Dynamic Interpretable Change Point Detection for Physiological Data Analysis

Jennifer Yu[*] JENNIFER.YU@MAIL.UTORONTO.CA
Tina Behrouzi[*] TINA.BEHROUZI@MAIL.UTORONTO.CA
Kopal Garg[*] KOPAL.GARG@MAIL.UTORONTO.CA
Anna Goldenberg ANNA.GOLDENBERG@UTORONTO.CA
Sana Tonekaboni SANA.TONEKABONI@MAIL.UTORONTO.CA
University of Toronto, Toronto, Ontario, Canada
The Hospital for Sick Children, Toronto, Ontario, Canada
Vector Institute, Toronto, Ontario, Canada

Abstract

Identifying change points (CPs) in time series is crucial to guide better decision-making in healthcare, and facilitating timely responses to potential risks or opportunities. In maternal health, monitoring health signals in pregnant women allows healthcare providers to promptly respond to complications like preeclampsia or enhance delivery time detection, improving overall maternal care. Existing Change Point Detection (CPD) methods often fail to generalize effectively due to diverse underlying changes that can cause a CP. We propose **Ti**me **Va**rying **CPD** (TiVaCPD), a change point detection method that captures different types of changes in the underlying distribution of multidimensional data. It combines a dynamic window MMD test with a graphical Lasso estimator of feature covariance to measure both changes in the joint distribution of the observations as well as changes in feature dynamics. TiVaCPD generates a unifying CP score by evaluating the relative similarity of the statistical tests. Additionally, TiVaCPD score enhances interpretability by offering insight into the underlying causes of CPs through a detailed analysis of feature dynamics, which is especially valuable in healthcare applications. We evaluate the performance of TiVaCPD on both simulated and real-world data, showing that it can outperform state-of-the-art methods. We further demonstrate the appliance of TiVaCPD in a pregnancy-related case study, showcasing the joint shifts in physiological signals that facilitate the detection of delivery time.

Keywords: Change point detection, Time series, Dynamic window, Health monitoring

1. Introduction

Identifying change points (CP) in healthcare time series data is of significant importance that enables the detection of changes in health state, plays a pivotal role in risk management and enhancing clinical decision-making (Truong et al., 2018; Aminikhanghahi and Cook, 2017), and more. Given the exponential growth in healthcare time-series data (Yang et al., 2007), and the increase in popularity of technologies such as wearable devices for recording physiological signals, the need for automated change point detection (CPD) methods has significantly increased. CPD enables prompt identification of variations in signals like heart rate or oxygen saturation levels, which results in the identification of critical changes in health state and can ultimately be used to alert for potential health concerns.

Many existing CPD methods often overlook the underlying variability in CP properties and its root causes, limiting their effectiveness in handling time-series with complex change dynamics. CPs are typically characterized as shifts in the distribution of measurements over time. Alternatively, they can result from changes in the correlation structure between features. While the former is well-studied, the latter is also important in various applications. For instance, in physiological signals, an increasing negative correlation between Heart Rate Variability (HRV) and Heart Rate (HR) measurements can signal increased nervous system activity caused by stress or anxiety, requiring medical attention (Sacha, 2014). Methods that solely focus on detecting changes in marginal distributions may fail to identify such scenarios.

[*] These authors contributed equally

In this paper, we propose a statistical CPD method called TiVaCPD that captures different types of CPs in time series without the need for labeled instances of change. TiVaCPD offers a non-parametric solution that does not require any distributional assumption of the generative process, and can therefore generalize to various scenarios. TiVaCPD assigns a CP score at every time-point that measures 2 types of shift: 1) change in the correlation of features and 2) change in the underlying distribution of the time-series features (Figure 1). Each part of the score is interpretable, allowing us to characterize and classify CPs with similar underlying properties and understand the cause of the change. To identify changes in the joint distribution of features over time, TiVaCPD employs a dynamic network inference method to acquire sparse time-varying precision matrices for feature interactions to detect changes in feature correlation patterns. To identify changes in the probability distributions of adjacent windows, TiVaCPD builds on the theory of non-parametric two-sample MMD tests (Schrab et al., 2021), and augments existing methods by dynamically adapting the window size to accommodate variable state lengths between CPs. We show that this alleviates major issues with fixed size windows. Namely having small windows that can weaken statistical power, and larger ones that may mix different distributions. Finally, different components of the TiVaCPD score are combined to detect CPs using an ensemble method that adaptively assigns weights to scores based on their dissimilarity, placing greater emphasis on scores that capture changes not detected by other components. We evaluate the performance of TiVaCPD across simulated and real-life datasets, comparing it to state-of-the-art CPD methods and show that our method outperforms competitors in all datasets. We also study the application of CPD in wearable data for a cohort of pregnant individuals. This study explores the potential utility of daily wearable data for aiding clinicians in identifying significant pregnancy-related events.

2. Related Work

There is abundant literature on CPD methods (Truong et al., 2018; Aminikhanghahi and Cook, 2017; Reeves et al., 2007). CPD methods consider a time-series to be a collection of random variables with abrupt changes in distributional properties over time. Most of these methods are parametric (Yamanishi and Takeuchi, 2002; Kawahara et al., 2007) and involve estimating the underlying probability density function of the signal, which limits detection to certain types of distributions and is usually computationally expensive. Non-parametric methods (Chang et al., 2019; Cheng et al., 2020; Matteson and James, 2014) are employed when time-series dynamics are hard to model, and prior data distribution assumptions are not feasible. An optimal transport-based approach by Cheng et al. (2020) conducts two-sample Wasserstein tests between cumulative distributions of contiguous subsequences, utilizing fixed-size sliding windows. However, relying on local maxima of this statistic can lead to a higher false positive rate. Additionally, this method projects data to one dimension and uses mean statistics, potentially reducing detection power. In contrast, Roerich Hushchyn and Ustyuzhanin (2021) adopts classical machine learning and regression models to detect distribution changes. Deep learning-based non-parametric methods have gained popularity, driven by the increasing availability of data. For instance, Time-Invariant Representation (TIRE) (De Ryck et al., 2021) is an autoencoder-based CPD approach that learns a partially time-invariant representation of time series and computes CPs using a dissimilarity measure. Another method, $TS-CP^2$ (Deldari et al., 2021a), employs contrastive learning with representations from temporal convolutional networks for CP detection. Some deep learning-based CPD approaches incorporate kernel functions (Li et al., 2015) for greater flexibility in representing density functions. One such method, KLCPD (Chang et al., 2019) uses deep generative models to enhance the test power of the kernel two-sample MMD test statistic (Gretton et al., 2007). It overcomes the limitations of prior kernel-based CPD methods by removing the need of a fixed number of CPs or prior knowledge of a reference or training set for kernel calibration. However, its performance depends on the choice of kernel and kernel bandwidths. Deep learning-based CPD methods lack interpretability, hindering our understanding of their predictions. Current methods also struggle to capture changes in correlation patterns in evolving multivariate time series. To address this, Gibberd and Nelson (2015) introduced GraphTime, a Group-Fused Graphical Lasso estimator for CP estimation in time series dependency structures. This method is suitable for detecting abrupt changes but produces excessive false positives for gradual CPs due to its piece-wise constant graph topology. Our method not only offers interpretability by quantifying the magnitude and direction of changes in correlation between variables

