Optimized Accelerated Algorithms for Large-Scale Dynamic Low-Rank Matrix Recovery

Bin Liu*[1], Yufeng Liu[2], Zhiqiang Liu[3], and Liwen Zhang[3]

[1]Department of Statistics and Data Science, School of Management at Fudan University, Shanghai, China

[2]Department of Statistics, University of Michigan, U.S.A

[3]School of Statistics and Data Science, Shanghai University of Finance and Economics, China

Abstract

Dynamic low-rank matrix recovery is essential for uncovering evolving low-dimensional structures from high-dimensional data, but existing approaches often incur prohibitive computational costs, limiting their scalability in practice. In this paper, we study the trace regression model with single and multiple change-points under low-rank structures, and propose accelerated algorithms that integrate parameter estimation with adaptive change-point search. For the single change-point setting, we develop LR-PGA algorithm, which alternates between coefficient estimation and change-point updates, and establish theoretical guarantees under both exact and near low-rank settings. We further propose LR-POA algorithm, which incorporates an optimistic search strategy with a novel gain function, achieving lower computational complexity while providing guarantees for the exact low-rank case. For the multiple change-points setting, we propose LR-POBS algorithm, an extension of our framework based

*Corresponding author: liubin0145@gmail.com

on seeded binary segmentation, and prove consistency for both the estimated number and locations of change-points. Simulation studies demonstrate that our methods achieve up to two orders of magnitude speedup over the traditional two-step method without loss of accuracy, and empirical applications to New York City taxi trip data highlight their effectiveness in capturing dynamic structural changes in real-world systems.

Keywords: Change-point detection, Low-rank recovery, Trace regression, Accelerated algorithm

1 Introduction

In real world, there exists low-rank structure in many high dimensional data, which represents the main natural information of the data. Because of the incomplete or noisy observed data, the traditional model can hardly capture its features. low-rank matrix recovery is a strong tool for recovering the original low-rank structure with incomplete observed data and has applications in various industries.Recent years, there are various researches on the low-rank matrix recovery, such as matrix completion (Zhu et al., 2022), robust principle component analysis (Kao and Van Roy, 2014), compressed sensing (Golbabaee and Vandergheynst, 2012), low-rank tensor recovery (Cai et al., 2022) and so on. Various methods have been proposed in the literature. Convex optimization methods replace the rank constrained function with the nuclear norm and solve the problem through convex optimization techniques (Candes and Recht, 2012). Non-convex optimization methods directly optimize the non-convex rank function or the Schatten-p norm (Mohan and Fazel, 2012). Geometric methods estimate the low-rank matrix within the rank-constrained space (Absil et al., 2008; Vandereycken, 2013). Bayesian methods assume a prior distribution on the matrix and estimate the low-rank structure through Bayesian inference (Babacan et al., 2012).

Although numerous studies have investigated low-rank matrix recovery, most of them focus on static low-rank structures. However, dynamic low-rank structures are prevalent in real-world data. For example, Bai et al. (2023) analyzed EEG data to capture task-related low-rank structures and studied macroeconomic data to identify structural changes based on 19 key macroeconomic variables. Shi et al. (2024) examined particulate matter (PM) data to detect structural changes in the composition of chemical species within PM. In this paper, we analyze traffic flow data to capture variations in travel demand structures across different time periods within a day, and video data to capture structural changes induced by motion. Figure 1 illustrates the dynamic low-rank structures embedded in the data. The presence of such latent dynamic low-rank structures poses a fundamental challenge to traditional low-rank matrix recovery methods: it is necessary not only to recover the low-rank components but also to estimate the time points at which these structures change. Moreover, the high dimensionality of large-scale data introduces significant computational challenges for developing efficient solution methods.

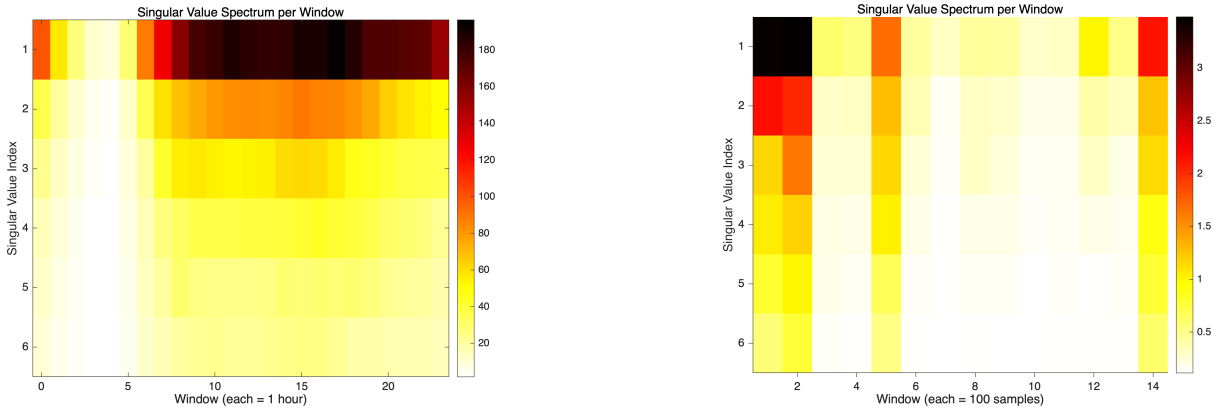

Figure 1: Singular value spectra from trace regression. ***Left panel:*** *Hourly New York City taxi origin-destination (OD) data (March 2025).* ***Right panel:*** *Consecutive-frame differences from the video dataset of Wang et al. (2014). In both cases, the top six singular values are shown.*

To describe the data with latent dynamic low-rank structure, we study the trace regression model in this paper. Suppose we have samples $\{(Y_i, \boldsymbol{X}_i, t_i)\}_{i=1}^N$, where $\boldsymbol{X}_i \in \mathbb{R}^{d_1 \times d_2}$, and $\boldsymbol{\Theta}^*_{(i)}$ is the regression matrix for the i-th observation. Then we consider the trace

regression model as follows:

$$Y_i = \langle \boldsymbol{X}_i, \boldsymbol{\Theta}^*_{(i)} \rangle + \epsilon_i,$$

where ϵ_i denotes the random error term. Let $\kappa^* \geq 0$ be the true number of unknown change-points, and denote their locations by the set $\{\tau^*_\kappa\}_{\kappa=0}^{\kappa^*+1}$ with $0 = \tau^*_0 < \tau^*_1 < \cdots < \tau^*_{\kappa^*} < \tau^*_{\kappa^*+1} = 1$. These change-points divide the N observations into $\kappa^* + 1$ segments, and the underlying regression matrices satisfy $\boldsymbol{\Theta}^*_{(i)} = \boldsymbol{\Theta}^*_j$, if $\tau^*_{j-1} < t_i \leq \tau^*_j$, $j = 1, \ldots, \kappa^*+1$, where $\boldsymbol{\Theta}^*_j \in \mathbb{R}^{d_1 \times d_2}$ denotes the true regression matrix in the j-th segment. In this framework, both the number of change-points κ^* and the dimensions d_1, d_2 are allowed to grow with the sample size N, reflecting the high-dimensional and dynamic nature of modern data. Our objective is to develop a comprehensive methodology for dynamic low-rank matrix recovery under the trace regression model, which simultaneously estimates the change-points locations, and the segment-specific low-rank coefficient matrices $\boldsymbol{\Theta}^*_j$.

To study dynamic low-rank matrix recovery problem, researchers combine the methods of change-point detection and propose new methods. Bai et al. (2020) assumed a vector auto-regressive (VAR) model with a regression matrix composed of a stable low-rank component and a time-varying sparse component, and proposed a three-step algorithm to solve it. Both the dynamic low-rank and sparse components are allowed in Bai et al. (2023), which proposed a two-step algorithm for VAR model with a single change-point and developed a rolling window strategy for detecting multiple change-points. Enikeeva et al. (2025) used information criteria to construct a test statistic for change-point detection in VAR model. Shi et al. (2024) studied a more general trace regression model and proposed a two-step algorithm to address the dynamic low-rank matrix recovery problem.

Although several methods have been proposed for dynamic low-rank matrix recovery, both methodological developments and theoretical guarantees remain relatively limited, and are primarily restricted to a few specific models, such as VAR (Bai et al., 2020, 2023; Enikeeva et al., 2025). Consequently, these approaches are difficult to generalize to other

problem settings. More importantly, modern large-scale data often involve not only massive sample sizes but also high-dimensional structures. Existing methods for dynamic low-rank matrix recovery generally follow a two-step procedure, estimating the change-point locations first and then fitting the corresponding coefficient matrices. In such large-scale scenarios, this approach can incur prohibitive computational costs. To illustrate the computational challenge, consider the two-step method proposed by Shi et al. (2024) under the trace regression model. As analyzed in Section 2.2, the total computational cost is given by $O(d_1 d_2 N^2/\alpha + (d_1 d_2^2 \wedge d_1^2 d_2)N/\alpha)$. When both the sample size N and the matrix dimensions d_1, d_2 are large, the computational cost becomes prohibitively high.

To overcome the challenges for computation and theory, we propose accelerated algorithms for low-rank matrix recovery with change-point and provide theoretical guarantees for estimation errors of regression matrices and change-point. Moreover, we try to propose accelerating solution method for multiple change-points scenario. The main algorithms of this paper along with their corresponding theoretical results and computational costs are summarized in Table 1. Our contributions can be summarized as follows.

In this paper, we study the dynamic low-rank matrix recovery problem under a general trace regression model with change-points. Our framework fully accommodates the presence of either single or multiple change-points, and allows the number of change-points, dimensions, and ranks to scale with the sample size N. This general formulation covers a wide range of practical scenarios, including compressed sensing, multi-task learning, and other high-dimensional applications where dynamic low-rank structures naturally arise.

For the dynamic low-rank matrix recovery problem with a single change-point, we propose the LR-PGA (**L**ow-**R**ank **P**roximal Update and **G**rid Search **A**lternating) algorithm (Algorithm 1).The key idea is to jointly update parameter estimation and change-point search by exploiting their mutual information within an iterative framework. In contrast, the conventional two-step method solves $O(N)$ independent least-squares problems with

nuclear norm penalty, one for each candidate change-point. This is highly redundant that nearby candidate points typically produce very similar solutions, while points far from the true change-point yield poor fits. Our algorithm instead leverages information from coefficient estimates to guide the selection of more informative candidate change-points, leading to more meaningful subproblems. Moreover, we observe that even single-step updates of the subproblems contain valuable information about the true parameters and the change-point. Building on this, LR-PGA couples one-step parameter updates with adaptive change-point search, reducing the per-iteration complexity from $O(d_1d_2N^2 + d_1d_2dN)$ to $O(d_1d_2N + d_1d_2d + N^2)$. However, this alternating strategy still incurs an additional $O(N^2)$ grid search cost per iteration, which becomes prohibitive for large N. To address this, we develop an optimistic search (OS) strategy tailored to the trace regression model. Specifically, we construct a gain function based on iteratively updated coefficients and replace full grid scanning with OS, yielding the LR-POA (**L**ow-**R**ank **P**roximal Update and **O**ptimistic Search **A**lternating) algorithm (Algorithm 2). This refinement further reduces the per-iteration complexity to $O(d_1d_2N + d_1d_2d + N\log N)$, while maintaining the same statistical accuracy. We establish theoretical guarantees for both algorithms, showing that the coefficient estimation error attains the same convergence rate as in the no change-point case. We further derive upper bounds for the change-point estimation error, providing the first such result for the compressed sensing (CS) case and matching existing rates for the multiresponse regression (MR) setting.

For the multiple change-point scenario, we extend our accelerated framework by integrating the seeded binary segmentation (SBS) strategy (Kovács et al., 2023), leading to the LR-POBS (**L**ow-**R**ank **P**roximal **O**ptimistic **B**inary **S**egmentation) algorithm (Algorithm 3). This method decomposes the multiple change-point detection task into a sequence of simpler single change-point problems by partitioning the grid search domain into subsets, applying the single change-point algorithm within each subset, and subsequently

selecting the final change-points from the resulting candidates. Theoretical guarantees are established for both the number and the locations of the estimated change-points, as summarized in Theorem 5.2. In addition, our simulation studies demonstrate that the proposed methods are up to two orders of magnitude faster than the two-step approach with comparable accuracy, and empirical analysis of New York City taxi data illustrates its practical effectiveness in capturing dynamic structural changes.

