# Fit First and Detect Later: A Unified Decoupled Framework for Change Detection in High Dimensions

Zhi Yang*, Liwen Zhang*, Bin Liu†, Yufeng Liu‡

**Abstract**

Change point detection for high-dimensional data is increasingly important across scientific domains. Despite progress, most existing methods remain model-specific and assume independence, and they struggle to reconcile detection accuracy with computational efficiency due to the curse of dimensionality. Accordingly, we propose a unified framework for detecting multiple change points in high-dimensional parametric models that accommodates a broad class of structural breaks (e.g., shifts in means or regression coefficients). To cut computation, we design a two-step algorithm that adopts a decoupled strategy to separate parameter estimation from change point localization, reducing computational cost from multiplicative to additive and achieving sub-second runtimes on large-scale data. Using mean and regression models as guiding examples, we establish theoretical guarantees for change point estimation under independence and extend them to time-dependent settings. The final refined change point estimators obtained by our algorithm achieve optimal convergence rate and permit simultaneously inference via their limiting distributions. We also develop a fully data-driven variant for practice. Numerical experiments demonstrate that our method achieves both high statistical accuracy and computational efficiency.

**Keywords:** Decoupled strategy, data segmentation, dynamic programming, global refinement, limiting distribution

*School of Statistics and Data Science, Shanghai University of Finance and Economics, China
†Department of Statistics and Data Science, School of Management at Fudan University, China
‡Department of Statistics, University of Michigan, U.S.A.

# 1 Introduction

High-dimensional data have become a central focus in modern statistics, driven by advances in computational techniques and widespread applications in finance, genomics, climatology, and neuroscience. One key feature of such data is heterogeneity, where the data-generating mechanism may exhibit structural breaks over time, space, or other sequential indices. As a result, methods that rely on homogeneity often break down, spurring extensive research on change point analysis to identify these structural breaks. Moreover, detecting these change points has a variety of real applications. For example, Liu et al. (2020) detected change points in tumor DNA segments indicative of copy number abnormalities, which are typically associated with the underlying disease; in macroeconomics, Cho et al. (2025) observed that parameters unavoidably change over time when predicting the growth rate of the industrial production index in factor-based forecasts.

We study a general framework for high-dimensional change-point detection. Let $[x] := \{1, \ldots, x\}$ for any integer $x > 0$. We observe the sequence $\mathcal{Z} = \{\mathbb{Z}_i\}_{i=1}^n$ generated by

$$\mathbb{Z}_i \sim \mathbb{P}(\boldsymbol{\theta}_j^*), \quad \text{for } \; t_{j-1}^* < i \leq t_j^*, \quad j \in [k^* + 1], \tag{1}$$

where $k^* \geq 0$ denotes the number of change points located at $\mathcal{T}_{k^*}^* = \{t_1^*, \ldots, t_{k^*}^*\}$ with $t_0^* = 0$ and $t_{k^*+1}^* = n$, such that the $j$-th segment follows distribution $\mathbb{P}(\boldsymbol{\theta}_j^*)$ with underlying parameter $\boldsymbol{\theta}_j^*$ satisfying $\boldsymbol{\theta}_{j-1}^* \neq \boldsymbol{\theta}_j^*$. To clarify form (1), we present two representative examples: the mean model (Example 1) and the linear regression model (Example 2).

**Example 1** (Mean model)**.** *Specializing model* (1)*, let data point* $\mathbb{Z}_i = \boldsymbol{X}_i \in \mathbb{R}^p$ *and parameter* $\boldsymbol{\theta}_j^* = \boldsymbol{\mu}_j^* \in \mathbb{R}^p$*. We then have the change-point mean model*

$$\boldsymbol{X}_i = \boldsymbol{\mu}_j^* + \boldsymbol{\varepsilon}_i, \quad \textit{for } \; t_{j-1}^* < i \leq t_j^*, \quad j \in [k^* + 1], \tag{2}$$

*where* $\boldsymbol{\mu}_j^*$ *is the mean vector for the* $j$*-th segment and* $\boldsymbol{\varepsilon}_i$*'s are noise term in* $\mathbb{R}^p$*.*

**Example 2** (Linear regression model)**.** *Specializing model* (1)*, let data point* $\mathbb{Z}_i = (y_i, \boldsymbol{X}_i) \in \mathbb{R} \times \mathbb{R}^p$ *and parameter* $\boldsymbol{\theta}_j^* = \boldsymbol{\beta}_j^* \in \mathbb{R}^p$*. We then have the change-point regression model*

$$y_i = \boldsymbol{X}_i^\top \boldsymbol{\beta}_j^* + \varepsilon_i, \quad \textit{for } \; t_{j-1}^* < i \leq t_j^*, \quad j \in [k^* + 1], \tag{3}$$

*where* $\boldsymbol{\beta}_j^*$ *is the regression coefficient for the* $j$*-th segment and* $\varepsilon_i$*'s are noise term in* $\mathbb{R}$*.*

It is worth mentioning that beyond the two above extensively studied examples, many other interesting models can also be encoded into this form, such as changes in quantile regression (Lee et al. 2018), covariance and network sequences (Bybee & Atchadé 2018, Dette et al. 2022), vector autoregressive processes (Bai et al. 2023), and trace regression (Shi et al. 2024), among others. For analytical convenience, we restrict data point $\mathbb{Z}_i$ to denote either a $p$-dimensional vector or a pair $(y_i, \boldsymbol{X}_i)$, where $y_i$ is a scalar response and $\boldsymbol{X}_i$ is a $p$-dimensional covariate. We focus on the high-dimensional setting $p \gg n$. Now we introduce notation used throughout the paper. Given an integer $v \in [\![n]\!]$[1], the space of candidate change point sets with $k$ locations is defined as

$$\mathbb{T}_k(v) := \Big\{ \{t_1, \ldots, t_k\} : 0 = t_0 < t_1 < \cdots < t_k < t_{k+1} = v \Big\}. \tag{4}$$

Change point detection in model (1) has been extensively studied under fixed-dimensional settings (Killick et al. 2012, Chan et al. 2014, Zou et al. 2020). For comprehensive reviews, we refer the reader to Csörgö & Horváth (1997) and Truong et al. (2020). Driven by contemporary statistical applications, methods tailored for high-dimensional data have rapidly emerged. Most recent works start by constructing test statistics under single change point model (i.e., $k^* \leq 1$) using well-known cumulative sum (CUSUM)-type methods (e.g. Zhang et al. 2022, Wang et al. 2022, Moen et al. 2024, Cho et al. 2025), and then extend to multiple change points via faster greedy search algorithms such as binary segmentation (BS, Fryzlewicz 2014, Kovács et al. 2023) or moving-window scans (Eichinger & Kirch 2018, Chen et al. 2022). However, several limitations arise when applying these methods under the unified framework (1). First, the construction of such test statistics typically relies heavily on specific model structures and assumptions, requiring separate designs for different settings and thus lacking a unified framework. For example, although Wang & Samworth (2018) and Wang et al. (2021) both employ projection statistics, they differ substantially in form and essence, tailored to models (2) and (3), respectively. Although Liu

[1] $v$ denotes the segment's right boundary, and $k$ is the number of interior change point locations. We assume $v > k$ throughout.

et al. (2020) proposed a CUSUM-based unified framework for multiple types of moment changes, our simulations show that its performance deteriorates when the independence assumption is violated. Second, some necessary steps involved in constructing test statistics, such as additional bootstrap procedures (Yu & Chen 2021) or extensive lasso-type fittings (Wang et al. 2021), substantially increase the computational cost for large-scale data.

Many flexible approaches rely on loss functions to guide segmentation, identifying optimal partitions by minimizing a penalized loss criterion over a discrete grid. Such approaches are advantageous for their adaptability, allowing for varied loss functions tailored to distinct modeling contexts and assumptions. One widely adopted method is to solve an $\ell_0$-regularized objective via the dynamic programming (DP) algorithm, known for its optimality. This approach has been applied to model (3) by Leonardi & Bühlmann (2016); by Xu et al. (2024), who addressed temporal dependence; and by Yang et al. (2025), who incorporated a composite quantile loss. However, this advantage comes at the cost of extremely high computation, with overall complexity $O\big(n^2\,\texttt{Lasso}(n,p)\big)$, where $\texttt{Lasso}(n,p)$ refers to the worst-case complexity of a lasso-type fit on a generic interval $(u,v] \subseteq \{1,\ldots,n\}$ with a $p$-dimensional parameter space. The difficulty intensifies for more complex model classes (e.g., quantile regression, Wang et al. 2025) and in settings where regularization demands are nontrivial (e.g., when low-rank structures require a nuclear-norm penalty, Bai et al. 2023, Shi et al. 2024). Not to mention that in practice, cross-validation (CV) is often run hundreds of times or more to select tuning parameters, so the total CV runtime can quickly become practically unaffordable. Fundamentally, the main computational bottleneck of the loss-function-based approach arises from the *coupling* between two sub-tasks: **parameter estimation** (e.g., lasso-type fitting) and the loss evaluations involved in estimating change point locations (hereafter referred to as the **location search**). In other words, one needs to frequently repeat lasso fitting over different intervals $(u,v]$. Due to this coupling, the total computational cost grows multiplicatively with the number of evaluations.

## 1.1 Our contribution

To overcome the aforementioned limitations, we propose an ultra-fast and unified method for multiple change point detection under the general high-dimensional parametric framework (1), which is broadly applicable to various models with independent and time-dependent data. As mentioned earlier, existing loss-function-based methods typically involve a *coupled* dual-objective problem that simultaneously solves two sub-tasks: (i) **parameter estimation**, such as lasso fitting on generic intervals $(u, v]$; and (ii) **location search**, i.e., locating change points over a full or coarse grid on the discrete domain $[\![n]\!]$. In contrast, we introduce a novel *decoupling strategy* that performs these sub-tasks sequentially—parameter estimation first, followed by location search. To implement this idea, we develop a two-step algorithm named `DPFree` (Dynamic Programming Free of parameter estimation) to estimate multiple change points. The detailed procedure is provided in Algorithm 2, where both steps systematically adopt the decoupling strategy. This design reduces the computational cost from multiplicative $O(n^2 \cdot \texttt{Lasso}(n, p))$ to additive $O(n^2 + \texttt{Lasso}(n, p))$. As a result, the proposed algorithm achieves sub-second computational performance in Examples 1–2, even when $n$ and $p$ are both large. Figure 1 displays selected simulation results, demonstrating the strong statistical accuracy and computational efficiency of our method.

While our proposed decoupling strategy can conceptually subsume the *refinement* step for change point estimation used in prior studies (Xu et al. 2024, Cui et al. 2024), that step depends on consistent estimators of $\mathcal{T}^*_{k^*}$. Obtaining such estimators typically requires coupling-based brute-force search approaches (e.g., DP), thereby defeating the original aim of reducing computational complexity. When $k^* \leq 1$ (the single-change-point case), one can apply the decoupling strategy by providing only one suitable guessed location, as in Kaul et al. (2019*b*). However, when the true number $k^* > 0$ is unknown, the decoupling strategy faces fundamental obstacles, yielding three core challenges:

**(Q1)** How should we choose the number $\tilde{k}$ and the guessed locations $\widetilde{\mathcal{T}}_{\tilde{k}} := \{\tilde{t}_1, \ldots, \tilde{t}_{\tilde{k}}\} \in \mathbb{T}_{\tilde{k}}(n)$ (hereafter called *initializers*) so as to carry out the parameter estimation subtask?

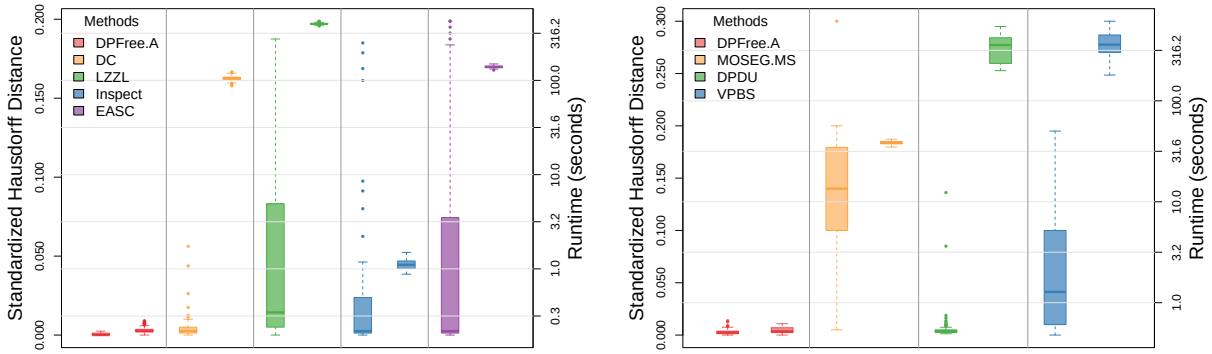


Figure 1: Performance comparison under two settings: the left panel shows the mean model (2) with independent data (M1), and the right panel shows the regression model (3) with temporal dependence (R2), both with $(n, p, k^*) = (800, 400, 4)$. Each method group includes two box plots: the left box plot shows detection accuracy measured by the standardized Hausdorff distance (left y-axis), and the right box plot shows runtime in seconds (right y-axis with logarithmic scale ticks). `DPFree.A` denotes our method, with four competitors in the model (2)(left) and three in the model (3)(right). Further details on the simulation setup and competing methods are provided in Section 4.

**(Q2)** Given $\tilde{k}+1$ segment-wise estimated parameters induced by $\widetilde{\mathcal{T}}_{\tilde{k}}$, how can we recover $\mathcal{T}^*_{k^*}$ by searching the full domain $[n]$? In other words, what regularized objective produces near-optimal change point estimators $\widehat{\mathcal{T}}_{\hat{k}} = \{\hat{t}_1, \ldots, \hat{t}_{\hat{k}}\}$ and which algorithm solve it efficiently?

**(Q3)** Can we design a *global* and adaptive refinement that relies solely on the $\hat{k}+1$ consistent parameter estimators, yielding refined change point estimators $\widehat{\mathcal{T}}^r_{\hat{k}} = \{\hat{t}^r_1, \ldots, \hat{t}^r_{\hat{k}}\}$, while avoiding the pointwise *local* refinement and interval-by-interval adjustments commonly used in prior studies (e.g., Li et al. 2023, Cho & Owens 2024, Cui et al. 2024)?

The main contributions are as follows. `DPFree` addresses the three challenges outlined above through two step: a preliminary step and a refinement step. The preliminary step takes as input the initializers $\widetilde{\mathcal{T}}_{\tilde{k}}$ that satisfy the weak condition in Assumption 1, and solves the proposed $\ell_0$-regularized objective (10) using a modified DP algorithm without additional lasso fits. This step does not sacrifice statistical accuracy, and the resulting preliminary estimators $\widehat{\mathcal{T}}_{\hat{k}}$ consistently recover $k^*$ and attain a near-optimal localization rate of $O_p(s\log(np)/\kappa^2)$, where $s$ is the sparsity and $\kappa^2$ is the minimum jump size between true parameters. The refinement step further sharpens all estimated change points by a global optimization objective, achieving the optimal rate $O_p(1/\kappa^2)$. Finally, through a novel transformation of the joint problem into a sequence of single-change-point problems,

we derive the limiting distributions of the refined change point estimators (Remark 4). We establish statistical guarantees under independence using mean and regression models as guiding examples, and then extend them to temporal dependence analyzed under the functional dependence framework of Wu (2011).