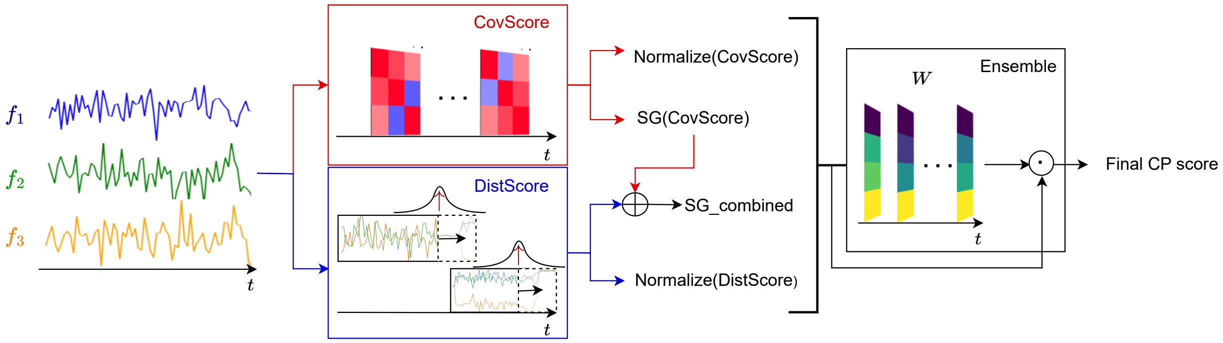

Figure 1: Overview of TiVaCPD that illustrates the score generation process and all the components including DistScore, CovScore, and the ensemble weights W. SG stands for Savitzky-Golay filter.

that trigger a specific CP but also utilizes a novel ensemble technique to effectively aggregate unsupervised CP scores from various statistical tests, ensuring better accuracy.

3. Method

3.1. Problem Formulation

Consider a multivariate time-series sample $\mathbf{X} \in \mathbb{R}^{d \times T}$ to be a sequence of random variables $[X_1, X_2, ..., X_T]$ with d indicating the number of features and T representing the total number of measurements over time. To identify change points in time steps of a data sample, a score $S[t]$, $\forall t \in [T]$ is estimated for each time step that measures the amount of change in the underlying generative process of the data.

3.2. Our CPD Algorithm - TiVaCPD

In this section, we introduce our CPD algorithm called Time Variable Change Point Detection (TiVaCPD). CPD detects 2 types of change in the underlying distribution over time: 1) change in correlation between features, i.e. change in the joint distribution of features over time 2) change in distribution of measurements over time, i.e. the marginal distribution of the observations. In the rest of the section we introduce TiVaCPDscore, explain how it captures a variety of CP types, and demonstrate how to interpret the score to better understand the CPs.

Detecting changes in feature correlation A CP can be caused by a change in the correlation between features. This can be identified through a change in the covariance of the joint distribution of the feature network. The evolving dynamics of features can be

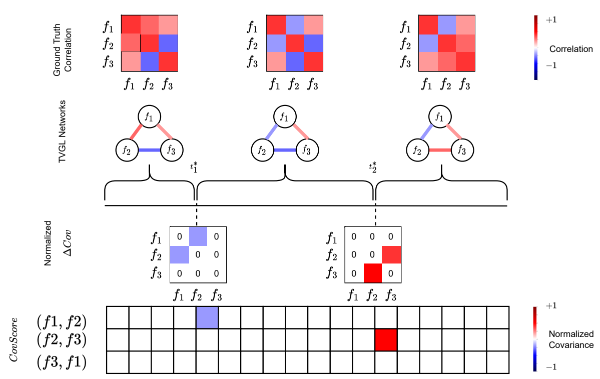

Figure 2: Temporal changes in 3 feature interactions using a correlation matrices heatmap. Each row represents correlation shifts between feature pairs, with colours indicating their direction.

modeled using graphical models, i.e. at every time point, the interactions of features can be modeled as a graph network, with nodes and links corresponding to each feature and correlation between sets of variables, respectively (Figure 2). However, to detect CPs, we need to calculate the covariance matrix and therefore the graphical network at every time step, which is computationally challenging. To overcome this, we use a Graphical Lasso estimator to estimate the sparse inverse covariance matrix (precision matrix) of the multivariate time-series with time-varying structures. The precision matrix aids in numerical stabilization, ensuring that the resulting matrix is positive semi-definite. This helps in better representing the conditional dependence among the features. We use the dynamic extension called the Time-Varying Graphical Lasso (TVGL) (Hallac et al., 2017) that can model the varying covariance over time. Additional details on TVGL can be found in Appendix D.

Using the inverse covariance matrices estimated at every time step P_t, we can estimate the partial correlation of each pair of features $\hat{P}_t$ as $-\frac{P_t(i,j)}{\sqrt{P_t(i,i)P_t(j,j)}}$ for $i \neq j$ and $P_t(i,i)$ otherwise. The absolute value of the difference of consecutive correlations over time quantifies change points caused by the change in the features' dynamics as a score we call **CovScore**.

Detecting shift in distribution Detecting changes in feature correlation is an important indication of a CP, but it would not be enough to explain all changes. In some cases, the distribution of all features can undergo concurrent shifts over time, without any change in feature interaction. To address such cases, we introduce the **DistScore**, as elaborated in the following. Assuming in a time series sample $\mathbf{X}$, each X_t is generated from a joint probability distribution $p_t(\cdot)$, a CP can occur at time t^* if observations after t^* are generated from a different distribution. To compare the probability distributions of adjacent windows, we employ a non-parametric two-sample testing procedure called MMD Aggregate (MMDAgg), introduced in Schrab et al. (2021). Kernel-based MMD tests serve as a measure between two probability distributions. With the statistical test threshold α, if the null hypothesis H_0, is rejected, the time-series may be partitioned by a CP at t^*, signifying that measurements in the past window of size Δ^- ($X_{t^*-\Delta^-:t^*}$) come from a different distribution than measurements in $X_{t^*:t^*+\Delta^+}$. MMDAgg aggregates multiple MMD tests using different kernel bandwidths, ensuring maximized test power over the collection of kernels used and eliminating the need for data splitting or arbitrary kernel selection. One shortcoming of MMDAgg is its lack of consideration for dynamic intervals of change points. Considering constant values for Δ^- can lead to increased false alarms when applied to real-world data without a predefined schedule. We propose to dynamically establish the window size based on the presence of CPs (Figure 3). Let Δ^- represent the size of the dynamic window of data points from the last estimated CP ($\tilde{t}$), up until the current time point (t). Starting with a constant Δ^+ and a small Δ^- window, the length of the running window Δ increases with each new observation until a new CP occurs, according to the MMD test. If a significant change in distribution is not detected by the MMD test, i.e. the MMD score is smaller than a pre-defined threshold ϵ, the two sub-sequences are combined and compared against the next sub-sequence in the series. Our dynamic windowing method eliminates the need for repetitive fixed-window comparisons and utilizes a growing sample set for the MMD test. In the evaluation section, we also demonstrate how dynamic windowing can significantly improve the performance of the statistical test in identifying CPs.