Table 1: Summary of algorithms

Algorithm	Complexity (per PGD iteration)	Theoretical Guarantees
Two-step	$O(d_1d_2N^2 + d_1d_2dN)$	General case: Theorem 7 in Shi et al. (2024) MR case: Theorem 11 in Shi et al. (2024)
LR-PGA	$O(d_1d_2N + d_1d_2d + N^2)$	General case: Theorem 3.3 CS case: Theorem 3.4 MR case: Theorem **??**
LR-POA	$O(d_1d_2N + d_1d_2d + N\log N)$	CS case: Theorem 4.2 MR case: Theorem **??**
LR-POBS		CS case: Theorem 5.2

The complexity corresponding to the algorithms is discussed without consideration of iteration complex. The two-step method is regarded as baseline of computation cost and its theoretical guarantees are provided by Shi et al. (2024). The CS case represents the settings of compressed sensing and MR case represents the settings of multiresponse regression.

1.1 Related works

Our work is closely related to Bybee and Atchadé (2018), who studied change-point detection in Gaussian graphical models and proposed the AMM algorithm based on alternating minimization with grid search. While AMM uses iterative updates to refine candidate change-point locations, its reliance on repeated grid search leads to high computational cost. In contrast, we study dynamic low-rank matrix recovery under the trace regression model and develop an accelerated framework that avoids exhaustive grid search while preserving statistical accuracy. We provide theoretical guarantees by establishing bias bounds for iteratively updated coefficient matrices under restricted region conditions, and further extend the proposed fast single change-point procedure to the multiple change-

point setting using the SBS framework, with consistency guarantees for both the number and locations of change-points.

Our approach is also related to the optimistic search method of Kovács et al. (2024) for Gaussian mean change models, which reduces search complexity using CUSUM-based gain functions. Since such statistics are computationally prohibitive in regression settings, we design a new gain function based on single-step proximal updates of coefficient matrices and embed it into an alternating minimization framework for efficient joint estimation. Compared with the two-step method of Shi et al. (2024), our framework allows coefficient matrices across change-points to have different low-rank structures and accommodates both CS and MR settings, leading to broader applicability and substantial improvements in computational efficiency.

1.2 Article organization and notation

The remainder of this paper is organized as follows. Section 2 introduces the low-rank trace regression model with a single change-point and reviews the traditional two-step method. In Section 3, we present the proposed LR-PGA algorithm for dynamic low-rank matrix recovery with a single change-point and establish its theoretical guarantees under both exact and near low-rank settings. Section 4 develops the LR-POA algorithm, together with its theoretical results under the exact low-rank setting. In Section 5, we extend our framework to the multiple change-point scenario and propose the LR-POBS algorithm along with its theoretical guarantees. Section 6 reports simulation studies and demonstrates the practical effectiveness of our methods through real data applications. Finally, Section 7 concludes the paper.

Notation. For any two scalars a and $b \in \mathbb{R}$, we write $a \vee b := \max(a, b)$ and $a \wedge b := \min(a, b)$, and define $(a)_+ := \max(a, 0)$. For a vector $\boldsymbol{x}$, let $\|\boldsymbol{x}\|_2$ denote its ℓ_2 norm. For a matrix $\boldsymbol{X} \in \mathbb{R}^{d_1 \times d_2}$, we use $\|\boldsymbol{X}\|_{op}$, $\|\boldsymbol{X}\|_F$, and $\|\boldsymbol{X}\|_*$ to denote its operator norm,

Frobenius norm and nuclear norm, respectively. The vectorization of a matrix $\boldsymbol{X}$ is denoted by $\text{vec}(\boldsymbol{X}) := (\boldsymbol{X}_1^\top, \boldsymbol{X}_2^\top, \ldots, \boldsymbol{X}_{d_2}^\top)^\top \in \mathbb{R}^{d_1 d_2}$, where $\boldsymbol{X}_j$ is the j-th column of $\boldsymbol{X}$. For any two matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{d_1 \times d_2}$, we define their inner product as $\langle \boldsymbol{A}, \boldsymbol{B} \rangle := \text{tr}(\boldsymbol{A}^\top \boldsymbol{B})$, where $\text{tr}(\cdot)$ denotes the trace operator. For a matrix $\boldsymbol{X}$, we let $\rho_l(\boldsymbol{X})$ denote its l-th largest singular value, and $\lambda_{\max}(\boldsymbol{X})$ and $\lambda_{\min}(\boldsymbol{X})$ denote its largest and smallest eigenvalues, respectively. For a differentiable function L, we use ∇L to denote its gradient. For two dimensions d_1 and d_2, we define $d := \max(d_1, d_2)$. For a finite set $\mathcal{T}$, we use $|\mathcal{T}|$ to denote its cardinality. For a random vector $\boldsymbol{x} \in \mathbb{R}^d$, we define its sub-Gaussian norm as $\|\boldsymbol{x}\|_{\psi_2} := \sup_{\boldsymbol{u} \in \mathcal{S}^{d-1}} \|\boldsymbol{u}^\top \boldsymbol{x}\|_{\psi_2}$, where $\mathcal{S}^{d-1}$ denotes the unit sphere in $\mathbb{R}^d$. For a random matrix $\boldsymbol{X} \in \mathbb{R}^{d_1 \times d_2}$, its sub-Gaussian norm is defined as $\|\boldsymbol{X}\|_{\psi_2} := \sup_{\boldsymbol{u} \in \mathcal{S}^{d_1-1}} \sup_{\boldsymbol{v} \in \mathcal{S}^{d_2-1}} \|\boldsymbol{u}^\top \boldsymbol{X} \boldsymbol{v}\|_{\psi_2}$. In this paper, constants with numerical subscripts, such as C_1, C_2 and c_1, c_2, represent generic positive constants whose values may differ across different theoretical results. The notation N denotes the sample size under the standard trace regression model, while n refers to the sample size in the MR model.

2 Model setup and methodology

In this section, we introduce the trace regression model with a single change-point and the traditional two-step method for estimating the change-point location and coefficient matrices. The model settings are presented in Section 2.1 and the traditional two-step method is introduced in Section 2.2.

2.1 Model setup

Firstly, we consider the trace regression model with a single change-point. Suppose that we have N independent samples $\{(Y_i, \boldsymbol{X}_i, t_i)\}_{i=1}^N$, where $Y_i \in \mathbb{R}$ is a response variable, $\boldsymbol{X}_i \in \mathbb{R}^{d_1 \times d_2}$ is a matrix of covariates and $t_i = i/N$ represents a threshold variable. The

trace regression model with a single change-point is given by

$$Y_i = \langle \boldsymbol{X}_i, \boldsymbol{\Theta}_1^* \rangle \mathbf{1}_{\{t_i \le \tau^*\}} + \langle \boldsymbol{X}_i, \boldsymbol{\Theta}_2^* \rangle \mathbf{1}_{\{t_i > \tau^*\}} + \epsilon_i = [\mathfrak{X}(\boldsymbol{\Gamma}^*; \tau)]_i + \epsilon_i, \tag{2.1}$$

where $\boldsymbol{\Theta}_1^*, \boldsymbol{\Theta}_2^* \in \mathbb{R}^{d_1 \times d_2}$ are low-rank matrices to be estimated before and after the change-point τ^*, and $\epsilon_i \in \mathbb{R}$ denotes independent noise. The transformed formulation is derived by defining $\boldsymbol{\Gamma}^* = [\boldsymbol{\Theta}_1^*, \boldsymbol{\Theta}_2^*] \in \mathbb{R}^{2d_1 \times d_2}$, $\boldsymbol{\mathcal{X}}_i(\tau) = \left(\boldsymbol{X}_i^\top \mathbf{1}_{\{t_i \le \tau\}}, \boldsymbol{X}_i^\top \mathbf{1}_{\{t_i > \tau\}}\right)^\top \in \mathbb{R}^{2d_1 \times d_2}$, and the linear operator $\mathfrak{X}(\boldsymbol{\Gamma}; \tau) \in \mathbb{R}^N$ whose i-th entry is given by $[\mathfrak{X}(\boldsymbol{\Gamma}; \tau)]_i = \langle \boldsymbol{\mathcal{X}}_i(\tau), \boldsymbol{\Gamma} \rangle$.

In this paper, we focus on the trace regression model because of its generality. While our main interest lies in the setting with change-points, we first examine the case without change-points to illustrate the model's flexibility, in which several important special cases can be recovered. Figure 2 depicts these transformations. The primary objective of trace regression is to estimate the latent low-rank structure $\boldsymbol{\Theta}^*$. Having illustrated the generality

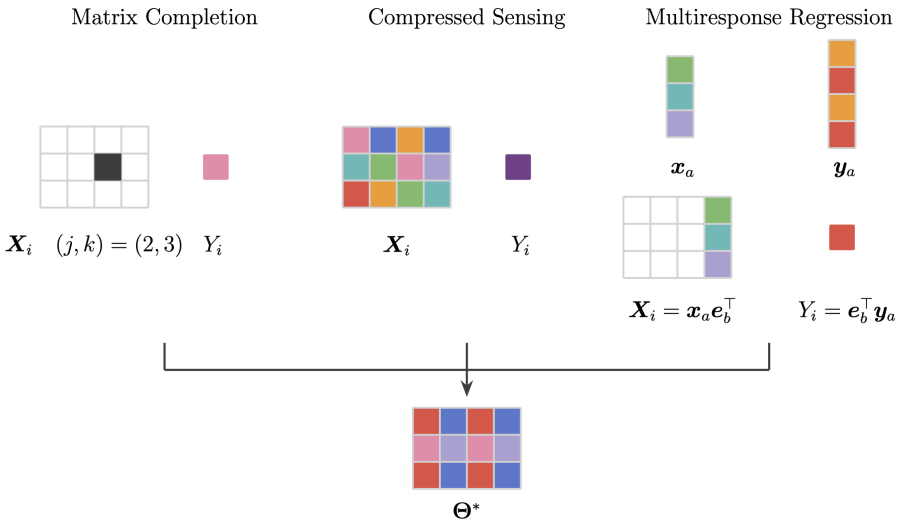

Figure 2: Special examples for trace regression model. *The boxes with different colors indicate distinct values.* ***Matrix completion:*** *each* $\boldsymbol{X}_i$ *is a matrix with a single entry equal to* 1 *at position* (j, k)*, and all other entries equal to* 0*.* ***Compressed sensing:*** *each* $\boldsymbol{X}_i$ *represents a sensing matrix that encodes richer structural information.* ***Multiresponse regression:*** *the first layer corresponds to the original MR model, while the second layer shows its trace regression representation. The mapping between the two layers is given by* $(a, b) \mapsto i = (a-1)d_2 + b$*. In the illustration above, the canonical basis vector* $\boldsymbol{e}_b$ *is specified with* $b = 4$*.*

of the trace regression model through its special cases, we now turn to an important structural assumption imposed on the coefficient matrices. In many applications, the coefficient matrices are believed to exhibit an underlying low-rank structure. However, requiring an exact low-rank form is often too restrictive for real-world data, where the rank of each true coefficient matrix $\mathbf{\Theta}_j^*$, $j = 1, 2$, is typically bounded by a small integer. To relax this restriction, we adopt a more flexible *near low-rank* framework. Specifically, we assume that each $\mathbf{\Theta}_j^*$ belongs to the set $\mathcal{B}(R_{j,q}) = \{\mathbf{\Theta} \in \mathbb{R}^{d_1 \times d_2} | \sum_{l=1}^{d} [\rho_l(\mathbf{\Theta})]^q \leq R_{j,q}\}$, where $0 \leq q \leq 1$ is a relaxation parameter and the radius $R_{j,q} > 0$ controls the ball. When $q = 0$, this condition reduces to the standard low-rank constraint $\text{rank}(\mathbf{\Theta}_j^*) \leq R_{j,0}$. More generally, the constraint $\mathcal{B}(R_{j,q})$ enforces a sufficiently fast decay of the singular values of $\mathbf{\Theta}_j^*$, ensuring that the matrix can be well approximated by a low-rank component while still allowing small residual singular values. Moreover, this formulation naturally accommodates the possibility that the coefficient matrices before and after the change-point possess different low-rank structures.