In practice, it is inherently difficult to choose an appropriate set of initializers without prior knowledge of the number and locations of $\mathcal{T}^*_{k^*}$. Given a fixed $\tilde{k}$, we adopt uniform partitioning as the default. We propose `DPFree.A`, a data-driven extension of `DPFree` that adaptively selects the optimal $\tilde{k} \in [k_{\max}]$ based on a user-defined upper bound $k_{\max}$. In addition, we introduce a new CV procedure (Algorithm 3) that automatically selects the tuning parameters. Simulation studies demonstrate that our method achieves strong statistical and computational performance under various model settings. Real-data applications in the Section A (supplementary materials) further validate the effectiveness of our method.

## 1.2 Organization and notations

The rest of this paper is organized as follows. Section 2 introduces the decoupled strategy and presents the proposed two-step algorithm. Section 3 establishes the theoretical guarantees of the estimated change points obtained via `DPFree`. Section 4 develops a data-driven algorithm and reports key numeric results under various settings. Section 5 concludes with a brief discussion. Further discussions, including algorithmic implementation, technical assumptions, complete proofs, additional numerical results, and real-data applications, are provided in the supplementary material.

**Notation.** For $\boldsymbol{v} \in \mathbb{R}^p$ and $q \in (0, \infty]$, let $\|\boldsymbol{v}\|_q$ denote the $\ell_q$ norm. For any set $\mathcal{S}$, let $|\mathcal{S}|$ denote its cardinality. For any integer $x > 0$, let $[x] := \{1, \ldots, x\}$ and $[x]^- := \{x, \ldots, 1\}$. For real sequences $\{a_n\}$ and $\{b_n\}$, write $a_n \lesssim b_n$ or $a_n = O(b_n)$ if $a_n \leq C b_n$ for some constant $C > 0$, and write $a_n \asymp b_n$ if both $a_n \lesssim b_n$ and $b_n \lesssim a_n$. We say a sequence of random variable $Z_n = O_p(a_n)$ if $\sup_n \mathbb{P}(|Z_n| \geq M a_n) \to 0$ as $M \to \infty$, and $Z_n = o_p(a_n)$ if $\sup_n \mathbb{P}(|Z_n| \geq M a_n) \to 0$ for all $M > 0$. The convergence in distribution is denoted by $\xrightarrow{\mathscr{D}}$.

# 2 A decoupled framework for change point detection

We begin by introducing some notations and formulating the parameter estimation problem. Let $l(\mathbb{Z}_i; \boldsymbol{\theta})$ denote a generic loss function for data point $\mathbb{Z}_i$ under given parameter $\boldsymbol{\theta}$. For any integers $0 \leq u \leq v \leq n$, the cumulative cost over segment $\mathcal{Z}_u^v := \{\mathbb{Z}_i\}_{i=u+1}^v$ is given by $\mathcal{L}(u, v; \boldsymbol{\theta}) = \sum_{i=u+1}^v l(\mathbb{Z}_i; \boldsymbol{\theta})$. Note that $\mathcal{L}(u, v; \boldsymbol{\theta}) = 0$ for any $\boldsymbol{\theta}$ if $u = v$. We primarily focus on the squared loss function (with $l(\mathbb{Z}_i; \boldsymbol{\theta})$ specified as $\|\boldsymbol{X}_i - \boldsymbol{\mu}\|_2^2$ in Example 1, and $(y_i - \boldsymbol{X}_i^\top \boldsymbol{\beta})^2$ in Example 2), which is applicable to both examples mentioned. Another closely related example is the Gaussian graphical model, where node-wise regression also employs a squared loss (e.g., Liu et al. 2021). While the squared loss is standard, other loss functions may better suit specific contexts: least absolute loss for Example 1, Huber or composite quantile loss for Example 2, and log-likelihood function for graphical model. For more complex cases, such as quantile regression, the check loss is employed. For any integer $k \geq 1$, define the parameter set $\boldsymbol{\Theta}(k+1) = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{k+1})$ with each $\boldsymbol{\theta}_j \in \mathbb{R}^p$ for $j \in [k+1]$. In particular, the true parameter set is denoted by $\boldsymbol{\Theta}^*(k^*+1) = (\boldsymbol{\theta}_1^*, \ldots, \boldsymbol{\theta}_{k^*+1}^*)$. For any $\mathcal{T}_k \in \mathbb{T}_k(n)$, we obtain an estimated parameter set $\check{\boldsymbol{\Theta}}(k+1; \mathcal{T}_k, \lambda) = \big(\check{\boldsymbol{\theta}}((t_0, t_1], \lambda), \ldots, \check{\boldsymbol{\theta}}((t_k, t_{k+1}], \lambda)\big)$, where each $\check{\boldsymbol{\theta}}((t_{j-1}, t_j], \lambda)$ is computed via

$$\check{\boldsymbol{\theta}}\big((t_{j-1}, t_j], \lambda\big) = \underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\arg\min} \Big\{ \mathcal{L}(t_{j-1}, t_j; \boldsymbol{\theta}) + \lambda \sqrt{t_j - t_{j-1}} \cdot \mathsf{Pen}(\boldsymbol{\theta}) \Big\}, \tag{5}$$

where $\lambda$ is a tuning parameter and $\mathsf{Pen}(\cdot)$ is a suitable penalty. We define the **initial parameters** $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1) := \check{\boldsymbol{\Theta}}(\tilde{k}+1; \widetilde{\mathcal{T}}_{\tilde{k}}, \lambda_1)$ by solving (5) over the segments defined by the initializers $\widetilde{\mathcal{T}}_{\tilde{k}}$ with tuning parameter $\lambda_1 > 0$. In this work, we adopt the $\ell_1$-penalty (lasso), motivated by the assumed sparsity of $\boldsymbol{\theta}_j^*$. Other choices for penalty can also be considered (e.g., SCAD). For generality, we assume at least one true change point, i.e., $k^* \geq 1$.

## 2.1 Grouping map and initializers

Before stating the assumption on the initializers $\widetilde{\mathcal{T}}_{\tilde{k}} \in \mathbb{T}_{\tilde{k}}(n)$ with $\tilde{k} \geq k^*$, we describe the rationale behind the construction of the initial parameters $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1)$. From now on, indices

on the set $\{1, \ldots, \tilde{k}+1\}$ are denoted by $\tilde{\jmath}$, so that $1 \leq \tilde{\jmath} \leq \tilde{k}$. Our goal is to construct the initial parameters $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1) = (\tilde{\boldsymbol{\theta}}_1, \ldots, \tilde{\boldsymbol{\theta}}_{\tilde{k}+1})$ corresponding to the true parameters $\boldsymbol{\Theta}^*(k^*+1) = (\boldsymbol{\theta}^*_1, \ldots, \boldsymbol{\theta}^*_{k^*+1})$ via the following grouping mapping way.

First, divide the extended index set $\{1, \ldots, \tilde{k}+1\}$ into $k^*+1$ non-empty, mutually exclusive *index groups* $\{\mathcal{G}_1, \ldots, \mathcal{G}_{k^*+1}\}$ via *marker* set $\mathcal{M} := \{m(1), \ldots, m(k^*)\} \subseteq \{1, \ldots, \tilde{k}\}$ such that

$$\{1, \ldots, \tilde{k}+1\} = \bigcup_{j=1}^{k^*+1} \mathcal{G}_j,$$

where each group $\mathcal{G}_j$ is defined as

$$\mathcal{G}_j = \Big\{\tilde{\jmath} \in \{1, \ldots, \tilde{k}+1\} : m(j-1) < \tilde{\jmath} \leq m(j)\Big\}, \text{ for } j \in [k^*+1], \tag{6}$$

with convention of $m(0) = 0$ and $m(k^*+1) = \tilde{k}+1$. Since each group $\mathcal{G}_j$ must contain its corresponding marker $m(j)$, it satisfies $|\mathcal{G}_j| \geq 1$.

Second, we require that the initial estimated parameters $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1) = (\tilde{\boldsymbol{\theta}}_1, \ldots, \tilde{\boldsymbol{\theta}}_{\tilde{k}+1})$ satisfy condition:

$$\big\|\tilde{\boldsymbol{\theta}}_{\tilde{\jmath}} - \boldsymbol{\theta}^*_j\big\|_2 = O_p(a_n) \quad \text{for all } \tilde{\jmath} \in \mathcal{G}_j; \;\; j \in [k^*+1], \tag{7}$$

where $a_n \to 0$ as $n \to \infty$ and the convergence rate can be arbitrarily slow. The grouping map guarantees that the initial parameters $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1)$ simultaneously fulfill two criteria: comprehensive coverage of the true parameters $\boldsymbol{\Theta}^*(k^*+1)$, and estimation of each $\boldsymbol{\theta}^*_j$ through possibly multiple analogous initial parameters $\{\tilde{\boldsymbol{\theta}}_{\tilde{\jmath}}\}_{\tilde{\jmath} \in \mathcal{G}_j}$ within each group. Figure 2 offers an intuitive instance that meets both criteria. Condition (7) is deliberately weak and only requires convergence in probability with no rate constraints, reflecting our lack of prior knowledge on the change points $\mathcal{T}^*_{k^*}$. Theorem 1 links $a_n$ to the jump size.

We obtain $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1) = \breve{\boldsymbol{\Theta}}(\tilde{k}+1; \widetilde{\mathcal{T}}_{\tilde{k}}, \lambda_1)$ by solving (5) on the initializers $\widetilde{\mathcal{T}}_{\tilde{k}} \in \mathbb{T}_{\tilde{k}}(n)$. Below we specify Assumption 1 about the initializers. We denote $\Delta = \min_{j \in [k^*+1]}(t^*_j - t^*_{j-1})$.

**Assumption 1.** *Define $\{\Upsilon_n\}_{n \in \mathbb{N}_+}$ as a diverging sequence. There exist constants $\iota \in (0, 1/2]$ and $C_d, C_{\mathfrak{r}} > 0$ such that the initializers $\widetilde{\mathcal{T}}_{\tilde{k}} \in \mathbb{T}_{\tilde{k}}(n)$ satisfies the following conditions.*

***a.*** *The number of initializers is no less than the true number of change points, $\tilde{k} \geq k^*$.*

***b.*** *All initializers are well-separated,* $\min_{\tilde{\jmath}\in[\tilde{k}+1]}(\tilde{t}_{\tilde{\jmath}} - \tilde{t}_{\tilde{\jmath}-1}) \geq d_{\min}$ *where* $d_{\min} \geq C_d\Delta/\Upsilon_n^{1-2\iota}$.

***c.*** *Each true change point has at least one initializer in its fractional neighborhood. That is, for any constant* $\tilde{\iota} \in (1-2\iota,\, 1-\iota]$, *there exists a marker set* $\mathcal{M}$ *such that* $\max_{j\in[k^*]} |\tilde{t}_{m(j)} - t_j^*| \leq \tilde{\mathfrak{r}} := C_{\mathfrak{r}}\Delta\kappa/\sqrt{\{s\Upsilon_n^{2\tilde{\iota}}\}}$.

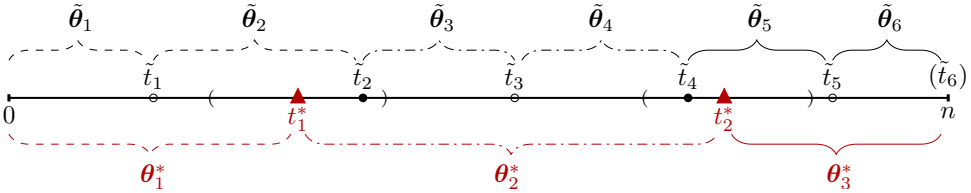


Figure 2: Illustration of grouping map mechanism. Red triangles indicate $\mathcal{T}_2^*$; red braces indicate corresponding parameter $\boldsymbol{\Theta}^*(3)$. Parentheses depict neighborhoods of radius $\tilde{\mathfrak{r}}$. Black dots denote initializers $\widetilde{\mathcal{T}}_5$, with solid ones indicating markers in $\mathcal{M}$. Black braces represent the $\widetilde{\boldsymbol{\Theta}}(6)$, styled differently for each group (dashed, dash-dotted, and solid).

Having introduced the grouping map mechanism and the conditions on initializers, we illustrate these concepts with the example in Figure 2. Consider model (1) with $k^* = 2$ change points $\mathcal{T}_2^* = \{t_1^*, t_2^*\}$, and $\tilde{k} = 5$ initializers $\widetilde{\mathcal{T}}_5 = \{\tilde{t}_1, \ldots, \tilde{t}_5\}$ satisfying Assumption 1. Here $|\tilde{t}_2 - t_1^*| \leq \tilde{\mathfrak{r}}$ and $|\tilde{t}_4 - t_2^*| \leq \tilde{\mathfrak{r}}$, so we choose the marker set $\mathcal{M} = \{2, 4\}$. By definition (6), the resulting index groups are $\mathcal{G}_1 = \{1, 2\}$, $\mathcal{G}_2 = \{3, 4\}$, and $\mathcal{G}_3 = \{5, 6\}$. This setup also shows that the initial parameters $\widetilde{\boldsymbol{\Theta}}(6)$ satisfy the grouping consistency condition (7) via the grouping map, namely $\|\tilde{\boldsymbol{\theta}}_{\tilde{\jmath}} - \boldsymbol{\theta}_1^*\|_2 = O(a_n)$ for $\tilde{\jmath} \in \{1, 2\}$, $\|\tilde{\boldsymbol{\theta}}_{\tilde{\jmath}} - \boldsymbol{\theta}_2^*\|_2 = O(a_n)$ for $\tilde{\jmath} \in \{3, 4\}$, and $\|\tilde{\boldsymbol{\theta}}_{\tilde{\jmath}} - \boldsymbol{\theta}_3^*\|_2 = O(a_n)$ for $\tilde{\jmath} \in \{5, 6\}$.

