Contrast Pattern Mining in Paired Multivariate Time Series of a Controlled Driving Behavior Experiment

QINGZHE LI, LIANG ZHAO, YI-CHING LEE, and JESSICA LIN, George Mason University

Q1

The controlled experiment is an important scientific method for researchers seeking to determine the influence of the intervention, by interpreting the contrast patterns between the temporal observations from control and experimental groups (i.e., paired multivariate time series (PMTS)). Due to recent technological advances and the growing popularity of sensing technology such as in-vehicle sensors and activity trackers, time series data is experiencing explosive growth in both size and complexity. This is threatening to overwhelm the interpretation of control experiments, which conventionally rely on human analysts. Thus, it is imperative to develop automated methods that are expected to simultaneously characterize and detect the interpretable contrast patterns in PMTS generated by controlled experiments. However, a few challenges prohibit existing methods from directly addressing this problem: (1) handling the coupling of contrast identification and pattern characterization, (2) dynamically characterizing the patterns in PMTS, and (3) mining the contrast patterns in multiple PMTS with ubiquitous individual differences. Therefore, we propose a novel framework to mine interpretable contrast patterns based on the dynamic feature dependencies for PMTS through optimization. The proposed framework simultaneously characterizes the dynamic feature dependency networks for PMTS and detects the contrast patterns. Specifically, we characterize the generative process of PMTS as a probabilistic model defined by pairwise Markov random fields whose likelihoods are maximized using our group graphical lasso. The model is then generalized to handle multiple PMTS and solved by proposing a customized algorithm based on the expectation-maximization framework. Extensive experiments demonstrate the effectiveness, scalability, and interpretability of our approach.

CCS Concepts: • **Information systems → Data stream mining**;

Additional Key Words and Phrases: Contrast pattern, dynamic feature dependency, controlled experiment, driving behavior, multivariate time series

ACM Reference format:
Qingzhe Li, Liang Zhao, Yi-Ching Lee, and Jessica Lin. 2020. Contrast Pattern Mining in Paired Multivariate Time Series of a Controlled Driving Behavior Experiment. *ACM Trans. Spatial Algorithms Syst.* 6, 4, Article 25 (May 2020), 28 pages.
https://doi.org/10.1145/3397272

25

1 INTRODUCTION

Controlled experiments, which are also known as randomized experiments and A/B tests, are widely used in many domains, such as medicine [van Geffen et al. 2011] and biology [Agrawal and Kotanen 2003]. Their primary purpose is to identify and interpret possible differences caused by the intervention between control and experimental groups. In controlled experiments, the

Authors' addresses: Q. Li, L. Zhao, Y.-C. Lee, and J. Lin, George Mason University, 4400 University Drive, Fairfax, VA 22030; emails: {qli10, lzhao9, ylee65, jessica}@gmu.edu.

2374-0353/2020/05-ART25 $15.00
https://doi.org/10.1145/3397272

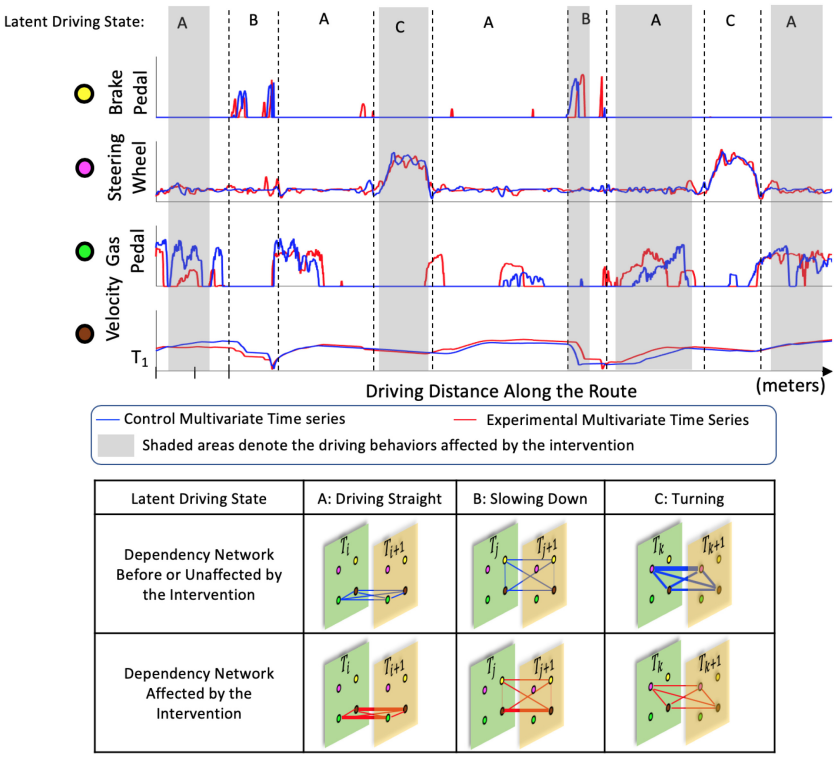

Fig. 1. The contrast patterns in PMTS (the PMTS are plotted at the top). Both time series in the PMTS correspond to the same route with identical traffic conditions as control factors. Thus, they should experience the same driving state at all locations. The unaffected and affected dependency networks corresponding to three latent states are plotted at the bottom. The node in the dependency network denotes the same-colored sensor within a small sliding window (i.e., size = 2). The widths of the edges denote the strengths of the dependencies between the connected sensors (better seen in color).

multivariate time series generated from the control group usually needs to be exactly paired with the multivariate time series generated from the experimental group. Here we call the control multivariate time series and experimental multivariate time series altogether as *paired multivariate time series* (PMTS). In this article, we focus on quantitatively analyzing the effects of an intervention (e.g., alcohol, medicine) on drivers' driving behaviors through the PMTS data.

The dynamically changed driving behaviors depend on the mixture of many factors, such as traffic conditions, weather, and driving skills. It is inappropriate and extremely difficult to universally predefine or label the driving behaviors from the multivariate times series data, which motivates us to model the driving behaviors in an unsupervised way. We found that the dependency networks of in-vehicle sensors inferred from the multivariate time series can precisely characterize the driving behaviors with high interpretability. For example, the co-occurred increasing steering wheel values and decreasing velocity values infer the dependency network of Latent State C shown in Figure 1, which can be easily interpreted as the "turning" behavior. However, when "steering wheel" and "gas pedal" have very small or zero values along with the high values on "brake" and steep decreasing values on "velocity," the structure of the inferred dependency networks are shown in Latent State B of Figure 1. Such a dependency network can be interpreted as slowing down.

These structural patterns should be shared by all drivers under all conditions if they try to drive safely. To know the effect of the intervention on the driving behavior, a driver is asked to drive twice with and without intervention on exactly the same route under an identical traffic environment (i.e., the control factors), so she or he should experience the same sequence of latent states (e.g., corresponding to curves, stop signs) as shown in Figure 1. Hence, although cross the controlled and experimental time series, the structural patterns of the dependency networks, the strengths of dependencies could be changed by the intervention. For example, as the dependency networks shown in the Turning column of the table in Figure 1, the dependency between "steering wheel" and "velocity" is weaker, which indicates a lower capability of adjusting the steering wheel according to the velocity caused by drinking alcohol. We say that there exists a contrast pattern if the dependency network that characterizes the same latent driving state is changed after the intervention. In addition, the driver may still be unaffected by the alcohol for some turns but affected for other turns. For example, because alcohol can increase the probability of making a bad turn, but it is unlikely to guarantee to make bad turns for all turning states. Therefore, the research goal of this work is to automatically identify whether and how much the intervention makes a difference in causing some "affected driving behaviors" under the same sequence of latent states.

Unsupervised identification and characterization of driving behaviors are an active research topic. The reason unsupervised approaches [Fugiglando et al. 2019; Hallac et al. 2018, 2017b] are much more popular than supervised approaches is that it is extremely difficult to obtain comprehensive, accurate, and sufficient labels. First, domain experts are still far away from having formal definitions or interpretations for all of the actual driving [Vilaca et al. 2017]. Even if we were given fully- and clearly defined states, it is still highly challenging and extremely labor intensive for domain experts to accurately label the raw multivariate time series data. Moreover, it is also extremely difficult to prepare sufficient labeled data to train powerful classification models, which typically require large amounts of data. The controlled experiment described previously typically contains millions of time points, tens of participants, and an exponential number of node combinations in the dependency networks. This is typical for PMTS in controlled experiments, which tend to increase rapidly in terms of their data size and complexity, quickly going far beyond the capacity of data analysts using traditional statistics to process or interpret directly. It is therefore imperative to develop new techniques capable of automatically (1) recognizing and characterizing the driving states (e.g., turning) by learning the dynamic dependency networks in PMTS and (2) discovering contrast patterns in PMTS for each driving state.

Although some previous works are partially related to our problem, such as time series subsequence clustering [Goldin et al. 2006; Hallac et al. 2017b], time series segmentation [Matsubara et al. 2014], and contrast pattern mining [Lee et al. 2017; Liu et al. 2017], none of them can simultaneously handle both of the previously mentioned subproblems for PMTS. Several challenges prevent the existing work from being directly utilized or combined to handle this problem. A first challenge is 1) difficulty in the coupling of latent state characterization and contrast pattern mining for PMTS. For example, to characterize one latent state using the dependency network(s), we need to know whether there exists a contrast pattern. If a contrast pattern exists, we need to learn two dependency networks from the control and experimental time series, respectively; otherwise, we just need to learn one dependency network from both control and experimental time series. Conversely, to mine the contrast pattern, we need to know how the dependency network(s) characterize the latent state. The patterns learned by existing works that address the first and second subproblems separately cannot maintain the consistency and optimality of learning. A second challenge is difficulty in joint dynamic dependency networks learning for PMTS. As Figure 1 shows, the dependency networks for a common latent state should always share a unique structural pattern, but adding this constraint typically leads to a nonconvex problem when learn-

ing the dependency networks. A third challenge is difficulty in jointly learning multiple PMTS. In controlled experiments, domain experts usually need to see contrast patterns with statistical significance in a group of individuals. There is a big challenge of eliminating the contingency caused by the ubiquitous individual differences when detecting the shared contrast pattern.

To simultaneously address the preceding challenges, we propose a novel framework to mine the contrast patterns of dynamic dependency networks for PMTS with interpretability. The main contributions of this work are as follows:

- *Developing a novel framework to mine contrast patterns in the dynamic dependency networks of the PMTS.* A novel contrast dynamic feature dependency (CDFD) pattern mining problem for PMTS is formulated that simultaneously optimizes latent state recognition and characterization, as well as CDFD pattern detection problems.
- *Proposing a new group graphical lasso based on a probabilistic model of PMTS.* We creatively model the subsequence pairs in PMTS as multiple Gaussian Markov random field (MRF) pairs to simultaneously capture the identical conditional dependency structures and contrast the patterns in each MRF pair. To achieve this, a new group graphical lasso is proposed by adding an $L_{2,1}$-norm regularization term to our probabilistic model.
- *Generalizing the proposed graphical lasso to mine the shared contrast patterns in multiple PMTS.* To mine the meaningful contrast pattern among multiple PMTS without contingency caused by the individual differences, we extend the proposed group graphical lasso model from one PMTS to multiple PMTS. To the best of our knowledge, this model is the first unified model that can simultaneously mine the shared contrast patterns and eliminate the influences of individual differences.
- *Developing an efficient algorithm to solve a new nonconvex and noncontinuous optimization problem.* To optimize the proposed model, which contains both nonconvex and discrete terms, we propose a new algorithm based on expectation-maximization (EM) [Dempster et al. 1977] and the alternating direction method of multipliers (ADMM) [Boyd et al. 2011] that solves the proposed model efficiently and is guaranteed to converge to a locally optimal solution.
- *Conducting comprehensive experiments to validate the effectiveness, efficiency, robustness, and interpretability of our proposed approach.* Extensive experiments on eight synthetic datasets demonstrate the effectiveness, scalability, and robustness of the proposed models and algorithms. The experiments on two real-world datasets qualitatively demonstrate the effectiveness and interpretability of the proposed methods on the mined CDFDs.

The rest of this article is organized as follows. Section 2 reviews the related work. Section 3 formulates the problem of CDFD pattern mining for PMTS. Section 4 presents two models to mine CDFD patterns in one and multiple PMTS, respectively. Our optimization algorithms are elaborated in Section 5. In Section 6, extensive experiments are conducted to evaluate the effectiveness, scalability, and interpretability of the proposed models and algorithms. The entire work is summarized and concluded in Section 7.

2 RELATED WORK

The previous work related to the research presented in this article is summarized in the following.

Contrast pattern mining for time series. There are only a few works on contrast pattern mining for time series, which can be divided into two categories: distance-based contrast patterns and model-based contrast patterns. Distance-based contrast patterns are defined based on some time series distance measures. For example, Lin and Keogh [2006] extended the notion of contrast sets for time series that identified the subsequence that differentiates two time series based on

Euclidean distance. Other distance-based contrast patterns in times series such as shapelets [Ye and Keogh 2009] and representative patterns [Wang et al. 2016] are developed exclusively for supervised learning tasks. Unlike the CDFD pattern, their definitions are all based on some distance measure. However, methods based on such definitions are unable to identify and interpret latent states in controlled experiments. For model-based contrast patterns, a few researchers have begun to utilize multivariate time series generated in fMRI to mine the contrast patterns by proposing various network inference models [Lee et al. 2017; Liu et al. 2017]. For instance, Lee et al. [2017] proposed a CNN-based deep neural network to identify contrasting dependency networks inferred from the entire time series without considering the contrast pattern occurring in subsequence level under a common latent state. Similarly, Liu et al. [2017] proposed a contrast graphical lasso model for whole time series that derives a single contrast dependency network that corresponds to two groups of time series. However, neither of these methods is able to explicitly identify subsequence pairs in PMTS with CDFD patterns.