2.2 Method

To recover the dynamic low-rank matrices under the above model setting, Shi et al. (2024) proposed a two-step method and demonstrated its effectiveness. The key idea of this method is to first fix the change-point and estimate the corresponding low-rank matrices, and then determine the optimal location of the change-point via a full grid search. In this paper, we adopt this approach as a baseline to compare computational efficiency. For clarity, we briefly describe the two-step procedure. We divide the samples into two subsets and denote by $\mathcal{I}_{1,\tau}$ ($\mathcal{I}_{2,\tau}$) the indices of the samples before (after) the change-point τ, with corresponding sample sizes $N_{1,\tau}$ and $N_{2,\tau}$. In the first step, the coefficient matrices are

estimated by solving the nuclear-norm penalized least squares problem

$$\widehat{\boldsymbol{\Theta}}_{j,\tau} = \underset{\boldsymbol{\Theta}_j}{\operatorname{argmin}} \sum_{i \in \mathcal{I}_{j,\tau}} \left(Y_i - \langle \boldsymbol{X}_i, \boldsymbol{\Theta}_j \rangle\right)^2 + \lambda_{j,\tau} \left\|\boldsymbol{\Theta}_j\right\|_*, \tag{2.2}$$

where $j = 1, 2$ indicates the subset before and after the change-point τ, and $\lambda_{j,\tau}$ is the regularization parameter corresponding to the location τ and part index j. We simply denote $L_{j,\tau}(\boldsymbol{\Theta}) = \sum_{i \in \mathcal{I}_{j,\tau}} \left(Y_i - \langle \boldsymbol{X}_i, \boldsymbol{\Theta} \rangle\right)^2$, $\quad \phi_{j,\tau}(\boldsymbol{\Theta}) = L_{j,\tau}(\boldsymbol{\Theta}) + \lambda_{j,\tau} \|\boldsymbol{\Theta}\|_*$, and then perform the grid search in the second step as

$$\widehat{\tau} = \underset{\tau \in \mathcal{T}}{\operatorname{argmin}} \left[\phi_{1,\tau}\left(\widehat{\boldsymbol{\Theta}}_{1,\tau}\right) + \phi_{2,\tau}\left(\widehat{\boldsymbol{\Theta}}_{2,\tau}\right)\right], \tag{2.3}$$

where $\mathcal{T} \subset [0, 1]$ is a finite search domain for the change-point.

To solve problem (2.2), we adopt the proximal gradient descent method to handle the non-smooth nuclear norm penalty. Each iteration consists of a gradient step on the smooth squared loss followed by a proximal update for the nuclear norm (Parikh and Boyd, 2014), which shrinks the iterate toward a low-rank matrix in terms of the Frobenius norm. Formally, fixing τ and given the coefficient matrix at the k-th iteration $\boldsymbol{\Theta}_{j,\tau}^{(k)}$, the update is $\boldsymbol{\Theta}_{j,\tau}^{(k+1)} = Prox_{\gamma\lambda_{j,\tau}}\left(\boldsymbol{\Theta}_{j,\tau}^{(k)} - \gamma \nabla L_{j,\tau}(\boldsymbol{\Theta}_{j,\tau}^{(k)})\right)$, where $\gamma > 0$ is the step size for proximal gradient update. For any matrix $\boldsymbol{G} \in \mathbb{R}^{d_1 \times d_2}$ with singular value decomposition $\boldsymbol{G} = \boldsymbol{U}_{\boldsymbol{G}}^\top \operatorname{diag}\{\rho_i(\boldsymbol{G})\}\boldsymbol{V}_{\boldsymbol{G}}$, $i = 1, \ldots, d$, the proximal operator is given by

$$Prox_{\gamma\lambda_{j,\tau}}(\boldsymbol{G}) = \boldsymbol{U}_{\boldsymbol{G}}^\top \operatorname{diag}\{(\rho_i(\boldsymbol{G}) - \gamma\lambda_{j,\tau})_+\}\boldsymbol{V}_{\boldsymbol{G}}, \tag{2.4}$$

where $x_+ = \max\{x, 0\}$.

The main computational burden of PGD arises from evaluating the gradient of $L_{j,\tau}(\boldsymbol{\Theta})$ and computing the proximal operator. The complexity of $\nabla L_{1,\tau}(\boldsymbol{\Theta})$ and $\nabla L_{2,\tau}(\boldsymbol{\Theta})$ is $O(d_1 d_2 N)$, while that of the proximal operator for the nuclear norm penalty is $O(d_1 d_2 d)$.

Hence, a single PGD iteration costs $O(d_1d_2N + d_1d_2d)$. Considering the iteration count, standard PGD requires $O(1/\alpha)$ iterations to achieve an α-optimal solution (i.e., $\|\boldsymbol{\Theta} - \widehat{\boldsymbol{\Theta}}\|_F^2 \leq \alpha$) (Beck and Teboulle, 2009). Thus, solving (2.2) with a fixed τ has a total complexity of $O((d_1d_2N + d_1d_2d)/\alpha)$. To locate the best τ, each candidate in $\mathcal{T}$ must be evaluated via grid search. Since the cardinality of the candidate set is $|\mathcal{T}| = O(N)$, the overall computational complexity of the two-step method for solving (2.3) is $O\big((d_1d_2N^2 + d_1d_2dN)/\alpha\big)$.

In effect, the traditional two-step method amounts to solving $O(N)$ nuclear-norm penalized least squares problems, one for each candidate change-point. This leads to substantial redundancy that nearby candidates often produce highly similar solutions, while candidates far from the true change-point rarely yield satisfactory fits. Consequently, treating all subproblems as completely independent incurs excessive computational cost without proportional benefit. These observations highlight the need for more efficient strategies, and in the next section we introduce an accelerated approach that addresses these inefficiencies.

3 Low-rank proximal update and grid search alternating algorithm

Motivated by the inefficiency of the conventional two-step procedure, we propose LR-PGA for dynamic low-rank matrix recovery with a change-point. The method alternates between a single proximal update of the coefficient matrices and a grid-based update of the change-point, reusing information across candidate locations to avoid redundant estimation. The one-step proximal update serves as an informative surrogate that guides the change-point search and yields substantial computational gains. Algorithmic details and theoretical guarantees are given in Sections 3.1 and 3.2.

3.1 Method

In this section, we present the LR-PGA algorithm in detail and analyze its computational complexity. Algorithm 1 summarizes the procedure and makes the alternating structure explicit. At iteration k, given the current change-point $\tau^{(k-1)}$ and coefficient matrix $\mathbf{\Theta}_j^{(k-1)}$, LR-PGA (i) performs a single proximal gradient update to obtain the coefficient matrices $\{\mathbf{\Theta}_j^{(k)}\}$, and then (ii) updates the change-point by a grid scan over $\mathcal{T}$ using the residual information produced in step (i). This yields a mutually reinforcing estimation scheme: as the coefficient estimates improve, the change-point criterion becomes more discriminative; in turn, the updated $\tau^{(k)}$ induces a partition that better matches the underlying regimes, thereby reducing bias in the subsequent proximal update of $\mathbf{\Theta}_j^{(k+1)}$.

Operationally, the residuals computed immediately after step (i) (line 5 of Algorithm 1) serve as the bridge between the two steps. For each sample i, the residuals relative to $\mathbf{\Theta}_1^{(k)}$ and $\mathbf{\Theta}_2^{(k)}$ are defined as $res_{i,j}^{(k)} = Y_i - \langle \boldsymbol{X}_i, \mathbf{\Theta}_j^{(k)} \rangle$, where $\mathbf{\Theta}_j^{(k)}$ is the coefficient matrix updated in line 4 of Algorithm 1. Leveraging the residuals $\{res_{i,j}^{(k)}\}$, the grid search reduces to aggregating squared residuals for each candidate τ,

$$\begin{aligned}\tau^{(k)} &\in \arg\min_{\tau \in \mathcal{T}} \phi_{1,\tau}(\mathbf{\Theta}_1^{(k)}) + \phi_{2,\tau}(\mathbf{\Theta}_2^{(k)}) \\ &= \arg\min_{\tau \in \mathcal{T}} \left\{ \sum_{i \in \mathcal{I}_{1,\tau}} (res_{i,1}^{(k)})^2 + \lambda_{1,\tau} \|\mathbf{\Theta}_1^{(k)}\|_* + \sum_{i \in \mathcal{I}_{2,\tau}} (res_{i,2}^{(k)})^2 + \lambda_{2,\tau} \|\mathbf{\Theta}_2^{(k)}\|_* \right\},\end{aligned}$$

where the nuclear-norm terms are already computed through the proximal update introduced in (2.4), and their computational cost is therefore not counted again.

From a computational perspective, the proximal update of $\mathbf{\Theta}_j^{(k-1)}$ has the same per-iteration cost as PGD, namely $O(d_1 d_2 N + d_1 d_2 d)$. Given the sample-wise residuals computed during this step, updating the change-point via a grid scan over $\mathcal{T}$ requires $O(N)$ operations per candidate, leading to an overall cost of $O(N^2)$. Since the residuals are reused from the proximal update, no additional computation is required. As a result, each

LR-PGA iteration has a complexity of $O(d_1 d_2 N + d_1 d_2 d + N^2)$. While the total number of iterations is difficult to characterize due to the dependence of the loss function on τ, each iteration of Algorithm 1 is substantially faster than that of the conventional two-step method, suggesting a significant improvement in overall computational efficiency.

Algorithm 1 LR-PGA for dynamic low-rank matrix recovery with single change-point

Require: Samples $\{(Y_i, \boldsymbol{X}_i, t_i)\}_{i=1}^N$, grid search domain $\mathcal{T}$, regularization constant λ_c, step size γ, max iteration K, convergence threshold ζ.

Init: $\boldsymbol{\Theta}_1^{(0)}, \boldsymbol{\Theta}_2^{(0)} \in \mathbb{R}^{d_1 \times d_2}$, $\tau^{(0)} \in \mathcal{T}$.
function LR-PGA($\tau^{(0)}, \boldsymbol{\Theta}_1^{(0)}, \boldsymbol{\Theta}_2^{(0)}$)
  for $k = 1 : K$ **do**
    Proximal update: $\boldsymbol{\Theta}_j^{(k)} \leftarrow Prox_{\gamma \lambda_{j,\tau^{(k-1)}}}\big(\boldsymbol{\Theta}_j^{(k-1)} - \gamma \nabla L_{j,\tau^{(k-1)}}(\boldsymbol{\Theta}_j^{(k-1)})\big)$, $j = 1, 2$.
    Compute residuals: $res_{i,j}^{(k)} = Y_i - \langle \boldsymbol{X}_i, \boldsymbol{\Theta}_j^{(k)} \rangle$.
    Grid search: $\tau^{(k)} \leftarrow \arg\min_{\tau \in \mathcal{T}} \big[\phi_{1,\tau}(\boldsymbol{\Theta}_1^{(k)}) + \phi_{2,\tau}(\boldsymbol{\Theta}_2^{(k)})\big]$.
    Compute stopping criterion: $\zeta_k = \sum_{j=1}^{2} \frac{\|\boldsymbol{\Theta}_j^{(k)} - \boldsymbol{\Theta}_j^{(k-1)}\|_F^2}{\|\boldsymbol{\Theta}_j^{(k-1)}\|_F^2}$.
    if $\zeta_k < \zeta$ **then break**
    end if
  end for
  return $\check{\tau} = \tau^{(k)}$, $\check{\boldsymbol{\Theta}}_j = \boldsymbol{\Theta}_j^{(k)}$, $j = 1, 2$.
end function

Remark 3.1. Notice that the regularization constant λ_c differs from the parameter $\lambda_{j,\tau}$, which is scaled according to the sample size as $\lambda_{j,\tau} = \sqrt{N_{j,\tau}}\, \lambda_c$. Moreover, λ_c itself depends on the model dimensions and is specified as $\lambda_c = c_\lambda \sqrt{d_1 + d_2}$, with c_λ a constant, for both the CS and MR settings.