We next examine Assumption 1 in detail. First, Assumption 1**a** poses no real concern, as $\tilde{k}$ is user-specified and can be safely overestimated without affecting the algorithm. This over-specification is common in practice and supported by prior studies (e.g., Zou et al. 2020, Bai et al. 2023). However, our setup is fundamentally different from the above Bayesian information criterion (BIC)-based studies in terms of output. The latter can be viewed as selecting $\hat{k}$ initializers from $\widetilde{\mathcal{T}}_{\tilde{k}}$, which may include very slowly converging ones (e.g., $\tilde{t}_2, \tilde{t}_4$). In contrast, our method directly outputs $\widehat{\mathcal{T}}_{\hat{k}}$, which enjoys a fast convergence rate. Second, Assumption 1**b** avoids anomalous situations, such as several initializers being densely packed or concentrated near the boundaries. In particular, for model (3) one may

require $d_{\min} \gg s\log(np)$ to guarantee sufficient separation. This ensures that each segment contains enough samples to produce reliable initial parameters satisfying condition (7), which is critical for the success of the subsequent location search task. Similar spacing constraints are common in change point literature (e.g., Roy et al. 2017, Xu et al. 2024, Liu et al. 2026). Lastly, and most importantly, Assumption 1**c** requires that at least one initializer lies within a neighborhood of each true change point. Although this requirement may appear overly stringent, since the initializers are often derived from uninformed guesses without prior information, it turns out to be relatively mild under common regimes. To address this, we analyze the requirement from both theoretical and practical perspectives. Theoretically, when $\kappa/\sqrt{s} = O(1)$ and $\Delta_n := \Delta/n = O(1)$, this condition ensures each true change point has an initializer within an $O(\Upsilon_n^{-\tilde{\iota}}n)$ radius, i.e., $|\tilde{t}_{m(j)} - t_j^*| = o(n)$ for all $j \in [k^*]$. The radius is depends by $\Upsilon_n$, which links to the detection boundary assumption (Assumption 3) discussed later. Note that we impose no requirements on initializers outside the marker set, i.e. on $\{\tilde{t}_{\tilde{\jmath}} : \tilde{\jmath} \notin \mathcal{M}\}$, as illustrated by the black hollow points in Figure 2. Hence, with a sufficiently large number of initializers, there will typically exist a marker initializer $\tilde{t}_{m(j)}$ close to each true change point $t_j^*$, which is naturally reminiscent of the wild binary segmentation (WBS) proposed by Fryzlewicz (2014). W.l.o.g., if several initializers lie near $t_j^*$, we set $m(j)$ as the index achieving $\min_{\tilde{\jmath}} |\tilde{t}_{\tilde{\jmath}} - t_j^*|$ over the neighborhood. In practice, Remark 1 shows that equal spacing serves as an effective initialization strategy.

**Remark 1** (Construction of $\widetilde{\mathcal{T}}_{\tilde{k}}$ in practice)**.** *To obtain an effective marker initializers* $\{\tilde{t}_j\}_{j\in\mathcal{M}}$ *for the true change points* $\mathcal{T}_{k^*}^*$*, a simple approach is to construct a coarse grid: (i)Choose a slowly diverging sequence* $G$ *(e.g.,* $G = c\log n$ *for some constant* $c > 0$*); (ii)Partition* $(0, n] = \{1, \ldots, n\}$ *into* $G$ *equally spaced intervals, ensuring* $G \geq k^* + 1$*. By the classical pigeonhole principle*[2]*, at least one initializer must lie within an* $O(n/\log(n))$ *neighborhood of each change point. Notably, this neighborhood is of order* $o(n)$*. However,*

[2]if $n$ items are put into $m$ containers, with $n > m$, then at least one container must contain more than one item.

*a key distinction between our framework and the literature on search space reduction (Bai & Safikhani 2023, Li et al. 2023, Cho & Owens 2024) is that those methods still perform parameter estimation and location search simultaneously. Due to the search domain reduction, the convergence rate of the estimated change points is inherently limited to $O_p(n/G)$. Achieving the suboptimal rate would require enlarging $G$, thus creating a trade-off between statistical accuracy and computational efficiency.*

## 2.2 $\ell_0$-regularized objective and dynamic programming algorithm

Building on the construction of initial parameters $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1)$, we propose an $\ell_0$-regularized optimization framework to obtain *preliminary estimators* $\widehat{\mathcal{T}}_{\hat{k}} = \{\hat{t}_1, \ldots, \hat{t}_{\hat{k}}\}$. Before that, we denote some useful notation. For any $v \in [\![n]\!]$ and a candidate number of locations $k > 0$, we define a relaxed space of candidate locations that allows adjacent locations to coincide, i.e.,

$$\mathbb{U}_k(v) := \Big\{\{u_1, \ldots, u_k\} : 0 = u_0 \le u_1 \le \cdots \le u_k \le u_{k+1} = v\Big\}. \tag{8}$$

Then for any $\mathcal{U} \in \mathbb{U}_k(v)$ and given $\boldsymbol{\Theta}(k+1) = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{k+1})$, the total loss over all segments is given by

$$\mathrm{TL}(\mathcal{U}, \boldsymbol{\Theta}(k+1)) := \sum_{j=1}^{k+1} \mathcal{L}(u_{j-1}, u_j; \boldsymbol{\theta}_j). \tag{9}$$

Analogously, for any $\mathcal{T} \in \mathbb{T}_k(v)$ defined in (4), the corresponding total loss over the segmentation induced by $\mathcal{T}$ is denoted by $\mathrm{TL}(\mathcal{T}, \boldsymbol{\Theta}(k+1))$.

Unlike classical partitioning problems (e.g., Rinaldo et al. 2021), which simultaneously perform parameter estimation and location search, our method first leverages the given $\tilde{k}+1$ initial parameters, and then reassigns $\tilde{k}$ locations through global search, allowing for overlap between adjacent locations. Then, the best reassigned $\tilde{k}$ locations, referred to as the *raw estimated locations* $\widehat{\mathcal{U}}_{\tilde{k}} := \{\hat{u}_1, \ldots, \hat{u}_{\tilde{k}}\}$, are obtained by solving the $\ell_0$-regularized optimization:

$$\widehat{\mathcal{U}}_{\tilde{k}} = \underset{\mathcal{U} \in \mathbb{U}_{\tilde{k}}(n)}{\arg\min} \left\{ \mathrm{TL}\Big(\mathcal{U}, \widetilde{\boldsymbol{\Theta}}(\tilde{k}+1)\Big) + \gamma \sum_{\tilde{j}=1}^{\tilde{k}+1} \mathbb{I}(u_{\tilde{j}-1} \neq u_{\tilde{j}}) \right\}, \tag{10}$$

where $\gamma > 0$ is the penalty parameter controlling the number of intervals. Next we explain why (10) enables effective detection of change points. Consider a special case where the

index group $\mathcal{G}_j = \{m(j) - 1, m(j)\}$ satisfies $|\mathcal{G}_j| = 2$ and $m(j-1) = m(j) - 2$. From the grouping map (6) and condition (7), the parameter $\boldsymbol{\theta}_j^*$ is associated with two consistent initial parameters $\{\tilde{\boldsymbol{\theta}}_{m(j)-1}, \tilde{\boldsymbol{\theta}}_{m(j)}\}$. According to the construction of the total loss in (9), the two related costs are: (i) $\mathcal{L}(\hat{u}_{m(j-1)}, \hat{u}_{m(j)-1}; \tilde{\boldsymbol{\theta}}_{m(j)-1})$ and (ii) $\mathcal{L}(\hat{u}_{m(j)-1}, \hat{u}_{m(j)}; \tilde{\boldsymbol{\theta}}_{m(j)})$. These two costs are small only near the $j$-th segment, as implied by condition (7) and the fact that $\mathcal{Z}_{t_{j-1}^*}^{t_j^*} \sim \mathbb{P}(\boldsymbol{\theta}_j^*)$. A sufficiently large $\gamma$ forces $\hat{u}_{m(j)-1}$ and $\hat{u}_{m(j)}$ to merge, i.e., $\hat{u}_{m(j)-1} = \hat{u}_{m(j)}$, so that only cost (i) remains in the objective. Consequently, cost (ii) vanishes and the indicator sum $\sum_{\tilde{j}=m(j)-1}^{m(j)} \mathbb{I}(\hat{u}_{\tilde{j}-1} \neq \hat{u}_{\tilde{j}}) = 1$, corresponding to a single valid segment $(\hat{u}_{m(j-1)}, \hat{u}_{m(j)-1}]$. More generally, if $|\mathcal{G}_j| > 2$, all locations in $\{\hat{u}_{m(j-1)+1}, \ldots, \hat{u}_{m(j)}\}$ merge, since their associated $\{\tilde{\boldsymbol{\theta}}_{\tilde{j}}\}_{\tilde{j} \in \mathcal{G}_j}$ converge to $\boldsymbol{\theta}_j^*$. Likewise, the penalty $\gamma$ guarantees that each index group $\mathcal{G}_j$ retains exactly one related initial parameter for all $j \in [k^* + 1]$. By merging duplicate locations, the number of valid partitions formed by $\widehat{\mathcal{U}}_{\tilde{k}}$ converges to $k^* + 1$.

**Remark 2** (Challenges of solving the $\ell_0$-regularized objective)**.** *Solving* (10) *presents significant challenges as a discrete optimization problem with non-smooth, non-convex characteristics. Its solution space grows exponentially with* $n^{\tilde{k}}$*, often containing multiple global optima. While the simulated annealing algorithm has been employed for similar problems, notably by Kaul et al. (2019a) for high-dimensional linear regression model with multiple change point and Bybee & Atchadé (2018) for single change point detection in large graphical models, it still faces practical limitations. The performance of this algorithm is highly sensitive to the choice of hyperparameters, such as the initial temperature and cooling rate. Additionally, finite iterations often result in this algorithm getting stuck at local optima due to its stochastic acceptance mechanism.*

We employ a modefied DP procedure to solve (10), which offers an optimal mechanism without requiring any hyperparameter. As previously noted, (10) can be viewed as a partitioning problem, assuming the empty solution is included in the feasible set. A commonly used DP is the segment neighborhood (SN) approach by Auger & Lawrence (1989). One

can readily see that, the optimization problem (10) closely resembles the SN procedure. The key distinction is that our method explicitly targets segmentations with exactly $\tilde{k}$ "change points" (hereafter referred to as locations), while permitting flexible overlap. Specifically, for any $v \in [\![n]\!]$ and number of locations $k \in [\![\tilde{k}]\!]$, along with data $\mathcal{Z}_0^v$ and initial parameters $\widetilde{\boldsymbol{\Theta}}(k+1) = (\tilde{\boldsymbol{\theta}}_1, \ldots, \tilde{\boldsymbol{\theta}}_{k+1})$, we introduce the Bellman function $\mathcal{F}$:

$$\mathcal{F}\left(v; \widetilde{\boldsymbol{\Theta}}(k+1)\right) = \min_{\mathcal{U} \in \mathbb{U}_k(v)} \left\{ \sum_{\tilde{\jmath}=1}^{k+1} \left[ \mathcal{L}\big(u_{\tilde{\jmath}-1}, u_{\tilde{\jmath}}; \tilde{\boldsymbol{\theta}}_{\tilde{\jmath}}\big) + \gamma \mathbb{I}(u_{\tilde{\jmath}-1} \neq u_{\tilde{\jmath}}) \right] \right\}. \tag{11}$$

It is evident that $\mathcal{F}(n; \widetilde{\boldsymbol{\Theta}}(\tilde{k}+1))$ represents the minimum value attained by (10). By extracting the rightmost partition, we derive the following decomposition:

$$\begin{aligned}
(11) &= \min_{u \leq v} \left\{ \min_{\mathcal{U} \in \mathbb{U}_{k-1}(u)} \sum_{\tilde{\jmath}=1}^{k} \left[ \mathcal{L}\big(u_{\tilde{\jmath}-1}, u_{\tilde{\jmath}}; \tilde{\boldsymbol{\theta}}_{\tilde{\jmath}}\big) + \gamma \mathbb{I}(u_{\tilde{\jmath}-1} \neq u_{\tilde{\jmath}}) \right] + \mathcal{L}\big(u, v; \tilde{\boldsymbol{\theta}}_{k+1}\big) + \gamma \mathbb{I}(u \neq v) \right\} \\
&= \min_{u \leq v} \left\{ \mathcal{F}\left(u; \widetilde{\boldsymbol{\Theta}}(k)\right) + \mathcal{L}\big(u, v; \tilde{\boldsymbol{\theta}}_{k+1}\big) + \gamma \mathbb{I}(u \neq v) \right\}.
\end{aligned}$$

This provides a recursion that gives the minimal cost for data $\mathcal{Z}_0^v$ in terms of the minimal cost for data $\mathcal{Z}_0^u$ for $u \leq v$. In this way, one can compute sequentially for $v \in [\![n]\!]$ by

$$\begin{aligned}
&\mathcal{F}\left(v; \widetilde{\boldsymbol{\Theta}}(1)\right) = \mathcal{L}(0, v; \tilde{\boldsymbol{\theta}}_1) + \gamma, \\
&\mathcal{F}\left(v; \widetilde{\boldsymbol{\Theta}}(k+1)\right) = \min_{u \leq v} \left\{ \mathcal{F}\left(u; \widetilde{\boldsymbol{\Theta}}(k)\right) + \mathcal{L}\big(u, v; \tilde{\boldsymbol{\theta}}_{k+1}\big) + \gamma \mathbb{I}(u \neq v) \right\}, \quad k \in [\![\tilde{k}]\!].
\end{aligned} \tag{12}$$

Then, the estimated locations $\widehat{\mathcal{U}}_{\tilde{k}} \in \mathbb{U}_{\tilde{k}}(n)$ are obtained through the backward recursion:

$$\hat{u}_{\tilde{\jmath}} = \operatorname*{arg\,min}_{u \leq \hat{u}_{\tilde{\jmath}+1}} \left\{ \mathcal{F}\left(u; \widetilde{\boldsymbol{\Theta}}(\tilde{\jmath})\right) + \mathcal{L}\big(u, \hat{u}_{\tilde{\jmath}+1}; \tilde{\boldsymbol{\theta}}_{\tilde{\jmath}+1}\big) + \gamma \mathbb{I}(u \neq \hat{u}_{\tilde{\jmath}+1}) \right\}, \quad \tilde{\jmath} \in [\![\tilde{k}]\!]^- = \{\tilde{k}, \ldots, 1\}, \tag{13}$$

where $\hat{u}_0 = 0$ and $\hat{u}_{\tilde{k}+1} = n$ following (8). After obtaining the optimized solution $\widehat{\mathcal{U}}_{\tilde{k}}$, which may contain redundant locations, it is necessary to construct a unique identification operator to extract valid change point estimators. We define the operator for any $\mathcal{U} = \{u_1, \ldots, u_k\} \in \mathbb{U}_k(v)$ with $k \in [\![\tilde{k}]\!]$ and $v \in [\![n]\!]$ as follow,

$$\mathsf{Unique}\big(\mathcal{U}\big) = \Big\{ u_{\tilde{\jmath}} \,:\, u_{\tilde{\jmath}} \neq u_{\tilde{\jmath}+1}, \tilde{\jmath} \in [\![k]\!] \Big\}. \tag{14}$$

If $u_{\tilde{\jmath}} \neq u_{\tilde{\jmath}+1}$, then $u_{\tilde{\jmath}}$ is kept, whereas if $u_{\tilde{\jmath}} = u_{\tilde{\jmath}+1}$, the lower-indexed location is removed.

Note that the rightmost endpoint $v$ is also excluded. Accordingly, we define the preliminary estimator for the change points as $\widehat{\mathcal{T}}_{\hat{k}} = \{\hat{t}_1, \ldots, \hat{t}_{\hat{k}}\} = \mathsf{Unique}(\widehat{\mathcal{U}}_{\tilde{k}})$, where $\hat{k} = |\mathsf{Unique}(\widehat{\mathcal{U}}_{\tilde{k}})|$.

We summarize (10)–(14) in Algorithm 1, denoted as `DPFreePartI`. Given the initial parameters $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1)$ and tuning parameter $\gamma$, it returns $\widehat{\mathcal{T}}_{\hat{k}}$. For better understanding of `DPFreePartI`'s operational logic for the de-duplication step, we present a detailed example tracking the partition record matrix $\mathbf{P}$ in Section C. Focusing on the merged DP step in Algorithm 1 (lines 2–7), we propose a more efficient and memory-friendly implementation, termed `DynamicUpdate`, which is outlined in Algorithm C.1 (Supplementary material).