Time series subsequence clustering. Mining the CDFD pattern requires identifying the latent states in PMTS that could be achieved by clustering the subsequences of PMTS. Clustering all overlapped time series subsequences produces meaningless results [Keogh et al. 2003] due to the reuse of the data points in the overlapping subsequences. Since then, some meaningful distance-based approaches have been proposed that avoided the preceding pitfall. For example, Rakthanmanon et al. [2012] proposed a parameter-free minimum description length framework to meaningfully cluster time series subsequences by ignoring some data. The distance-based approaches cluster time series subsequences by their "shapes" as opposed to our dependency-base patterns. There are also model-based time series subsequence clustering approaches such as those based on ARMA [Xiong and Yeung 2004], the Gaussian Mixture model (GMM) [Banfield and Raftery 1993], and hidden Markov models [Smyth 1997]. These typically consider the whole sequence, except for Toeplitz inverse covariance-based clustering (TICC), proposed recently by Hallac et al. [2017b], which clusters the subsequences in a single multivariate time series according to structural patterns estimated by a graphical lasso. However, TICC only focuses on single time series and can neither take into account the correlations among pairs of time series nor mine their contrast patterns.

Graphical lasso for time series. Lasso [Tibshirani 1996] is an important feature selection technique in the sparse feature learning domain [Du et al. 2018; Gao and Zhao 2018; Wang et al. 2018b; Zhao et al. 2019]. The graphical lasso [Friedman et al. 2008] is validated as an effective and efficient technique of inferring the sparse graphs [Hallac et al. 2017b; Liu et al. 2017], feeding to the graph mining domain [Hassan et al. 2016]. Many graphical lasso–based models have been applied to time series sparse inverse covariance matrix estimation problems [Hallac et al. 2017a, 2017b; Jung et al. 2015; Veeriah et al. 2015; Yuen et al. 2018], some of which estimated sparse Gaussian inverse covariance matrices for multivariate time series subsequences [Hallac et al. 2017a, 2017b], although they are only able to detect the "latent states" but did not consider the contrast patterns. Others [Jung et al. 2015; Veeriah et al. 2015; Yuen et al. 2018] estimated sparse Gaussian inverse covariance matrices across the entire sequences of multiple univariate time series or one multivariate time series. Jung et al. [2015] proposed a graphical model selection scheme based on graphical lasso for stationary time series, but they applied the graphical lasso to the entire time series, which also failed to capture the contrast patterns on the subsequence level under the common latent state required by the controlled experiments.

Driving behavior modeling. Modeling driving behaviors is one of the hottest topics in multiple domains, such as urban computing and autonomous driving. Most of these models are focusing on modeling the driving behaviors using one of the following data types: (1) trajectory data recorded by portable GPS devices and (2) multivariate time series data recorded by a controller area network

Table 1. Notations

Notations	Descriptions
m	Count of observations in a multivariate time series
n	Dimensionality of each observation
C	Multivariate time series data C
w	Window size parameter
X	One control multivariate time series
$\hat{X}$	One experimental multivariate time series
$(X, \hat{X})$	One PMTS
$(\mathcal{X}, \hat{\mathcal{X}})$	Multiple PMTS
P	Count of PMTS in $(\mathcal{X}, \hat{\mathcal{X}})$
$\theta_k, \hat{\theta}_k$	One pair of MRFs of the k^{th} latent state to be learned
$\Theta^{(k)}, \hat{\Theta}^{(k)}$	P pairs of MRFs of the k^{th} latent state to be learned
Y	Latent state assignments to be learned
Z	Contrast pattern indicator to be learned
K	Parameter of the latent state count
β	Penalty parameter of switching between contrast and noncontrast latent states
γ	Penalty parameter of switching among different latent states
λ	Regularization parameter that controls the sparsity level in the MRFs
ρ	ADMM penalty parameter
$U, \hat{U}$	Scaled dual variables in the ADMM algorithm

(CAN). Each of these data types has some advantages and disadvantages in terms of granularity and whether or not they consider the spatial information. The models using the trajectory data [Wang et al. 2018a, 2019] are able to model the driving behaviors with the spatial information. However, they suffer from precisely characterizing the driving behaviors due to the coarse granularity of the trajectory data. However, the models using the multivariate time series data [Fugiglando et al. 2019; Hallac et al. 2018; Li et al. 2019] are good at capturing inconspicuous driving behaviors but are unable to take the spatial information into account. Our problem requires both fine granularity to identify the inconspicuous contrast pattern and consideration of spatial information as one of the controlling factors.

3 PROBLEM SETUP

In this section, we first define the relevant concepts and then present the new problem of CDFD pattern mining for PMTS. The key notations, with brief descriptions, are listed in Table 1.

Consider the multivariate time series shown in Figure 2. A multivariate time series $C = [C_1, C_2, \ldots, C_m]$ is a time-ordered sequence of m vectors where each time point $C_t \in \mathbb{R}^{n\times 1}$ is a multivariate observation that contains n dimensions. Unlike the data that follows independent and identically distributed (iid) assumption, the observation of a time point t is also dependent on its context, which is captured by the *subsequences*. Given a sliding window of size $w \ll m$, we define a multivariate time series *subsequence* $X_t \in \mathbb{R}^{1\times nw}$ as $X_t = [C_t^\intercal, \ldots, C_{t+w-1}^\intercal]$, which consists of a concatenation of w consecutive n-d vectors starting from the t^{th} time point. We call each dimension in X_t a *feature*, so there are nw features in X_t. Next, we denote $X = [X_1^\intercal, X_2^\intercal, \ldots, X_T^\intercal]^\intercal$, which stacks all subsequences of size w in C, where $X \in \mathbb{R}^{T\times nw}$ and $T = m - w + 1$ is the count of

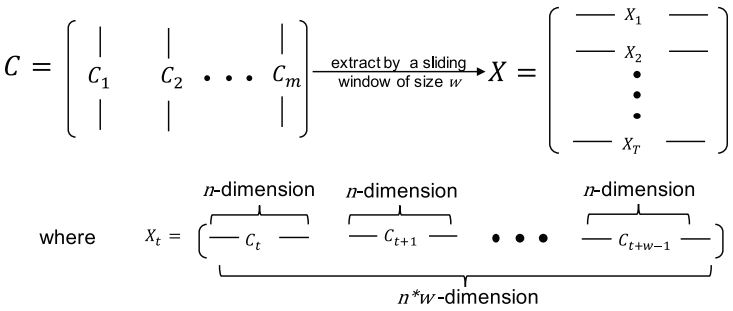

Fig. 2. Multivariate time series data representation.

subsequences in C. For any given w, there is a one-to-one mapping relationship between C and X, so we will directly use X to denote a multivariate time series in this article.

In multivariate time series, the sensors in each dimension can be correlated to each other, and neighboring data points of the same dimension have temporal dependency. Therefore, these dependencies may exist between any two features. The structural pattern of the feature dependency network exclusively characterizes a *latent state* as seen in Figure 1, and a multivariate time series data X can be generated from K latent states (e.g., turning, slowing down), where the parameter K is determined by users. Naturally, the feature dependency pattern of each latent state is characterized by an MRF [Kindermann and Snell 1980] among the nw features in X. Specifically, we denote $G_k = (X_t, \theta_k)$ as the Gaussian MRF that generates the subsequences belonging to the k^{th} latent state, where $\theta_k \in \mathbb{R}^{nw \times nw}$ is the inverse covariance matrix that defines G_k and encodes the structural representation of the conditional independency among the features. We use $Y \in \{0,1\}^{T \times K}$ to denote the assignments of the latent state for all time points. Specifically, $Y_{t,k} = 1$ if X_t belongs to the k^{th} latent state; otherwise, $Y_{t,k} = 0$.

In controlled experiments, time series commonly come in pairs, so the *paired multivariate time series* is formally defined as follows.

Definition 3.1 (Paired Multivariate Time Series). We denote two multivariate time series as X and $\hat{X}$, where X is defined as the *control time series* and $\hat{X}$ is the *experimental time series* of X such that (1) X and $\hat{X}$ have the same size T (i.e., the count of subsequences for a given w); (2) each pair of X_t and $\hat{X}_t$ shares the same assignment of latent state Y_t; and (3) for all $k = 1, \ldots, K$, their k^{th} latent states are always identical in their conditional independency structure such that $\text{supp}(\theta_k) = \text{supp}(\hat{\theta}_k)$, where the matrix support "supp" is defined as the index set of nonzero elements.

For the example shown in Figure 1, two time series in PMTS contain the same count of subsequences for a given w, and the subsequences in the same road segment are defined to share the same latent state assignments. Here, the latent state refers to the situation when a driver should brake at one location, given the identical route and traffic condition, which may be different from the actual driving state. We define our contrast pattern as the differences in the dependency strengths by letting both drives share the same latent state assignments. Formally, our CDFD pattern is defined as follows.

Definition 3.2 (Contrast Dynamic Feature Dependency). Given a PMTS $(X, \hat{X})$, for each subsequence pair $(X_t, \hat{X}_t)$ where $t = 1, \ldots, T$, a *contrast dynamic feature dependency* pattern exists if and only if X_t and $\hat{X}_t$ are generated from different MRFs defined by θ_k and $\hat{\theta}_k$, where $\text{supp}(\theta_k) = \text{supp}(\hat{\theta}_k)$ and $\theta_k \neq \hat{\theta}_k$. The existence of the CDFD patterns are denoted by a *contrast*

indicator $Z \in \{0,1\}^{T\times 1}$. Specifically, $Z_t = 0$ when there is CDFD pattern between X_t and $\hat{X}_t$, and $Z_t = 1$ when there is no CDFD pattern (i.e., X_t and $\hat{X}_t$ are generated from identical MRFs).

For example, as the dependency networks shown in the bottom of Figure 1, the CDFD pattern refers to the characterization of the dependency networks θ_k and $\hat{\theta}_k$ (i.e., within each column or latent state) that have an identical structural pattern but different feature dependency strengths. To capture the fact that the intervention (e.g., alcohol) is likely to increase the probability of the occurrences of CDFD patterns but unlikely to guarantee the occurrences of CDFD patterns, we define the problem in a more general way by introducing the contrast indicator Z to be learned from PMTS. Our assumption is weaker since we do not enforce all instances to presumably have contrast patterns but learn the patterns from the data. The problem of CDFD pattern mining for one PMTS is formulated as follows.

Problem formulation. Given a PMTS $(X, \hat{X})$, our goal is to mine its CDFD patterns that can be interpreted through K MRFs, which requires to (1) characterize the K latent states by learning their MRFs $\theta = \{\theta_k\}_k^K$ and $\hat{\theta} = \{\hat{\theta}_k\}_k^K$, (2) determine the latent state assignments Y, and (3) decide the Z assignments by detecting the CDFD pattern for each subsequence.

For the example in Figure 1, given a PMTS $(X, \hat{X})$ obtained from the driving simulator without (i.e., X) and with (i.e., $\hat{X}$) an intervention, mining the CDFD patterns involves (1) characterizing the K latent states encoded by θ and $\hat{\theta}$, (2) determining the latent state assignments Y for all road segments, and (3) deciding on the Z assignments based on whether the driving behaviors have been changed for each road segment.

The preceding problem poses the following main technical challenges. The first challenge is difficulty in jointly learning all of the variables $\theta_k, \hat{\theta}_k, Y, Z$ for each PMTS. These variables are correlated with each other, and thus must be jointly learned. However, there is no existing model that can jointly characterize them in a unified framework. The second challeng is difficulty in maintaining the dependencies among the paired MRFs in PMTS. As stated in Definition 3.1, the constraint requiring identical patterns for the conditional independency structures between the MRFs in each latent state, namely $\text{supp}(\theta_k) = \text{supp}(\hat{\theta}_k)$, must be protected during the parameter optimization process. This constraint is inherently nonconvex, which is difficult to maintain effectively and efficiently during the optimization process.

4 METHODOLOGY

The models for mining CDFD patterns in PMTS are proposed in this section. We first propose a new probabilistic modeling method for PMTS in Section 4.1. Then a novel model of CDFD pattern mining for PMTS (CMP) is proposed to mine the CDFD in one PMTS in Section 4.2. The CMP model is generalized to a group CMP (GCMP) model that mines the CDFD in multiple PMTS in Section 4.3.