3.2 Main results

We now present the theoretical analysis of LR-PGA in the single change-point setting. We first establish general convergence results and then specialize them to the CS and MR cases. For each setting, we state the required assumptions and derive convergence guarantees for both the estimated change-point and the coefficient matrices under exact and approximate low-rank structures.

3.2.1 General case

We start from a general setting without imposing specific structure on the design matrices $\boldsymbol{X}_i$. To establish convergence, we require several mild regularity conditions, which fall into two categories: assumptions on nuisance parameters that hold with high probability under suitable choices for both CS and MR designs, and assumptions on model-specific parameters that ensure algorithmic guarantees. We first present the nuisance parameter conditions in Assumption A.

Assumption A (Assumptions on nuisance parameter)**.**

(A.1) (URSC and URSM)**.** For parameters $0 < c_l \leq c_u$ and $0 < \kappa_l \leq \kappa_u$, the URSC and URSM conditions hold over $\boldsymbol{\Omega}'$ such that for all $\boldsymbol{\Delta} \in \boldsymbol{\Omega}'$ and all $\tau \in \mathcal{T}$, $N_{j,\tau} c_l \|\boldsymbol{\Delta}\|_F^2 - \kappa_l \|\boldsymbol{\Delta}\|_* \leq \sum_{i \in \mathcal{I}_{j,\tau}} \langle X_i, \boldsymbol{\Delta} \rangle^2 \leq N_{j,\tau} c_u \|\boldsymbol{\Delta}\|_F^2 + \kappa_u \|\boldsymbol{\Delta}\|_*, j = 1, 2.$

(A.2) (Regularization level)**.** For each $\tau \in \mathcal{T}$, $\lambda_{j,\tau} \geq 2\|\nabla L_{j,\tau}(\boldsymbol{\Theta}_j^*)\|_{op}, j = 1, 2$, where $\lambda_{j,\tau} = \lambda_c \sqrt{N_{j,\tau}}$.

(A.3) (Identifiability)**.** Assume $\boldsymbol{\Theta}_j^* \in \mathcal{B}_q(R_{j,q})$ for some $q \in [0, 1)$, and $\delta_{\boldsymbol{\Theta}^*} = \boldsymbol{\Theta}_2^* - \boldsymbol{\Theta}_1^* \neq 0$. There exist the minimal detection length $\eta(N, d_1, d_2) > 0$ and the constant $c_0 \in (0, 1)$ such that for any $\tau \in \mathcal{T}$ with $|\tau - \tau^*| > \eta(N, d_1, d_2)$, $\|\mathfrak{X}(\boldsymbol{\Gamma}^*; \tau) - \mathfrak{X}(\boldsymbol{\Gamma}^*; \tau^*)\|_2^2 > c_0 c_l \|\delta_{\boldsymbol{\Theta}^*}\|_F^2 |\tau - \tau^*| N$.

As outlined above, Assumption (A.1) specifies the uniformly restricted strong convexity (URSC) and uniformly restricted smoothness (URSM) conditions. The left-hand side of the inequality corresponds to URSC, ensuring sufficient local curvature of the loss, while the right-hand side enforces URSM to control its local smoothness. In high-dimensional settings such as low-rank matrix recovery, global strong convexity is typically too stringent; instead, imposing convexity on a suitable restricted set is sufficient for theoretical analysis (Negahban et al., 2012). Following this principle and the framework of Agarwal et al. (2012), our URSC/URSM conditions incorporate tolerance terms $\kappa_l |\boldsymbol{\Delta}|$ and $\kappa_u |\boldsymbol{\Delta}|$,

which allow for mild deviations from ideal curvature and smoothness. The restricted region $\mathbf{\Omega}'$, which takes a cone-like form constructed from the leading r_j singular vectors in the SVD of $\mathbf{\Theta}_j^*$ (with $r_j \in \{1, \ldots, d\}$), will be fully specified in the proof of Theorem 3.3. Assumption (A.2) complements the above by ensuring that the error $\mathbf{\Delta}$ indeed lies within $\mathbf{\Omega}'$. This requires the regularization parameter to dominate the stochastic gradient fluctuations. Choosing $\lambda_{j,\tau}$ at the stated order guarantees that the nuclear-norm penalty controls the estimation error and keeps the iterates in the region where the URSC/URSM conditions apply. Assumption (A.3) provides the identifiability condition necessary for consistent change-point detection. It requires that once a candidate τ deviates from the true τ^* by more than $\eta(N, d_1, d_2)$, the resulting representation $\mathfrak{X}(\Gamma^*; \tau)$ must differ from the truth by an amount proportional to $\|\boldsymbol{\delta}_{\mathbf{\Theta}^*}\|_F^2 |\tau - \tau^| N$. In essence, this condition ensures that the model exhibits a detectable structural discontinuity at τ^*, so that the true change-point is uniquely identifiable. Next, the assumptions on model-specific parameters are stated as Assumption B.

Assumption B (Assumptions on model parameters)**.**

(B.1) (Sample size)**.** Assume the sample size N satisfies $\xi_{\min} N \geq \max\{288(r_1 + r_2)\kappa_l / c_l, 288(r_1 + r_2)\kappa_u / c_u\}$, where $\xi_{\min} = \min\{\tau^*, 1 - \tau^*\}$ is the minimal segment length, r_1, r_2 define the restricted region, and (c_l, κ_l), (c_u, κ_u) are the URSC/URSM constants.

(B.2) (Jump size)**.** Assume the jump size $\|\delta_{\mathbf{\Theta}^*}\|_F$ satisfies $\|\delta_{\mathbf{\Theta}^*}\|_F \geq c_\delta \max(\|\mathbf{\Theta}_1^*\|_*, \|\mathbf{\Theta}_2^*\|_*)$, where c_δ is an enough large constant.

(B.3) (Sub-Gaussian noises)**.** The noise variables $\epsilon_1, \ldots, \epsilon_N$ are independent, mean-zero, sub-Gaussian random variables with $K = \max_i \|\epsilon_i\|_{\psi_2}$.

We next offer additional clarification on the specific conditions stated above. A sufficiently large sample size is typically required for reliable change-point detection. In our setting, this requirement is formalized in Assumption (B.1). The bound ensures that the

sample size is adequate for standard low-rank matrix recovery even in the absence of a change-point, aligning with the sample complexity stated in Proposition 10.6 of Wainwright (2019). Assumption (B.2) imposes a lower bound on the jump size $\|\boldsymbol{\delta}_{\boldsymbol{\Theta}^*}\|_F$, which is essential for distinguishing the pre- and post-change structures. Assumption (B.3) specifies the distributional properties of the noise variables.

Now we define the grid search domian $\mathcal{T}$. For the grid search step, we do not scan over all samples but restrict attention to a set of candidate change-points $\mathcal{T} \subset \{1/N, 2/N, \ldots, 1\}$. Throughout the analysis we work with a search set $\mathcal{T}$ satisfying, for every $\tau \in \mathcal{T}$, $N_{j,\tau} \geq \max\Big\{288(r_1+r_2)\kappa_l/c_l,\ 288(r_1+r_2)\kappa_u/c_u\Big\}, j = 1, 2,$ and $|\tau - \tau^*|\, N \leq A_0\sqrt{N_{j,\tau}\kappa_l}$, where A_0 is a sufficiently large constant. The choice of $\mathcal{T}$ in practice is discussed in Remark 3.2.

Remark 3.2. The search domain condition above is mild in high-dimensional settings. Since κ_l typically scales as $O(d_1 + d_2)$, the bound $|\tau - \tau^*|N \leq A_0\sqrt{N_{j,\tau}\kappa_l}$ allows a neighborhood of τ^* whose width grows with the model dimension, making the requirement compatible with the usual scaling of low-rank matrix recovery. In practice, this motivates restricting the grid to an interior region, such as $[\rho, 1-\rho]$ with a small constant $\rho > 0$ to avoid boundary segments with insufficient sample size.

After introducing the assumptions for the general case, we establish the theoretical properties of the proposed estimators, including both the coefficient matrices and the location of the change-point. We consider both the exact low-rank and near low-rank settings, and present the general theoretical results in Theorem 3.3.

Theorem 3.3. Suppose Assumptions A and B hold. $\boldsymbol{\Theta}_j^{(k)}$ and $\tau^{(k)}$ are the output of Algorithm 1 in k-th iteration. Denote $\mathcal{I}_0 = \{i \mid \min(\tau, \tau^*) \leq t_i \leq \max(\tau, \tau^*), |\tau - \tau^*| > \eta(N, d_1, d_2)\}$, $h_0 = \sum_{i\in\mathcal{I}_0}\langle \boldsymbol{X}_i, \delta_{\boldsymbol{\Theta}^*}\rangle^2/|\mathcal{I}_0|$. Assume the low-rank structure that $\boldsymbol{\Theta}_j^* \in \mathcal{B}(R_{j,q})$ and the regularization constant λ_c satisfying $\lambda_c \geq 4\max\{\sqrt{\kappa_u}K, \sqrt{\kappa_u c_l}\|\boldsymbol{\delta}_{\boldsymbol{\Theta}^*}\|_F, \sqrt{\kappa_u}\|\boldsymbol{\delta}_{\boldsymbol{\Theta}^*}\|_F\}$

. Denote some positive constants C_1, C_2, c_1–c_7 and

$$\eta^* = \frac{c_1 c_u^2 \lambda_c^2}{\kappa_u c_0^2 c_l^4 \|\delta_{\mathbf{\Theta}^*}\|_F^4 N} \left[\frac{17\lambda_c^2(r_1+r_2)}{c_l} + 4\lambda_c\sqrt{N}\sum_{j=1}^{2}\sum_{l=r_j+1}^{d}\rho_l(\mathbf{\Theta}_j^*) + 40\kappa_u \sum_{j=1}^{2}\left(\sum_{l=r_j+1}^{d}\rho_l(\mathbf{\Theta}_j^*)\right)^2 \right].$$

Then with the probability at least $1-N\exp\{-C_1\lambda_c^2/\|\delta_{\mathbf{\Theta}^*}\|_F^2\}-2N\exp\{-C_2\lambda_c^2\|\delta_{\mathbf{\Theta}^*}\|_F^2/h_0\}$, we have

$$\lim_{k\to\infty}\left\|\mathbf{\Theta}_j^{(k)}-\mathbf{\Theta}_j^*\right\|_F^2 \le \frac{c_2\lambda_c^2 r_j}{c_l^2 N_{j,\tau^*}} + \frac{c_3\lambda_c}{c_l\sqrt{N_{j,\tau^*}}}\sum_{l=r_j+1}^{d}\rho_l(\mathbf{\Theta}_j^*) + \frac{c_4\kappa_l}{c_l N_{j,\tau^*}}\left(\sum_{l=r_j+1}^{d}\rho_l(\mathbf{\Theta}_j^*)\right)^2, \quad (3.1)$$

$$\lim_{k\to\infty}\left\|\mathbf{\Theta}_j^{(k)}-\mathbf{\Theta}_j^*\right\|_*^2 \le \frac{c_5\lambda_c^2 r_j^2}{c_l^2 N_{j,\tau^*}} + \frac{c_6\lambda_c r_j}{c_l\sqrt{N_{j,\tau^*}}}\sum_{l=r_j+1}^{d}\rho_l(\mathbf{\Theta}_j^*) + c_7\left(\sum_{l=r_j+1}^{d}\rho_l(\mathbf{\Theta}_j^*)\right)^2, \quad (3.2)$$

and

$$\lim_{k\to\infty}|\tau^{(k)}-\tau^*| \le \eta^*. \quad (3.3)$$

Theorem 3.3 establishes the convergence rates for the outputs of Algorithm 1. For the coefficient matrices before and after the change-point, we provide separate error bounds under the Frobenius and nuclear norms. Directly deriving an upper bound for $\mathbf{\Theta}_j^{(k)}-\mathbf{\Theta}_j^*$ in these norms is challenging. To overcome this difficulty, we introduce an intermediate estimator $\widehat{\mathbf{\Theta}}_{j,\tau^{(k)}}$, obtained by solving the nuclear-norm penalized least squares problem in (2.2), so that the difference $\mathbf{\Theta}_j^{(k)}-\mathbf{\Theta}_j^*$ can be separated into two parts. The bounds for $\widehat{\mathbf{\Theta}}_{j,\tau^{(k)}}-\mathbf{\Theta}_j^*$ and $\mathbf{\Theta}_j^{(k)}-\widehat{\mathbf{\Theta}}_{j,\tau^{(k)}}$ can then be established separately. The upper bound for the change-point estimation error is derived via a contradiction argument, using the error bounds for the coefficient matrices as the key ingredient. The proof of Theorem 3.3 is deferred to Appendix **??** of the supplementary materials. Focusing on all of the upper bounds, each bound decomposes into three interpretable components: (i) an *estimation error term*, reflecting the statistical difficulty of recovering the low-rank structure; (ii) an *approximation error term*, depending on how well the true matrix can be approximated by

rank r_j; and (iii) a *tolerance term* arising from the URSC condition. In particular, when the true matrix has rank at most r_j, both the approximation error and the tolerance term vanish.