**Algorithm 1** `DPFreePartI`$\big(\mathcal{Z}, \widetilde{\boldsymbol{\Theta}}(\tilde{k}+1), \gamma\big)$

**Input:** Data $\mathcal{Z}$, initial parameters $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1)$, and tuning parameter $\gamma > 0$.
**Output:** Estimated number $\hat{k}$, preliminary change point estimators $\widehat{\mathcal{T}}_{\hat{k}}$.

**Initialization**
1: Location set $\mathcal{U} \leftarrow \emptyset$, partition record matrix $\mathbf{P} \leftarrow (0) \in \mathbb{R}^{\tilde{k} \times n}$
**(Merged) dynamic programming step**
2: **for** $k \in [\tilde{k}]$ **do**
3: **for** $v \in [n]$ **do**
4: $\mathcal{F}\left(v; \widetilde{\boldsymbol{\Theta}}(k+1)\right) = \min_{u \le v} \left\{ \mathcal{F}\left(u; \widetilde{\boldsymbol{\Theta}}(k)\right) + \mathcal{L}\big(u, v; \tilde{\boldsymbol{\theta}}_{k+1}\big) + \gamma \mathbb{I}(u \neq v) \right\}$
5: $\mathbf{P}(k, v) \leftarrow \arg\min_{u \le v} \left\{ \mathcal{F}\left(u; \widetilde{\boldsymbol{\Theta}}(k)\right) + \mathcal{L}\big(u, v; \tilde{\boldsymbol{\theta}}_{k+1}\big) + \gamma \mathbb{I}(u \neq v) \right\}$
6: **end for**
7: **end for**
**Backtracking step**
8: $(k, v) \leftarrow (\tilde{k}, n)$
9: **while** $k > 0$ **do**
10: $\mathcal{U} \leftarrow \mathcal{U} \cup \{\mathbf{P}(k, v)\}, \ (k, v) \leftarrow (k-1, \mathbf{P}(k, v))$
11: **end while**
**De-duplication step**
12: $\hat{k} \leftarrow |\mathsf{Unique}\big(\mathcal{U}\big)|$, $\widehat{\mathcal{T}}_{\hat{k}} \leftarrow \mathsf{Unique}(\mathcal{U})$

## 2.3 Global refinement for change point estimators

Since the initial parameters $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1)$ converge at a slow rate and their number significantly exceeds $k^*$, change point estimation efficiency is inevitably affected. To achieve a sharper convergence rate, we perform a refinement of the estimators $\widehat{\mathcal{T}}_{\hat{k}}$, as outlined below. With $\widehat{\mathcal{T}}_{\hat{k}}$ dividing the data into $\hat{k}+1$ segments, we update the parameters by solving (5) with tuning parameter $\lambda_2 > 0$, yielding $\widehat{\boldsymbol{\Theta}}(\hat{k}+1) := \breve{\boldsymbol{\Theta}}(\hat{k}+1; \widehat{\mathcal{T}}_{\hat{k}}, \lambda_2)$. Based on updated parameters $\widehat{\boldsymbol{\Theta}}(\hat{k}+1) = (\hat{\boldsymbol{\theta}}_1, \ldots, \hat{\boldsymbol{\theta}}_{\hat{k}+1})$, we *simultaneously* obtain the refined estimators

$\widehat{\mathcal{T}}^r_{\hat{k}} = \{\hat{t}^r_1, \ldots, \hat{t}^r_{\hat{k}}\}$ by solving

$$\widehat{\mathcal{T}}^r_{\hat{k}} = \underset{\mathcal{T} \in \mathbb{T}_{\hat{k}}(n)}{\arg\min}\, \mathrm{TL}\Big(\mathcal{T}, \widehat{\boldsymbol{\Theta}}(\hat{k}+1)\Big), \tag{15}$$

where the space $\mathbb{T}_{\hat{k}}(n)$ is defined in (4), and the total loss function $\mathrm{TL}(\cdot, \cdot)$ is defined similarly in (9). This method is motivated by the theoretical insight that knowing the underlying parameters $\boldsymbol{\Theta}^*(k^*+1)$ (i.e., both the number and each true parameter) would yield the oracle change point estimators via $\arg\min_{\mathcal{T} \in \mathbb{T}_{k^*}(n)} \mathrm{TL}\big(\mathcal{T}, \boldsymbol{\Theta}^*(k^*+1)\big)$. In other words, the goal is to allocate $\boldsymbol{\Theta}^*(k^*+1)$ within the data $\mathcal{Z}$ to minimize the total cost, ensuring that each oracle change point estimator remains sufficiently close to its true counterpart to achieve the optimal convergence rate. Consequently, as long as $\widehat{\boldsymbol{\Theta}}(\hat{k}+1)$ converge sufficiently fast, the refinement effect can be attained. More specifically, we provide the estimators $\widehat{\mathcal{T}}^r_{\hat{k}}$ that are asymptotically equivalent to the oracle counterparts.

**Novelty of global refinement.** Recent developments in high-dimensional multiple change point analysis increasingly focus on refining preliminary estimators, as optimal rates are essential for limiting distributions and valid inference. Existing methods typically rely on componentwise local refinement: Cho & Owens (2024), Cui et al. (2024), and Xu et al. (2024) utilized updated parameters from both sides of each true change point; Rinaldo et al. (2021) and Li et al. (2023) applied group-lasso on rescaled intervals. In stark contrast, our objective (15) adopts a global refinement mechanism that simultaneously refits all preliminary change point estimators, eliminating local search intervals and further additional parameter estimation, while guaranteeing optimal simultaneous convergence rates.

We propose a simplied DP procedure to solve (15). For any right endpoint $v \in [n]$ and number $k \in [\hat{k}]$, we define the refined Bellman function $\mathcal{F}^r$ as

$$\mathcal{F}^r\left(v; \widehat{\boldsymbol{\Theta}}(k+1)\right) = \min_{\mathcal{T} \in \mathbb{T}_k(v)} \sum_{j=1}^{k+1} \mathcal{L}(t_{j-1}, t_j; \hat{\boldsymbol{\theta}}_j).$$

By setting $\mathcal{F}^r\big(v; \widehat{\boldsymbol{\Theta}}(1)\big) = \mathcal{L}(0, v; \hat{\boldsymbol{\theta}}_1)$, a simple decomposition leads to the recurrence formula

$$\mathcal{F}^r\left(v; \widehat{\boldsymbol{\Theta}}(k+1)\right) = \min_{t<v} \left\{ \mathcal{F}^r\left(t; \widehat{\boldsymbol{\Theta}}(k)\right) + \mathcal{L}\big(t, v; \hat{\boldsymbol{\theta}}_{k+1}\big) \right\}, \quad k \in [\hat{k}]. \tag{16}$$

Then, the refined estimator $\widehat{\mathcal{T}}_{\hat{k}}^r = \{\hat{t}_1^r, \ldots, \hat{t}_{\hat{k}}^r\}$ are obtained by solving the following optimization problem via backward recursion

$$\hat{t}_j^r = \underset{t<\hat{t}_{j+1}^r}{\arg\min} \left\{ \mathcal{F}^r\left(t; \widehat{\boldsymbol{\Theta}}(j)\right) + \mathcal{L}\left(t, \hat{t}_{j+1}^r; \hat{\boldsymbol{\theta}}_{j+1}\right) \right\}, \quad j \in [\hat{k}]^-. \tag{17}$$

Algorithm C.2 (in the supplementary materials), denoted as `DPFreePartII`, summarizes the procedure described in (15)–(17). Given the $\widehat{\boldsymbol{\Theta}}(\hat{k}+1)$, it returns the refined change point estimators $\widehat{\mathcal{T}}_{\hat{k}}^r$. Furthermore, the refined DP step within `DPFreePartII` adopts an accelerated, memory-efficient implementation, which is outlined in Algorithm C.3.

## 2.4 Algorithm `DPFree` and its Strengths

**Algorithm 2** `DPFree`$\left(\mathcal{Z}, \widetilde{\mathcal{T}}_{\tilde{k}}, \gamma, \lambda_1, \lambda_2\right)$

**Input:** Data $\mathcal{Z}$, initializers $\widetilde{\mathcal{T}}_{\tilde{k}}$, and tuning parameters $\lambda_1, \lambda_2, \gamma > 0$.

**Output:** (i) Change point estimation: estimated number $\hat{k}$, preliminary change point estimators $\widehat{\mathcal{T}}_{\hat{k}}$, refined change point estimators $\widehat{\mathcal{T}}_{\hat{k}}^r$; (ii) parameter estimation: initial parameters $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1)$, updated parameters $\widehat{\boldsymbol{\Theta}}(\hat{k}+1)$.

**Preliminary step**

**Step 1-1:** Compute initial parameters $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1)$ by solving (5) with initializers $\widetilde{\mathcal{T}}_{\tilde{k}}$ and $\lambda_1$.

**Step 1-2:** Run `DPFreePartI`$\left(\mathcal{Z}, \widetilde{\boldsymbol{\Theta}}(\tilde{k}+1), \gamma\right)$ in Algorithm 1 to obtain $\widehat{\mathcal{T}}_{\hat{k}}$.

**Refinement step**

**Step 2-1:** Compute updated parameters $\widehat{\boldsymbol{\Theta}}(\hat{k}+1)$ by solving (5) with $\widehat{\mathcal{T}}_{\hat{k}}$ and $\lambda_2$.

**Step 2-2:** Run `DPFreePartII`$\left(\mathcal{Z}, \widehat{\boldsymbol{\Theta}}(\hat{k}+1)\right)$ in Algorithm C.2 to obtain refined estimators $\widehat{\mathcal{T}}_{\hat{k}}^r$.

The two step procedure is formally described in Algorithm 2, referred to as `DPFree`. For better comprehension of the full procedure, we present a visual roadmap of `DPFree` following the setting in Figure 2, illustrated in Figure 3. The first line in Figure 3 mirrors Figure 2,

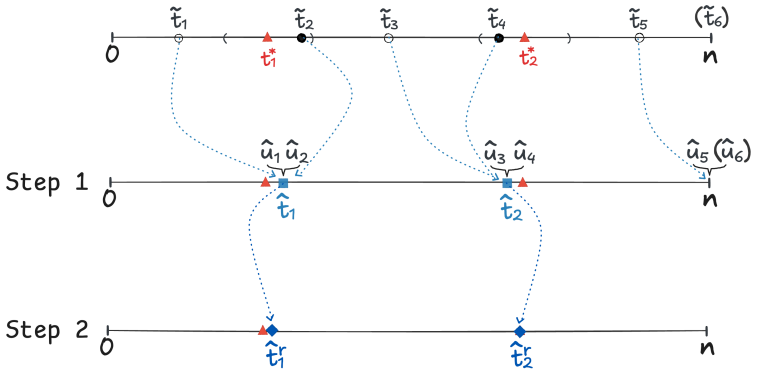


Figure 3: Illustration of the working mechanism of Algorithm 2 with $\tilde{k} = 5$ and $k^* = 2$.

with parameter details omitted. The blue rectangles indicate the preliminary estimators $\widehat{\mathcal{T}}_2$ obtained in **Step 1**, while their corresponding raw estimated locations $\widehat{\mathcal{U}}_5 = \{\hat{u}_1, \ldots, \hat{u}_5\}$ are shown above the second line. The blue diamonds represent the refined estimators $\widehat{\mathcal{T}}_2^r$ produced in **Step 2**. The main advantage of `DPFree` lies in its computational efficiency, with detailed complexity results given in Lemma 1.

**Lemma 1.** *Consider* $\texttt{DPFree}\big(\mathcal{Z}, \widetilde{\mathcal{T}}_{\tilde{k}}, \gamma, \lambda_1, \lambda_2\big)$ *applied to observations* $\mathcal{Z}$ *from either model* (2) *or* (3)*, initializers* $\widetilde{\mathcal{T}}_{\tilde{k}}$ *with* $\tilde{k} \ll n$ *and any* $\gamma, \lambda_1, \lambda_2 > 0$*. Then the worst-case computational time complexity is* $O(n^2 + \texttt{Lasso}(n, p))$ *and the memory requirement is* $O(n + p)$*.*

Since the decoupling strategy performs parameter estimation and location search sequentially, the computational cost naturally amounts to $O(n^2 + \texttt{Lasso}(n, p))$. The DP step is implemented via the algorithm `DynamicUpdate`, which only stores quantities of dimension $p$ and $n$, leading to a memory cost that is significantly lower than the $O(n + p^2)$ required by standard DP (Xu et al. 2024). Section C.2 (Supplementary materials) provides a detailed analysis of Lemma 1 and further discusses additional strengths of our algorithm.

# 3 Theoretical results

Although `DPFree` is applicable to a variety of model settings, a unified theoretical analysis of its detection accuracy requires specific assumptions. This section presents unified results for two representative models: the mean model (2) and the regression model (3). We begin this section by outlining the model assumptions underlying our framework.

Assumption 2 unifies the assumptions across the two models. Due to space constraints, the detailed forms are deferred to the supplementary material: Section D.2 for the mean mode and Section D.3 for the regression model.

**Assumption 2.** *Let* $\mathcal{Z}$ *be generated from model* (1)*, where the data either follow the mean model* (2) *under Assumption D.1, or the regression model* (3) *under Assumption D.2.*

Below is a concise summary. Assumptions D.1-D.2**a** specify sub-Gaussian or sub-Exponential tail behavior for the underlying independent distributions. Assumptions D.1-D.2**b** constrain the minimum and maximum eigenvalues of the covariance matrices, implying a form of restricted eigenvalue conditions. The sparsity $s$ of the underlying parameters is specified in Assumptions D.1–D.2**c**, while Assumptions D.1-D.2**d** define the minimal jump size $\kappa = \min_{j\in[k^*]} \kappa_j = \min_{j\in[k^*]} \left\|\boldsymbol{\theta}^*_{j+1} - \boldsymbol{\theta}^*_j\right\|_2$, and impose an upper bound on $\kappa_j$.

**Assumption 3.** *There exists a sufficiently large constant $C_{snr}$ and a diverging sequence $\{\Upsilon_n\}_{n\in\mathbb{N}_+}$ such that:* ***a.*** $\Delta\kappa^2 \geq C_{snr}\Upsilon_n s\log(np)$; ***b.*** $\Upsilon_n \gg s\log^2(np)$.

Assumption 3 imposes a lower bound on the signal-to-noise ratio (SNR) quantity, allowing $s$ and $p$ to increase as long as the jump size $\kappa$ adequately offsets the growing dimensionality. Identifying the preliminary estimators $\widehat{\mathcal{T}}_{\hat{k}}$ relies solely on Assumption 3**a**, without imposing any specific requirement on $\Upsilon_n$. Moreover, establishing optimal convergence rate and valid inference for the refined estimators $\widehat{\mathcal{T}}^r_{\hat{k}}$ requires Assumption 3, which means that the diverging sequence must satisfy $\Upsilon_n \gg s\log^2(np)$. Notably, it encompasses both the non-vanishing jump and the vanishing jump where $\kappa \to 0$, provided that $s^2\log^3(np) = o(\Delta)$.