4.1 Probabilistic Modeling of PMTS

As X_t and $\hat{X}_t$ are continuous variables, they are defined to be sampled from multivariate Gaussian distributions. When $Z_t = 0$ (i.e., existing CDFD), X_t and $\hat{X}_t$ are generated from the multivariate Gaussian distributions defined by different inverse covariance matrices θ_k and $\hat{\theta}_k$, respectively: $X_t \sim \mathcal{N}(X_t|\theta_k, \mu_k)$ and $\hat{X}_t \sim \mathcal{N}(X_t|\hat{\theta}_k, \hat{\mu}_k)$ such that the conditional joint distribution of $(X_t, \hat{X}_t)$ is

$$p(X_t, \hat{X}_t|Y_{t,k} = 1, Z_t = 0) = \mathcal{N}(X_t|\theta_k, \mu_k) \cdot \mathcal{N}(\hat{X}_t|\hat{\theta}_k, \hat{\mu}_k). \quad (1)$$

Otherwise, when $Z_t = 1$, X_t and $\hat{X}_t$ are generated from the multivariate Gaussian distributions defined by the same inverse covariance matrix $\Theta^{(k)}$: $X_t \sim \mathcal{N}(X_t|\theta_k, \mu_k)$ and $\hat{X}_t \sim \mathcal{N}(X_t|\theta_k, \hat{\mu}_k)$

such that the conditional joint distribution of $(X_t, \hat{X}_t)$ is

$$p(X_t, \hat{X}_t | Y_{t,k} = 1, Z_t = 1) = \mathcal{N}(X_t | \theta_k, \mu_k) \cdot \mathcal{N}(\hat{X}_t | \theta_k, \hat{\mu}_k). \quad (2)$$

Based on the preceding equations, for all time points $t = 1, \ldots, T$, the likelihood of $(X, \hat{X})$ conditioned on the parameters Y, Z, θ, and $\hat{\theta}$ is

$$p(X, \hat{X} | Y, Z, \theta, \hat{\theta}) = \prod_{k,t}^{K,T} [\mathcal{N}(X_t | \theta_k, \mu_k)^{Y_{t,k}} \mathcal{N}(\hat{X}_t | \theta_k, \hat{\mu}_k)^{Y_{t,k}}]^{Zt} \cdot [\mathcal{N}(X_t | \theta_k, \mu_k)^{Y_{t,k}} \mathcal{N}(\hat{X}_t | \hat{\theta}_k, \hat{\mu}_k)^{Y_{t,k}}]^{(1-Z_t)}. \quad (3)$$

4.2 CDFD Pattern Mining for One PMTS

This section presents our proposed model of CDFD pattern mining for one PMTS (CMP), which optimizes the parameters of the probabilistic model for a single PMTS. To achieve this, three considerations must be taken into account: (1) the maximal likelihood of the probabilistic model for PMTS, (2) regularization on the structure of the paired MRFs for PMTS, and (3) the temporal dependency of the latent state assignments. These are discussed in turn in the following.

4.2.1 Loss Function. Given a PMTS $(X, \hat{X})$, maximizing the likelihood of Equation (3) is equivalent to minimizing the negative log likelihood, leading to our loss function:

$$\mathcal{L}(Y, Z, \theta, \hat{\theta}) = \sum_{t,k}^{T,K} Y_{t,k} [Z_t(-\ell\ell(X_t, \theta_k) - \ell\ell(\hat{X}_t, \theta_k)) + (1 - Z_t)(-\ell\ell(X_t, \theta_k) - \ell\ell(\hat{X}_t, \hat{\theta}_k))], \quad (4)$$

where $\ell\ell(A, B) = -\frac{1}{2}(A - \mu)^\intercal B(A - \mu) + \frac{1}{2}\log\det B - \frac{n}{2}\log(2\pi))$ denotes the log likelihood that the multivariate subsequence A comes from the Gaussian distribution with inverse covariance matrix B.

4.2.2 Structural and Temporal Regularization. Due to the identical conditional independency structure constraint required in Definition 1, the widely used L_1-norm regularization term [Hallac et al. 2017b] would not satisfy such constraint. We thus propose an $L_{2,1}$-norm regularization term that enforces the identical sparsity pattern in the contrast MRF pair defined by θ_k and $\hat{\theta}_k$, so the zero values correspond to the conditional independent relationship between the two features. Our $L_{2,1}$-norm regularization term is defined as $\sum_k^K \|\lambda \cdot [v(\theta_k), v(\hat{\theta}_k)]\|_{2,1}$, where $v(\cdot)$ is a vectorization function for any input matrix and λ is the regularization parameter that determines the sparsity level in the MRFs. To distinguish the dependency patterns for different latent states, the values of λ should be always greater than zero since $\lambda = 0$ will lead to a clique for all MRFs and should not be too large, as this will cause some learned θ_k and $\hat{\theta}_k$ are both equal to $\mathbf{0}$. Typically, any λ value between 0.1 and 50 works well for normalized PMTS.

Due to the nature of temporal continuity in time series, neighboring points tend to have consistent latent state assignments as suggested in the work of Hallac et al. [2017b]. The contrast pattern has temporal dependency as well. We thus penalize the divergence between neighboring time points on both the Y and Z assignments by proposing the following smoothing term:

$$\boldsymbol{h}_{\beta,\gamma}(Y, Z) = \sum_t^T (\beta \mathbb{1}(Z_t \neq Z_{t-1}) + \gamma \mathbb{1}(Y_t \neq Y_{t-1})),$$

where $\mathbb{1}(\cdot)$ is an indicator function that maps "true" values to 1 and "false" values to 0, β is the penalty if $Z_t \neq Z_{t-1}$, and γ is the penalty of switching among the K latent states. Typically, setting β and γ to any values between 0 and 50 will work for z-normalized PMTS.

4.2.3 Objective Function. Based on the loss function and the regularization terms proposed earlier, our overall objective function is to jointly minimize them all:

$$\arg\min_{Y,Z,\{\theta,\hat{\theta}\}>0} \sum_{k}^{K} \|\lambda \circ [v(\theta_k), v(\hat{\theta}_k)]\|_{2,1} + \boldsymbol{h}_{\beta,\gamma}(Y,Z) + \mathcal{L}(Y,Z,\theta,\hat{\theta}).$$

In addition to the regularization parameters λ, β, and γ discussed in Section 4.2.2, K and w can be chosen based on prior knowledge through cross validation or by a principled method such as the Bayesian information criterion [Schwarz et al. 1978]. If the count of subsequences assigned to any latent state is too small (e.g., less than 30) to learn a good θ_k and $\hat{\theta}_k$, this indicates that the value of K should be decreased. Since the short-term temporal dependency is much stronger than the long-term temporal dependency in real-world applications, the window size w should be small (e.g., $w < 10$).

4.3 CDFD Pattern Mining for Multiple PMTS

The CMP model proposed previously focuses on discovering the patterns for a single PMTS, but in many situations there are actually multiple PMTS. For example, when testing an intervention, multiple participants typically will be invited to test for common effects on the population based on all of their corresponding PMTS. In addition, it is required to collectively discover the contrast patterns between control and experimental time series shared by multiple PMTS.

We therefore focus on mining the collective patterns of multiple PMTS by generalizing CMP to a new model named *group CMP*. Given P PMTS, all of the control time series are denoted as $\mathcal{X} = [\mathcal{X}_1, \ldots, \mathcal{X}_P]$, whereas the experimental time series are $\hat{\mathcal{X}} = [\hat{\mathcal{X}}_1, \ldots, \hat{\mathcal{X}}_P]$. For each PMTS $(\mathcal{X}_p, \hat{\mathcal{X}}_p)$, $\hat{\mathcal{X}}_p$ is the experimental time series corresponding to its control $\mathcal{X}_p$. We denote $\Theta_p^{(k)}$ and $\hat{\Theta}_p^{(k)}$ as the contrast inverse covariance matrices of the k^{th} latent state, where $k = 1, \ldots, K$, $p = 1 \ldots, P$ and define $\Theta = \{\Theta_p^{(k)}\}_{p,k}^{P,K}$ and $\hat{\Theta} = \{\hat{\Theta}_p^{(k)}\}_{p,k}^{P,K}$; to discover shared patterns across multiple PMTS, the same latent state assignment and the contrast indicator must be shared by all P pairs and are thus still denoted as Y and Z, respectively. Moreover, as the conditional independencies of the MRFs across all PMTS share the same structure, for any two different pairs p and q, we have

$$\text{supp}(\Theta_q^{(k)}) = \text{supp}(\Theta_p^{(k)}) = \text{supp}(\hat{\Theta}_p^{(k)}) = \text{supp}(\hat{\Theta}_q^{(k)}). \tag{5}$$

Therefore, the problem of GCMP can be formally defined as follows. Given P PMTS, GCMP (1) characterizes the MRFs $\Theta_p^{(k)}$ and $\hat{\Theta}_p^{(k)}$ for each state K and each pair p, (2) detects the shared latent state assignment Y, and (3) identifies the unified contrast indicator Z.

The loss function for P PMTS can be generalized from the loss function for one PMTS defined in Equation (4): $\sum_p^P \mathcal{L}(Y, Z, \Theta_p, \hat{\Theta}_p)$. As defined in Equation (5), the MRFs for different PMTS share the same sparsity pattern, enabling us to propose a new group-based regularization term to enforce the identical sparcity pattern on all $\Theta_p^{(k)}$ and $\hat{\Theta}_p^{(k)}$ such that $\sum_k^K g(\Theta^{(k)}, \hat{\Theta}^{(k)})$, where

$$g(\Theta^{(k)}, \hat{\Theta}^{(k)}) = \|\lambda \circ [v(\Theta_1^{(k)}), v(\hat{\Theta}_1^{(k)}), \ldots, v(\Theta_P^{(k)}), v(\hat{\Theta}_P^{(k)})]\|_{2,1}.$$

Finally, imposing a similar penalty over the latent state assignment Y and contrast indicator Z also enforces their temporal continuity. The overall objective function for the GCMP problem can now be defined as

$$\arg\min_{Y,Z,\Theta,\hat{\Theta}} \sum_{k}^{K} g(\Theta^{(k)}, \hat{\Theta}^{(k)}) + \boldsymbol{h}_{\beta,\gamma}(Y,Z) + \sum_{p}^{P} \mathcal{L}(Y,Z,\Theta_p,\hat{\Theta}_p). \tag{6}$$

ALGORITHM 1: Parameter Optimization for GCMP

Require: $\mathcal{X}$, $\hat{\mathcal{X}}$, λ, β, γ, w
Ensure: solution Y, Z, Θ, $\hat{\Theta}$
1: **repeat**
2: **for** $K = 1$ to K **do**
3: initialize Θ, $\hat{\Theta}$, Q, $\hat{Q}$, U, $\hat{U} \leftarrow \mathbf{0}$
4: **repeat**
5: **for** $p = 1$ to P **do**
6: $\Theta_p^{(k)} \leftarrow$ Equation (9) // update $\Theta_p^{(k)}$
7: $\hat{\Theta}_p^{(k)} \leftarrow$ Equation (10) //update $\hat{\Theta}_p^{(k)}$
8: **end for**
9: **for** $i = 1$ to nw **do**
10: **for** $j = 1$ to i **do**
11: $[Q_{0,i,j}^{(k)}, \hat{Q}_{0,i,j}^{(k)}] \leftarrow$ Equation (11) //update the lower entries
12: $[Q_{0,j,i}^{(k)}, \hat{Q}_{0,j,i}^{(k)}] \leftarrow [Q_{0,i,j}^{(k)}, \hat{Q}_{0,i,j}^{(k)}]$ //make the matrices symmetric
13: **end for**
14: **end for**
15: $U^{(k)}, \hat{U}^{(k)} \leftarrow$ Equation (12)
16: **until** convergence
17: **end for**
18: E-step: optimizing Y and Z is described in Section 5.2
19: **until** Y and Z assignments are stationary

Comparing the objective function in Equation (6) for GCMP with the objective function introduced in Section 4.2.3 for CMP reveals that the GCMP model is actually the generalization of the CMP model and that when $P = 1$, GCMP reduces to CMP.

4.4 Relationship to the Related State-of-the Art Approach

In this section, we show that the current state-of-the-art approach—the TICC [Hallac et al. 2017b] model—is actually a special case of the proposed model.

The TICC approach is only able to solve the second subproblem defined in Section 3 (i.e., determine the latent state assignment Y). In the proposed CPM model, let $Z_t = 1$ for all $t = 1, \ldots T$, which means that no contrast pattern is allowed, and the model is thus reduced to the TICC model:

$$\arg\min_{Y,\theta>0} \sum_{t,k}^{T,K} Y_{t,k}[-\ell\ell(X_t, \theta_k) - \ell\ell(\hat{X}_t, \theta_k)] + \sum_{t}^{T} \gamma \mathbb{1}(Y_t \neq Y_{t-1}) + \sum_{k}^{K} \|\lambda \circ v(\theta_k)\|_1.$$

However, it would not be able to mine the contrast pattern anymore.

5 PARAMETER OPTIMIZATION

In this section, the parameter optimization algorithm for GCMP is presented and its special case CMP solved by simply setting $P = 1$ in our algorithm. Equation (6) is a mixture of the combinational optimization of discrete variables (i.e., Y, Z) and nonconvex nonsmooth optimization of continuous variables (i.e., $\Theta, \hat{\Theta}$). As there is no existing algorithm capable of solving this problem efficiently and effectively, we propose a new algorithm based on EM [Moon 1996] and ADMM [Boyd et al. 2011]. The details are summarized in Algorithm 1 that alternately optimize the continual variables and discrete variables until stationary. The maximization step (M-step) described in lines 3 through 17 jointly optimizes Θ and $\hat{\Theta}$ by adapting the ADMM framework; the expectation step (E-step) is performed in line 18. The M-step and E-step are described in more detail in Section 5.1 and 5.2, respectively.