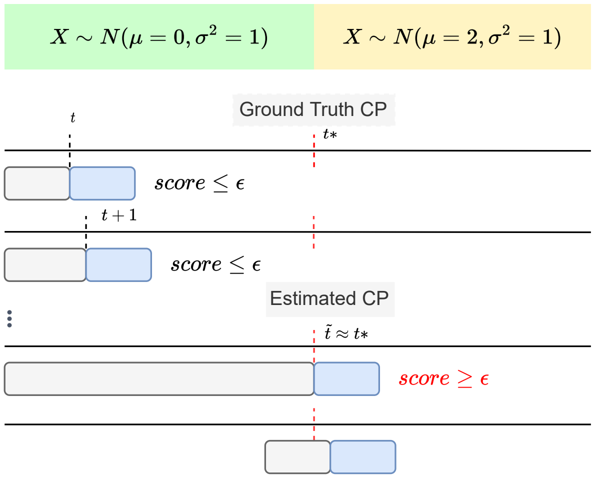

Figure 3: Dynamic window procedure. The expanding window (gray) and fixed-size future observation window (blue) enlarge to include more samples from the generative distribution as the algorithm proceeds. Once a CP is detected at $\tilde{t}$, the window size reverts to its initial size.

For determining the final CP score, we need to meaningfully ensemble the MMD score with the cor-

Algorithm 1 Estimating TiVaCPD score

```
Input: $X$, $\alpha$ (Statistical threshold), $\Delta^-$ (Initial past window size), $\Delta^+$ (Future window size), $\epsilon$ (Score threshold)
Output: $Y_S$ (TiVaCPDscore)
$\Delta = \Delta^-$   // Size of the running window
$P = Conv^{-1}(X)$   // Sparse inverse covariance
$\hat{P}$ Partial Correlation
for all $t \in [1, ..., T]$ do
    $S[t]$ = MMDAgg($X_{t-\Delta:t}$, $X_{t:t+\Delta^+}$, $\alpha$)
    if $S[t] \geq \epsilon$ then
        DistScore[t]= $S[t]$   &   $\Delta = \Delta^-$
    else
        DistScore[t]= 0   &   $\Delta = \Delta + 1$
    end if
    CovScore[t] = $\begin{cases} \sum|\hat{P}[t] - \hat{P}[t-1]| & t > 0 \\ 0 & t = 0 \end{cases}$
end for
CovScore = Smoothing(CovScore)
SG$_{Combined}$ = SG(|CovScore|+|DistScore|)
All_Score = [SG$_{Combined}$, Normalize(DistScore), Normalize(CovScore), CovScore]
for all $t\_win \in [T]$ do
    $W = \sum_j mean$(|All_Score − All _Score$_j$|)
    $\hat{W}$: Update $W$ based on #CPs difference
    $S = \hat{W} \cdot$ All_Score
end for
return $Y_S$
```

relation change score, which is challenging because the correlation score is bounded while MMD is a positive unbounded score. Hence, TiVaCPD incorporates kernel normalization in the MMDAgg algorithm. We use a generalization of Cosine normalization (Ah-Pine, 2010) to normalize our kernels so as to have a similarity index. For a given kernel function, $K^{z=1}(x,y)$ represents the normalized kernel of order z. We use the generalized mean with exponent $z = 1$ (arithmetic mean), which means $K^{z=1}(x,y) = \frac{K(x,y)}{M^{z=1}(K(x,x),K(y,y))}$, where $M^{z=1}(a_{i=1}^{p}) = \frac{1}{p}\sum_{i=1}^{p}(a_i^z)$. This normalization technique projects the objects from the feature space to a unit hypersphere and guarantees $|K^{z=1}(x,y)| \leq 1$.

Ensemble CP Score TiVaCPD identifies change points based on the **CovScore** and **DistScore** as defined and presented in Algorithm 1, using an ensemble method that utilizes the score differences to highlight the scores that contribute the most to representing the CPs. We use four score variants to generate dynamic dissimilarity weight W to effectively aggregate the scores into a unified score (As shown in Figure 1). The four scores used are as follows: *a*) *b*) Normalize(CovScore) and Normalize(DistScore): standardized CP scores using z-score normalization to bring them into the same scale, *c*) Smoothing(CovScore): Since CovScore is sensitive to small distributional changes that can lead to false change point detection, we mitigate the risk of detecting spurious CPs by applying Savitzky-Golay (SG) smoothing filter (Press and Teukolsky, 1990), which is a widely used method for smoothing patterns and reducing noise in time series data. *d*) SG$_{Combined}$: We also apply the filter to the sum of filtered CovScore and DistScore, which effectively reduces noise and improves the performance of CPD. The ensemble approach utilizes a weighted average of the aforementioned four scores. The importance weight W is calculated based on the mean absolute difference between scores. This approach assigns a higher weight to scores that exhibit greater dissimilarity, enhancing the detection of CPs that might be overlooked by other scoring methods. The weights are calculated for each time-point window, as the distribution of scores' importance may vary over time. To reduce the number of false positives, we modify the weights to $\hat{W}$. The detail on how $\hat{W}$ is calculated is in Appendix D. Finally, to locate the exact time of the CPs, we identify peaks in ensemble scores by searching for local maxima with a threshold to reduce false positives created by noise.

Understanding change points and interpreting TiVaCPD score TiVaCPD offers valuable insights into the underlying nature of the observed change points. In real-world applications such as the healthcare domain, understanding the cause for a change point has significant importance, as it may represent very different circumstances. For instance, a change in patient state caused by a shift in the distribution of blood pressure has different clinical implications compared to when heart rate and blood pressure measures change from being negatively correlated or uncorrelated to positively correlated. Different components of TiVaCPD score, and more specifically analyzing the generated correlation graphs, provide a detailed analysis of the feature dynamics at each time step. This information allows for the categorization of CPs, thereby enhancing interpretability, as shown in Figure 4. This figure presents a multivariate time series sample showcasing TiVaCPD results, featuring CovScore, DistScore, and a weighted ensemble score. In addition, CovScore's heatmap illustrates the feature pairs that caused a CP, and TiVaCPD identifies the direction of correlation change at each CP. The first two CPs (CP_1 and CP_2) are caused by changes in the correlation between features 0 and 1, where CP_1 corresponds to a positive change in correlation and CP_2 corresponds to a negative change in correlation. CP_3 is caused by changes in the mean between features 0 and 1 where DistScore is the highest and CP_4 is caused by changes in a combination of variance, correlation, and mean.