**Remark 3.** *The diverging sequence $\Upsilon_n$ serves two purposes. First, it ensures the localization error rate $\max_{1\leq j\leq \hat{k}} |\hat{t}_j - t^*_j| = o(\Delta)$ (see Theorem 2), where Rinaldo et al. (2021) and Wang et al. (2025) specifically used $\Upsilon_n = \log(n\vee p)^\xi$ for arbitrarily small $\xi > 0$. Second, it also plays a role in Assumption 1**c** for initializers: larger SNR require a smaller radius $\tilde{\mathfrak{r}}$. To see this, recall the example in Figure 2. If the jump size $\kappa_1 = \|\boldsymbol{\theta}^*_2 - \boldsymbol{\theta}^*_1\|_2$ is large, then since the data $\mathcal{Z}^{\tilde{t}_2}_{\tilde{t}_1}$ contains information from $\mathcal{Z}^{\tilde{t}_2}_{t^*_1} \sim \mathbb{P}(\boldsymbol{\theta}^*_2)$, the marker initializer $\tilde{t}_2$ that lies too far from the true change point $t^*_1$ would cause the initial parameter $\tilde{\boldsymbol{\theta}}_2$ to fail to converge to $\boldsymbol{\theta}^*_1$. We discuss possible forms of $\Upsilon_n$ in Remark C.2 (Supplementary material).*

## 3.1 Results for the preliminary step

As a starting point, we focus on the initialization requirement given in (7), which is essential for triggering the algorithm. To this end, we introduce Theorem 1.

**Theorem 1.** *Suppose Assumptions 2 and 3**a** hold. Let* $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1) = \breve{\boldsymbol{\Theta}}(\tilde{k}+1; \widetilde{\mathcal{T}}_{\tilde{k}}, \lambda_1)$ *denote the solution to (5) given initializer set* $\widetilde{\mathcal{T}}_{\tilde{k}}$ *and tuning parameter* $\lambda_1$*. Further assume* $\widetilde{\mathcal{T}}_{\tilde{k}} \in \mathbb{T}_{\tilde{k}}(n)$ *satisfies Assumption 1 with* $\iota^\star = \tilde{\iota} - (1-2\iota)$*, and let* $\lambda_1 = 2(\lambda' + \lambda'')$ *where* $\lambda' = C'\sqrt{\log(np)}$ *and* $\lambda'' = C''\tilde{\mathfrak{r}}/\sqrt{d_{\min}}$ *for absolute constants* $C', C'' > 0$*. Then for any* $\tilde{j} \in \mathcal{G}_j$ *with* $j \in [k^*+1]$*, the following holds with probability at least* $1 - o(1)$*:*

*(i)* $\|(\tilde{\boldsymbol{\theta}}_{\tilde{j}})_{S_{\tilde{j}}^c}\|_1 \le 3\|(\tilde{\boldsymbol{\theta}}_{\tilde{j}} - \boldsymbol{\theta}_j^*)_{S_j}\|_1$*;*

*(ii)* $\|\tilde{\boldsymbol{\theta}}_{\tilde{j}} - \boldsymbol{\theta}_j^*\|_2 \le C_1\lambda_1(s/d_{\min})^{1/2} \le C_2\Upsilon_n^{-\iota^\star}\kappa$*, for some absolute constants* $C_1, C_2 > 0$*.*

Theorem 1 ensures that all estimators in group $\mathcal{G}_j$ converge to $\boldsymbol{\theta}_j^*$, under controlled sparsity. The $\ell_2$ error can be further expressed as $\left\|\tilde{\boldsymbol{\theta}}_{\tilde{j}} - \boldsymbol{\theta}_j^*\right\|_2 \le C_2\Upsilon_n^{-\iota^\star}\max\{\kappa_{j-1}, \kappa_j\}$. This bound reflects that when the segment $(\tilde{t}_{\tilde{j}-1}, \tilde{t}_{\tilde{j}}]$ fully covers the segment $(t_{j-1}^*, t_j^*]$, the estimator $\tilde{\boldsymbol{\theta}}_{\tilde{j}}$ incorporates information from three different parameters: $\boldsymbol{\theta}_{j-1}^*$, $\boldsymbol{\theta}_j^*$, and $\boldsymbol{\theta}_{j+1}^*$. Naturally, larger jumps between these parameters (quantified by $\kappa_{j-1}$ and $\kappa_j$) lead to greater estimation errors, hence the error bound depends on their maximum. The penalty parameter $\lambda_1$ consists of two distinct terms. The first term $\lambda'$ aligns with standard lasso theory and contributes an error of order $O_p(\lambda'\sqrt{s/d_{\min}})$, which simplifies to $O_p(\sqrt{s\log(np)/d_{\min}})$. Under Assumption 3**a** and the scaling $d_{\min} \ge C_d\Delta/\Upsilon_n^{1-2\iota}$, this satisfies $\sqrt{s\log(np)/d_{\min}} \lesssim \Upsilon_n^{-\iota}\kappa$. The second term $\lambda''$ introduces an additional $\ell_2$ error of order $O_p(\lambda''\sqrt{s/d_{\min}}) = O_p(\tilde{\mathfrak{r}}\sqrt{s}/d_{\min})$. Substituting $\tilde{\mathfrak{r}} \asymp \Upsilon_n^{-\tilde{\iota}}\Delta\kappa s^{-1/2}$ from Assumption 1**c** yields $O_p(\Upsilon_n^{-\{\tilde{\iota}-(1-2\iota)\}}\kappa)$. Combining both terms gives an overall error bound of $O_p(\Upsilon_n^{-\min\{\{\tilde{\iota}-(1-2\iota)\},\iota\}}\kappa)$, explicitly linking it to the minimal jump size $\kappa$. Selecting $\tilde{\iota} \le 1-\iota$ ensures rate dominance by the second term $\lambda''$. While $\tilde{\iota} > 1-\iota$ would change dominance to the first term $O_p(\lambda'\sqrt{s/d_{\min}})$, this would require stronger conditions on $\tilde{\mathfrak{r}}$. Our theory only needs to satisfy (7), making $\tilde{\iota} \le 1-\iota$ the natural choice.

**Theorem 2.** *Suppose Assumptions 2 and 3**a** hold, and assume that the initial parameters* $\widetilde{\boldsymbol{\Theta}}(\tilde{k}+1)$ *satisfy the properties in Theorem 1(i) and (ii). Let the estimated change points* $\widehat{\mathcal{T}}_{\tilde{k}}$ *be obtained from Algorithm 1, using a tuning parameter* $\gamma$ *satisfying* $C_\gamma n\lambda_1^2 s/d_{\min} \le \gamma \le$

$c_\gamma \Delta \kappa^2$, *for some constants* $C_\gamma, c_\gamma > 0$. *Then, as* $n, p \to \infty$, *the following results hold:*

*(i)* $\mathbb{P}\{\hat{k} = k^*\} \to 1$, *as* $n, p \to \infty$;

*(ii)* $\max_{1 \le j \le \hat{k}} |\hat{t}_j - t_j^*| = O_p(s \log(np)/\kappa^2)$.

Theorem 2 establishes that, with high probability, even when the provided parameter estimates converge much more slowly than the optimal rate, the algorithm `DPFreePartI` is nevertheless able to accurately recover the true number of change points and yield location estimates that fall within optimal (off by a logarithmic factor) neighborhoods of the true locations. We then denote this sub-optimal rate $\hat{\mathfrak{r}} := C_\epsilon s \log(np)/\kappa^2$ for some constant $C_\epsilon > 0$. We next provide the proof sketch for Theorem 2, with additional aspects such as tuning parameter $\gamma$ and the no-change-point scenario discussed in Section C.3.

**Technical novelty in proving Theorem 2.** The proof proceeds in two steps. First, we show that the solution $\widehat{\mathcal{U}}_{\tilde{k}}$ from (10) ensures that each marker raw estimated location (i.e., $\hat{u}_{m(j)}$) lies within a $\Delta/4$ neighborhood of the corresponding true change point $t_j^*$, and all estimated locations with indices in the same group $\mathcal{G}_j$ are merged to this marker raw estimated location $\hat{u}_{m(j)}$, yielding $\hat{k} = k^*$ with high probability, as shown in Proposition 1.

**Proposition 1.** *Under the same conditions of Theorem 2, let* $\widehat{\mathcal{U}}_{\tilde{k}}$ *be the solution to* (10) *and* $\hat{k} = |\mathsf{Unique}(\widehat{\mathcal{U}}_{\tilde{k}})|$. *Then with probability at least* $1 - o(1)$*: (i) for all* $j \in [\![k^*]\!]$, $|\hat{u}_{m(j)} - t_j^*| \le \Delta/4$*; (ii) for all* $j \in [\![k^* + 1]\!]$, $\hat{u}_{h(j)} = \cdots = \hat{u}_{m(j)}$ *with* $h(j) = m(j-1) + 1$, *and thus* $\hat{k} = k^*$.

Second, relying on Proposition 1 and its extension Proposition 1', we transform the nested backward recursion (13) into a sequence of independent single change point minimization problems:

$$\hat{u}_{m(j)} = \underset{u \le \hat{u}_{m(j+1)}}{\arg\min} \begin{cases} \mathcal{L}(t_j^*, u; \tilde{\boldsymbol{\theta}}_{h(j)}) - \mathcal{L}(t_j^*, u; \tilde{\boldsymbol{\theta}}_{h(j+1)}), & \text{when } u \ge t_j^*; \\ \mathcal{L}(u, t_j^*; \tilde{\boldsymbol{\theta}}_{h(j+1)}) - \mathcal{L}(u, t_j^*; \tilde{\boldsymbol{\theta}}_{h(j)}), & \text{when } u < t_j^*. \end{cases} \tag{18}$$

We recursively apply this minimization over $j \in [\![k^*]\!]^-$, progressively tightening the bound $|\hat{u}_{m(j)} - t_j^*|$ from $O_p(\Delta/4)$ to $O_p(s \log(np)/\kappa_j^2)$. Unlike existing techniques (e.g., Kaul et al. 2019*a*, Wang et al. 2025) focusing solely on analyzing the $\ell_0$-regularized objective (10),

our analysis further incorporates the recursion (13) of the DP algorithm, a distinctive and novel technique in theoretical analysis.

## 3.2 Results for the refinement step

As shown in Theorem 2, the event $\mathcal{E}_1 := \{\hat{k} = k^* \text{ and } \max_{j\in[k^*]} \kappa_j^2|\hat{t}_j - t_j^*| \le C_\epsilon s\log(np)\}$ holds with high probability. Under $\mathcal{E}_1$, the estimators computed on segments induced by $\widehat{\mathcal{T}}_{\hat{k}}$ are of higher quality than the initial parameters, as formalized in the following theorem.

**Theorem 3.** *Suppose Assumptions 2 and 3 hold. Let the updated parameters* $\widehat{\boldsymbol{\Theta}}(\hat{k}+1) = (\hat{\boldsymbol{\theta}}_1, \ldots, \hat{\boldsymbol{\theta}}_{\hat{k}+1}) = \breve{\boldsymbol{\Theta}}(\hat{k}+1; \widehat{\mathcal{T}}_{\hat{k}}, \lambda_2)$ *be the solution to (5), computed with preliminary change point estimators* $\widehat{\mathcal{T}}_{\hat{k}}$ *and tuning parameter* $\lambda_2 = 2C'\sqrt{\log(np)}$ *for some constant* $C' > 0$. *Then for any* $j \in [\![k^*+1]\!]$, *the following results hold with probability at least* $1 - o(1)$*:*
*(i)* $\big\|(\hat{\boldsymbol{\theta}}_j)_{S_j^c}\big\|_1 \le 3\big\|(\hat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j^*)_{S_j}\big\|_1$;
*(ii)* $\big\|\hat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j^*\big\|_2 \le C_1(s\log(np)/\Delta)^{1/2}$ *for some absolute constant* $C_1 > 0$.

Theorem 3 shows that the parameter estimators over intervals $\{(\hat{t}_{j-1}, \hat{t}_j]\}_{j=1}^{\hat{k}+1}$ achieve the minimax optimal rate when $\Delta_n = O(1)$. To understand why the small tuning suffices, consider the case $(t_{j-1}^*, t_j^*] \subset (\hat{t}_{j-1}, \hat{t}_j]$ under model (3). The interval $\mathcal{I} = (\hat{t}_{j-1}, t_{j-1}^*] \cup (t_{j-1}^*, t_j^*] \cup (t_j^*, \hat{t}_j]$ splits into $\mathcal{I}_1 \cup \mathcal{I}_2 \cup \mathcal{I}_3$. Here, $\lambda' \asymp \sqrt{\log(np)}$ controls $\|\sum_{i\in\mathcal{I}} \varepsilon_i \boldsymbol{X}_i\|_\infty$, while $\lambda'' \asymp \{\log(np) \vee \hat{\mathfrak{r}}\}/\Delta^{1/2}$ controls $\|\sum_{i\in\mathcal{I}_1}(\boldsymbol{\beta}_j^* - \boldsymbol{\beta}_{j-1}^*)^\top \boldsymbol{X}_i\boldsymbol{X}_i^\top\|_\infty$ and $\|\sum_{i\in\mathcal{I}_3}(\boldsymbol{\beta}_j^* - \boldsymbol{\beta}_{j+1}^*)^\top \boldsymbol{X}_i\boldsymbol{X}_i^\top\|_\infty$ (see Corollary E.6). Under Assumption 3 and the bound $\hat{\mathfrak{r}}$, we have $\lambda'' \ll \lambda'$, validating the use of a reduced penalty.

**Theorem 4.** *Suppose Assumptions 2 and 3 hold, and that the updated parameters* $\widehat{\boldsymbol{\Theta}}(\hat{k}+1)$ *satisfy the bounds in Theorem 3(i) and (ii). Let the refined estimators* $\widehat{\mathcal{T}}_{\hat{k}}^r$ *be obtained by solving the problem (15). Then we have* $\max_{1\le j\le\hat{k}} |\hat{t}_j^r - t_j^*| = O_p(1/\kappa^2)$, *as* $n, p \to \infty$.

Theorem 4 shows that the use of updated parameters leads to a minimax lower bound for the localization rate (e.g., Wang & Samworth 2018, Rinaldo et al. 2021). While the global refinement procedure (15) is conducted on $\widehat{\boldsymbol{\Theta}}(\hat{k}+1)$, which depends on $(s, p)$, Theorem 4

establishes that the refined estimator satisfies $|\hat{t}_j^r - t_j^*| = O_p(1/\kappa_j^2)$ for all $j \in [k^*]$, with a rate that is independent of $s$, $p$, and $n$. This holds uniformly across high-dimensional settings, even when $\kappa$ vanishes.