5.1 M-step: Optimizing $\Theta^{(k)}$ and $\hat{\Theta}^{(k)}$

5.1.1 Decomposing GCMP into K Subproblems. In the M-step, we fix the latent state assignment Y and contrast indicator Z, and optimize $\Theta^{(k)}, \hat{\Theta}^{(k)}$ in parallel, for all K latent states. We therefore rewrite the joint likelihood term as

$$\sum_{p}^{P} \mathcal{L}(Y, Z, \Theta_p, \hat{\Theta}_p) = \sum_{k=1}^{K} \sum_{p=1}^{P} (f(\Theta_p^{(k)}) + \hat{f}(\hat{\Theta}_p^{(k)})) + CONST, \tag{7}$$

where

$$\begin{aligned} f(\Theta_p^{(k)}) =& \frac{1}{2}[|\mathcal{X}_p^{(k,1)}| tr(S(\mathcal{X}_p^{(k,1)})\Theta_p^{(k)}) + |\hat{\mathcal{X}}_p^{(k,1)}| tr(S(\hat{\mathcal{X}}_p^{(k,1)})\Theta_p^{(k)}) \\ &+ |\mathcal{X}_p^{(k,0)}| tr(S(\mathcal{X}_p^{(k,0)})\Theta_p^{(k)}) - (|\mathcal{X}_p^{(k,1)}| + |\hat{\mathcal{X}}_p^{(k,1)}| + |\mathcal{X}_p^{(k,0)}|) \log\det \Theta_p^{(k)}] \\ \hat{f}(\hat{\Theta}_p^{(k)}) =& \frac{1}{2}|\hat{\mathcal{X}}_p^{(k,0)}|[tr(S(\hat{\mathcal{X}}_p^{(k,0)})\hat{\Theta}_p^{(k)}) - \log\det \hat{\Theta}_p^{(k)}]. \end{aligned}$$

Here, P is the count of PMTS; $\mathcal{X}_p^{(k,z)} \in \mathbb{R}^{c \times nw}$ is the matrix that stacks all of the subsequences belonging to the k^{th} latent state with (i.e., $z = 0$) or without (i.e., $z = 1$) CDFD in $\mathcal{X}_p$, where $c = |\mathcal{X}_p^{(k,z)}|$ is the count of these subsequences. In addition, $tr(\cdot)$ is the trace of the matrix, and $S(\cdot)$ is a function that computes the empirical covariance matrix: $S(A) = \frac{1}{|A|}\sum_{r=1}^{|A|} A_r A_r^{\intercal}$.

According to Equation (7), Equation (6) can be optimized separately for each pair of covariances $(\Theta^{(k)}, \hat{\Theta}^{(k)})$ to formulate a graphical lasso problem [Friedman et al. 2008]:

$$\underset{\{\Theta_p^{(k)}, \hat{\Theta}_p^{(k)}\} \succ 0}{\arg\min} \quad g(\Theta^{(k)}, \hat{\Theta}^{(k)}) + \sum_{p}^{P} (f(\Theta_p^{(k)}) + \hat{f}(\hat{\Theta}_p^{(k)})).$$

5.1.2 Solving Graphical Lasso. Solving each graphical lasso problem involves exploring all of the sparse patterns for $(nw)^2$ elements, and there are the K graphical lasso problems to be solved dozens of times before the E-M algorithm converges. However, we notice that the graphical lasso problem can be solved efficiently by adapting the ADMM framework after reformulating into its equivalent form by introducing the consensus variables $Q^{(k)}$ and $\hat{Q}^{(k)}$:

$$\begin{aligned} \underset{\{Q^{(k)}, \hat{Q}^{(k)}, \Theta_p^{(k)}, \hat{\Theta}_p^{(k)}\} \succ 0}{\arg\min} \quad & g(\Theta^{(k)}, \hat{\Theta}^{(k)}) + \sum_{p}^{P} (f(\Theta_p^{(k)}) + \hat{f}(\hat{\Theta}_p^{(k)})) \\ s.t., \; & Q^{(k)} = \Theta^{(k)}, \hat{Q}^{(k)} = \hat{\Theta}^{(k)}, \end{aligned}$$

of which the augmented Lagrangian form [Boyd et al. 2011] is

$$\begin{aligned} L_\rho(\Theta^{(k)}, \hat{\Theta}^{(k)}, Q^{(k)}, \hat{Q}^{(k)}, U^{(k)}, \hat{U}^{(k)}) =& \; g(Q^{(k)}, \hat{Q}^{(k)}) \\ &+ \sum_{p}^{P} (f(\Theta_p^{(k)}) + \hat{f}(\hat{\Theta}_p^{(k)})) - \frac{\rho}{2}\|[U^{(k)}, \hat{U}^{(k)}]\|_F^2 \\ &+ \frac{\rho}{2}\|[\Theta^{(k)}, \hat{\Theta}^{(k)}] - [Q^{(k)}, \hat{Q}^{(k)}] + [U^{(k)}, \hat{U}^{(k)}]\|_F^2, \end{aligned} \tag{8}$$

where $\rho > 0$ is the ADMM [Boyd et al. 2011] penalty parameter and U and $\hat{U}$ are the scaled dual variables.

Equation (8) can be solved by iteratively updating $[\Theta, \hat{\Theta}]$, $[Q, \hat{Q}]$ and $[U, \hat{U}]$ until convergence. Due to the convexity of the objective function and the simplicity of the linear equality constraint,

the convergence is theoretically guaranteed to the global optimal solution. Each subproblem can be solved as described below:

Updating $\Theta^{(k)}$ *and* $\hat{\Theta}^{(k)}$. All P pairs of $\Theta_p^{(k)}$ and $\hat{\Theta}_p^{(k)}$ can be updated in parallel. $\Theta_p^{(k)}$ is updated by solving the following objective function:

$$\arg\min_{\Theta_p^{(k)}} \boldsymbol{f}(\Theta_p^{(k)}) + \frac{\rho}{2}\|\Theta_p^{(k)} - Q_p^{(k)} + U_p^{(k)}\|_F^2.$$

We first set the partial derivative of the target variable $\Theta_p^{(k)}$ to 0, then move the terms with known variables to the right-hand side:

$$2\rho\Theta_p^{(k)} - [|\mathcal{X}_p^{(k,1)}| + |\hat{\mathcal{X}}_p^{(k,1)}| + |\mathcal{X}_p^{(k,0)}|]\Theta_p^{(k)^{-1}} = 2\rho(Q_p^{(k)} - U_p^{(k)}) - [|\mathcal{X}_p^{(k,1)}|S(\mathcal{X}_p^{(k,1)}) + |\hat{\mathcal{X}}_p^{(k,1)}|S(\hat{\mathcal{X}}_p^{(k,1)}) + |\mathcal{X}_p^{(0)}|S(\mathcal{X}_p^{(k,0)})].$$

After performing the eigendecomposition on the right-hand side of the preceding equation, the solution is

$$\Theta_p^{(k)} = D\tilde{\Theta}^{(k)}D^\intercal, \tag{9}$$

where the square matrix D and diagonal matrix Λ are the resulting eigenvectors and eigenvalues of the eigendecomposition, respectively. In addition, $\tilde{\Theta}_{p,ii}^{(k)} = (\Lambda_{ii} + \sqrt{\Lambda_{ii}^2 + 8\rho(|\mathcal{X}_p^{(k,1)}| + |\hat{\mathcal{X}}_p^{(k,1)}| + |\mathcal{X}_p^{(k,0)}|)})/4\rho$.

We update $\hat{\Theta}_p^{(k)}$ by solving the objective function:

$$\arg\min_{\hat{\Theta}_p^{(k)}} \hat{\boldsymbol{f}}(\hat{\Theta}_p^{(k)}) + \frac{\rho}{2}\|\hat{\Theta}_p^{(k)} - \hat{Q}_p^{(k)} + \hat{U}_p^{(k)}\|_F^2.$$

This can be solved as for $\Theta_p^{(k)}$. The solution is

$$\hat{\Theta}_p^{(k)} = D\tilde{\hat{\Theta}}^{(k)}D^\intercal, \tag{10}$$

where the square matrix D and the diagonal matrix Λ are obtained by eigendecomposition: $2\rho(\hat{Q}_p^{(k)} - \hat{U}_p^{(k)}) - |\hat{\mathcal{X}}_p^{(k,0)}| \cdot S(\hat{\mathcal{X}}_p^{(0)}) = D\Lambda D^\intercal$ and $\tilde{\hat{\Theta}}_p^{(k)}$ is the diagonal matrix whose i^{th} element $\tilde{\hat{\Theta}}_{p,ii}^{(k)}$ on the diagonal is $(\Lambda_{ii} + \sqrt{\Lambda_{ii}^2 + 8\rho|\hat{\mathcal{X}}_p^{(k,0)}|})/4\rho$.

Updating $[Q^{(k)}, \hat{Q}^{(k)}]$. $[Q^{(k)}, \hat{Q}^{(k)}]$ is updated by solving the optimization function:

$$\arg\min_{Q^{(k)}, \hat{Q}^{(k)}} g(Q^{(k)}, \hat{Q}^{(k)}) + \frac{\rho}{2}\|[\Theta^{(k)}, \hat{\Theta}^{(k)}] - [Q^{(k)}, \hat{Q}^{(k)}] + [U^{(k)}, \hat{U}^{(k)}]\|_F^2.$$

This minimization problem can be solved by a group soft thresholding operators [Boyd et al. 2011]:

$$[Q_{0,i,j}^{(k)}, \hat{Q}_{0,i,j}^{(k)}] \leftarrow \eta_{\lambda/rho}([\Theta_{0,i,j}^{(k)}, \hat{\Theta}_{0,i,j}^{(k)}] + [U_{0,i,j}^{(k)}, \hat{U}_{0,i,j}^{(k)}]). \tag{11}$$

Here, $B_{0,i,j} \in \mathbb{R}^P$ denotes the vector in a third-order tensor of size $P \times nw \times nw$ where $B \in \{\Theta^{(k)}, \hat{\Theta}^{(k)}, Q^{(k)}, \hat{Q}^{(k)}, U^{(k)}, \hat{U}^{(k)}\}$, $i = 1, 2, \ldots, (nw)$, and $j = 1, \ldots, i$. The group soft thresholding function [Donoho et al. 1993] is defined as $\eta_{\lambda/\rho}(\boldsymbol{a}) = (1 - \frac{\lambda}{\rho\|\boldsymbol{a}\|_2})_+\boldsymbol{a}$.

Updating $[U^{(k)}, \hat{U}^{(k)}]$. $[U^{(k)}, \hat{U}^{(k)}]$ is updated by

$$[U^{(k)}, \hat{U}^{(k)}] \leftarrow [U^{(k)}, \hat{U}^{(k)}] + [\Theta^{(k)}, \hat{\Theta}^{(k)}] - [Q^{(k)}, \hat{Q}^{(k)}]. \tag{12}$$

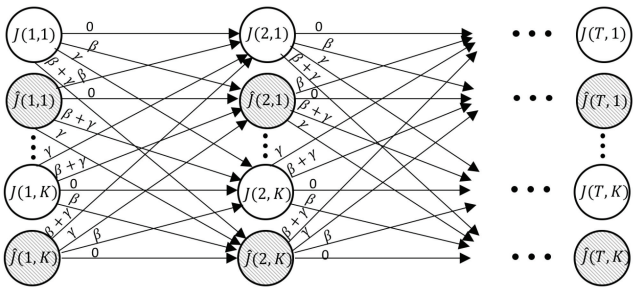

Fig. 3. E-step. Optimizing Y and Z assignments can be solved by selecting one node from each layer (i.e., each column) to minimize the amount cost spent on the nodes and edges.

Q5

5.2 E-step: Optimizing the Y, Z Assignments

In the E-step, we fix $\Theta^{(k)}$ and $\hat{\Theta}^{(k)}$ for all $k = 1, \ldots, K$, and vary the Y and Z assignment for each index t to minimize

$$\arg\min_{Y,Z} \sum_{t}^{T} (\beta \mathbb{1}\{Z_t \neq Z_{t-1}\} + \gamma \mathbb{1}\{Y_t \neq Y_{t-1}\}) + \sum_{t,K}^{T,K} Y_{t,k}[Z_t \hat{J}(t,k) + (1 - Z_t) J(t,k)], \quad (13)$$

where $J(t,k) = \sum_p^P (-\ell\ell(\mathcal{X}_{p,t}, \Theta_p^{(k)}) - \ell\ell(\hat{\mathcal{X}}_{p,t}, \Theta_p^{(k)}))$, $\hat{J}(t,k) = \sum_p^P (-\ell\ell(\mathcal{X}_{p,t}, \Theta_p^{(k)}]) - \ell\ell(\hat{\mathcal{X}}_{p,t}, \hat{\Theta}_p^{(k)}))$.

The assignment optimization problem in the preceding equation can be formulated and solved as a classic problem of finding the minimum cost Viterbi path [Viterbi 1967] in a fully connected network, as shown in Figure 3. Each layer/column t represents the index of the series, and each row represents unique Y and Z assignments. For instance, the node $J(t,k)$ denotes the cost of assigning $Y_{t,k} = 1$ and $Z_t = 1$, and node $\hat{J}(t,k)$ denotes the cost of assigning $Y_{t,k} = 1$ and $Z_t = 0$. The optimization problem in E-step can be solved by finding an optimal path from $t = 1$ to T such that the total cost at the edges and the nodes is minimal, which can be solved by dynamic programming in $O(KT)$ time where the current cost at each node is updated by

$$\begin{aligned} J(t+1,k) &= \min(J_{\min}(t)+\gamma, \hat{J}_{\min}(t)+\beta+\gamma, J(t,k), \hat{J}(t,k)+\beta) \\ \hat{J}(t+1,k) &= \min(\hat{J}_{\min}(t)+\gamma, J_{\min}(t)+\beta+\gamma, \hat{J}(t,k), J(t,k)+\beta), \end{aligned}$$

where $J_{\min}(t)$ and $J_{\min}(t)$ are the minimal costs to the t^{th} layer of all J-nodes and all $\hat{J}$-nodes, respectively. Finally, the shortest path through the network from left to right with minimal cost is recovered by backtracking.