As introduced above, Theorem 3.3 provides a general convergence result. By choosing different values of the integer r_j used to construct the restricted regions, we obtain Corollaries 3.1 and 3.2. In particular, when r_j is chosen to be exactly the rank of $\mathbf{\Theta}_j^*$, the constructed subspace fully captures the information of the true matrix. Consequently, both the approximation error term and the tolerance term vanish, and Theorem 3.3 specializes to Corollary 3.1.

Corollary 3.1 (Exact low-rank matrix recovery)**.** Suppose the conditions in Theorem 3.3 hold and assume $\mathbf{\Theta}_j^*$ has exact rank r_j. Then there exists some constants C_1, C_2, and c_1–c_3, with the probability at least $1 - N\exp\left\{-C_1\lambda_c^2/\|\delta_{\mathbf{\Theta}^*}\|_F^2\right\} - 2N\exp\left\{-C_2\lambda_c^2\|\delta_{\mathbf{\Theta}^*}\|_F^2/h_0\right\}$, we have

$$\lim_{k\to\infty}\left\|\mathbf{\Theta}_j^{(k)} - \mathbf{\Theta}_j^*\right\|_F^2 \le \frac{c_1\lambda_c^2 r_j}{c_l^2 N_{j,\tau^*}}, \quad \lim_{k\to\infty}\left\|\mathbf{\Theta}_j^{(k)} - \mathbf{\Theta}_j^*\right\|_*^2 \le \frac{c_2\lambda_c^2 r_j^2}{c_l^2 N_{j,\tau^*}},$$

and

$$\lim_{k\to\infty}|\tau^{(k)} - \tau^*| \le \frac{c_3 c_u^2\lambda_c^4(r_1+r_2)}{\kappa_u c_0^2 c_l^5\|\delta_{\mathbf{\Theta}^*}\|_F^4 N}.$$

Corollary 3.1 shows that, when $r_j = \mathrm{rank}(\mathbf{\Theta}_j^*)$, the upper bounds involve only the estimation error term. To quantify the probability, we replace the parameters depending on the dimensions d_1 and d_2. Following Remark 3.1, we set $\lambda_c = c_\lambda\sqrt{d_1+d_2}$, where c_λ is a constant. In this case, the probability can be expressed as $1 - C_3\exp(-C_4(d_1+d_2))$, with C_3 and C_4 being positive constants. Hence, as d_1 and d_2 grow large, the estimation error bounds hold with high probability. The exact order of these bounds will be discussed in the special cases once all parameters are specified.

We now extend the analysis to the near low-rank setting. To accommodate this more general case, we introduce the notion of an *effective rank*, which measures the number

of sufficiently large singular values. In this framework, r_j is taken to be the effective rank of $\mathbf{\Theta}_j^*$. Theoretical guarantees for the near low-rank case then follow directly from Theorem 3.3.

Corollary 3.2 (Near low-rank matrix recovery)**.** Suppose the conditions in Theorem 3.3 hold and assume $\mathbf{\Theta}_j^*$ belongs to $\mathcal{B}_q(R_{j,q})$ for a radius $R_{j,q}$ such that $\frac{\kappa_u \lambda_c^{-q} R_{j,q}}{c_l^{1-q} N_{j,\tau^*}^{1-\frac{q}{2}}} \leq 1$. Then there exists some constants C_1, C_2, and c_1–c_3, with the probability at least $1 - N\exp\{-C_1\lambda_c^2/\|\delta_{\mathbf{\Theta}^*}\|_F^2\} - 2N\exp\{-C_2\lambda_c^2\|\delta_{\mathbf{\Theta}^*}\|_F^2/h_0\}$, we have

$$\lim_{k\to\infty}\left\|\mathbf{\Theta}_j^{(k)} - \mathbf{\Theta}_j^*\right\|_F^2 \leq \frac{c_1\lambda_c^{2-q}R_{j,q}}{c_l^{2-q}N_{j,\tau^*}^{1-\frac{q}{2}}}, \quad \lim_{k\to\infty}\left\|\mathbf{\Theta}_j^{(k)} - \mathbf{\Theta}_j^*\right\|_*^2 \leq \frac{c_2\lambda_c^{2-2q}R_{j,q}^2}{c_l^{2-2q}N_{j,\tau^*}^{1-q}},$$

and

$$\lim_{k\to\infty}|\tau^{(k)} - \tau^*| \leq \frac{c_3 c_u^2 \lambda_c^{4-q}(R_{1,q} + R_{2,q})}{\kappa_u c_0^2 c_l^{5-q}\|\delta_{\mathbf{\Theta}^*}\|_F^4 N^{1-\frac{q}{2}}}.$$

Because of the structural change, we adopt different effective thresholds, specified as $\frac{\lambda_c}{\sqrt{N_{j,\tau^*}}c_l}$, for the coefficient matrices before and after the change-point. Detailed derivations are provided in Appendix **??** of the supplementary materials. This choice reflects a trade-off between the estimation error and the approximation error. In particular, Corollary 3.2 generalizes Corollary 3.1 in the case $q = 0$.

3.2.2 Matrix compressed sensing

Having established convergence results under the general framework, we now specialize to the compressed sensing setting, where each design matrix $\boldsymbol{X}_i$ corresponds to a sensing operator. We focus on sub-Gaussian designs and begin by stating the assumptions required in this setting, which are essential for the analysis and parallel those in Assumption B for the general case.

Assumption C (Assumptions for the CS case)**.**

(C.1) (Sample size). Assume that the sample size N satisfies $\xi_{\min} N > C_N(r_1+r_2)(d_1+d_2)$, where $\xi_{\min} = \min\{\tau^*, 1-\tau^*\}$, the constant $C_N > 0$ is sufficiently large, and r_1, r_2 are the integers used to construct the restricted regions.

(C.2) (Jump size). Assume that the jump size satisfies $\|\delta_{\mathbf{\Theta}^*}\|_F^2 \geq c_\delta \max\{\|\mathbf{\Theta}_1^*\|_*, \|\mathbf{\Theta}_2^*\|_*\}$, for a sufficiently large constant $c_\delta > 0$.

(C.3) (Sub-Gaussian design and noise). The design vectors $\{\mathrm{vec}(\boldsymbol{X}_i)\}_{i=1}^N$ are i.i.d. mean-zero sub-Gaussian with $K_0 = \max_i \|\mathrm{vec}(\boldsymbol{X}_i)\|_{\psi_2}$ and covariance eigenvalues in $[\underline{\lambda}, \overline{\lambda}]$. The noise variables $\{\epsilon_i\}_{i=1}^N$ are independent, conditionally mean-zero ($\mathbb{E}[\epsilon_i|\boldsymbol{X}_i] = 0$), and sub-Gaussian with $K = \max_i \|\epsilon_i\|_{\psi_2}$.

The assumptions in Assumption C are the CS counterparts of those in the general setting. Assumption (C.1) provides a sample size requirement ensuring that both segments contain enough observations for reliable low-rank recovery and change-point estimation, while Assumption (C.2) imposes a lower bound on the jump size so that the structural change is detectable. Assumption (C.3) further strengthens the model by requiring a sub-Gaussian sensing design together with sub-Gaussian noise. These distributional conditions ensure the concentration properties needed for our high-probability bounds and allow the URSC and URSM conditions from the general framework to carry over to the CS setting.

Next, we specify the nuisance parameter settings and show that the conditions in Assumption A hold with high probability under CS setting. For the URSC and URSM conditions, we set $(c_l, \kappa_l) = (\underline{\lambda}/2,\, c'(d_1+d_2))$ and $(c_u, \kappa_u) = (3\overline{\lambda}/2,\, c'(d_1+d_2))$, where c' is a constant. Lemma **??** verifies that these choices satisfy the URSC and URSM conditions with high probability. For the regularization level, we take $\lambda_{j,\tau} = c_\lambda \sqrt{N_{j,\tau}(d_1+d_2)}$, where $c_\lambda = A\max\{\overline{\lambda}\|\delta_{\mathbf{\Theta}^*}\|_F,\ 3C_e(K_0K \vee K_0^2)\|\delta_{\mathbf{\Theta}^*}\|_F,\ 3C_e(K_0K \vee K_0^2)\}$, with the constant $A > 0$ and C_e the universal constant for sub-Gaussian to sub-exponential transitions. Lemma **??** shows that this condition holds with high probability. For the identifiability condition, we set the minimal detection length as $\eta(N, d_1, d_2) = C_\eta(d_1+d_2)/N$, where $C_\eta > 0$ is a

constant. Lemma **??** establishes that this condition also holds with high probability.

Under the assumptions outlined above, Theorem 3.4 provides convergence rates for the estimated change-point and coefficient matrices, which confirm the effectiveness of Algorithm 1 in the CS setting.

Theorem 3.4. Suppose Assumption C holds. Denote $\mathbf{\Theta}_j^{(k)}$ and $\tau^{(k)}$ as the outputs of Algorithm 1 in k-th iteration. Assume $\mathbf{\Theta}_j^*$ belongs to $\mathcal{B}_q(R_{j,q})$ for a radius $R_{j,q}$ such that $\frac{2^{1-q}c'c_\lambda^{-q}R_{j,q}}{\underline{\lambda}^{1-q}}\left(\frac{d_1+d_2}{\xi_{\min}N}\right)^{1-\frac{q}{2}} \leq 1$. Then there exists some constants C_1, C_2 and c_1–c_3, with the probability at least $1 - C_1 N \exp\{-C_2(d_1+d_2)\}$ we have

$$\lim_{k\to\infty}\left\|\mathbf{\Theta}_j^{(k)} - \mathbf{\Theta}_j^*\right\|_F^2 \leq \frac{c_1 c_\lambda^{2-q} R_{j,q}}{\underline{\lambda}^{2-q}}\left(\frac{d_1+d_2}{N_{j,\tau^*}}\right)^{1-\frac{q}{2}}, \tag{3.4}$$

$$\lim_{k\to\infty}\left\|\mathbf{\Theta}_j^{(k)} - \mathbf{\Theta}_j^*\right\|_*^2 \leq \frac{c_2 c_\lambda^{2-2q} R_{j,q}^2}{\underline{\lambda}^{2-2q}}\left(\frac{d_1+d_2}{N_{j,\tau^*}}\right)^{1-q}, \tag{3.5}$$

and

$$\lim_{k\to\infty}|\tau^{(k)} - \tau^*| \leq \frac{c_3\overline{\lambda}^2 c_\lambda^{4-q}(R_{1,q}+R_{2,q})}{\underline{\lambda}^{5-q}\|\delta_{\mathbf{\Theta}^*}\|_F^4}\left(\frac{d_1+d_2}{N}\right)^{1-\frac{q}{2}}. \tag{3.6}$$

Theorem 3.4 establishes upper bounds for the estimation errors of both the coefficient matrices and the change-point location in the near low-rank compressed sensing setting. By specifying appropriate parameter choices, we show that the Frobenius and nuclear norm errors of the coefficient matrix estimator attain the same order as in compressed sensing models without change-points (Negahban and Wainwright, 2011, Corollary 5). This demonstrates that Algorithm 1 achieves no loss in estimation accuracy due to the presence of a change-point. We also derive an upper bound for the change-point estimation error, which, to the best of our knowledge, is the first such result for compressed sensing models with change-points. The proof of Theorem 3.4 verifies that the assumptions of Theorem 3.3 and Corollary 3.2 hold with high probability under the specified parameter choices for the compressed sensing design. Substituting these choices then yields the desired bounds.

