed data<sup>4</sup>, but instead preprocessed data in a standard manner<sup>5</sup>. Specifically, we first removed email headers, signatures, nested text representing quotes from other users, and stop-words. We then extracted words that totally appear more than 100 times in the whole documents and extracted documents that include at least one above word. We obtained 11066 examples and 1370 boolean bag-of-words features with 20 news topics. The topics include "comp.graphics", "comp.os.ms-windows.misc", "comp.sys.ibm.pc.hardware", "comp.sys.mac.hardware", "comp.windows.x", "rec.autos", "rec.motorcycles", "rec.sport.baseball", "rec.sport.hockey", "sci.crypt", "sci.electronics", "sci.med", "sci.space", "misc.forsale", "talk.politics.misc", "talk.politics.guns", "talk.politics.mideast", "talk.religion.misc", "alt.atheism", and "soc.religion.christian".

The objective of this problem is to predict either the user is interested in email messages or not. We suppose that the user's interest changes gradually. We randomly splitted 20 batches. The user is interested in the  $k$ -th through  $(k + 9)$ -th topics at  $(2k - 1)$ -th and  $2k$ -th batches for  $k = 1, \dots, 10$ . Thus, the user's interest was stationary from  $(2k - 1)$  to  $2k$ -th step and was gradually drifted from  $2k$  to  $(2k + 1)$ -th step. We trained models using each batch and tested the next batch.

<sup>3</sup><https://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups.html>

<sup>4</sup>[http://lpis.csd.auth.gr/mlkd/concept\\_drift.html](http://lpis.csd.auth.gr/mlkd/concept_drift.html)

<sup>5</sup><https://www.tidytextmining.com/usenet.html>

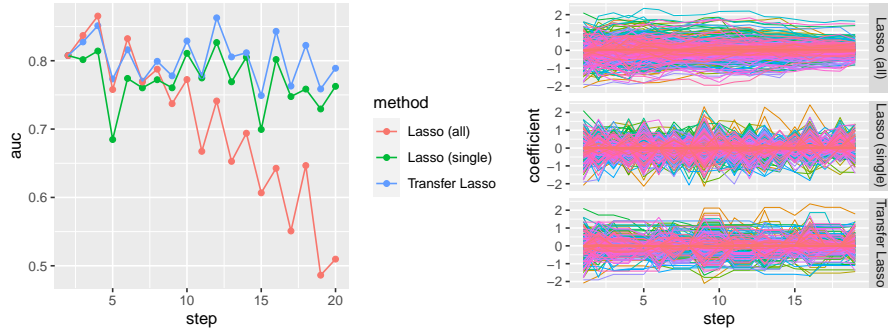


Figure 7: AUC (left) and coefficients (right) for newsgroup message data with gradual concept drift.

Figure 7 shows the results. Transfer Lasso outperformed others for almost all steps including stationary steps (even numbers) and gradual concept drift steps (odd numbers). Moreover, Transfer Lasso showed stable behaviors of the estimates, and some coefficients remained unchanged due to our regularization.