6 EXPERIMENTS

The performance of the proposed models is evaluated on 8 synthetic and 13 real-world datasets in Sections 6.1 and 6.2, respectively. All experiments were conducted on a 64-bit machine with an Intel processor (i7CPU@2.5 GHz) and 16 GB of memory.

6.1 Experiments on Synthetic Datasets

6.1.1 Experimental Setup. The generation process for the synthetic datasets, the comparison methods used, and the parameter settings and evaluation metrics are described in turn next.

Generating the synthetic datasets. The process used to generate four group datasets (i.e., datasets 5 through 8), where each dataset contains seven PMTS (i.e., $P = 7$), is described in the following. In addition, four individual experimental datasets (i.e., datasets 1 through 4) are generated by using

the same process by setting $P = 1$. Each dataset is generated 10 times, then the average performance of 10 repetitive experiments is reported.

(1) Generating the inverse covariance matrices Θ *and* $\hat{\Theta}$. $\Theta_p^{(k)}$ and $\hat{\Theta}_p^{(k)}$ need to be generated for all $p = 1 \ldots P$ and $k = 1 \ldots K$, where K is the number of latent states. To prevent the generated inverse covariance matrices biasing to our model, we follow the generation process described by Hallac et al. [2017b], which enforces the block Toeplitz constraint on the inverse covariance matrix. Specifically, we generate the inverse covariance matrices in three steps. In the first step, an unweighted undirected clique with $n = 5$ nodes is created. In the second step, as described in Figure 1, each latent driving behavior corresponds to a unique sparse structural pattern of its dependency network. To simulate this, $w \cdot K$ unweighted and undirected Erdős-Rényi random graphs $E^{(k,v)}$ [Erdős et al. 2013] are generated by randomly removing 80% of the edges in the clique, where $w = 5$ is the window size; $v = 1, \ldots, w$; and $k = 1, \ldots, K$. Each removed edge, which reflects the conditional independency in the MRFs between the nodes/features connected, lead to a zero value of the inverse covariance matrix that encodes the dependency network or MRF. In the third step, for each random graph $E^{(w,v)}$, P pairs of weighted graphs encoded by adjacent matrices $(\{W_p^{(k,v)}, \hat{W}_p^{(k,v)}\} \in \mathbb{R}^{n \times n})$ that share the identical zero entries are generated by assigning a random weight to every nonzero entry, which simulates various strengths of the dependencies caused by the individual differences on driving behaviors. In the fourth and final step, each pair of the inverse covariance matrices $(\Theta_p^{(k)}, \hat{\Theta}_p^{(k)}$ are generated by constructing a pair of $wn \times wn$ Toeplitz matrices using $(\{W_p^{(k,v)}, \hat{W}_p^{(k,v)}\}$. To ensure inversibility, the values in the generated inverse covariance matrices are adjusted by $\Theta_p^{(k)} = \Theta_p^{(k)} + (0.1 + |e|)I$ and $\hat{\Theta}_p^{(k)} = \hat{\Theta}_p^{(k)} + (0.1 + |\hat{e}|)I$, where e and $\hat{e}$ are the smallest eigenvalues of the corresponding $\Theta_p^{(k)}$ and $\hat{\Theta}_p^{(k)}$, respectively.

(2) Generating labels for the latent state assignment Y *and contrast pattern indicator* Z. To simulate the temporal dependency of the time series in the real world, we first select a sequence of segments for the Y assignments. For example, the sequence of "1,2,1" denotes three segments assigned to $K = 2$ latent states, where "1" and "2" denote the Latent States 1 and 2, respectively. Let each segment contain $100 * K$ time points. The latent state assignments $Y_{t,k}$ for $t = 1, \ldots, 200$ would be $Y_{t,1} = 1$, and for $t = 201, \ldots, 400$ and $t = 401, \ldots, 600$ would be $Y_{t,2} = 1$ and $Y_{t,1} = 1$, respectively. Following this rationale, the datasets used in this section are generated from four segment sequences: "1,2,1," "1,2,3,2,1," "1,2,3,4,1,2,3,4," and "1,2,2,1,3,3,3,1." The dataset for each sequence is generated 10 times for repeating the experiments to get the average result. To determine the sequence of the Z assignments, the time points that belong to the 1/4 to 3/4 interval of each segment are assigned to 0 (i.e., include CDFDs), and the remaining time points are assigned to 1. Finally, 50% CDFDs are intentionally removed from two out of seven PMTS to simulate the noise of which some PMTS do not contain CDFD.

(3) Generate PMTS. Given $\Theta, \hat{\Theta}$, Y, and Z, the process of generating PMTS is the same as that described in Section 4.1. Specifically, if $Y_{t,k} = 1$ and $Z_t = 1$, $\Theta_p^{(k)}$ is used to generate $\mathcal{X}_{p,t}^{(k)}$ and $\hat{\mathcal{X}}_{p,t}^{(k)}$. However, if $Y_{t,k} = 1$ and $Z_t = 0$, $\Theta_p^{(k)}$ is used to generate $\mathcal{X}_{p,t}^{(k)}$, and $\hat{\Theta}_p^{(k)}$ is used to generate $\hat{\mathcal{X}}_{p,t}^{(k)}$. After generating all of the PMTS data, the uniformly distributed noises between $[-0.5\sigma, 0.5\sigma]$ are added to all observations, where $\sigma \in \mathbb{R}^n$ is the standard deviation of each multivariate time series.

Evaluation metrics. To evaluate and compare the effectiveness of the proposed methods and other baseline methods on PMTS, the predicted Y and Z assignments are compared with the Y and Z assignments used to generate the PMTS. To ensure a fair comparison of the effectiveness of the baseline methods with our method, the number of latent states K in all of the methods is fixed to the corresponding K used to generate the datasets, thus ensuring that all methods would be evaluated as a K-class classification problem for Y assignments and a two-class classification problem for Z assignments. Therefore, the macro F1 scores for the Y assignments are computed

for all of the methods, where the macro F1 score is defined as the average of the K F1 scores where each is the harmonic mean of the precision and recall for predicting each class of Y assignment. The Z assignments are evaluated using F1 scores: the closer the (macro) F1 score to 1, the better the result.

Comparison methods. To the best of our knowledge, as yet there is no integrated method capable of mining CDFD for PMTS generated from controlled experiments. The baseline methods therefore require a two-step procedure to decide the Y assignments and Z assignments separately. For step 1 to determine the Y assignments, two methods are considered: GMM [Banfield and Raftery 1993] and the state-of-the-art TICC [Hallac et al. 2017b] introduced in Section 2. For step 2 to determine the Z assignments, this can be considered as a two-group partitioning problem over the subsequence pairs in PMTS. Three distance-based methods and one model-based method are compared with our approach. First, in distance-based methods, for each latent state obtained from step 1, the distances of all subsequence pairs are computed using three distance measures for multivariate time series, namely the Euclidean distance, dynamic time warping–dependent (DTW-D) distance [Shokoohi-Yekta et al. 2017], and dynamic time warping–independent (DTW-I) [Shokoohi-Yekta et al. 2017] distance. The computed distances are then sorted in descending order, and the pairs with the top-i largest distances are assigned to contain CDFDs (i.e., $Z_t = 0$). The macro F1 scores are computed for all possible values of i, and the maximal macro F1 scores of the baseline methods are reported in the tables. Second, in model-based methods, for each latent state obtained from step 1, the two-component GMM [Banfield and Raftery 1993] is used to partition all subsequences in both the control and experimental time series belonging to the same latent state into two groups. For each subsequence pair $(X_t, \hat{X}_t)$, if X_t and $\hat{X}_t$ are partitioned into different groups, Z_t is assigned to 0 (i.e., existing CDFD); otherwise, $Z_t = 1$. In other words, the values of Z_t are decided by an XNOR gate. Third, in the baselines using ground truth latent state assignment, to explore the performance of the distance-based and model-based methods only on the subproblem of contrast pattern detection (i.e., Z assignment), we also evaluate the comparison method by starting with the ground truth latent state assignments.

Parameter settings. In effectiveness evaluation, $\lambda = 0.5, \beta = 1, \gamma = 3$ are used for our methods. For the TICC method, the parameters are intensively tuned to achieve the best performance. For a fair evaluation of the effectiveness, the values of K and w are set the same as those used to generate the synthetic data for all methods.

6.1.2 Performance on Synthetic Datasets. In this section, the effectiveness of the baseline methods and the proposed CMP and GCMP are evaluated, and the scalability and the parameter sensitivities of the proposed approaches are tested.

Effectiveness evaluation. The results of the effectiveness evaluation on Y assignments are shown in Table 2(a) and (b) for the individual and group datasets, respectively. Table 2(c) and (d) list the effectiveness evaluation results for Z assignments, where the two-step comparison methods witha plus sign show the results of Z assignments based on the Y assignments predicted by the first step, and the comparison methods withouta plus sign show the results of Z assignments based on the ground truth latent state assignments.

As the results show, our integrated methods outperform the comparison methods for both the Y and Z assignments, whereas none of the other methods perform well on the Z assignments because they are unable to capture the dependency between the latent states and the CDFD patterns. As shown in Table 2(a) and (b), the macro F1 scores of our models on Y (i.e., latent state) assignments achieve the highest macro F1 scores of 0.960 on average, whereas the best comparison method can only achieve 0.860. These results are impressive considering that the data are noisy and are generated by the Toeplitz inverse covariance matrix that is not assumed by our models. In contrast,

Table 2. Effectiveness Performance

(a) Macro F1 scores and running time in seconds of latent state assignments Y on one PMTS

Individual Datasets	Dataset 1		Dataset 2		Dataset 3		Dataset 4	
Method	F1	Time	F1	Time	F1	Time	F1	Time
TICC	0.519	3.83 s	0.375	7.61 s	0.284	13.13 s	0.355	9.80 s
GMM	0.954	0.02 s	0.798	0.08 s	0.596	0.12 s	0.766	0.07 s
CMP (ours)	**0.992**	5.54 s	**0.940**	12.83 s	**0.889**	22.81 s	**0.885**	12.25 s

(b) Macro F1 scores and running time in seconds of latent state assignment Y on multiple PMTS

Group Datasets	Dataset 5		Dataset 6		Dataset 7		Dataset 8	
Method	F1	Time	F-1	Time	F1	Time	F1	Time
TICC	0.945	1.61 s	0.560	21.35 s	0.366	29.91 s	0.531	22.32 s
GMM	0.989	0.02 s	0.943	0.04 s	0.876	0.10 s	0.956	0.06 s
GCMP (ours)	**0.989**	6.47 s	**0.995**	12.83 s	**0.995**	19.55 s	**0.996**	15.12 s

(c) F1 scores and running time in seconds of contrast pattern indicator Z on one PMTS

Individual Datasets	Dataset 1		Dataset 2		Dataset 3		Dataset 4	
Method	F1	Time	F1	Time	F1	Time	F1	Time
GMM+DTW-I	0.391	6.78 s	0.410	11.73 s	0.402	19.84 s	0.386	21.33 s
GMM+Euclidean	0.434	0.43 s	0.436	1.33 s	0.44	2.24 s	0.393	3.29 s
GMM+DTW-D	0.392	2.59 s	0.415	4.69 s	0.390	8.60 s	0.393	10.05 s
TICC+Euclidean	0.491	5.43 s	0.476	10.50 s	0.475	19.64 s	0.497	18.54 s
TICC+DTW-D	0.465	6.51 s	0.470	12.28 s	0.468	22.51 s	0.498	20.92 s
TICC+GMM-XNOR	0.490	3.89 s	0.444	7.73 s	0.461	13.29 s	0.371	9.93 s
TICC+DTW-I	0.451	10.73 s	0.471	19.33 s	0.474	33.74 s	0.437	32.27 s
GMM+GMM-XNOR	0.765	0.11 s	0.706	0.22 s	0.603	0.31 s	0.591	0.27 s
Euclidean	0.462	1.51 s	0.502	2.74 s	0.515	4.88 s	0.477	6.55 s
DTW-D	0.421	2.56 s	0.469	4.46 s	0.484	7.68 s	0.481	9.34 s
DTW-I	0.421	6.68 s	0.479	11.37 s	0.482	18.81 s	0.477	20.29 s
GMM-XOR	0.810	0.08 s	0.799	0.14 s	0.824	0.19 s	0.778	0.21 s
CMP (ours)	**0.869**	5.54 s	**0.882**	10.95 s	**0.886**	22.81 s	**0.843**	12.25 s

(d) F1 scores and running time in seconds of contrast pattern indicator Z on multiple PMTSs

Group Datasets	Dataset 5		Dataset 6		Dataset 7		Dataset 8	
Method	F1	Time	F1	Time	F1	Time	F1	Time
GMM-Euclidean	0.478	0.47 s	0.416	1.24 s	0.391	5.03 s	0.388	4.95 s
GMM-DTW-D	0.472	0.99 s	0.416	2.18 s	0.393	7.31 s	0.402	7.42 s
GMM-DTW-I	0.454	2.43 s	0.411	6.63 s	0.415	17.32 s	0.423	13.12 s
TICC-Euclidean	0.388	2.15 s	0.471	24.86 s	0.543	39.51 s	0.433	29.77 s
TICC-DTW-D	0.388	2.61 s	0.481	25.99 s	0.550	42.14 s	0.440	31.74 s
TICC-DTW-I	0.386	4.40 s	0.473	30.35 s	0.555	51.76 s	0.453	41.05 s
TICC-GMM-XNOR	0.495	1.90 s	0.388	23.42 s	0.419	34.62 s	0.320	25.00 s
GMM-GMM-XNOR	0.469	0.08 s	0.279	0.13 s	0.350	0.29 s	0.342	0.17 s
GCMP (ours)	**0.842**	6.82 s	**0.976**	14.38 s	**0.866**	23.77 s	**0.975**	17.31 s

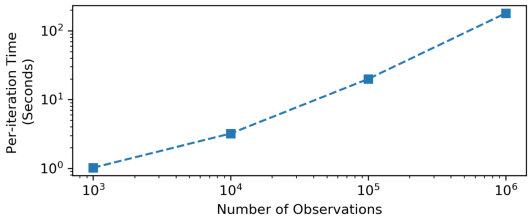

Fig. 4. Per-iteration running time of our algorithm (both E-step and M-step) using a single-thread Python program. Our proposed algorithm scales linearly with the number of time points.