Details are provided in Appendix ?? of the supplementary materials.

4 Low-rank proximal update and optimistic search alternating algorithm

The LR-PGA algorithm discussed in Section 3.1 avoids solving $O(N)$ least-squares problems with nuclear-norm penalty by modifying the iteration strategy. However, this comes at the expense of an $O(N^2)$ computational cost due to the grid search required at each iteration. When the sample size N is large, this cost becomes non-negligible. To overcome this issue, we improve upon Algorithm 1 and propose the LR-POA algorithm, which is inspired by the divide-and-conquer principle. The detailed methodology of LR-POA is presented in Section 4.1, and its theoretical guarantees are established in Section 4.2.

4.1 Method

As discussed in Section 3.1, performing a full grid search at each iteration incurs an $O(N^2)$ computational cost, which becomes prohibitive for large N. This cost arises from evaluating the loss for all candidate change-points. Conceptually, full grid search is equivalent to a sequential search, suggesting that divide-and-conquer strategies can be used to accelerate the procedure by recursively narrowing the search interval.

This idea was exploited by Kovács et al. (2024), who proposed an optimistic search method for the changing-means model based on CUSUM-derived gain functions. However, extending this strategy to regression models is nontrivial. Constructing CUSUM-type gain functions requires repeated coefficient estimation and residual computation around probe points, leading to substantial additional cost. Moreover, sequential evaluation within the reduced interval further undermines computational efficiency. These challenges motivate the development of a new optimistic search strategy tailored to regression models.

To address these challenges, we combine an optimized change-point search strategy with LR-PGA and propose the LR-POA algorithm. As summarized in Algorithm 2, each iteration alternates between a proximal update of the coefficient matrices and an optimistic search for the change-point, with residuals computed as an intermediate step. The key distinction from LR-PGA lies in the search strategy: LR-PGA performs a full grid search, whereas LR-POA employs optimistic search to progressively narrow the candidate interval by comparing gain functions at selected probe points. The search is initialized at each iteration using the previous estimate $\tau^{(k-1)}$, ensuring a monotonic decrease of the total loss under the updated coefficients. Further algorithmic details are provided in Remark 4.1.

Algorithm 2 LR-POA for dynamic low-rank matrix recovery with single change-point

Require: Samples $\{(Y_i, \boldsymbol{X}_i, t_i)\}_{i=1}^N$, grid search domain $\mathcal{T}$, search interval $I = [s, e]$, regularization constant λ_c, step size γ, gain function G_I, search step size $w \in (0, 1)$, minimal sample size n_m, max iteration K, convergence threshold ζ.

Init: $\tau^{(0)} \in \mathcal{T}$, random $\boldsymbol{\Theta}_j^{(0)} \in \mathbb{R}^{d_1 \times d_2}$ $(j = 1, 2)$

function LR-POA$(\tau^{(0)}, \boldsymbol{\Theta}_1^{(0)}, \boldsymbol{\Theta}_2^{(0)})$

  for $k = 1 : K$ **do**

    Proximal update: $\boldsymbol{\Theta}_j^{(k)} \leftarrow Prox_{\gamma\lambda_{j,\tau^{(k-1)}}}\big(\boldsymbol{\Theta}_j^{(k-1)} - \gamma \nabla L_{j,\tau^{(k-1)}}(\boldsymbol{\Theta}_j^{(k-1)})\big)$.

    Compute residuals: $res_{i,j}^{(k)} = Y_i - \langle \boldsymbol{X}_i, \boldsymbol{\Theta}_j^{(k)} \rangle$.

    Optimistic search: $(V, u, U) \leftarrow (s, \tau^{(k-1)}, e)$.

    while $|\mathcal{T}_{(V,U)}| > n_m$ **do**

      $\delta = \mathbf{1}\{U - u > u - V\}$.

      $v = \arg\min_{x \in \mathcal{T}} \big| x - [\delta(U - (U-u)w) + (1-\delta)(V + (u-V)w)] \big|$.

      $$(V, u, U) \leftarrow \begin{cases} (u, v, U), & \text{if } \delta = 1 \text{ and } G_I(v, \boldsymbol{\Theta}_1^{(k)}, \boldsymbol{\Theta}_2^{(k)}) \geq G_I(u, \boldsymbol{\Theta}_1^{(k)}, \boldsymbol{\Theta}_2^{(k)}). \\ (V, u, v), & \text{if } \delta = 1 \text{ and } G_I(v, \boldsymbol{\Theta}_1^{(k)}, \boldsymbol{\Theta}_2^{(k)}) < G_I(u, \boldsymbol{\Theta}_1^{(k)}, \boldsymbol{\Theta}_2^{(k)}). \\ (V, v, u), & \text{if } \delta = 0 \text{ and } G_I(v, \boldsymbol{\Theta}_1^{(k)}, \boldsymbol{\Theta}_2^{(k)}) \geq G_I(u, \boldsymbol{\Theta}_1^{(k)}, \boldsymbol{\Theta}_2^{(k)}). \\ (v, u, U), & \text{if } \delta = 0 \text{ and } G_I(v, \boldsymbol{\Theta}_1^{(k)}, \boldsymbol{\Theta}_2^{(k)}) < G_I(u, \boldsymbol{\Theta}_1^{(k)}, \boldsymbol{\Theta}_2^{(k)}). \end{cases}$$

    end while

    $\tau^{(k)} = \arg\min_{\tau \in \mathcal{T}_{(V,U)}} [\phi_{1,\tau}(\boldsymbol{\Theta}_1^{(k)}) + \phi_{2,\tau}(\boldsymbol{\Theta}_2^{(k)})]$.

    $\zeta_k = \sum_{j=1}^2 \frac{\|\boldsymbol{\Theta}_j^{(k)} - \boldsymbol{\Theta}_j^{(k-1)}\|_F^2}{\|\boldsymbol{\Theta}_j^{(k-1)}\|_F^2}$; if $\zeta_k < \zeta$, break.

  end for

  return $\check{\tau} = \tau^{(k)}$, $\check{\boldsymbol{\Theta}}_j = \boldsymbol{\Theta}_j^{(k)}$

end function

Remark 4.1. In Algorithm 2, the search interval $I = [s, e]$ is fixed. The shortening of the interval in the optimistic search implies that the domain of candidate change points

becomes smaller. In line 8, δ is the parameter presenting which side of the probe point is longer. And line 9 selects another probe point in the longer side. For the comparison process of gain function, the first term of gain function is same for different probe points and we only need to compare the total loss with different probe points.

To avoid the additional computational cost associated with explicitly estimating the coefficient matrices at each probe point, we construct the gain function in line 10 of Algorithm 2 based on the coefficient estimates $\mathbf{\Theta}_j^{(k)}$ obtained at the current iteration. Specifically, for the original interval $I = [s, e]$, $0 \leq s < e \leq 1$, we define

$$G_I(\tau, \mathbf{\Theta}_1^{(k)}, \mathbf{\Theta}_2^{(k)}) = \min\{\phi_I(\mathbf{\Theta}_1^{(k)}), \phi_I(\mathbf{\Theta}_2^{(k)})\} - \phi_{1,\tau}(\mathbf{\Theta}_1^{(k)}) - \phi_{2,\tau}(\mathbf{\Theta}_2^{(k)}), \tag{4.1}$$

where $\phi_I(\mathbf{\Theta}) = \sum_{i\in\mathcal{I}}(Y_i - \langle \boldsymbol{X}_i, \mathbf{\Theta}\rangle)^2 + \lambda_{1,1}\|\mathbf{\Theta}\|_*, \phi_{j,\tau}(\mathbf{\Theta}) = \sum_{i\in\mathcal{I}_{j,\tau}}(Y_i - \langle \boldsymbol{X}_i, \mathbf{\Theta}\rangle)^2 + \lambda_{j,\tau}\|\mathbf{\Theta}\|_*, j = 1, 2$, and τ is the probe point belonging to $\mathcal{T} \subset [\rho, 1-\rho]$. Here, $\mathcal{I} = \{i \mid s \leq t_i \leq e\}$, $\mathcal{I}_{1,\tau} = \{i \mid s \leq t_i < \tau\}$, and $\mathcal{I}_{2,\tau} = \{i \mid \tau \leq t_i \leq e\}$ are the index sets. For the single change-point detection problem, the original interval I is always equal to $[0, 1]$. Intuitively, the first term of the gain function represents the total loss under the no-change-point hypothesis, while the second and third terms correspond to subtracting the total loss under the change-point hypothesis. Thus, the gain function effectively quantifies the improvement in total loss when introducing a change-point compared to assuming no change-point. Analyzing its computation, as discussed in Section 3.1, the evaluation of the gain function reduces to summing the squared residuals at different probe points using the pre-computed residuals, which leads to a computational cost of $O(N)$.

To analyze the computational complexity of Algorithm 2, the first step (proximal update) and the intermediate step (residual computation) in each iteration are identical to those in the LR-PGA algorithm, yielding the same cost of $O(d_1 d_2 N + d_1 d_2 d)$. For the optimistic search step, if the search step size is set to $w = 1/2$, then at each iteration the current interval is reduced by at least $1/4$, and at most $O(\log N)$ interval reductions are

required. Each reduction involves computing the gain function, which costs $O(N)$. Hence, the total computational complexity of the optimistic search step is $O(N \log N)$. Combining these, the overall per-iteration computational complexity of the LR-POA algorithm is $O(d_1 d_2 N + d_1 d_2 d + N \log N)$.

4.2 Main results

In this section, we establish the theoretical convergence guarantees of the LR-POA algorithm under both the CS and MR settings. The detailed analysis for the CS case is presented below, while the results for the MR case are deferred to Appendix **??**. For clarity, we refer to the entire iterative procedure of the algorithm as the *outer loop*, and to the interval reduction steps within the optimistic search as the *inner loop*. Throughout this section, we focus on the exact low-rank recovery setting, which both simplifies the exposition and allows a more transparent analysis. Thus, the integers r_1 and r_2 used to construct the restricted regions coincide with the true ranks r_1^* and r_2^*.

Next, we present the theoretical results for the LR-POA algorithm. Compared with Algorithm 1, the outer loop remains essentially the same, while the key difference lies in the inner loop, where the full grid search is replaced by optimistic search. This distinction implies that the analysis of the outer loop can follow the proof strategy of the LR-PGA algorithm, whereas the analysis of the inner loop relies on the proof techniques of Theorem 2 in Kovács et al. (2024). Theorem 4.2 establishes the upper bounds for the estimation errors of both the coefficient matrices and the change-point location.