**A sequence of single-change-point problems.** Similar to Proposition 1, under event $\mathcal{E}_1$, the solutions from (15) satisfy $|\hat{t}_j^r - t_j^*| \leq \Delta/4$ for any $j \in [k^*]$ (see Proposition F.1). Similar to transformation (18), and relying by Proposition F.1 and its extension Proposition F.1', we obtain a sequence of independent single change point minimization problems:

$$\begin{aligned} \hat{t}_j^r &= \underset{t \leq t_j^* + \Delta/4}{\arg\min} \Big\{ \underbrace{\mathcal{L}\big(t_j^*, t; \hat{\boldsymbol{\theta}}_j\big) - \mathcal{L}\big(t_j^*, t; \hat{\boldsymbol{\theta}}_{j+1}\big)}_{-\mathbb{Q}_j^+(r; \hat{\boldsymbol{\theta}}_j, \hat{\boldsymbol{\theta}}_{j+1}) \text{ with } r = t - t_j^*} \Big\}, \text{ if } t \geq t_j^*; \\ \hat{t}_j^r &= \underset{t \geq t_j^* - \Delta/4}{\arg\min} \Big\{ \underbrace{\mathcal{L}\big(t, t_j^*; \hat{\boldsymbol{\theta}}_{j+1}\big) - \mathcal{L}\big(t, t_j^*; \hat{\boldsymbol{\theta}}_j\big)}_{-\mathbb{Q}_j^-(r; \hat{\boldsymbol{\theta}}_j, \hat{\boldsymbol{\theta}}_{j+1}) \text{ with } r = t - t_j^*} \Big\}, \text{ if } t < t_j^*. \end{aligned} \tag{19}$$

Given the boundary $t_j^* - \Delta/4$ and $t_j^* + \Delta/4$, along with parameter estimators $\hat{\boldsymbol{\theta}}_j$ and $\hat{\boldsymbol{\theta}}_{j+1}$ before and after the change point $t_j^*$, the expression (19) is equivalent to finding the maximizer of

$$\mathbb{Q}_j(r; \hat{\boldsymbol{\theta}}_j, \hat{\boldsymbol{\theta}}_{j+1}) = \mathbb{Q}_j^-(r; \hat{\boldsymbol{\theta}}_j, \hat{\boldsymbol{\theta}}_{j+1}) \cdot \mathbb{I}(r < 0) + \mathbb{Q}_j^+(r; \hat{\boldsymbol{\theta}}_j, \hat{\boldsymbol{\theta}}_{j+1}) \cdot \mathbb{I}(r > 0).$$

We also define the oracle quantity $\mathbb{Q}_j(r; \boldsymbol{\theta}_j^*, \boldsymbol{\theta}_{j+1}^*)$ based on the above construction. Finally, for $r = \hat{t}_j^r - t_j^* > 0$ (with the symmetric case handled similarly), we can decompose $\mathbb{Q}_j(r; \hat{\boldsymbol{\theta}}_j, \hat{\boldsymbol{\theta}}_{j+1})$ as $\mathbb{Q}_j(r; \boldsymbol{\theta}_j^*, \boldsymbol{\theta}_{j+1}^*)$ plus a remainder term. Lemma F.1 (supplementary material) shows that this remainder term is of order $o_p(\max\{r\kappa_j^2, \sqrt{(r\kappa_j^2)}, 1\})$, which is ensured under the stronger SNR condition in Assumption 3**b**. By applying a contradiction argument we can establish that $|\hat{t}_j^r - t_j^*| = O_p(1/\kappa_j^2)$. Note that Wang et al. (2025) adopts a similar optimization formulation to (15) in their second step for high-dimensional quantile regression, their convergence analysis of $\widehat{\mathcal{T}}_{\hat{k}}^r$ is derived globally via Theorem 3.4.1 of van der Vaart & Wellner (2023). In contrast, we analyze a sequence of single change point prob-

lems in (19), yielding a localized and more interpretable proof framework. Importantly, the expression (19) serves two purposes: (i) proving the optimal convergence rate, and (ii) simultaneously constructing limiting distributions for all change point estimators.

**Simultaneous inference results.** While removing the $s\log(np)$ term may seem a minor gain over prior near-optimal rates, it is crucial from an inferential perspective. Theorem 4 gives a high-probability bound for each $\kappa_j^2|\hat{t}_j^r - t_j^*|$ with $j \in [\hat{k}]$, and together with the single change point problem in (19), provides a theoretical basis for simultaneously constructing confidence intervals. While our main theoretical contributions center on simultaneous change point estimation, we treat inferential results more concisely since they follow directly from our result (19). The full results appear in Remark 4. The proofs of limiting distribution primarily rely on Slutsky's theorem and the argmax continuous mapping theorem (see Theorem 3.2.2 in van der Vaart & Wellner 2023). Section F.3 in the supplementary materials includes notation, inference assumptions, and brief proofs.

**Remark 4** (Limit distributions)**.** *Under the same conditions as Theorem 4, and assuming one of Assumptions F.1–F.2 holds for the corresponding model, we simultaneously establish the limiting distributions of $\widehat{\mathcal{T}}_{\hat{k}}^r$ under two regimes for the jump sizes $\{\kappa_j\}_{j=1}^{k^*}$: (i) the vanishing regime with $\kappa_j \to 0$ as $n \to \infty$; and (ii) the non-vanishing regime where $\kappa_j \to \varrho_j$ converges to some constant $\varrho_j$. Specifically, we obtain the following for all $j \in [k^*]$.*

*(i) (Vanishing regime) The estimation error satisfies that $|\hat{t}_j^r - t_j^*| = O_p(\kappa_j^{-2})$, as $n \to \infty$. Let $\mathbb{M}(r)$ be a two-sided standard Brownian motion defined as*

$$\mathbb{M}(r) = \mathbb{B}_1(-r) \cdot \mathbb{I}(r < 0) + 0 \cdot \mathbb{I}(r = 0) + \mathbb{B}_2(r) \cdot \mathbb{I}(r > 0),$$

*with $\mathbb{B}_1(r)$ and $\mathbb{B}_2(r)$ being two independent standard Brownian motions defined on $\mathbb{Z}_+$. As $n \to \infty$, we have*

$$\kappa_j^2(\hat{t}_j^r - t_j^*) \xrightarrow{\mathscr{D}} \arg\max_{r \in \mathbb{R}} \left\{ \mathbb{M}(r) - \frac{\varpi_j}{2\sigma_j}|r| \right\}, \tag{20}$$

*where $\varpi_j$ and $\sigma_j$ are model-specific constants defined explicitly in Section F.3.*

*(ii) (Non-vanishing regime) The estimation error satisfies that $|\hat{t}_j^r - t_j^*| = O_p(1)$. Let $\mathbb{W}_j(r)$*

*denote a two-sided random walk process defined by*

$$\mathbb{W}_j(r) = \mathbb{I}(r < 0) \sum_{i=1}^{-r} Z_{j,i} + \mathbb{I}(r = 0) \cdot 0 + \mathbb{I}(r > 0) \sum_{i=1}^{r} Z^*_{j,i},$$

*where* $Z_{j,i} \overset{i.i.d.}{\sim} \mathsf{Dist}(\mu_j, \sigma_j^2)$[3] *and* $Z^*_{j,i} \overset{i.i.d.}{\sim} \mathsf{Dist}(\mu^*_j, (\sigma^*_j)^2)$ *are mutually independent sequences. The parameters* $(\mu_j, \sigma_j^2)$ *and* $(\mu^*_j, (\sigma^*_j)^2)$ *depend on the underlying model, specified in Section F.3. As* $n \to \infty$*, we have*

$$\hat{t}^r_j - t^*_j \xrightarrow{\mathscr{D}} \arg\max_{r \in \mathbb{Z}} \mathbb{W}_j(r).$$

*Moreover, we can prove that* $\kappa_j^2(\hat{t}^r_j - t^*_j)$ *with* $j \in [k^*]^-$ *are asymptotically independent, and* $\hat{t}^r_j - t^*_j$ *with* $j \in [k^*]^-$ *are asymptotically independent. We refer to Remark C.3 for a discussion on constructing valid confidence intervals around estimated change points.*

### 3.3 Under temporal dependence and general tail behavior

While independence is often assumed in change point analysis for technical ease, real-world data typically exhibit temporal dependence. Instead of using traditional mixing conditions, we adopt the functional dependence framework (Wu 2011, Xu et al. 2024), which accommodates a wide class of dependent processes and aligns well with our setting. Definition D.1 introduces functional dependence for a nonstationary multivariate sequence $\boldsymbol{Z}_i = (Z_{i1}, \ldots, Z_{ip})^\top = G_i(\cdot)$, where $G_i(\cdot)$ is a time-varying $\mathbb{R}^p$-valued measurable function[4]. This setup covers the stationary noise vectors $\{\boldsymbol{\varepsilon}_i\}$ in model (2) and covariates $\{\boldsymbol{X}_i\}$ in model (3), both generated from time-invariant functions $G_\varepsilon(\cdot)$ and $G_X(\cdot)$, respectively. Likewise, the scalar noise $\{\varepsilon_i\}$ in model (3) arises from a time-invariant $\mathbb{R}$-valued function $g_\varepsilon(\cdot)$. See Definitions D.3–D.5 in the supplementary material for details.

Let $\{\boldsymbol{Z}_i\}_{i=1}^n \subset \mathbb{R}^p$ be a consecutive segment of $\{\boldsymbol{Z}_i\}_{i \in \mathbb{Z}}$ with mean zero, following Definition D.1. We require that $\{\boldsymbol{Z}_i\}_{i \in \mathbb{Z}}$ satisfy the following conditions: (i) temporal dependence: for absolute constants $\zeta_1, c_0, C_{Z1} > 0$, $\sup_{m \geq 0} \exp(c_0 m^{\zeta_1}) \Delta^{\boldsymbol{Z}}_{m,4} \leq C_{Z1}$, where $\Delta^{\boldsymbol{Z}}_{\cdot,\cdot}$ is dependence measure; (ii) tail behaviour: for absolute constants $\zeta_2, C_{Z2} > 0$, satisfying

[3] $\mathsf{Dist}(\mu, \sigma^2)$ represents continuous distribution supported in $\mathbb{R}$, with mean $\mu$ and finite variance $\sigma^2$.
[4] In the special case of $p = 1$, we write $Z_i = g_i(\cdot)$. Refer to Definition D.2.

$\sup_i \sup_{\|\boldsymbol{v}\|_2=1} \mathbb{P}(|\boldsymbol{v}^\top \boldsymbol{Z}_i| > t) \leq 2\exp\{-(t/C_{Z2})^{\zeta_2}\}$. Specifically, we relax the i.i.d. sub-Exponential condition in Assumption D.1**a** to Assumption D.1a' in model (2). In model (3), we relax the i.i.d. sub-Gaussian conditions on $\{\boldsymbol{X}_i\}$ and $\{\varepsilon_i\}$ in Assumption D.2**a** to Assumptions D.2a' and D.2a", respectively. The constant $\zeta_1$ controls dependence strength: larger values indicate weaker dependence, with $\zeta_1 = \infty$ corresponding to independence. The tail parameter $\zeta_2 \leq 2$ places distributions in the sub-Weibull family, encompassing sub-Exponential ($\zeta_2 = 1$) and sub-Gaussian ($\zeta_2 = 2$) cases. We introduce the composite parameter $\zeta^{-1} = \zeta_1^{-1} + \zeta_2^{-1}$ (similar to Assumption 8 in Wong et al. 2020), which balances dependence and tail behavior. Restricting $\zeta < 1$ ensures the applicability of Bernstein's inequality under temporal dependence (Theorem 4 in Xu et al. 2024). These relaxed assumptions inevitably lead to increased sample size requirements for estimation, scaling as the order of $O(\{s\log(np)\}^{2/\zeta-1})$. This sample size inflation, in turn, demands more stringent SNR conditions, which we formalize subsequently in Assumption 3'.

**Assumption 3'.** *Let $C_{snr} > 0$ be a sufficiently large constant and $\{\Upsilon_n\}_{n\in\mathbb{N}_+}$ a diverging sequence such that:* ***a.*** $\Delta\kappa^2 \geq C_{snr}\Upsilon_n\{s\log(np)\}^{2/\zeta-1}$; ***b.*** $\Upsilon_n \gg s\log^{2/\zeta}(np)$.

Such assumptions consequently yield the convergence rate $O_p(\{s\log(np)\}^{2/\zeta-1}/\kappa^2)$ for preliminary change point estimators $\widehat{\mathcal{T}}_{\hat{k}}$, without materially affecting the remaining theoretical results. The proofs of Theorems 1–4 under the independence assumption repeatedly use a collection of lemmas that provide deviation bounds for partial sums derived from specific sequences. These sequences include $\{\boldsymbol{\varepsilon}_i\}_{i\in\mathbb{Z}}$ in model (2) (see Lemmas D.1–D.2), and $\{\boldsymbol{X}_i\boldsymbol{X}_i^\top\}_{i\in\mathbb{Z}}$ and $\{\varepsilon_i\boldsymbol{X}_i^\top\}_{i\in\mathbb{Z}}$ in model (3) (see Lemmas D.4–D.7). Each deviation bound concerns partial sums constructed from these sequences. For the dependent setting, we develop time-dependent analogues of these lemmas, where each deviation bound inherits an additional dependence on the constant $\zeta$. It is crucial to note that our core proof technique for change point estimation does not rely on these deviation bounds. Therefore, we omit the detailed proofs under temporal dependence, as interested readers can readily substitute the relevant lemmas with corresponding dependent versions.

# 4 Numerical experiments

This section provides extensive numerical studies. Section 4.1 outlines a new tuning method and a fully data-driven version of `DPFree`. To save space, simulation designs and metrics (including $\mathbb{E}[\hat{k}]$ for estimated number, $\mathbf{A}(\hat{k})$ for accuracy, and $\mathbb{E}[d_{\mathrm{H}}]$[5], $\mathbf{Std}(d_{\mathrm{H}})$ for standardized Hausdorff distance) are described in Section B.1. In addition, we compare with DC (Cho 2016), Inspect (Wang & Samworth 2018), LZZL (Liu et al. 2020), and ESAC (Moen et al. 2024) for mean shifts; VPBS (Wang et al. 2021), DPDU (Xu et al. 2024), and MOSEG.MS (Cho & Owens 2024) for regression changes. Technical implementation details for these methods are provided in Section B.2. Key results are reported in Section 4.2, with additional experiments reported in Sections B.3–B.4.

## 4.1 Implementation details of `DPFree`

Without prior knowledge of $\mathcal{T}^*_{k^*}$, we suggest equally spaced initializers: $\widetilde{\mathcal{T}}_{\tilde{k}} = \{\tilde{j}\lfloor n/(\tilde{k}+1)\rfloor : \tilde{j} \in [\tilde{k}]\}$ for any $\tilde{k} \geq k^*$. We consider two cases: (**C1**) $k^* = 2$ with change points at $\{0.25n, 0.6n\}$ and (**C2**) $k^* = 4$ with change points at $\{0.1n, 0.3n, 0.7n, 0.8n\}$, both with unequally spaced change points that pose greater challenges than conventional setups.

### 4.1.1 Tuning parameter selection

A good choice of tuning parameters is essential for accurate estimation results. Our proposed procedure in **Step 1** of Algorithm 2 involves the selection of two tuning parameters $\lambda_1$ and $\gamma$. For their simultaneous selection, we adopt a CV method and partition the data into observations with odd and even indices, which we denote by $\mathcal{Z}^{\mathrm{odd}} = \{\mathbb{Z}_i^{\mathrm{odd}}\}_{i=1}^{n_0}$ and $\mathcal{Z}^{\mathrm{even}} = \{\mathbb{Z}_i^{\mathrm{even}}\}_{i=1}^{n_0}$, with $n_0 = n/2$. Without loss of generality, assume $n$ is even. We present our CV procedure for selecting $(\lambda_1, \gamma)$ in Algorithm 3. The intuitive reasoning behind the proposed CV tuning procedure is elaborated in Remark B.1. For $\lambda_2$ in **Step 2**, we

[5]For clarity, we use $d_{\mathrm{H}}$ to denote the standardized Hausdorff distance for both $\widehat{\mathcal{T}}_{\hat{k}}$ and $\widehat{\mathcal{T}}^r_{\hat{k}}$, with values in tables shown side by side separated by a vertical bar |.

recommend $\lambda_2 = \sqrt{2}\lambda_0^*$, where $\lambda_0^*$ is the optimal auxiliary tuning parameter that minimizes (21), with the factor $\sqrt{2}$ accounting for the sample size difference. Thanks to the substantial reduction in computational cost, our method permits extensive tuning over a broad and finely discretized search grid, without incurring prohibitive computational burden.