TICC only achieves a macro F1 score at most 0.52 even after we intensively tuned its parameters. GMM runs very fast but performs worse than our models due to the absence of the temporal and structural regularization terms. Notice that the running time of our algorithm, as an integrated method, is not only for Y assignments but also for Z assignments.

The results on Z assignments for one PMTS are shown in Table 2(c) and (d). Our methods achieve an average F1 score of 0.896, whereas the best two-step methods only achieve an average F1 score of 0.513. Even starting with the ground truth Y assignments, the best comparison method only achieves the average F1 score of 0.803, which is still 10% worse than our methods. The distance-based methods are all close to random guess because they are unable to mine the dependency patterns.

In addition, the results for the group datasets validate that our GCMP model is robust enough to capture the CDFDs in noisy data. Furthermore, when the datasets include multiple PMTS, our GCMP model performs even better than the CMP model. This is because by adding an $L_{2,1}$-norm regularization term to the probabilistic model, the GCMP model is able to take all of the PMTS data into account while maintaining the dependency pattern among all MRFs. It is very important to utilize all available data in controlled experiments that typically require the data generated by a group of participants.

Scalability analysis. One iteration of our E-M-style algorithm consists of optimizing the Y and Z assignments in the E-step whose complexity is $O(KT)$ as described in the previous section, and optimizing Θ and $\hat{\Theta}$ in the M-step of whose complexity is $O(T)$ for computing the empirical covariances plus $O((nw)^2)$ for our ADMM algorithm. Typically, our ADMM algorithm will give a good enough solution [Boyd et al. 2011] after a few tens of iterations, so the number of iterations in our ADMM algorithm is considered as a constant number. Moreover, T can potentially be in the millions, which is much larger than K and nw. The total number of iterations of our E-M algorithm depends on the data but typically converges in dozens of iterations and thus can also be considered as a constant number. Therefore, the overall complexity of our algorithm can be considered as $O(T)$ in practice. To validate the scalability of the proposed algorithm, we vary T and compute the running time over one E-M iteration. A large dataset is generated by using n= 10, w = 3, K = 10, and $T_{\max} = 10^6$. The per-iteration running time, which contains both the E-step and M-step, is plotted using a log-log scale in Figure 4. Our algorithm grows almost linearly over T and is able to optimize the PMTS with 2 million data points in about 100 seconds per iteration using a single thread.

Sensitivity tests. The sensitivities of the hyper-parameters, such as w, λ, β, and γ, are tested separately by using a basic setting of $K = 4, \lambda = 10, \beta = 1, \gamma = 3, w = 5$ and varying a single parameter each time. The individual and group datasets used here are all generated by the same sequence, namely datasets 3 and 7. The results of the sensitivity test are plotted in Figure 5. As the figure shows, both of our CMP and GCMP models are relatively insensitive to all parameters within the

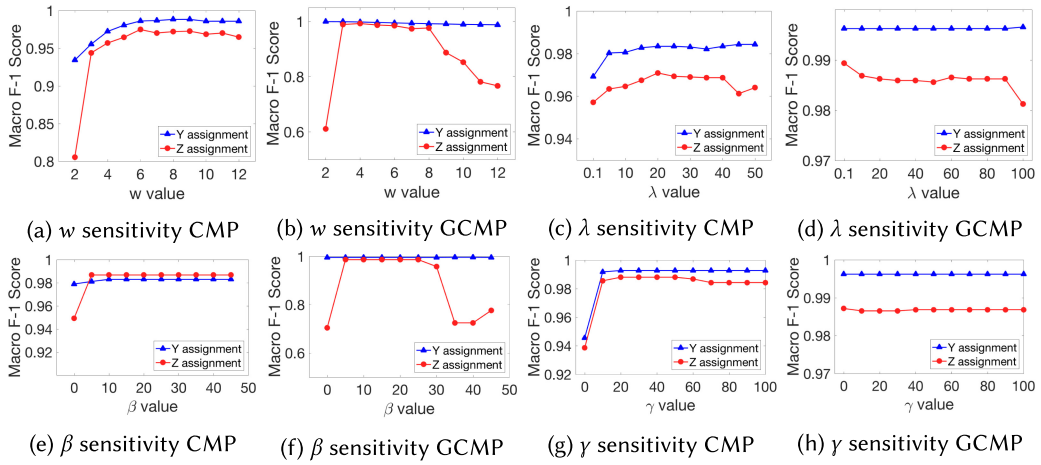

Fig. 5. Sensitivity tests.

range shown. The sensitivities for window sizes w ranging from 2 to 12 are plotted in Figure 5(a) and (b) for the individual and group datasets, respectively. Recall that the "true" window size of the datasets is 5, so when $w = 2$, the macro F1 scores are relatively low since neither models take long-term dependencies into account. When $w > 8$, the performance starts to decrease since the model seeks to estimate long-term dependencies that do not exist in the datasets. The sensitivity for the three regularization parameters are plotted in Figure 5(c) through (h), which demonstrate that any values between 0.1 and 50 work well on the proposed models.

6.2 Experiments on Real-World Datasets

To demonstrate the utility of the CDFD pattern mining task, the proposed CMP and GCMP are applied to a study of contrast driving behaviors by participants diagnosed with attention deficit hyperactivity disorder (ADHD), a disease that influences human driving behaviors, before and after taking their ADHD medication.

6.2.1 Experiment Setup. Thirteen real-world datasets were obtained by monitoring 13 ADHD participants whose driving behaviors were recorded by a high-fidelity driving simulator. Each dataset contains a pair of multivariate time series of driving data under identical traffic scenarios collected before and after the participants took their ADHD medication after a few weeks so that they were unlikely to memorize the previous scenarios. Adding this requirement could prevent influence on driving behaviors caused by memorization, which was an unrelated factor of the controlled experiment. The other detailed protocols of this controlled experiment are described in Lee et al. [2018].

Translating to PMTS. Even though all multivariate time series were generated under the same scenarios, due to the various velocities, these time series did not not perfectly match each other along the time axis. However, the spatial trajectories recorded by their coordinates were very similar, so instead of using timestamp values for the X axis of these PMTS, we used locations ordered by time to bind the multivariate time series to form the PMTS defined in Section 3. These PMTS were therefore translated from the original multivariate time series using the same trajectory to bind all time series. Specifically, all PMTS were dynamically rescaled along the X axis from equal time intervals to equal distance intervals in two steps. The first step entailed randomly selecting one trajectory, then translating it to a *step-invariant trajectory* (SIT) [Li et al. 2017] to serve as

the template trajectory such that the distances between any consecutive points were equal to the step distance parameter δ. Here, we set $\delta = 1$ foot. In the second step, for each spatial point in the template trajectory, the corresponding values of the other sensors were then estimated by linear interpolation to obtain a PMTS dataset whose multivariate time series were all indexed by the same sequence of locations ordered by time.

6.2.2 Performance of CMP. To validate the effectiveness of CMP, the model is applied to an individual dataset with one PMTS. For any value of $K \geq 4$, the model assigns most of the points to four latent states, so let $K = 4$ for this dataset. Each of the resulting latent states can be naturally interpreted as a unique driving state that can be validated by observing the trajectory and the PMTS in Figure 6. For example, the latent state plotted in red in PMTS View can be interpreted as slowing down since the values of the red segments are high in the brake dimension and decreased in the velocity dimension; the orange latent state can be interpreted as turning since all of the orange segments correspond to corners, as highlighted in Trajectory View; the green latent state can be interpreted as driving in a straight line since the values of green segments are high in the gas pedal dimension and close to 0 in the steering dimension, and the blue latent state can be interpreted as switching lanes since the values of the blue segments are high in the steering dimension, then change rapidly to the other direction.

To locate the CDFD, the segments containing CDFD (i.e., $Z_t = 0$) are shaded. Recall that the edge in an MRF represents a *partial correlation* (PC) [Rue and Held 2005] between two connected features. The PC between feature F_1 and feature F_2, denoted as $\boldsymbol{pc}(F_1, F_2)$, measures their "true" correlation, which excludes the effect of the other features. We thus visualize the MRFs by plotting their PC networks. Due to the limited space, only the PC networks corresponding to "turning" are plotted in MRF View in Figure 6. Each node in the PC network represents a feature, and each solid/dashed edge represents a positive/negative PC. Naturally, the CDFD patterns can be visualized by plotting the differences between $\boldsymbol{pc}(\cdot, \cdot)$ (i.e., before medication) and $\widehat{\boldsymbol{pc}}(\cdot, \cdot)$ (i.e., after medication) in the residual PC View in Figure 6 whose weight of the edge between F_1 and F_2 is defined as $\boldsymbol{r}(F_1, F_2) = \widehat{\boldsymbol{pc}}(F_1, F_2) - \boldsymbol{pc}(F_1, F_2)$. All negative/positive weights in the residual PC network are plotted in blue/red, respectively.

The CDFD can be interpreted as the different driving behaviors collected before and after medication. For example, after medication, $\boldsymbol{r}(B_t, B_{t+1})$, $\boldsymbol{r}(G_t, G_{t+1})$, and $\boldsymbol{r}(V_t, V_{t+1})$ are all positive while turning, which means that these sensors at index t are more correlated to themselves at the next index after medication. This could be interpreted as this ADHD driver controlling the gas and brake pedals more smoothly after taking her or his medication, whereas $\boldsymbol{r}(S_t, S_{t+1}) < 0$ suggests that the steering wheel is less correlated to the steering wheel at the next index, indicating that after taking medication, the ADHD participant is more likely to adjust the steering wheel proactively. In addition, $\boldsymbol{r}(V_t, S_t)$ and $\boldsymbol{r}(V_{t+1}, S_{t+1})$ are both negative, which indicates that the velocity is less correlated with the steering wheel, and thus safe handling of the steering wheel, when the velocity is high.

To conclude, the CDFDs showed that before medication, this ADHD participant is more likely to turn the vehicle primarily by adjusting the gas and brake pedals. In contrast, after medication, the same participant is more likely to turn the vehicle by proactively adjusting the steering wheel based on current velocity and adjusting the gas and brake pedals more smoothly.

6.2.3 Performance of GCMP. To validate the effectiveness of GCMP, the model is then applied to the group dataset with all 13 PMTS. The experimental settings are the same as those described Section 6.2.1. As shown in Figure 7, the results of the Y assignments and the interpretations are very similar to those seen previously, as all participants drove under identical traffic scenarios, so the drivers are mostly under the same driving state at the same location. To validate and interpret