4. Experiments

4.1. Datasets

We evaluate the performance of our method compared to multiple baselines on the following datasets:

Simulated Data: We created 4 different datasets to simulate different types of CPs for which we know the ground truth cause. In all datasets, each time series sample $X \in \mathbb{R}^{d \times T}$ consists of $d = 3$ features, and X is sampled from a d-dimensional Gaussian distribution $N_d(\mu_i, \sigma_i^2)$. The time at which each change point occurs is randomly chosen within a range of 50 to 100 time points. The properties of the 4 datasets are as follows and the description and results on another real-world dataset are in Appendix A:

- **Jumping Mean**: The variance is assumed to be constant ($\sigma^2 = 1$) over time and across all features. The ground-truth CPs correspond to abrupt jumps in the mean that can happen independently in any of the features.
- **Changing Variance**: All three features are generated with constant mean $\mu = 1$, but their distribution variance changes over time. CPs are indicated as time points with changes in σ^2.
- **Changing Correlation**: This data set consists of a multivariate time-series generated with constant σ^2 and μ. The correlation between features (1, 2) and (2, 3) can change between {-1, 0, 1}. Here, the ground truth CPs correspond to points in time where the correlation ρ_t changes.
- **Arbitrary CPs**: This dataset consists of a multivariate time series with CPs due to varying μ, or σ^2, or correlations between pairs of variables, resulting in a mixture of CPs scattered over time.

Human Activity Recognition (HAR) (Anguita et al., 2013): We use a subset of HAR which includes periods of 6 activities such as normal walking and standing. These activities are measured with 3-axial linear acceleration and angular velocity sensors, for a total of 6 features. The ground-truth CPs are labeled as the transitions between activities.

Pregnancy study (BUMP) (Goodday et al., 2022): Better Understanding of Metamorphosis of Pregnancy (BUMP) is a longitudinal feasibility study aimed to gain a deeper understanding of the pregnancy experience using digital tools, including the Oura ring, Garmin watch, and Bodyport scale alongside study apps to capture various physiological and psychological symptoms.

4.2. Baseline Methods

We compare the performance of TiVaCPD with a number of CPD methods commonly used in the literature.[1] The selected approaches include those that measure change in distribution (KL-CPD and Roerich) and those that focus on the graphical structure of features over time (GraphTime and TIRE). More detail on all methods is provided in Appendix E and Appendix H, along with a description of best parameters and sensitivity to hyperparameters. All hyper-parameters are determined based on a random search over 10% of the datasets (more details on best parameters and sensitivity to hyperparameter change are provided in Appendix H). We utilized the SciPy peak detection method to identify the CP location. This approach was consistently applied to all models to compute precision, recall, and F1 scores, ensuring a fair basis for comparison.

4.3. Evaluation

Tables 1-3 show the performance of TiVaCPD and all baselines on detecting CP locations measured by F_1 scores, Precision, and Recall. To estimate performance metrics, we define a margin of error M for the exact location of CP, which is common practice in the CPD literature (van den Burg and Williams, 2020; Deldari et al., 2021b). Given a user-defined margin of error, $M > 0$, an estimated CP is a True Positive (TP) if the distance between the ground truth (t^*) and the estimated CP ($\widetilde{t}$) is smaller than the margin, i.e. $|t^* - \widetilde{t}| \leq M$. As explained in Figure 5, if an estimated CP falls outside the margin, then it is considered False Positive (FP), i.e. $\widetilde{t} \notin [t^* - M/2, t^* + M/2]$. We show the impact of margin values $M = \{5, 10\}$ on the performance of all baselines in Appendix B. TiVaCPD outperforms all other baselines, achieving

[1] The implementation of the baselines was done using publicly released source code by the authors.

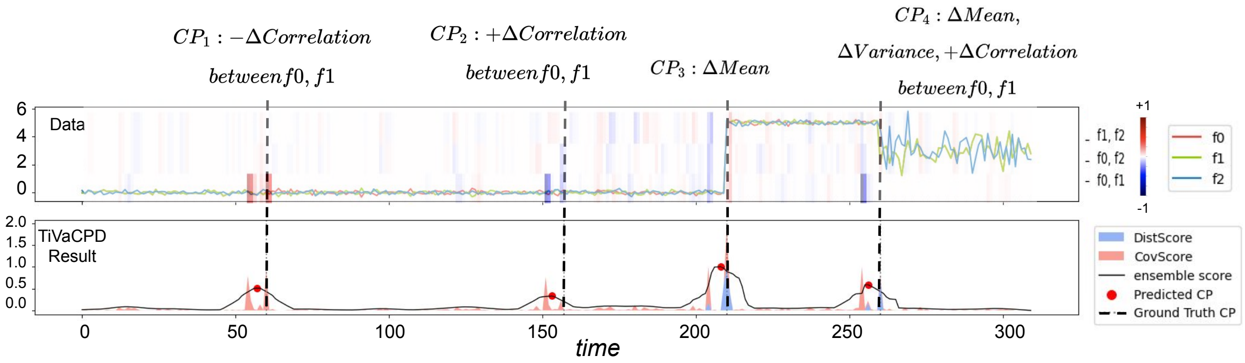

Figure 4: This example shows a simulated time-series with various CPs. Row 1: raw data with heatmap interpretation (blue: negative correlation change, red: positive correlation change). Rows 2: TiVaCPD score, along with a breakdown of the DistScore and CovScore. The black dotted lines represent the ground truth CPs and the red dots are predicted CPs. Please see the results of other baselines in the appendix Figure 8.

Table 1: Performance of CPD methods on the Jumping Mean and Changing Variance with $M = 5$.

	Jumping Mean			Changing Variance		
Method	Precision	Recall	F1	Precision	Recall	F1
KL-CPD	0.84 (0.24)	0.42 (0.14)	0.49 (0.15)	0.13 (0.02)	0.72 (0.13)	0.21 (0.03)
Roerich	0.15 (0.01)	0.97 (0.05)	0.26 (0.02)	0.16 (0.02)	0.92 (0.12)	0.27 (0.03)
GraphTime	0.38 (0.08)	0.90 (0.17)	0.50 (0.10)	0.10 (0.03)	1.00 (0.00)	0.17 (0.03)
TIRE	0.9 (0.12)	0.87 (0.17)	0.88 (0.16)	0.30 (0.34)	0.08 (0.08)	0.12 (0.14)
TiVaCPD	1.00 (0.00)	0.90 (0.12)	**0.93 (0.08)**	0.8 (0.16)	0.85 (0.12)	**0.82 (0.15)**

Table 2: Performance of CPD methods on the Changing Correlations and Arbitrary CPs with $M = 5$.