Theorem 4.2. Suppose Assumption C holds. Let $\mathbf{\Theta}_j^{(k)}$ and $\tau^{(k)}$ denote the outputs of Algorithm 2 at the k-th iteration. Then there exist constants C_1–C_4 and c_1–c_3 such that, with probability at least $1 - C_1 N \log N \exp\{-C_2(d_1+d_2) - C_3 N (\log N)^2 \exp\{-C_4(d_1+d_2)\}$,

we have

$$\lim_{k\to\infty}\left\|\boldsymbol{\Theta}_j^{(k)}-\boldsymbol{\Theta}_j^*\right\|_F^2\le\frac{c_1c_\lambda^2r_j^*(d_1+d_2)}{\underline{\lambda}^2N_{j,\tau^*}},\quad\lim_{k\to\infty}\left\|\boldsymbol{\Theta}_j^{(k)}-\boldsymbol{\Theta}_j^*\right\|_*^2\le\frac{c_2c_\lambda^2r_j^{*2}(d_1+d_2)}{\underline{\lambda}^2N_{j,\tau^*}},\tag{4.2}$$

and

$$\lim_{k\to\infty}|\tau^{(k)}-\tau^*|\le\frac{c_3\overline{\lambda}^2c_\lambda^4(r_1^*+r_2^*)(d_1+d_2)}{\underline{\lambda}^5\|\delta_{\boldsymbol{\Theta}^*}\|_F^4N}.\tag{4.3}$$

The theoretical guarantees for the LR-POA algorithm are largely consistent with those established in Theorem 3.4, indicating that LR-POA attains the same statistical accuracy as LR-PGA while simultaneously offering substantial computational savings through the optimistic search strategy. As discussed in Section 3.2.2, the upper bounds for the coefficient matrix estimation error remain of the same order as in the no-change-point setting. The complete proof of Theorem 4.2 is provided in Appendix **??** of the supplementary materials.

5 Multiple change-points scenario

We have introduced accelerated algorithms for low-rank matrix recovery with a single change-point and established their theoretical guarantees. In this section, we extend LR-POA algorithm to the multiple change-points scenario. The trace regression model with multiple change-points, along with LR-POBS algorithm proposed to address this problem, is presented in Section 5.1. Theoretical guarantees for the convergence of both the estimated change-points and coefficient matrices are provided in Section 5.2.

5.1 Method

As the model setup for the trace regression model with a single change-point, suppose that we have N independent samples $\{(Y_i,\boldsymbol{X}_i,t_i)\}_{i=1}^N$ and there exists κ^* change-points.

Thus, the trace regression model with multiple change-points is given by

$$Y_i = \langle \boldsymbol{X}_i, \boldsymbol{\Theta}_1^* \rangle \mathbf{1}_{\{t_i \le \tau_1^*\}} + \langle \boldsymbol{X}_i, \boldsymbol{\Theta}_2^* \rangle \mathbf{1}_{\{\tau_1^* < t_i \le \tau_2^*\}} + \cdots + \langle \boldsymbol{X}_i, \boldsymbol{\Theta}_{\kappa^*+1}^* \rangle \mathbf{1}_{t_i > \tau_{\kappa^*}^*} + \epsilon_i,$$

where $\boldsymbol{\Theta}_\kappa^* \in \mathbb{R}^{d_1 \times d_2}$, $\kappa = 1, \ldots, \kappa^* + 1$, are low-rank matrices to be estimated and $\{\tau_\kappa^*\}_{\kappa=0}^{\kappa^*+1}$ are the true change-points. We define $\tau_0^* = 0$ and $\tau_{\kappa^*+1}^* = 1$.

To address the multiple change-point detection problem, we integrate the SBS algorithm of Kovács et al. (2023) with the LR-POA algorithm proposed in this paper. The key idea of the SBS procedure is to search for a single change-point within carefully constructed seeded intervals, under the expectation that some intervals will contain at most one change-point. By decomposing the multiple change-point detection task into a sequence of simpler single change-point problems, the overall computational complexity is significantly reduced. Utilizing the seeded intervals constructed, we apply Algorithm 2 within each interval to detect candidate change-points. The overall procedure for multiple change-point detection is summarized in Algorithm 3, with further discussion provided in Remark 5.1.

Algorithm 3 LR-POBS for dynamic low-rank matrix recovery with multiple change-points

Require: Independent observations $\{(Y_i, \boldsymbol{X}_i, t_i)\}_{i=1}^N$, candidate change-point set $\mathcal{T}$, initial interval $(0, 1]$, decay rate $a \in [1/2, 1)$, minimal segment size m, search boundary ratio $\rho' \in (0, 1/2)$.

1: Construct seeded intervals $\mathcal{S}$ by geometric decay with rate a; keep those with $\ge m$ samples
2: **for** $(s, e] \in \mathcal{S}$ **do**
3: Random init $\boldsymbol{\Theta}_1^{(0)}, \boldsymbol{\Theta}_2^{(0)} \in \mathbb{R}^{d_1 \times d_2}$, pick $\tau^{(0)} \in \{\, \tau \in \mathcal{T} \mid s + \rho'(e-s) < \tau < e - \rho'(e-s) \,\}$
4: Candidate change-point: $\check{\tau}_\kappa \leftarrow \texttt{LR-POA}(\tau^{(0)}, \boldsymbol{\Theta}_1^{(0)}, \boldsymbol{\Theta}_2^{(0)})$
5: **end for**
6: Select final change-points $\{\check{\tau}_1, \ldots, \check{\tau}_{\check{\kappa}}\}$ via model selection
7: Partition samples by $\{\check{\tau}_j\}$ and re-estimate coefficient matrices $\{\check{\boldsymbol{\Theta}}_j\}_{j=1}^{\check{\kappa}+1}$
8: **Output:** $\{\check{\tau}_j\}_{j=1}^{\check{\kappa}}$, $\{\check{\boldsymbol{\Theta}}_j\}_{j=1}^{\check{\kappa}+1}$

Remark 5.1. In lines 3 and 4, the LR-POA algorithm is applied to each seeded interval. Note that the local boundary ratio ρ' differs from the global parameter ρ. The global ρ confines all candidate change-points to $(\rho, 1-\rho]$ within the full interval $(0, 1]$, whereas ρ'

restricts candidates within each local interval $(s, e]$ to its interior $(s+\rho'(e-s), , e-\rho'(e-s)]$, reducing edge effects and improving stability. The model selection in line 6 can follow the narrowest-over-threshold (NOT) criterion Baranowski et al. (2019), which identifies the narrowest interval showing significant evidence of a change.

5.2 Main results

In this section, we present the theoretical results for the LR-POBS algorithm introduced above, restricting attention to the exact low-rank setting for clarity of exposition. As in previous sections, we begin by introducing the necessary conditions.

Assumption A (Assumptions for the MCP setting)**.**

(A.1) (Sample size)**.** Assume that the sample size N satisfies $\xi_{\min} N > C_N r^*_{\max}(d_1 + d_2)$, where $r^*_{\max} = \max\{\text{rank}(\boldsymbol{\Theta}^*_1), \ldots, \text{rank}(\boldsymbol{\Theta}^*_{\kappa^*+1})\}$ is the maximal rank among the coefficient matrices, $\xi_{\min} = \min\{|\tau^*_0 - \tau^*_1|, \ldots, |\tau^*_{\kappa^*} - \tau^*_{\kappa^*+1}|\}$ is the minimal spacing between adjacent change-points, and C_N is a sufficiently large constant.

(A.2) (Jump size)**.** For each segment, we define the jump size as $\delta_\kappa = \|\boldsymbol{\Theta}^*_{\kappa+1} - \boldsymbol{\Theta}^*_\kappa\|_F$. The maximal and minimal jump sizes are given by $\delta_{\max} = \max\{\delta_1, \ldots, \delta_{\kappa^*}\}$ and $\delta_{\min} = \min\{\delta_1, \ldots, \delta_{\kappa^*}\}$, respectively. Assume that the minimal jump size $\delta_{\min}$ satisfies $\delta_{\min} \geq c_\delta \max\{\|\boldsymbol{\Theta}^*_1\|_*, \ldots, \|\boldsymbol{\Theta}^*_{\kappa^*}\|_*\}$, for some positive constant c_δ.

(A.3) (Sub-Gaussian design and noise)**.** The design vectors $\{\text{vec}(\boldsymbol{X}_i)\}_{i=1}^N$ are i.i.d. mean-zero sub-Gaussian with sub-Gaussian norm bounded by K_0, and their covariance eigenvalues lie in $[\underline{\lambda}, \overline{\lambda}]$. The noise variables $\{\epsilon_i\}_{i=1}^N$ are independent, conditionally mean-zero ($\mathbb{E}[\epsilon_i|\boldsymbol{X}_i] = 0$), and sub-Gaussian with sub-Gaussian norm bounded by K.

We focus on the CS model and adopt the same model-setting as in Assumption (A.3). And Assumption (A.2) is the natural extension of Assumption (C.2) to the multiple change-point setting. It requires each true jump δ_κ to be sufficiently large so that all change-points

are detectable across segments. Assumption (A.1) provides the sample size requirement for the multiple change-point regime. In particular, it ensures that each seeded interval containing a single change-point meets the same sample size condition as in the single change-point case (Assumption (C.1)), thereby allowing the local analysis developed for the CS setting to be applied segment-wise.

Next, we introduce the nuisance parameter specifications required in the multiple change-point setting. As in the single change-point case, we first consider the URSC and URSM conditions and set $(c_l, \kappa_l) = (\underline{\lambda}/2,\ c'(d_1+d_2))$ and $(c_u, \kappa_u) = (3\overline{\lambda}/2,\ c'(d_1+d_2))$, where c' is a positive constant. For the basic regularization parameter, we take $\lambda_c = c_{\lambda,\max}\sqrt{d_1+d_2}$, where $c_{\lambda,\max} = A' \max\{\overline{\lambda}\,\delta_{\max}, 3C_e(K_0K \vee K_0^2)\delta_{\max}, 3C_e(K_0K \vee K_0^2)\}$, with $A' > 0$ a constant, C_e the universal constant for converting sub-Gaussian to sub-exponential bounds, and K_0 and K denoting the sub-Gaussian norms of the design matrices and noise variables, respectively. To define the search domain for each segment, we extend the construction used in the single change-point case to the seeded intervals obtained in the multiple change-point procedure. We first build a total candidate change-point set $\mathcal{T} = \{1/N, \ldots, 1\}$. Consider a seeded interval $I = (s, e]$. If there exists a true change-point $\tau^* \in (s, e]$, we select a boundary parameter $\rho' \in (0, 1/2]$ and restrict the candidate change-points to the interior region $\mathcal{T}_I = \{\tau \in \mathcal{T} | s + \rho'(e-s) \le \tau \le e - \rho'(e-s)\}$. For any candidate $\tau \in \mathcal{T}_I$, this construction ensures that $|\tau - \tau^*|N \le A\sqrt{\rho'(e-s)N(d_1+d_2)}$, for some sufficiently large constant $A > 0$, thereby guaranteeing that the search domain satisfies the required regularity conditions within each seeded interval.

Based on the assumptions introduced above, we now establish the theoretical guarantees for the LR-POBS algorithm, summarized in Theorem 5.2.

Theorem 5.2. Suppose that Assumption A holds. Denote the estimated change-points of Algorithm 3 with NOT selection (Baranowski et al., 2019) by $\check{\tau}_1, \ldots, \check{\tau}_{\check{\kappa}}$ as the iteration of LR-POA converges ($k \to \infty$), and let $\check{\kappa}$ be the number of estimated change-points. The

selection threshold γ in the NOT procedure is set as $\gamma = c_1 \underline{\lambda} |I| \delta_{\min}^2$ for some constant $c_1 > 0$, where I denotes the corresponding interval. Moreover, each seeded interval is assumed to have length at least $\xi_{\min}/3$ to ensure sufficient sample information for reliable detection. Then, there exist constants C_1, C_2, and c_2 such that, with probability at least $1 - C_1 N \exp\{-C_2(d_1 + d_2)\}$, we have

$$\check{\kappa} = \kappa^* \quad \text{and} \quad |\check{\tau}_\kappa - \tau_\kappa^*| \leq \frac{c_2 \overline{\lambda}^2 c_{\lambda,\max}^4 (r_\kappa^* + r_{\kappa+1}^*)(d_1 + d_2)}{\underline{\lambda}^5 \delta_{\min}^4 N}, \quad \kappa = 1, 2, \ldots, \kappa^*, \tag{5.1}$$

where r_κ^* and $r_{\kappa+1}^*$ are the true ranks of the coefficient matrices $\mathbf{\Theta}_\kappa^*$ and $\mathbf{\Theta}_{\kappa+1}^*$.