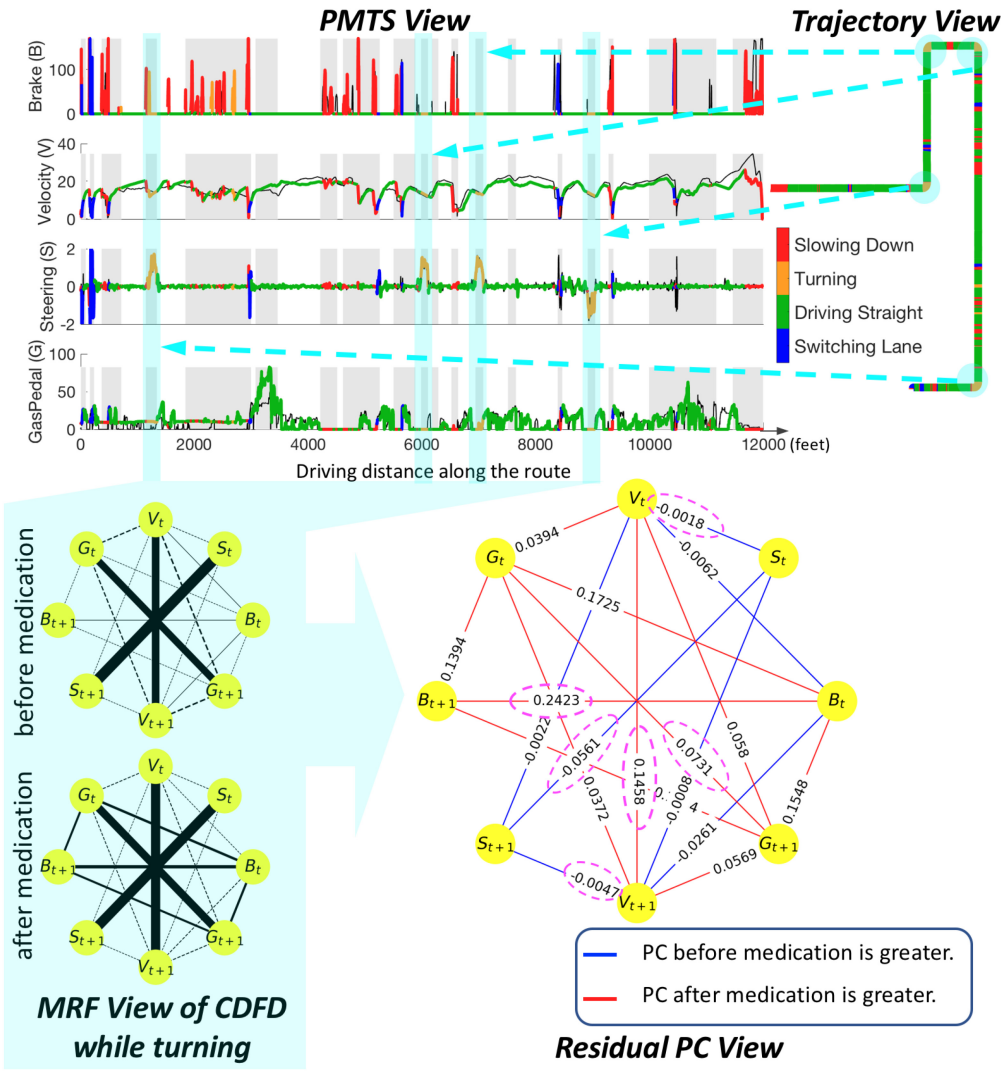

Fig. 6. The contrast patterns, which show some of the driving behaviors changed by the ADHD medication, are plotted in four views. Each latent state is plotted using a unique color in both the Trajectory and PMTS views. In PMTS View, the multivariate time series plotted in four colors recorded the driving behaviors after medication. Since the two multivariate time series in PMTS share the same latent state assignments, the multivariate time series before medication is plotted in black. The road segments with contrast patterns are shaded in grey (i.e., $Z_t = 0$) and/or highlighted in cyan (i.e., $Y_{t,\text{Turning}} = 1$ and $Z_t = 0$). The PC networks of latent state "turning" are plotted in MRF View, and the differences of the PC networks are plotted in Residual PC View.

the CDFD, which contains 13 pairs of MRFs, a paired t-test is performed and plotted in t-Test PC View in Figure 7. The edges in the network denote the existence of significant differences (i.e., the p-value is less than 0.05) between the corresponding PCs before and after medication. Similar to the residual PC network seen previously, the edges in the t-test network are plotted in red if the

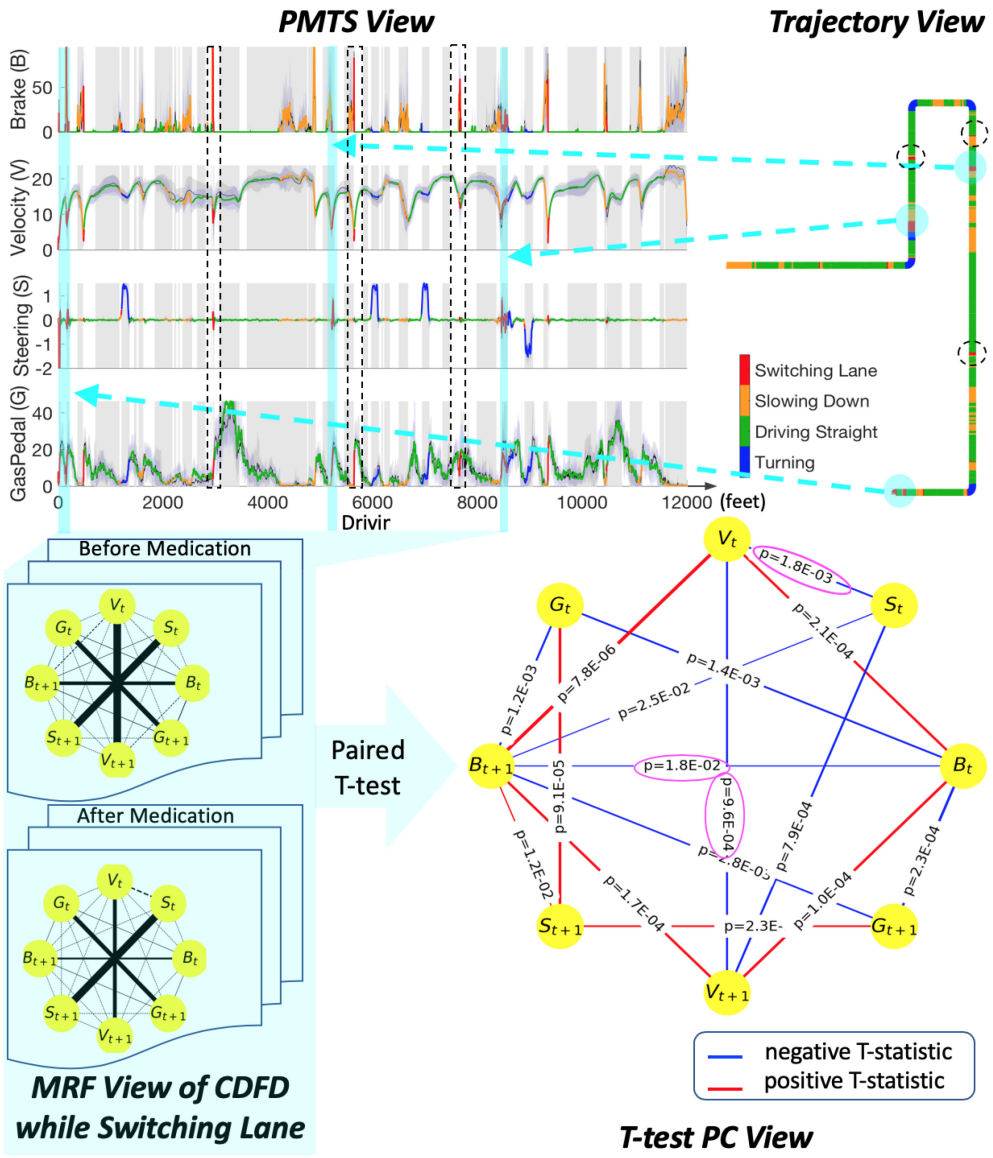

Fig. 7. The group contrast patterns, which show some of the driving behaviors of 13 participants, are changed by the ADHD medication. The views are similar to Figure 6, except (1) the mean and standard deviation of the 13 PMTSs are plotted in PMTS View, (2) 13 pairs of PC networks in the switching lane latent state are plotted in MRF View, and (3) a paired *t*-test is performed on these PC networks in *t*-Test PC View.

PCs increased significantly after medication (i.e., a positive *t*-statistic); otherwise, the edges are plotted in blue.

The driving state for line switching was then analyzed. Our model suggested that some segments that are circled in Trajectory View in Figure 7 do not contain CDFD, and others, which are highlighted in PMTS and Trajectory views, contain CDFD patterns. After examining the original videos, the switching lane state actually contained two cases: passing a slow vehicle and avoiding a sudden

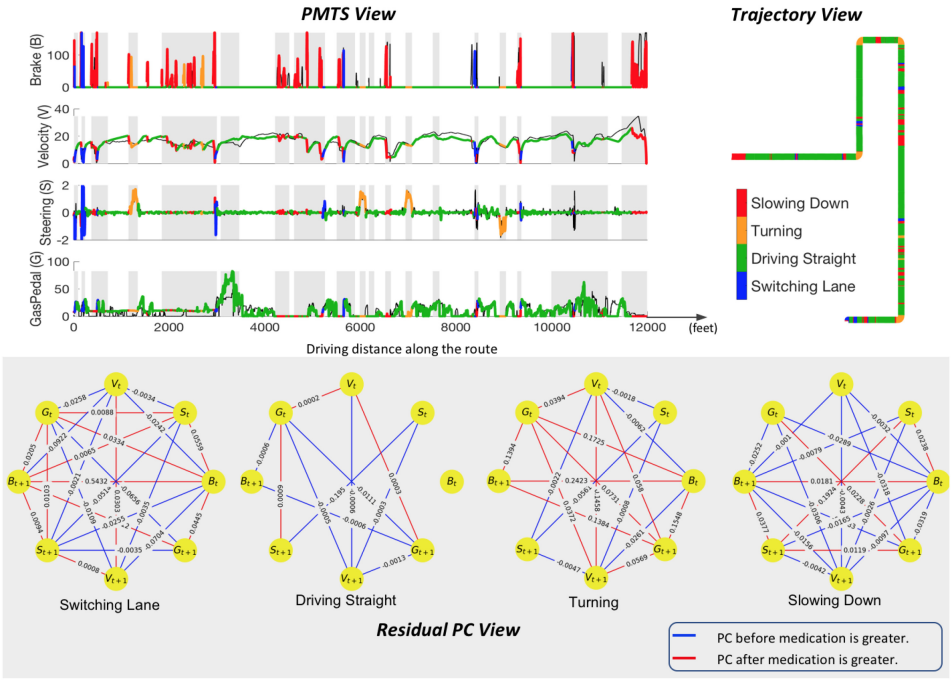

Fig. 8. Driver A.

cut-in vehicle. The segments marked as no CDFD (i.e., $Z_t = 0$) mostly correspond to the former cases, and the CDFD segments correspond to the latter cases. This indicates that the drivers mostly drive in a similar way when they are switching lanes to pass a slow vehicle in both medication conditions but switch lanes in different ways before and after medication when another vehicle suddenly cuts into their current lane. In this case, the $\widehat{PC}(B_t, B_{t+1})$ are significantly (i.e., the p-value is 0.018) less than $PC(B_t, B_{t+1})$, signifying a stronger reaction (i.e., a weaker PC) on the brake pedals when the ADHD participants switch lanes to avoid crashing into the cut-in vehicles after medication, and consequently the $\widehat{PC}(V_t, V_{t+1})$ are also significantly (i.e., the p-value is 0.00096) less than $PC(V_t, V_{t+1})$. Even though all participants successfully avoid crashes with the cut-in vehicles in both medication conditions, their ways of avoiding the cut-in vehicles are quite different between the before and after medication conditions. As t-Test PC View in Figure 7 illustrates, $\widehat{PC}(V_t, S_t)$ is significantly less than $PC(V_t, S_t)$, which means that these ADHD participants are more capable of stabilizing their vehicles when avoiding a crash with the cut-in vehicles after medication.

In conclusion, the CDFDs show that after medication, the ADHD participants react by braking strongly to slow down and stabilize their vehicles when interacting with cut-in vehicles, thus demonstrating better driving behaviors.

6.3 Additional Results on Real-World Datasets

The contrast patterns for 4 out of 13 ADHD participants, namely driver A through driver D, are plotted in Figures 8 through 11. As seen in these figures, although each PMTS is fed to our CMP model independently, the latent state assignments (i.e., Y assignments) and their interpretations are almost the same for all drivers, which validated the effectiveness of our CMP model again. However, their contrast patterns are quite different, which can potentially be used to quantify the

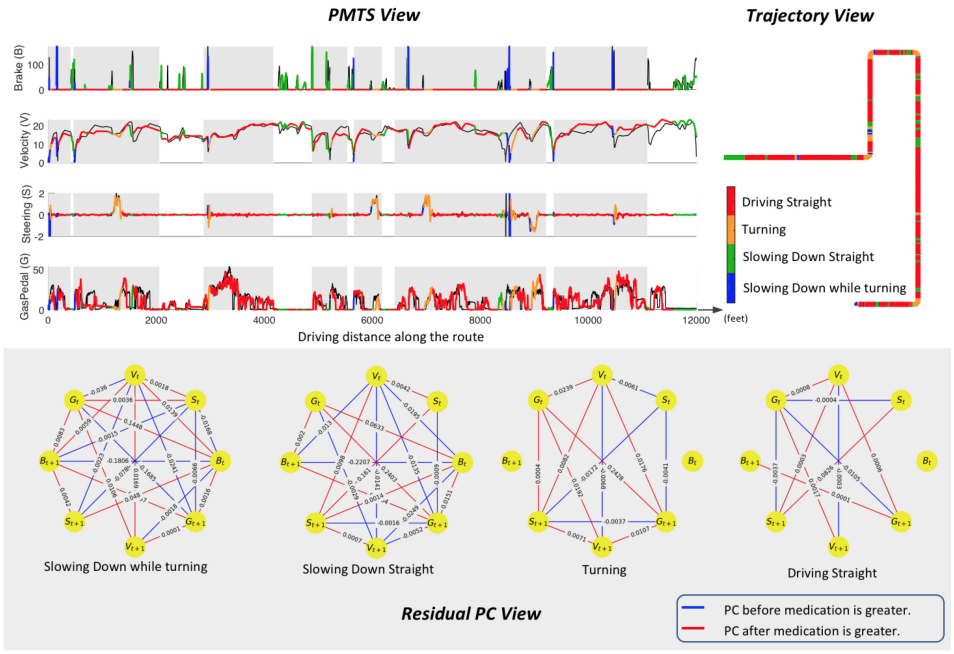

Fig. 9. Driver B.