	Changing Correlations			Arbitrary CPs		
Method	Precision	Recall	F1	Precision	Recall	F1
KL-CPD	0.11 (0.03)	0.75 (0.15)	0.19 (0.06)	0.72 (0.17)	0.56 (0.09)	0.63 (0.12)
Roerich	0.12 (0.03)	0.77 (0.10)	0.22 (0.05)	0.17 (0.02)	0.95 (0.08)	0.28 (0.02)
GraphTime	0.13 (0.03)	1.00 (0.00)	0.24 (0.06)	0.21 (0.07)	0.88 (0.13)	0.34 (0.10)
TIRE	0.14 (0.13)	0.10 (0.09)	0.12 (0.11)	1.00 (0.00)	0.55 (0.21)	0.67 (0.18)
TiVaCPD	0.33 (0.04)	0.88 (0.09)	**0.47 (0.06)**	0.93 (0.08)	0.77 (0.15)	**0.83 (0.11)**

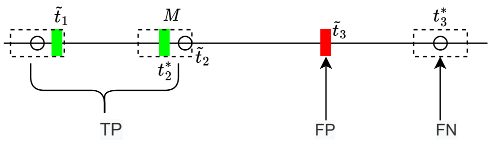

Figure 5: Definition of margin of error. The ground truth CPs (t^*) are shown as circles and $\widetilde{t}$ are the detected CPs. The prediction is a true positive if $\widetilde{t}$ falls within the margin around the ground truth.

Table 3: HAR dataset performance comparison

Method	Precision	Recall	F1(M=5)
KL-CPD	0.66 (0.11)	0.20 (0.03)	0.30 (0.04)
Roerich	0.69 (0.15)	0.11 (0.03)	0.18 (0.05)
GraphTime	0.04 (0.00)	0.96 (0.02)	0.08 (0.01)
TIRE	0.52 (0.19)	0.14 (0.05)	0.22 (0.08)
TiVaCPD	0.72 (0.06)	0.48 (0.06)	**0.58 (0.06)**

5% to 70% increase in F1 score. This difference is particularly noticeable in simulated data scenarios involving changes in variance and correlation, where other methods exhibit high Recall scores but low Precision. This highlights that in these cases, SOTA methods tend to detect numerous false change points due to their over-sensitivity to data variation. Similar results are observed in the HAR dataset.

4.4. Interpreting TiVaCPD scores

Figure 4 shows a graphical representation of TiVaCPD for a sample (row 1) from the Arbitrary CP dataset, and demonstrates how different methods generate scores for CPs. The different components of TiVaCPD are shown in rows 2 and 3, and highlight the types of CPs they identify. The heatmaps of CovScore offer interpretability, helping identify which pair of features experienced a change in correlation and the direction of that change. This visualization enhances our understanding of how TiVaCPD analyzes and interprets CPs in the data.

4.5. Ablation Study

We evaluate the importance of the different components TiVaCPD namely the CovScore, DistScore, and the dynamic windowing, through an ablation study in the Arbitrary dataset. First, we show that the dynamic windowing considerably improves the performance of TiVaCPD score, as shown by comparison of the TiVaCPD and without dynamic window columns in Figure 7. Dynamic windowing also improves the performance of the MMD test alone, as presented in the without Covscore column. Applying smoothing results in a 0.06 increase in the F1 score. Nevertheless, our method outperforms other baselines even without the addition of smoothing. Most importantly, the results in Figure 7 show that by carefully ensembling the scores for different types of CP, TiVaCPD better balances Precision and Recall. Overall, the results show that for various types of CPs that can occur, all parts of the method play a role in achieving its overall high performance.

4.6. Case Study - BUMP

In the BUMP study, identifying a significant change in the physiology of the mother during pregnancy through wearable measurements is of great importance. TiVaCPD score can be used to detect such changes, for instance, readiness for delivery, as this knowledge can help in enhancing the accuracy of delivery time prediction. Figure 6 shows an example of how TiVaCPD effectively identifies CPs, especially the ones associated with the event of delivery using seven daily features: total sleep time, total REM sleep time, total restless sleep time, deviation of skin temperature from the long-term average, total daytime resting time, metabolic-equivalent minutes (MET mins) during medium-intensity activities, and MET mins during high-intensity activities Erickson et al. (2023). The delivery date is marked as a black vertical line as the ground truth. The overlaid interpretation matrix shows changes in pairwise feature correlations. A value of -1 (blue) shows a shift from a positive to a negative correlation between variables, and a value of 1 (red) indicates the opposite, from a negative to a positive correlation. Notably, the correlation between total restless sleep time and total daytime rest time changes more frequently than other pairs, suggesting that sleep quality is a key indicator for the events detected in this study.

We evaluated the performance of our TiVaCPD in detecting changes in the delivery date using the F1 score on the final ten days leading up to the delivery. This timeframe provides a focused evaluation of alterations in the delivery date and ensures the F1 score's representativeness in computing performance. Our TiVaCPD scored **0.46** outperforming other methods: Roerich: 0.10, GraphTime: 0.21, TIRE: 0.0, and KL-CPD: 0.0. For the complete table please refer to Appendix G. These results demonstrate that our model is significantly more accurate in detecting delivery date changes compared to other methods.

With TiVaCPD , clinicians are empowered to discern the specific features linked to each detected CP. This identification not only helps with a deeper understanding of what caused these changes but also serves as a resource for clinical decision-making and improves actionability. While initially focused on delivery dates, the TiVaCPD method can also detect other significant events like lifestyle changes. However, evaluating those CPs is challenging and requires an extensive study with clinicians which is the next step of our work. This flexibility makes TiVaCPD a valuable tool for monitoring patients using wearable devices, not only for pregnancy check-ins but also for various aspects of healthcare.

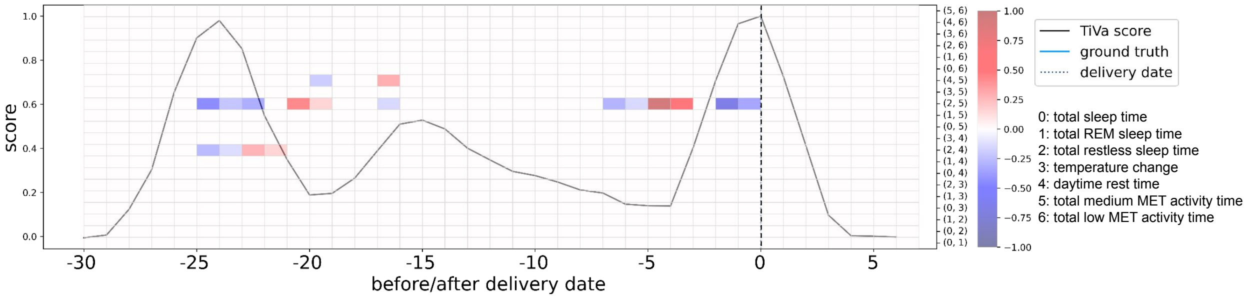

Figure 6: An example of CPD for a subject in the BUMP study. The paired numbers on the right side of the image represent feature numbers, while the color of the bars illustrates their partial correlation.