Theorem 5.2 establishes that, with high probability, the number of estimated change-points matches the true number of change-points, and it provides an upper bound on the estimation error for the locations of the change-points in the CS setting. Moreover, to save space, the detailed proofs are deferred to Appendix **??**, while the results under the MR setting are briefly discussed in Remark 5.3.

Remark 5.3. For the MR setting, the theoretical guarantees for the outputs of Algorithm 3 can be established by adapting the proof framework developed for the CS case. The key differences arise from the transformation between the MR model and the trace regression model, which primarily affect the choice of certain parameters. These modifications follow directly from the proof of Theorem **??**. For brevity, the corresponding results and proofs are omitted.

Finally, we summarize all the theoretical results established in this paper in Table 2. For the LR-PGA algorithm, we derive upper bounds for the estimation errors of both the change-point location and the coefficient matrices under the general near low-rank setting. For the LR-POA algorithm, the analysis is more complex, and we therefore focus on the exact low-rank setting. For the LR-POBS algorithm under the multiple change-point setting, we establish upper bounds for the estimation error of each individual change-point.

Table 2: Summary of theoretical results

Algorithm	Case	$\lim_{k\to\infty} \|\mathbf{\Theta}_j^{(k)} - \mathbf{\Theta}_j^*\|_F^2$	$\lim_{k\to\infty} \|\mathbf{\Theta}_j^{(k)} - \mathbf{\Theta}_j^*\|_*^2$	$\lim_{k\to\infty} \|\tau^{(k)} - \tau^*\|$
LR-PGA	CS	$R_{j,q}\left(\frac{d_1+d_2}{N_{j,\tau^*}}\right)^{1-\frac{q}{2}}$	$R_{j,q}^2\left(\frac{d_1+d_2}{N_{j,\tau^*}}\right)^{1-q}$	$(R_{1,q}+R_{2,q})\left(\frac{d_1+d_2}{N}\right)^{1-\frac{q}{2}}$
	MR	$R_{j,q}\left(\frac{d_1+d_2}{n_{j,\tau^*}}\right)^{1-\frac{q}{2}}$	$R_{j,q}^2\left(\frac{d_1+d_2}{n_{j,\tau^*}}\right)^{1-q}$	$(R_{1,q}+R_{2,q})\left(\frac{d_1+d_2}{n}\right)^{1-\frac{q}{2}}$
LR-POA	CS	$\frac{r_j^*(d_1+d_2)}{N_{j,\tau^*}}$	$\frac{r_j^{*2}(d_1+d_2)}{N_{j,\tau^*}}$	$\frac{(r_1^*+r_2^*)(d_1+d_2)}{N}$
	MR	$\frac{r_j^*(d_1+d_2)}{n_{j,\tau^*}}$	$\frac{r_j^{*2}(d_1+d_2)}{n_{j,\tau^*}}$	$\frac{(r_1^*+r_2^*)(d_1+d_2)}{n}$
SBS-OAMM	CS			$\frac{(r_\kappa^*+r_{\kappa+1}^*)(d_1+d_2)}{N}$

6 Simulation and real data application

In this section, we conduct simulation studies and a real-data applicaion to evaluate the performance of the proposed algorithms under both single and multiple change-point scenarios.

6.1 Simulation results

We first present simulation results for the single change-point setting under both the CS and MR cases. Different experimental setups are considered to illustrate the robustness of the methods across various conditions. For each setting, we report the estimation accuracy of the regression coefficients and change-point location, as well as the computational cost of different algorithms. Specifically, we compare the proposed LR-PGA (Algorithm 1) and LR-POA (Algorithm 2) with the traditional two-step method, highlighting the statistical accuracy and computational efficiency.

For the CS case, we generate the data as follows. The coefficient matrices satisfy $d_1 = d_2 = d$ and $r_1^* = r_2^* = r^* = 5$. We consider independent samples $\{(Y_i, \boldsymbol{X}_i, t_i)\}_{i=1}^N$, where $t_i = i/N$ are thresholding variables, and the entries of $\boldsymbol{X}_i \in \mathbb{R}^{d\times d}$ are independently drawn from the standard Gaussian distribution $N(0,1)$. The noise variables are i.i.d. samples from $N(0, 0.1^2)$. The true coefficient matrices $\mathbf{\Theta}_1^*$ and $\mathbf{\Theta}_2^*$ are constructed using the singular vectors of Gaussian random matrices, and we further generate large- and small-signal cases

by scaling their singular values. We vary $N \in \{1500, 2000, 2500, 3500, 4000, 5000\}$ and $d \in \{40, 80\}$, and set $\tau^* \in \{0.3, 0.5\}$ with the regularization constant $c_\lambda = 1.3$. In the MR model, we consider n independent samples $\{(\boldsymbol{y}_a, \boldsymbol{x}_a, t_a)\}_{a=1}^n$, where $\boldsymbol{y}_a \in \mathbb{R}^d$ and $\boldsymbol{x}_a \in \mathbb{R}^d$ denote the response and covariate vectors, respectively, and $t_a = a/n$ is the threshold variable. Each $\boldsymbol{x}_a$ is independently drawn from $N_d(\mathbf{0}, \boldsymbol{I}_d)$, and $\boldsymbol{\epsilon}_a \sim N_d(\mathbf{0}, 0.1^2 \boldsymbol{I}_d)$. We consider $n \in \{300, 500, 800, 1000\}$, $d \in \{50, 100\}$, $r_1^* = r_2^* = r^* = 5$, $\tau^* \in \{0.3, 0.5\}$, and fix the regularization parameter constant at $c_\lambda = 0.3$. Additional details and results for both models are reported in Appendix **??**.

We examine the estimators in the multiple change-point setting under both CS and MR frameworks. For the CS case, we set $d_1 = d_2 = d = 40$, $r^* = 5$ for each true coefficient matrix, and sample size $N = 6000$. The true coefficient matrices $\boldsymbol{\Theta}_\kappa^*$ ($\kappa = 1, \ldots, \kappa^*+1$) are generated from singular vectors of standard Gaussian matrices with large jumps. The design matrices and noise terms are i.i.d. from $N(0, 1)$ and $N(0, 0.1^2)$, respectively. For the MR case, we set $d_1 = d_2 = d = 50$, $r^* = 5$ for each true coefficient matrix, and $n = 2500$, and generate $\boldsymbol{\Theta}_\kappa^*$ similarly. The design and noise vectors follow $N_d(\mathbf{0}, \mathbf{I}_d)$ and $N_d(\mathbf{0}, 0.1^2 \mathbf{I}_d)$, respectively. In both cases, we consider $\kappa^* = 3$ true change-points at $\tau_1^* = 0.3$, $\tau_2^* = 0.5$, and $\tau_3^* = 0.7$. Additional details and results for both models are also reported in Appendix **??**.

6.2 An application to traffic data

In this section, we investigate a transportation science problem by identifying dynamic traffic patterns on weekdays. Specifically, we analyze the New York City taxi trip data maintained by the Taxi and Limousine Commission (TLC), which are publicly available at https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page. For each trip record, we focus on the date and time, pick-up and drop-off locations, and trip distance. Our goal is to uncover the relationship between traffic congestion levels and different travel routes across various times of the day, thereby providing useful insights for transportation

planning and travel behavior analysis. Based on our analysis, we find that taxi traffic patterns throughout a day can be divided into four distinct time periods. The detailed results and explanations are provided in Appendix **??**.

7 Conclusion

In this paper, we study dynamic low-rank matrix recovery under the trace regression model with change-points. We propose the LR-PGA algorithm for single change-point detection and further develop the more efficient LR-POA algorithm using an optimistic search strategy. Compared with the conventional two-step method, the proposed algorithms significantly reduce computational cost while preserving statistical accuracy. We establish consistency of change-point estimation and derive upper bounds for the estimation error of the coefficient matrices. By combining LR-POA with SBS strategy, we extend the method to multiple change-point scenarios and prove consistency for both the number and locations of change-points. Simulation studies demonstrate substantial computational gains with comparable estimation accuracy, and an application to New York City taxi data illustrates the effectiveness of our approach in capturing dynamic structural changes. Several limitations remain. Our theoretical analysis relies on sub-Gaussian error assumptions, which may not hold in the presence of heavy-tailed noise. Developing robust extensions of the proposed algorithms is an important direction for future work. In addition, our focus is limited to trace regression models, while related problems such as matrix completion with change-points are not considered. Extending the accelerated framework to these settings would further broaden its applicability.

References

Absil, P.-A., Mahony, R. and Sepulchre, R. (2008). *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton.

Agarwal, A., Negahban, S. and Wainwright, M. J. (2012). Fast global convergence of gradient methods for high-dimensional statistical recovery. *The Annals of Statistics* **40** 2452–2482.

Babacan, S. D., Luessi, M., Molina, R. and Katsaggelos, A. K. (2012). Sparse bayesian methods for low-rank matrix estimation. *IEEE Transactions on Signal Processing* **60** 3964–3977.

Bai, P., Safikhani, A. and Michailidis, G. (2020). Multiple change points detection in low rank and sparse high dimensional vector autoregressive models. *IEEE Transactions on Signal Processing* **68** 3074–3089.

Bai, P., Safikhani, A. and Michailidis, G. (2023). Multiple change point detection in reduced rank high dimensional vector autoregressive models. *Journal of the American Statistical Association* **118** 2776–2792.

Baranowski, R., Chen, Y. and Fryzlewicz, P. (2019). Narrowest-over-threshold detection of multiple change points and change-point-like features. *Journal of the Royal Statistical Society Series B: Statistical Methodology* **81** 649–672.

Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences* **2** 183–202.

Bybee, L. and Atchadé, Y. (2018). Change-point computation for large graphical models: A scalable algorithm for gaussian graphical models with change-points. *Journal of Machine Learning Research* **19** 1–38.

Cai, C., Li, G., Poor, H. V. and Chen, Y. (2022). Nonconvex low-rank tensor completion from noisy data. *Operations Research* **70** 1219–1237.

Candes, E. and Recht, B. (2012). Exact matrix completion via convex optimization. *Communications of the ACM* **55** 111–119.

Enikeeva, F., Klopp, O. and Rousselot, M. (2025). Change-point detection in low-rank var processes. *Bernoulli* **31** 1058–1083.

Golbabaee, M. and Vandergheynst, P. (2012). Hyperspectral image compressed sensing via low-rank and joint-sparse matrix recovery. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Kao, Y.-H. and Van Roy, B. (2014). Directed principal component analysis. *Operations Research* **62** 957–972.

Kovács, S., Bühlmann, P., Li, H. and Munk, A. (2023). Seeded binary segmentation: a general methodology for fast and optimal changepoint detection. *Biometrika* **110** 249–256.

Kovács, S., Li, H., Haubner, L., Munk, A. and Bühlmann, P. (2024). Optimistic search: Change point estimation for large-scale data via adaptive logarithmic queries. *Journal of Machine Learning Research* **25** 1–64.

Mohan, K. and Fazel, M. (2012). Iterative reweighted algorithms for matrix rank minimization. *The Journal of Machine Learning Research* **13** 3441–3473.

Negahban, S. and Wainwright, M. J. (2011). Estimation of (near) low-rank matrices with noise and high-dimensional scaling. *The Annals of Statistics* **39** 1069–1097.

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

Parikh, N. and Boyd, S. (2014). Proximal algorithms. *Found. Trends Optim.* **1** 127–239.

Shi, L., Wang, G. and Zou, C. (2024). Low-rank matrix estimation in the presence of change-points. *Journal of Machine Learning Research* **25** 1–71.

Vandereycken, B. (2013). Low-rank matrix completion by riemannian optimization. *SIAM Journal on Optimization* **23** 1214–1236.

Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint*, vol. 48. Cambridge university press.

Wang, Y., Jodoin, P.-M., Porikli, F., Konrad, J., Benezeth, Y. and Ishwar, P. (2014). Cdnet 2014: An expanded change detection benchmark dataset. In *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*.

Zhu, Z., Li, X., Wang, M. and Zhang, A. (2022). Learning markov models via low-rank optimization. *Operations Research* **70** 2384–2398.