**Algorithm 3** Cross-Validation for `DPFree`

**Input:** Candidate sets $\Lambda_1, \Gamma, \Lambda_0 \subset \mathbb{R}^+$, number of initializers $\tilde{k}$, training data $\mathcal{Z}^{\text{odd}}$, test data $\mathcal{Z}^{\text{even}}$.

**Output:** Optimal tuning parameters $\big(\lambda_1^\diamond(\tilde{k}), \gamma^\diamond(\tilde{k})\big)$ and corresponding change point estimators

1: **Initialize:** Set initializers $\{\tilde{j}\lfloor n_0/(\tilde{k}+1)\rfloor : \tilde{j} \in [\![\tilde{k}]\!]\}$

2: **for** each $(\lambda_1, \gamma) \in \Lambda_1 \times \Gamma$ **do**

3: Apply **Step 1** of Algorithm 2 on training set $\mathcal{Z}^{\text{odd}}$ with fixed $(\lambda_1, \gamma)$

4: Obtain change point estimators $\widehat{\mathcal{T}}^{\text{odd}}_{\hat{k}^{\text{odd}}} = \{\hat{t}_1^{\text{odd}}, \ldots, \hat{t}_{\hat{k}^{\text{odd}}}^{\text{odd}}\}$

5: Evaluate the residual sum of squares (RSS): Compute

$$\text{RSS}(\lambda_1, \gamma; \tilde{k}) = \min_{\lambda_0 \in \Lambda_0} \sum_{j=1}^{\hat{k}^{\text{odd}}+1} \sum_{i=\hat{t}_{j-1}^{\text{odd}}+1}^{\hat{t}_j^{\text{odd}}} l\left(\mathbb{Z}_i^{\text{even}}; \check{\boldsymbol{\theta}}\big((\hat{t}_{j-1}^{\text{odd}}, \hat{t}_j^{\text{odd}}], \lambda_0\big)\right), \tag{21}$$

6: where $\check{\boldsymbol{\Theta}}\big(\hat{k}^{\text{odd}}; \widehat{\mathcal{T}}^{\text{odd}}_{\hat{k}^{\text{odd}}}, \lambda_0\big) = \big(\check{\boldsymbol{\theta}}\big((0, \hat{t}_1^{\text{odd}}], \lambda_0\big), \ldots, \check{\boldsymbol{\theta}}\big((\hat{t}_{\hat{k}^{\text{odd}}}^{\text{odd}}, n_0], \lambda_0\big)\big)$ is defined similarly as (5) solved under $\mathcal{Z}^{\text{odd}}$

7: **end for**

8: **Select optimal parameters:**

$$\Big(\lambda_1^\diamond(\tilde{k}), \gamma^\diamond(\tilde{k})\Big) = \underset{(\lambda_1,\gamma)\in\Lambda_1\times\Gamma}{\arg\min}\ \text{RSS}(\lambda_1, \gamma; \tilde{k}) \tag{22}$$

9: **Final estimation:** Obtain estimators $\widehat{\mathcal{T}}^{\text{odd}}_{\hat{k}^{\text{odd}}}$ under optimal tuning pair $\big(\lambda_1^\diamond(\tilde{k}), \gamma^\diamond(\tilde{k})\big)$

10: **Scale to original:** Scale by factor of 2 to yield change point estimators on original scale

### 4.1.2 Constructing the data-adaptive procedure

When the true number of change points $k^*$ is unknown, the input initializer count may not satisfy $\tilde{k} \geq k^*$. To address this, we propose a data-adaptive version of `DPFree`, termed `DPFree.A`, which proceeds as follows: (i) Specify a sufficiently large upper bound $k_{\max} \geq 1$, tuning parameters $\lambda_1$, $\gamma > 0$, and an auxiliary parameter $\Lambda_0$; (ii) Choose the number of initializers via $\tilde{k}^\diamond(\lambda_1, \gamma) = \arg\min_{\tilde{k}\in[\![k_{\max}]\!]} \text{RSS}(\lambda_1, \gamma; \tilde{k})$; (iii) Run `DPFree` using

$\tilde{k}^{\diamond}(\lambda_1, \gamma)$ as the initializer count. This design leverages the observation that $\tilde{k} < k^*$ leads to underestimation and large residual error, while $\tilde{k}$ close to $k_{\max}$ yields unstable estimates due to small segment sizes, potentially violating theoretical guarantees (e.g., Theorem 1) and reducing localization accuracy. In simulations, we fix $k_{\max} = 8$ for case **(C1)** with two change points, and $k_{\max} = 10$ for cases **(C2)** and **(C3)** with four and five change points, respectively. When tuning is involved, we search over candidate sets $\Lambda_1$ and $\Gamma$, and define $\tilde{k}^{\diamond}(\Lambda_1, \Gamma) = \arg\min_{\tilde{k} \in [k_{\max}]} \mathrm{RSS}(\lambda_1^{\diamond}(\tilde{k}), \gamma^{\diamond}(\tilde{k}); \tilde{k})$, where $\big(\lambda_1^{\diamond}(\tilde{k}), \gamma^{\diamond}(\tilde{k})\big)$ is selected via (22).

### 4.2 Simulation results

This section presents representative results covering both independent and temporally dependent designs. We begin with statistical accuracy comparisons. Tables 1 and 2 report results for models (2) and (3) under four-change-point case **(C2)**, respectively. The `DPFree` performs well across a wide range of $\tilde{k}$, and the adaptive version `DPFree.A` generally outperforms fixed-$\tilde{k}$ `DPFree` by adaptively selecting a suitable $\tilde{k}^{\diamond}$. Compared to existing methods, we use `DPFree.A` for fair comparison. Across almost all cases, our approach demonstrates strong performance in estimating the number and locations of change points, with the global refinement effects consistently observed and minimal sensitivity to changes in $n$ or $p$. Our method achieves better detection performance in the first step, possibly due to its decoupled structure that involves at most $2(\tilde{k}+1)$ parameter estimations, thereby reducing the potential for estimation inaccuracies. Additionally, computational efficiency is shown in Figure 4 (runtime of `DPFree` under varying $\tilde{k}$) and Figure 5 (runtime of `DPFree.A`). All scenarios exhibit sub-second runtimes, nearly unaffected by $\tilde{k}$, $n$, or $p$.

## 5 Conclusions

We proposed `DPFree`, a scalable and theoretically grounded approach for multiple change point detection in high-dimensional settings. The method decouples parameter estimation from location search, enabling substantial computational savings. This decoupling

Table 1: Mean change point detection performance in four-change-point case **(C2)** with $n = 800$, under schemes **(M1)** and **(M3)**, based on 100 repetitions.

**Change point detection under independent scheme (M1)**

| | | | $p = 400$ | | | | $p = 500$ | | | | $p = 600$ | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Methods** | $\mathbb{E}[\hat{k}]$ | $\mathbf{A}(\hat{k})$ | $\mathbb{E}[d_{\mathrm{H}}]$ | $\mathbf{Std}(d_{\mathrm{H}})$ | $\mathbb{E}[\hat{k}]$ | $\mathbf{A}(\hat{k})$ | $\mathbb{E}[d_{\mathrm{H}}]$ | $\mathbf{Std}(d_{\mathrm{H}})$ | $\mathbb{E}[\hat{k}]$ | $\mathbf{A}(\hat{k})$ | $\mathbb{E}[d_{\mathrm{H}}]$ | $\mathbf{Std}(d_{\mathrm{H}})$ |
| $\tilde{k} = 4$ | 3.01 | 1 | 0.197\|0.198 | (0.019\|0.020) | 3.00 | 0 | 0.200\|0.200 | (0.003\|0.002) | 3.00 | 0 | 0.200\|0.200 | (0.003\|0.002) |
| $\tilde{k} = 5$ | 4.00 | 100 | 0.004\|0.003 | (0.002\|0.001) | 4.00 | 100 | 0.004\|0.003 | (0.002\|0.001) | 4.00 | 100 | 0.005\|0.003 | (0.002\|0.001) |
| $\tilde{k} = 6$ | 4.00 | 100 | 0.004\|0.003 | (0.002\|0.001) | 4.00 | 100 | 0.004\|0.003 | (0.002\|0.002) | 4.00 | 100 | 0.005\|0.003 | (0.002\|0.002) |
| $\tilde{k} = 7$ | 3.98 | 98 | 0.008\|0.005 | (0.014\|0.014) | 4.00 | 100 | 0.009\|0.003 | (0.008\|0.002) | 3.98 | 98 | 0.015\|0.005 | (0.015\|0.014) |
| $\tilde{k} = 8$ | 4.00 | 100 | 0.003\|0.003 | (0.002\|0.001) | 4.00 | 100 | 0.003\|0.003 | (0.002\|0.002) | 4.00 | 100 | 0.004\|0.003 | (0.002\|0.002) |
| $\tilde{k} = 9$ | 4.00 | 100 | 0.003\|0.003 | (0.002\|0.002) | 4.00 | 100 | 0.003\|0.003 | (0.002\|0.002) | 4.00 | 100 | 0.004\|0.003 | (0.002\|0.002) |
| $\tilde{k} = 10$ | 4.00 | 100 | 0.003\|0.003 | (0.002\|0.001) | 4.00 | 100 | 0.004\|0.003 | (0.002\|0.002) | 4.00 | 100 | 0.004\|0.003 | (0.002\|0.002) |
| `DPFree.A` | 4.01 | 99 | 0.001\|0.000 | (0.006\|0.001) | 4.00 | 100 | 0.000\|0.000 | (0.001\|0.001) | 4.01 | 99 | 0.003\|0.001 | (0.019\|0.001) |
| DC | 4.04 | 97 | 0.004 | (0.008) | 4.06 | 94 | 0.007 | (0.020) | 4.08 | 92 | 0.008 | (0.018) |
| LZZL | 4.59 | 56 | 0.046 | (0.053) | 4.47 | 61 | 0.037 | (0.048) | 4.45 | 65 | 0.035 | (0.048) |
| Insepct | 4.39 | 68 | 0.019 | (0.037) | 4.36 | 70 | 0.017 | (0.039) | 4.69 | 53 | 0.038 | (0.055) |
| EASC | 4.51 | 70 | 0.038 | (0.061) | 4.70 | 57 | 0.041 | (0.058) | 4.55 | 61 | 0.048 | (0.065) |

**Change point detection under temporal dependence scheme (M3)**

| | | | $p = 400$ | | | | $p = 500$ | | | | $p = 600$ | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Methods** | $\mathbb{E}[\hat{k}]$ | $\mathbf{A}(\hat{k})$ | $\mathbb{E}[d_{\mathrm{H}}]$ | $\mathbf{Std}(d_{\mathrm{H}})$ | $\mathbb{E}[\hat{k}]$ | $\mathbf{A}(\hat{k})$ | $\mathbb{E}[d_{\mathrm{H}}]$ | $\mathbf{Std}(d_{\mathrm{H}})$ | $\mathbb{E}[\hat{k}]$ | $\mathbf{A}(\hat{k})$ | $\mathbb{E}[d_{\mathrm{H}}]$ | $\mathbf{Std}(d_{\mathrm{H}})$ |
| $\tilde{k} = 4$ | 3.04 | 4 | 0.192\|0.192 | (0.036\|0.038) | 2.97 | 0 | 0.199\|0.200 | (0.004\|0.003) | 2.90 | 0 | 0.199\|0.200 | (0.004\|0.003) |
| $\tilde{k} = 5$ | 4.00 | 100 | 0.005\|0.004 | (0.003\|0.002) | 4.00 | 100 | 0.006\|0.004 | (0.003\|0.002) | 4.00 | 100 | 0.007\|0.004 | (0.005\|0.002) |
| $\tilde{k} = 6$ | 4.00 | 100 | 0.005\|0.004 | (0.003\|0.002) | 4.00 | 100 | 0.005\|0.004 | (0.003\|0.002) | 4.00 | 100 | 0.006\|0.004 | (0.004\|0.002) |
| $\tilde{k} = 7$ | 4.00 | 100 | 0.015\|0.003 | (0.012\|0.002) | 4.00 | 100 | 0.026\|0.004 | (0.020\|0.002) | 3.98 | 98 | 0.031\|0.006 | (0.022\|0.014) |
| $\tilde{k} = 8$ | 4.00 | 100 | 0.004\|0.003 | (0.002\|0.002) | 4.00 | 100 | 0.005\|0.004 | (0.002\|0.002) | 4.00 | 100 | 0.006\|0.004 | (0.004\|0.002) |
| $\tilde{k} = 9$ | 4.00 | 100 | 0.004\|0.004 | (0.002\|0.002) | 4.00 | 100 | 0.004\|0.004 | (0.002\|0.002) | 4.00 | 100 | 0.004\|0.004 | (0.002\|0.002) |
| $\tilde{k} = 10$ | 4.00 | 100 | 0.004\|0.004 | (0.002\|0.002) | 4.00 | 100 | 0.005\|0.004 | (0.003\|0.002) | 4.00 | 100 | 0.005\|0.005 | (0.003\|0.002) |
| `DPFree.A` | 4.00 | 100 | 0.001\|0.001 | (0.001\|0.001) | 4.00 | 100 | 0.001\|0.001 | (0.002\|0.001) | 4.00 | 100 | 0.001\|0.001 | (0.002\|0.001) |
| DC | 4.07 | 93 | 0.009 | (0.019) | 4.06 | 94 | 0.008 | (0.009) | 4.06 | 94 | 0.010 | (0.016) |
| LZZL | 13.06 | 0 | 0.180 | (0.015) | 13.15 | 0 | 0.182 | (0.012) | 13.12 | 0 | 0.179 | (0.015) |
| Inspect | — | — | — | — | — | — | — | — | — | — | — | — |
| EASC | — | — | — | — | — | — | — | — | — | — | — | — |

Inspect and ESAC perform acceptably only under normal distribution, but tend to produce an excessive number of spurious change points (often in the hundreds) under other settings, and are thus omitted in those cases.