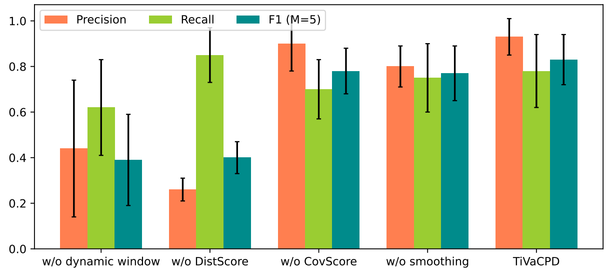

Figure 7: Ablation study on Arbitrary simulated dataset (M=5). wo stands for without.

5. Discussion

In this paper, we introduce TiVaCPD, a novel CPD method for detecting and characterizing various types of CPs in time-series data. By capturing changes in feature distribution, dynamics, and correlation networks, TiVaCPD provides valuable insights into the underlying causes of CPs, enhancing interpretability for end users. This is particularly crucial in domains like healthcare, where the type of CP significantly influences downstream decision-making. The method is currently designed for offline settings to retrospectively detect changes. For future work, we intend to extend TiVaCPD to the online setting for real-time measurements. Moreover, we plan to incorporate techniques for handling missing data by leveraging correlated features and temporal dynamics.

Acknowledgments

We like to express our sincere gratitude to 4YouandMe for supporting this project. Special thanks are extended to Dr. Stephen H. Friend and Dr. Sarah Goodday for their insightful consultation regarding the BUMP study and for providing us access to the valuable BUMP wearable database. Additionally, we sincerely appreciate the support of the Vector Institute in providing computing resources for conducting the experiments presented in this work.

References

Julien Ah-Pine. Normalized kernels as similarity indices. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 362–373. Springer, 2010.

Samaneh Aminikhanghahi and Diane J. Cook. A survey of methods for time series change point detection. *Knowledge and Information Systems*, 51: 339–367, 2017.

Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge L. Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones. In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2013.

Wei-Cheng Chang, Chun-Liang Li, Yiming Yang, and Barnabás Póczos. Kernel change-point detection with auxiliary deep generative models. In *International Conference on Learning Representations*, 2019.

Kevin C Cheng, Shuchin Aeron, Michael C Hughes, Erika Hussey, and Eric L Miller. Optimal transport based change point detection and time series segment clustering. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6034–6038. IEEE, 2020.

Tim De Ryck, Maarten De Vos, and Alexander Bertrand. Change point detection in time series data using autoencoders with a time-invariant representation. *IEEE Transactions on Signal Processing*, 2021.

Shohreh Deldari, Daniel V. Smith, Hao Xue, and Flora D. Salim. Time series change point detection with self-supervised contrastive predictive coding. In *Proceedings of The Web Conference 2021*, WWW '21. Association for Computing Machinery, 2021a. doi: 10.1145/3442381.3449903. URL https://doi.org/10.1145/3442381.3449903.

Shohreh Deldari, Daniel V. Smith, Hao Xue, and Flora D. Salim. Time series change point detection with self-supervised contrastive predictive coding. *WWW '21: Proceedings of the Web Conference*, 2021b.

Elise N. Erickson, Neta Gotlieb, Leonardo M. Pereira, Leslie Myatt, Clara Mosquera-Lopez, and Peter G. Jacobs. Predicting labor onset relative to the estimated date of delivery using smart ring physiological data. *npj Digital Medicine*, 6(1), 2023. doi: 10.1038/s41746-023-00902-y.

Alexander J. Gibberd and James D.B. Nelson. Estimating dynamic graphical models from multivariate time-series data. In *International Workshop on Advanced Analytics and Learning on Temporal Data*, 2015.

S M Goodday, E Karlin, A Brooks, C Chapman, D R Karlin, L Foschini, E Kipping, M Wildman, M Francis, H Greenman, Li Li, E Schadt, M Ghassemi, A Goldenberg, F Cormack, N Taptiklis, C Centen, S Smith, and S Friend. Better understanding of the metamorphosis of pregnancy (bump): protocol for a digital feasibility study in women from preconception to postpartum. *npj Digital Medicine*, 5:40, 2022. ISSN 2398-6352. doi: 10.1038/s41746-022-00579-9. URL https://doi.org/10.1038/s41746-022-00579-9.

Arthur Gretton, Karsten M Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex J Smola. A kernel method for the two-sample-problem. *International Conference on Neural Information Processing*, 2007.

David Hallac, Youngsuk Park, Stephen Boyd, and Jure Leskovec. Network inference via the time-varying graphical lasso. In *23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 205–213, 2017.

Mikhail Hushchyn and Andrey Ustyuzhanin. Generalization of change-point detection in time series data based on direct density ratio estimation. *Journal of Computational Science*, 53:101385, 2021.

Yoshinobu Kawahara, Takehisa Yairi, and Kazuo Machida. Change-point detection in time-series data based on subspace identification. *The IEEE International Conference on Data Mining (ICDM)*, 2007.

Shuang Li, Yao Xie, Hanjun Dai, and Le Song. M-statistic for kernel change-point detection. *International Conference on Neural Information Processing*, 2015.

David S. Matteson and Nicholas A. James. A nonparametric approach for multiple change point analysis of multivariate data. *Journal of the American Statistical Association*, 109, 2014.

Sang Min Oh, James M. Rehg, Tucker Balch, and Frank Dellaert. Learning and inferring motion patterns using parametric segmental switching linear dynamic systems. In *International Journal of Computer Vision (IJCV) Special Issue on Learning for Vision*, pages 103–124, 2008.

William H. Press and Saul A. Teukolsky. Savitzky-golay smoothing filters. *Computers in Physics 4, Volume 4*, 1990.

Jaxk Reeves, Jien Chen, Xiaolan L. Wang, Robert Lund, and Qi Qi Lu. A review and comparison of changepoint detection techniques for climate data. *Journal of Applied Meteorology and Climatology*, 46, 2007.

Jerzy Sacha. Interaction between heart rate and heart rate variability. *Annals of Noninvasive Electrocardiology*, 19:i–vi, 207–297, 2014.

Antonin Schrab, Ilmun Kim, Mélisande Albert, Béatrice Laurent, Benjamin Guedj, and Arthur Gretton. Mmd aggregated two-sample test, 2021. URL https://arxiv.org/abs/2110.15073.

Charles Truong, Laurent Oudre, and Nicolas Vayatis. Selective review of offline change point detection methods. *Signal Processing Volume 167*, 2018.

Gerrit J. J. van den Burg and Christopher K. I. Williams. An evaluation of change point detection algorithms. *arXiv:2003.06222v2 [stat.ML]*, 2020.

Kenji Yamanishi and Jun-ichi Takeuchi. A unifying framework for detecting outliers and change points from non-stationary time series data. *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data*, pages 676–681, 2002.

Ping Yang, Dumont Guy, and J Mark Ansermino. Adaptive change detection in heart rate trend monitoring in anesthetized children. *IEEE Transactions on Biomedical Engineering*, 2007.

Appendix A. Beedance Dataset

Bee dance (Min Oh et al., 2008): This dataset consists of six three-dimensional time-series of bees' positions while performing three-stage waggle dances. The bees communicate through actions like left/right turn, and waggle. The transition between the states represents ground truth CP.

As demonstrated in Table 4, our TiVaCPD method achieves the highest F1 score among other state-of-the-art methods, further highlighting its strong performance.

Table 4: Performance of multiple CPD methods on the Bee Dance dataset.

		Bee Dance	
Method	Precision	Recall	F1(M=5)
KL-CPD	0.24 (0.26)	0.10 (0.07)	0.13 (0.11)
Roerich	0.50 (0.34)	0.32 (0.26)	0.40 (0.30)
GraphTime	0.13 (0.04)	0.77 (0.13)	0.22 (0.07)
TIRE	0.34 (0.44)	0.14 (0.19)	0.20 (0.26)
TiVaCPD	0.36 (0.18)	0.59 (0.22)	**0.45 (0.15)**

Appendix B. Effect of the margin of error on CPD performance

The following tables demonstrate the performance of TiVaCPD and other baselines in detecting CPs with a margin of error of 10. In general, a larger margin allows for more flexibility in detecting CPs. By increasing the margin of error, the F1 score increases as it requires to be less exact in detecting the change point time. Once again, TiVaCPD outperforms all baselines in detecting the exact time of CPs in both simulated and real-world datasets.

Appendix C. Additional Results Visualization Comparison

Please see Figure 8.

Appendix D. Additional Structural Info

TVGL additional information: TVGL method Hallac et al. (2017) can be used for inferring multiple networks in time and in constructing adjacency matrices from local time windows. Two regularization parameters are used in TVGL - α and β. The parameter α controls the network's sparsity. A large α will lead to a precision matrix with fewer non-zero elements. The parameter β controls the temporal consistency by determining how strongly correlated adjacent network estimations should be. A large β will result in a more temporally consistent network, where the precision matrices at different time points can be similar. Another parameter is slice size which is the size of the window used to calculate the precision matrix. A larger window not only includes more information but also introduces more noise into the estimation. Moreover, to promote the identification of shifts in the covariance pattern of features, we integrated an L2-norm penalty function into the estimation of the matrix inverse.

Instruction on how $\hat{W}$ is calculated: Initially, $\hat{W}$ mirrors W. Then we calculate the number of peaks detected in each score individually. Subsequently, we calculate the number of detected peaks for each score individually. If a score identifies a significantly higher number of CPs compared to the number detected by other scores (where five CPs are considered significant), we update the weights of that specific score to zero.

Appendix E. Baseline methods

Kernel Change Point Detection (KLCPD) (Chang et al., 2019) is a kernel learning framework for CPD that uses a two-sample test and optimizes

Table 5: Performance of CPD methods on the Jumping Mean and Changing Variance with $M = 10$.

	Jumping Mean			Changing Variance		
Method	Precision	Recall	F1 (M=10)	Precision	Recall	F1 (M=10)
KL-CPD	0.90 (0.16)	0.50 (0.14)	0.60 (0.12)	0.22 (0.05)	0.75 (0.21)	0.34 (0.08)
Roerich	0.29 (0.03)	0.97 (0.06)	0.44 (0.03)	0.23 (0.04)	0.75 (0.17)	0.36 (0.07)
GraphTime	0.41 (0.11)	0.90 (0.07)	0.54 (0.09)	0.08 (0.02)	1.00 (0.00)	0.15 (0.03)
TIRE	0.97 (0.07)	0.87 (0.12)	0.90 (0.12)	0.30 (0.35)	0.08 (0.09)	0.12 (0.13)
TiVaCPD	1.00 (0.00)	0.95 (0.17)	**0.94 (0.14)**	1.0 (0.0)	0.98 (0.06)	**0.98 (0.03)**

Table 6: Performance of CPD methods on the Changing Correlations and Arbitrary CPs with $M = 5$.

	Changing Correlations			Arbitrary CPs		
Method	Precision	Recall	F1 (M=10)	Precision	Recall	F1 (M=10)
KL-CPD	0.22 (0.04)	0.78 (0.13)	0.35 (0.06)	0.84 (0.18)	0.6 (0.19)	0.64 (0.19)
Roerich	0.28 (0.04)	0.77 (0.18)	0.40 (0.07)	0.32 (0.04)	0.9 (0.12)	0.46 (0.05)
GraphTime	0.13 (0.04)	1.00 (0.00)	0.23 (0.07)	0.23 (0.08)	0.90 (0.09)	0.36 (0.10)
TIRE	0.16 (0.14)	0.15 (0.08)	0.15 (0.13)	1.00 (0.00)	0.58 (0.19)	0.70 (0.16)
TiVaCPD	0.33 (0.03)	1.00 (0.00)	**0.48 (0.05)**	0.96 (0.07)	0.85 (0.15)	**0.88 (0.10)**

Table 7: Performance of multiple CPD methods on the Bee Dance and HAR datasets with $M = 10$.

	Bee Dance			HAR		
Method	Precision	Recall	F1(M=10)	Precision	Recall	F1(M=10)
KL-CPD	0.26 (0.25)	0.15 (0.10)	0.17 (0.13)	0.66 (0.10)	0.24 (0.02)	0.31 (0.05)
Roerich	0.51 (0.34)	0.36 (0.38)	0.42 (0.32)	0.69 (0.16)	0.14 (0.04)	0.20 (0.06)
GraphTime	0.13 (0.03)	0.80 (0.11)	0.24 (0.08)	0.08 (0.001)	0.96 (0.01)	0.09 (0.01)
TIRE	0.36 (0.46)	0.17 (0.20)	0.23 (0.25)	0.55 (0.20)	0.16 (0.03)	0.25 (0.07)
TiVaCPD	0.36 (0.18)	0.62 (0.20)	**0.46 (0.14)**	0.72 (0.05)	0.51 (0.05)	**0.60 (05)**

a lower bound of test power via an auxiliary generative model. For this method, we used window sizes $w \in [10, 25]$ for all experiments and trained the model for 25 epochs, unless more training led to improved results. Consistent with our own post-processing steps, we performed peak detection to detect the exact time of change.

Roerich (Hushchyn and Ustyuzhanin, 2021) is a CPD method based on direct density ratio estimation to detect the change in distribution. We set all parameters to default and use window sizes $w \in [10, 25]$ for all experiments.

Group Fused Graph Lasso (GraphTime) (Gibberd and Nelson, 2015) is a time-varying graphical model based on the group fused-lasso. Similar to TiVaCPD it uses the graphical model to model the dependencies of variables in time series. GraphTime models the temporal dependencies between variables while TiVaCPD models the pairwise dependencies to identify CPs.