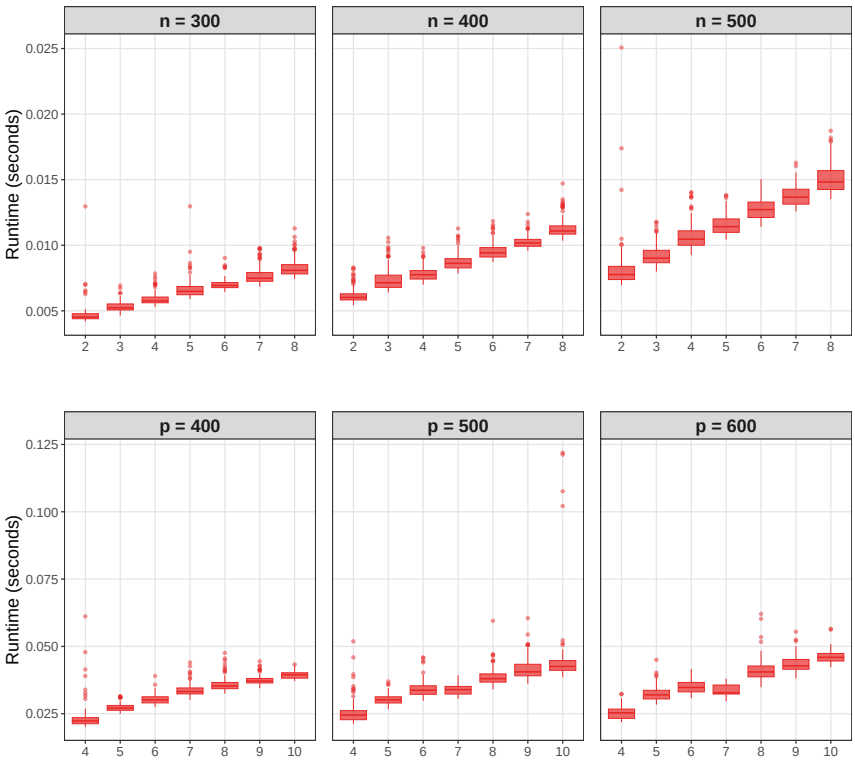


Figure 4: Runtime of the `DPFree` procedure with varying number of initializers $\tilde{k}$ for mean change point detection under scheme **(M1)**. (Top) Case **(C1)** with $p = 400$ and varying $n$; (Bottom) Case **(C2)** with $n = 800$ and varying $p$. The results are based on 100 independent replications.

Table 2: Regression change point performance for four-change-point case **(C2)** with $n = 800$ under schemes **(R1)** and **(R2)**, based on 100 repetitions.

| | | Independent scheme **(R1)** | | | | Dependent scheme **(R2)** | | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | **Methods** | $\mathbb{E}[\hat{k}]$ | $\mathbf{A}(\hat{k})$ | $\mathbb{E}[d_{\mathrm{H}}]$ | $\mathbf{Std}(d_{\mathrm{H}})$ | $\mathbb{E}[\hat{k}]$ | $\mathbf{A}(\hat{k})$ | $\mathbb{E}[d_{\mathrm{H}}]$ | $\mathbf{Std}(d_{\mathrm{H}})$ |
| 200 | `DPFree.A` | 4.04 | 96 | 0.009\|0.003 | (0.025\|0.002) | 4.00 | 100 | 0.005\|0.003 | (0.004\|0.002) |
| | MOSEG.MS | 6.89 | 4 | 0.145\|0.135 | (0.044\|0.051) | 5.91 | 13 | 0.134\|0.120 | (0.060\|0.065) |
| | DPDU | 4.06 | 94 | 0.013\|0.008 | (0.022\|0.019) | 4.10 | 91 | 0.013\|0.006 | (0.018\|0.011) |
| | VPBS | 4.22 | 50 | 0.062 | (0.058) | 4.43 | 55 | 0.064 | (0.063) |
| 400 | `DPFree.A` | 4.02 | 98 | 0.007\|0.004 | (0.015\|0.010) | 4.00 | 100 | 0.004\|0.003 | (0.004\|0.002) |
| | MOSEG.MS | 6.23 | 10 | 0.140\|0.133 | (0.051\|0.056) | 6.53 | 9 | 0.132\|0.122 | (0.055\|0.057) |
| | DPDU | 4.05 | 95 | 0.013\|0.006 | (0.022\|0.016) | 4.14 | 88 | 0.021\|0.014 | (0.035\|0.034) |
| | VPBS | 4.05 | 56 | 0.063 | (0.059) | 4.18 | 56 | 0.060 | (0.057) |
| 800 | `DPFree.A` | 4.01 | 99 | 0.007\|0.003 | (0.011\|0.002) | 4.04 | 96 | 0.007\|0.003 | (0.016\|0.007) |
| | MOSEG.MS | 5.79 | 10 | 0.134\|0.123 | (0.048\|0.052) | 5.39 | 22 | 0.113\|0.106 | (0.063\|0.064) |
| | DPDU | — | — | — | — | — | — | — | — |
| | VPBS | — | — | — | — | — | — | — | — |

Note: Due to excessive computational time, results for DPDU and VPWBS under $(n, p) = (800, 800)$ are omitted.

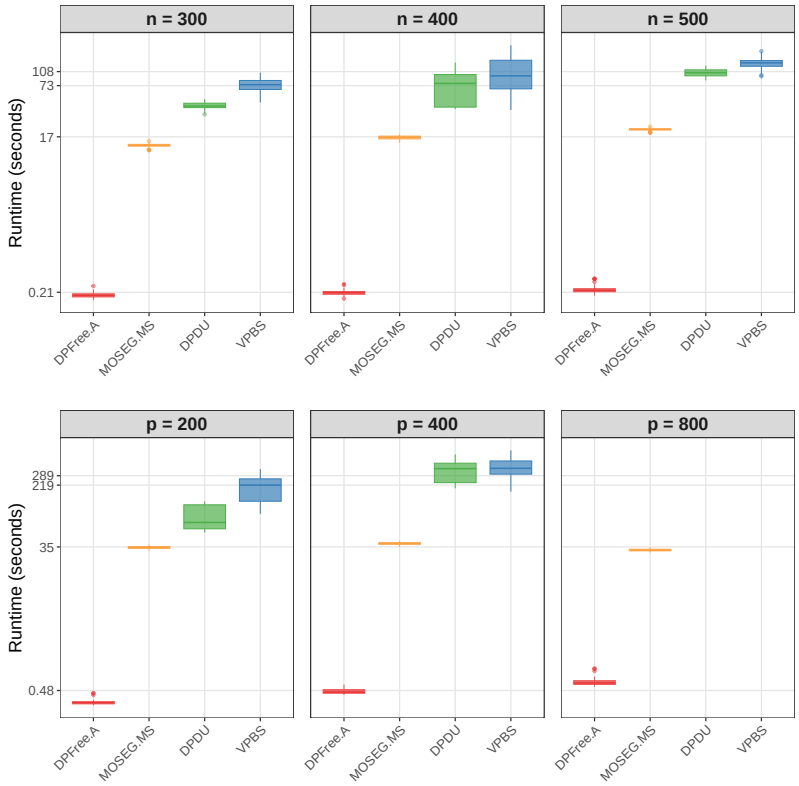


Figure 5: Runtime of `DPFree.A` and competing methods for regression change point detection under scheme **(R1)**. (Top) Case **(C1)** with $p = 400$, varying $n$; (Bottom) Case **(C2)** with $n = 800$, varying $p$. Each result averages 100 replications.The $y$-axis is in the logarithm scale.

strategy is broadly applicable beyond our specific settings, especially when parameter estimation is computationally expensive. Our theory guarantees that, under mild conditions on the initializers, preliminary change point estimators attain a near-optimal convergence rate, while refined change point estimators achieve the optimal rate, along with their limiting distributions. The extension to time series further demonstrates the generality of our framework. These contributions lay a solid foundation for future research, including extensions to nonparametric models, online detection, and functional data applications.

# Supplementary material

The supplementary material is provided to support the main article, covering: (i) real-data applications, (ii) additional simulation results and implementation details, (iii) algorithmic pseudocode with explanations, (iv) further theoretical discussions, (v) model

assumptions for several examples, and (vi) full technical proofs of all stated results.

# References


Auger, I. E. & Lawrence, C. E. (1989), 'Algorithms for the optimal identification of segment neighborhoods', *Bulletin of Mathematical Biology* **51**(1), 39–54.

Bai, P., Safikhani, A. & Michailidis, G. (2023), 'Multiple Change Point Detection in Reduced Rank High Dimensional Vector Autoregressive Models', *Journal of the American Statistical Association* **118**(544), 2776–2792.

Bai, Y. & Safikhani, A. (2023), 'A Unified Framework for Change Point Detection in High-Dimensional Linear Models', *Statistica Sinica* **33**(1), 1721–1748.

Bybee, L. & 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**(11), 1–38.

Chan, N. H., Yau, C. Y. & Zhang, R.-M. (2014), 'Group LASSO for Structural Break Time Series', *Journal of the American Statistical Association* **109**(506), 590–599.

Chen, L., Wang, W. & Wu, W. B. (2022), 'Inference of Breakpoints in High-dimensional Time Series', *Journal of the American Statistical Association* **117**(540), 1951–1963.

Cho, H. (2016), 'Change-point detection in panel data via double CUSUM statistic', *Electronic Journal of Statistics* **10**(2), 2000–2038.

Cho, H., Kley, T. & Li, H. (2025), 'Detection and inference of changes in high-dimensional linear regression with nonsparse structures', *Journal of the Royal Statistical Society Series B: Statistical Methodology* . To appear.

Cho, H. & Owens, D. (2024), 'High-dimensional data segmentation in regression settings permitting temporal dependence and non-Gaussianity', *Electronic Journal of Statistics* **18**(1), 2620–2664.

Csörgö, M. & Horváth, L. (1997), *Limit Theorems in Change-Point Analysis*, Wiley.

Cui, X., Geng, H., Wang, Z. & Zou, C. (2024), 'Robust Estimation of High-Dimensional Linear Regression with Changepoints', *IEEE Transactions on Information Theory* **70**(10), 7297–7319.

Dette, H., Pan, G. & Yang, Q. (2022), 'Estimating a Change Point in a Sequence of Very High-Dimensional Covariance Matrices', *Journal of the American Statistical Association* **117**(537), 444–454.

Eichinger, B. & Kirch, C. (2018), 'A MOSUM procedure for the estimation of multiple random change points', *Bernoulli* **24**(1), 526–564.

Fryzlewicz, P. (2014), 'Wild binary segmentation for multiple change-point detection', *The Annals of Statistics* **42**(6), 2243–2281.

Kaul, A., Jandhyala, V. K. & Fotopoulos, S. B. (2019*a*), 'Detection and estimation of parameters in high dimensional multiple change point regression models via $\ell_0/\ell_1$ regularization and discrete optimization', *arXiv preprint arXiv:1906.04396* .

Kaul, A., Jandhyala, V. K. & Fotopoulos, S. B. (2019*b*), 'An Efficient Two Step Algorithm for High Dimensional Change Point Regression Models Without Grid Search', *Journal of Machine Learning Research* **20**(111), 1–40.

Killick, R., Fearnhead, P. & Eckley, I. A. (2012), 'Optimal Detection of Changepoints With a Linear Computational Cost', *Journal of the American Statistical Association* **107**(500), 1590–1598.

Kovács, S., Bühlmann, P., Li, H. & Munk, A. (2023), 'Seeded binary segmentation: A general methodology for fast and optimal changepoint detection', *Biometrika* **110**(1), 249–256.

Lee, S., Liao, Y., Seo, M. H. & Shin, Y. (2018), 'Oracle Estimation of a Change Point in High-Dimensional Quantile Regression', *Journal of the American Statistical Association* **113**(523), 1184–1194.

Leonardi, F. & Bühlmann, P. (2016), 'Computationally efficient change point detection for high-dimensional regression', *arXiv preprint arXiv:1601.03704* .

Li, W., Wang, D. & Rinaldo, A. (2023), Divide and Conquer Dynamic Programming: An Almost Linear Time Change Point Detection Methodology in High Dimensions, *in* 'Proceedings of the 40th International Conference on Machine Learning', PMLR, pp. 20065–20148.

Liu, B., Zhang, X. & Liu, Y. (2021), 'Simultaneous Change Point Inference and Structure Recovery for High Dimensional Gaussian Graphical Models', *Journal of Machine Learning Research* **22**(274), 1–62.

Liu, B., Zhang, X. & Liu, Y. (2026), 'Simultaneous Change Point Detection and Identification for High Dimensional Linear Models', *Statistica Sinica* **36**. To appear.

Liu, B., Zhou, C., Zhang, X. & Liu, Y. (2020), 'A unified data-adaptive framework for high dimensional change point detection', *Journal of the Royal Statistical Society Series B: Statistical Methodology* **82**(4), 933–963.

Moen, P. A. J., Glad, I. K. & Tveten, M. (2024), 'Efficient sparsity adaptive changepoint estimation', *Electronic Journal of Statistics* **18**(2), 3975–4038.

Rinaldo, A., Wang, D., Wen, Q., Willett, R. & Yu, Y. (2021), Localizing Changes in High-Dimensional Regression Models, *in* 'Proceedings of The 24th International Conference on Artificial Intelligence and Statistics', PMLR, pp. 2089–2097.

Roy, S., Atchadé, Y. & Michailidis, G. (2017), 'Change Point Estimation in High Dimensional

Markov Random-Field Models', *Journal of the Royal Statistical Society Series B: Statistical Methodology* **79**(4), 1187–1206.

Shi, L., Wang, G. & Zou, C. (2024), 'Low-Rank Matrix Estimation in the Presence of Change-Points', *Journal of Machine Learning Research* **25**(220), 1–71.

Truong, C., Oudre, L. & Vayatis, N. (2020), 'Selective review of offline change point detection methods', *Signal Processing* **167**, 107299.

van der Vaart, A. W. & Wellner, J. A. (2023), *Weak Convergence and Empirical Processes: With Applications to Statistics*, Springer.

Wang, D., Zhao, Z., Lin, K. Z. & Willett, R. (2021), 'Statistically and computationally efficient change point localization in regression settings', *Journal of Machine Learning Research* **22**(248), 1–46.

Wang, R., Zhu, C., Volgushev, S. & Shao, X. (2022), 'Inference for change points in high-dimensional data via selfnormalization', *The Annals of Statistics* **50**(2), 781–806.

Wang, T. & Samworth, R. J. (2018), 'High dimensional change point estimation via sparse projection', *Journal of the Royal Statistical Society Series B: Statistical Methodology* **80**(1), 57–83.

Wang, X., Liu, B., Zhang, X. & Liu, Y. (2025), 'Efficient Multiple Change Point Detection and Localization For High-dimensional Quantile Regression with Heteroscedasticity', *Journal of the American Statistical Association* **120**(550), 976–989.

Wong, K. C., Li, Z. & Tewari, A. (2020), 'Lasso guarantees for $\beta$-mixing heavy-tailed time series', *The Annals of Statistics* **48**(2), 1124–1142.

Wu, W. B. (2011), 'Asymptotic theory for stationary processes', *Statistics and Its Interface* **4**(2), 207–226.

Xu, H., Wang, D., Zhao, Z. & Yu, Y. (2024), 'Change-point inference in high-dimensional regression models under temporal dependence', *The Annals of Statistics* **52**(3), 999–1026.

Yang, Z., Zhang, L., Sun, S. & Liu, B. (2025), 'Robust change point detection for high-dimensional linear models with tolerance for outliers and heavy tails', *Canadian Journal of Statistics* **53**(1), e11826.

Yu, M. & Chen, X. (2021), 'Finite Sample Change Point Inference and Identification for High-Dimensional Mean Vectors', *Journal of the Royal Statistical Society Series B: Statistical Methodology* **83**(2), 247–270.

Zhang, Y., Wang, R. & Shao, X. (2022), 'Adaptive Inference for Change Points in High-Dimensional Data', *Journal of the American Statistical Association* **117**(540), 1751–1762.

Zou, C., Wang, G. & Li, R. (2020), 'Consistent selection of the number of change-points via sample-splitting', *The Annals of Statistics* **48**(1), 413–439.