Time-Invariant Representation (TIRE) De Ryck et al. (2021) is an autoencoder-based CPD method, we used window sizes w between 10, 25 for all experiments. For the *domain* parameter, we chose *both* to include both time, and frequency domains. We used 200 epochs to train the model.

Appendix F. Time Complexity Analysis

We conducted a comprehensive set of experiments to evaluate how increasing data dimensionality impacts our methodology. Our analysis encompassed three distinct experiments, focusing on: 1. The influence of expanding the number of features. 2. The effects

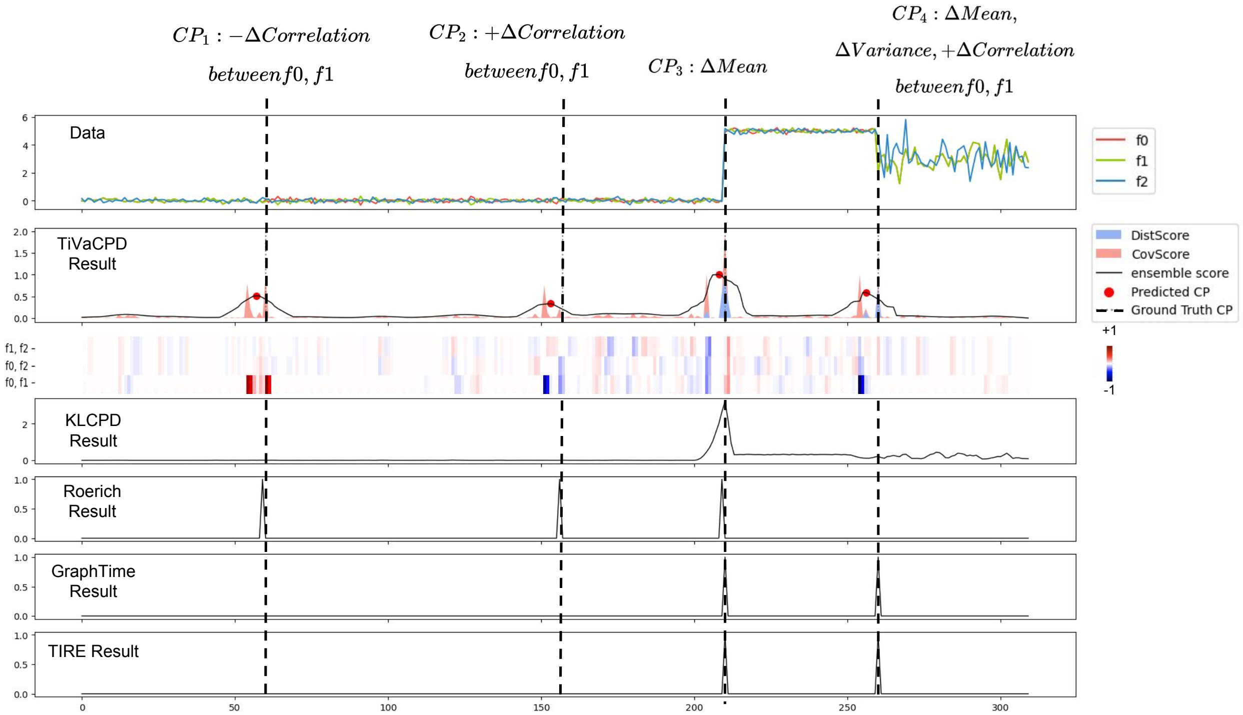

Figure 8: This example shows a simulated time-series with various CPs. Row 1: raw data. Rows 2-3: TiVaCPD score with heatmap interpretation. Rows 4-6: other CP baseline scores.

Table 8: Empirical Time Complexity Analysis on Arbitrary dataset (M=5). #Dim stands for a number of input features. Time x 2 has twice the length of recording and CP (10 change points with 728 time points) with 3 features.

Arbitrary dataset					
#Dim 3	Time x 2	#Dim 5	#Dim 10	#Dim 20	#Dim 40
24.78 (1.39)	43.00 (4.34)	25.09 (4.56)	50.75 (7.20)	116.2 (9.23)	453.18 (30.19)

of extending the temporal scope of the data. 3. The outcomes of dimensionality reduction, involving the removal of highly correlated data during preprocessing.

The results in table 8 demonstrate that our approach exhibits linear scalability over time. This property renders that our model is suitable for data streaming scenarios, where new data continually enters the system. Specifically, our algorithm takes 22 seconds to process each sample of arbitrary data with 368 time points and 3 variables. Furthermore, our investigations revealed that our model's performance remains consistent even when highly correlated features with a correlation of 0.9 are removed. Therefore, adding a feature reduction preprocessing step can help ensure that computational costs do not increase disproportionately with a growing number of features.

Our choice to employ TVGL comes from its dedicated efforts to mitigate the computational load associated with inverse covariance calculations by the introduction of parallelism and sparsity into the algorithm. Although TVGL's time complexity is cubic in relation to the number of features, it still proves considerably more efficient than the $O(n^6)$ complexity of some conventional methods for inverse covariance computation (Hallac et al., 2017). Based on table 8, when increasing the number of features from 5 to 10, and subsequently to 20 and 40, per-sample run time increases by 2.0, 2.3, and 3.9 times, respectively. Therefore, we recommend considering employing the mentioned correlation removal approach or other dimensionality reduction methods for high-dimensional data before passing it to our TiVaCPD model.

Appendix G. BUMP Result Table

Please see Table 9.

Table 9: Performance of multiple CPD methods on BUMP dataset (M=5).

	BUMP		
Method	Precision	Recall	F1(M=5)
KL-CPD	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)
Roerich	0.07 (0.16)	0.23 (0.27)	0.10 (0.17)
GraphTime	0.12 (0.07)	0.92 (0.17)	0.21 (0.11)
TIRE	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)
TiVaCPD	0.52 (0.30)	0.46 (0.30)	**0.46 (0.31)**

Appendix H. Random Search and Hyper-parameter Sensitivity

To determine the best hyper-parameters for finding CPs and evaluate the model's sensitivity to these parameters, we implemented a random search with 20 random combinations of the dynamic window threshold ϵ and CovScore's α, β, and $SliceSize$. The search is run over only 10% percent of each data, and the segmentation is done through the data file and not time. In our testing, we evaluated the impact of different window lengths on the computation of the covariance matrix, which was achieved by adjusting the algorithm slice size $SliceSize : \{14, 10, 5\}$. We also tested TiVaCPD with $\epsilon : \{.2, .02, .002\}$, $\alpha : \{5, 1, 0.4\}$, and $\beta : \{12, 6, 0.4\}$.

Through experimentation and parameter random search, we found that smaller α values lead to very dense networks that are less interpretable as the model overfits the data. Large β values lead to smoother network estimates over time that do not easily change from window to window. For all datasets, the standard deviation of the F1 score is between 0.003 and 0.1 and the CPD performance slightly varies depending on the parameters settings. In general, choosing a small threshold, medium alpha value, a small beta value, and a small slice size leads to more granular, timely, and interpretable CPs.