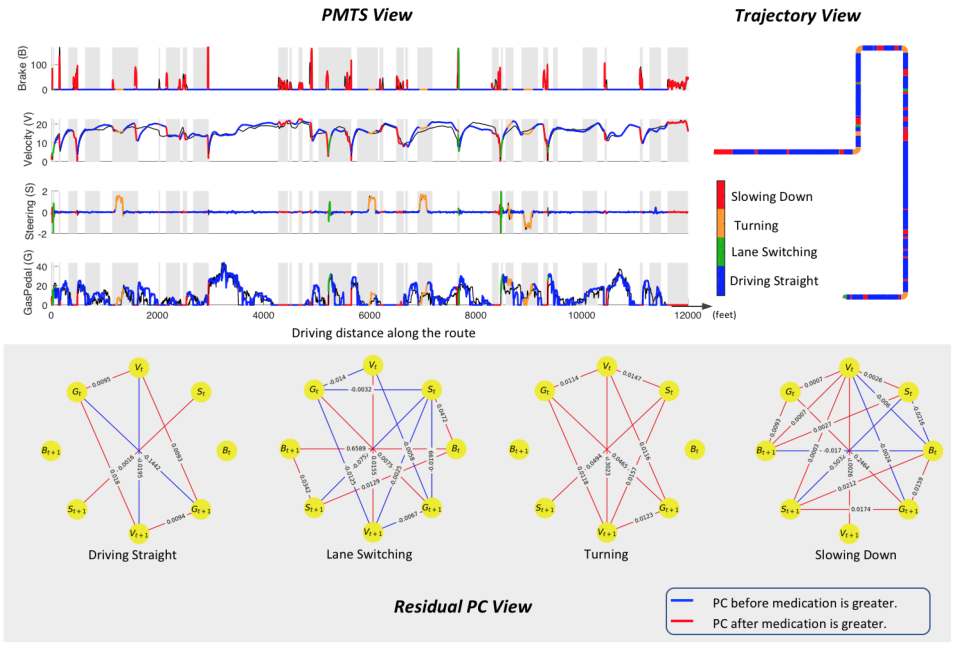

Fig. 10. Driver C.

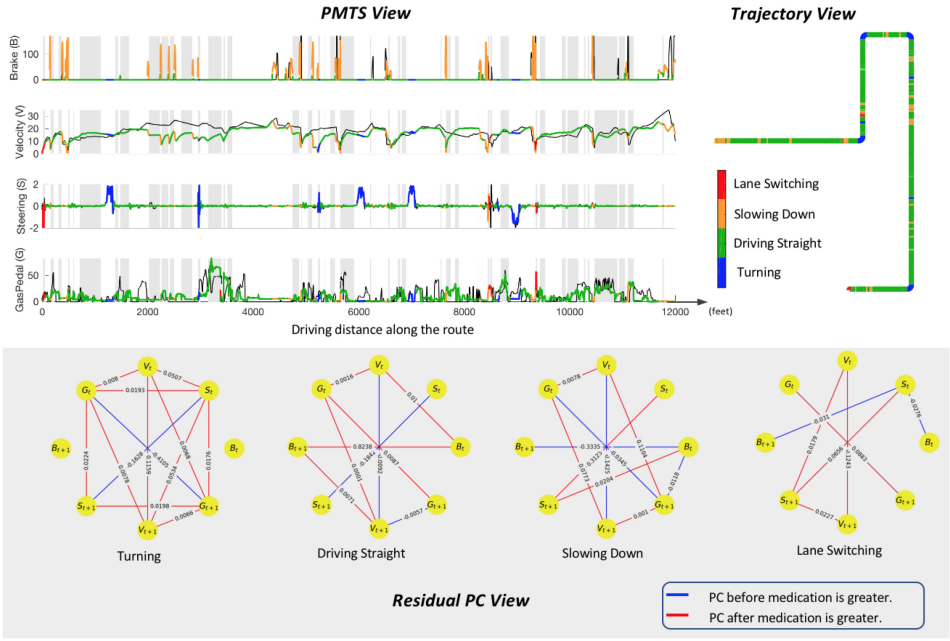

Fig. 11. Driver D.

Table 3. Percentages of Road Segments with Contrast Patterns

Driver	A	B	C	D
e	53.2%	83.4	46.9%	38.7%

effects of the ADHD medication on each ADHD driver's driving behavior. The effects of the ADHD medication can be quantified by our model in two aspects:

(1) $e = \frac{\text{count}(\{t|Z_t=0\})}{T} \times 100\%$, which is the percentage of the road segments with contrast patterns (i.e., the shaded parts plotted in Figures 8 through 11 indicating that the ADHD medication takes effect on the ADHD driver's driving behaviors). Different patients have different sensitivity to the medication: the higher the value of e, the more sensitive is the ADHD medication to the ADHD driver. The results are shown in Table 3. For example, our model suggests that driver B (i.e., $e_B = 83.4\%$) is more sensitive to the ADHD medication than driver D (i.e., $e_D = 38.7\%$).

(2) $r(\cdot,\cdot) = \hat{\boldsymbol{pc}}(\cdot,\cdot) - \boldsymbol{pc}(\cdot,\cdot)$, which quantifies how much difference there is between the driving behaviors before and after medication by the difference of the corresponding PCs. As seen in Figures 8 through 11, the ADHD medication changes the driving behaviors of different ADHD drivers in different ways—that is, after medication, some PCs remain the same, whereas other PCs increase or decrease. More importantly, it is only meaningful to quantify the changes by summarizing all of the subsequences under the same latent state for controlled experiments, which prohibited the traditional methods applied to the contrast pattern mining problem.

7 CONCLUSION

In this article, we proposed a novel framework to mine interpretable CDFD for PMTS in controlled experiments. In this framework, the CDFD pattern mining problem is formulated as an optimization problem that integrates latent state identification, paired dependency network inference, and contrast pattern detection. To model the optimization problem, we proposed a new probabilistic group graphical lasso that forces the identical structure constraint in paired inverse covariance matrices by adding an $L_{2,1}$-norm regularization term. An efficient algorithm based on E-M and ADMM frameworks was also proposed to solve the graphical lasso. Extensive experimental evaluations on synthetic datasets demonstrated the effectiveness, scalability, and robustness of the proposed approach. Additional experiments on real-world datasets demonstrated the utility and interpretability on the mined CDFDs patterns.

REFERENCES

Anurag A. Agrawal and Peter M. Kotanen. 2003. Herbivores and the success of exotic plants: A phylogenetically controlled experiment. *Ecology Letters* 6, 8 (2003), 712–715.

Jeffrey D. Banfield and Adrian E. Raftery. 1993. Model-based Gaussian and non-Gaussian clustering. *Biometrics* 49, 3 (1993), 803–821. http://www.jstor.org/stable/2532201

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning* 3, 1 (2011), 1–122. DOI: https://doi.org/10.1561/2200000016

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B* 39, 1 (1977), 1–38.

D. Donoho, I. Johnstone, and Iain M. Johnstone. 1993. Ideal spatial adaptation by wavelet shrinkage. *Biometrika* 81 (1993), 425–455.

Bowen Du, Yifeng Cui, Yanjie Fu, Runxing Zhong, and Hui Xiong. 2018. SmartTransfer: Modeling the spatiotemporal dynamics of passenger transfers for crowdedness-aware route recommendations. *ACM Transactions on Intelligent Systems and Technology* 9, 6 (2018), 1–26.

László Erdős, Antti Knowles, Horng-Tzer Yau, and Jun Yin. 2013. Spectral statistics of Erdős–Rényi graphs I: Local semicircle law. *Annals of Probability* 41, 3B (2013), 2279–2375.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. 2008. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* 9, 3 (2008), 432–441.

U. Fugiglando, E. Massaro, P. Santi, S. Milardo, K. Abida, R. Stahlmann, F. Netter, and C. Ratti. 2019. Driving behavior analysis through CAN bus data in an uncontrolled environment. *IEEE Transactions on Intelligent Transportation Systems* 20, 2 (Feb. 2019), 737–748. DOI: https://doi.org/10.1109/TITS.2018.2836308

Yuyang Gao and Liang Zhao. 2018. Incomplete label multi-task ordinal regression for spatial event scale forecasting. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*.

Dina Goldin, Ricardo Mardales, and George Nagy. 2006. In search of meaning for time series subsequence clustering: Matching algorithms based on a new distance measure. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management (CIKM'06)*. ACM, New York, NY, 347–356.

David Hallac, Suvrat Bhooshan, Michael Chen, Kacem Abida, Rok Sosic, and Jure Keskovec. 2018. Drive2Vec: Multiscale state-space embedding of vehicular sensor data. In *Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC'18)*. IEEE, Los Alamitos, CA, 3233–3238.

David Hallac, Youngsuk Park, Stephen Boyd, and Jure Leskovec. 2017a. Network inference via the time-varying graphical lasso. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'17)*. 205–213. DOI: https://doi.org/10.1145/3097983.3098037

David Hallac, Sagar Vare, Stephen Boyd, and Jure Leskovec. 2017b. Toeplitz inverse covariance-based clustering of multivariate time series data. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'17)*. 215–223. DOI: https://doi.org/10.1145/3097983.3098060

Mohamed S. Hassan, Walid G. Aref, and Ahmed M. Aly. 2016. Graph indexing for shortest-path finding over dynamic sub-graphs. In *Proceedings of the 2016 International Conference on Management of Data*. 1183–1197.

Alexander Jung, Gabor Hannak, and Norbert Goertz. 2015. Graphical lasso based model selection for time series. *IEEE Signal Processing Letters* 22, 10 (2015), 1781–1785.

E. Keogh, J. Lin, and W. Truppel. 2003. Clustering of time series subsequences is meaningless: Implications for previous and future research. In *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM'03)*. 115–122. DOI: https://doi.org/10.1109/ICDM.2003.1250910

Ross Kindermann and J. Laurie Snell. 1980. *Markov Random Fields and Their Applications*. Vol. 1. Contemporary Mathematics. American Mathematical Society, Washington, DC.

John Boaz Lee, Xiangnan Kong, Yihan Bao, and Constance Moore. 2017. Identifying deep contrasting networks from time series data: Application to brain network analysis. In *Proceedings of the 2017 SIAM International Conference on Data Mining (SIAM'17)*. 543–551.

Yi-Ching Lee, Chelsea Ward McIntosh, Flaura Winston, Thomas Power, Patty Huang, Santiago Ontañón, and Avelino Gonzalez. 2018. Design of an experimental protocol to examine medication non-adherence among young drivers diagnosed with ADHD: A driving simulator study. *Contemporary Clinical Trials Communications* 11 (2018), 149–155.

Qingzhe Li, Jessica Lin, Liang Zhao, and Huzefa Rangwala. 2017. A uniform representation for trajectory learning tasks. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL'17)*. Article 80, 4 pages. DOI: https://doi.org/10.1145/3139958.3140017

Qingzhe Li, Liang Zhao, Yi-Ching Lee, Yanfang Ye, Jessica Lin, and Lingfei Wu. 2019. Contrast feature dependency pattern mining for controlled experiments with application to driving behavior. In *Proceedings of the 2019 IEEE International Conference on Data Mining (ICDM'19)*. IEEE, Los Alamitos, CA, 1192–1197.

Jessica Lin and Eamonn Keogh. 2006. Group SAX: Extending the notion of contrast sets to time series and multimedia data. In *Knowledge Discovery in Databases: PKDD 2006*. Lecture Notes in Computer Science, Vol. 4213. Springer, 284–296.

Xinyue Liu, Xiangnan Kong, and Ann B. Ragin. 2017. Unified and contrasting graphical lasso for brain network discovery. In *Proceedings of the 2017 SIAM International Conference on Data Mining (SIAM'17)*. 180–188.

Yasuko Matsubara, Yasushi Sakurai, and Christos Faloutsos. 2014. AutoPlait: Automatic mining of co-evolving time sequences. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data (SIGMOD'14)*. ACM, New York, NY, 193–204. DOI: https://doi.org/10.1145/2588555.2588556

Todd K. Moon. 1996. The expectation-maximization algorithm. *IEEE Signal Processing Magazine* 13, 6 (1996), 47–60.

Thanawin Rakthanmanon, Eamonn J. Keogh, Stefano Lonardi, and Scott Evans. 2012. MDL-based time series clustering. *Knowledge and Information Systems* 33, 2 (2012), 371–399.

Havard Rue and Leonhard Held. 2005. *Gaussian Markov Random Fields: Theory and Applications*. Monographs on Statistics and Applied Probability. Chapman & Hall.

Gideon Schwarz. 1978. Estimating the dimension of a model. *Annals of Statistics* 6, 2 (1978), 461–464.

Mohammad Shokoohi-Yekta, Bing Hu, Hongxia Jin, Jun Wang, and Eamonn Keogh. 2017. Generalizing DTW to the multi-dimensional case requires an adaptive approach. *Data Mining and Knowledge Discovery* 31, 1 (2017), 1–31.

Padhraic Smyth. 1997. Clustering sequences with hidden Markov models. In *Advances in Neural Information Processing Systems 9*. 648–654. http://papers.nips.cc/paper/1217-clustering-sequences-with-hidden-markov-models.pdf.

Robert Tibshirani. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* 58, 1 (1996), 267–288.

Erica C. G. van Geffen, Daphne Philbert, Carla van Boheemen, Liset van Dijk, Marieke B. Bos, and Marcel L. Bouvy. 2011. Patients' satisfaction with information and experiences with counseling on cardiovascular medication received at the pharmacy. *Patient Education and Counseling* 83, 3 (2011), 303–309.

Vivek Veeriah, Rohit Durvasula, and Guo-Jun Qi. 2015. Deep learning architecture with dynamically programmed layers for brain connectome prediction. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'15)*. 1205–1214. DOI: https://doi.org/10.1145/2783258.2783399

Afonso Vilaca, Pedro Cunha, and André L. Ferreira. 2017. Systematic literature review on driving behavior. In *Proceedings of the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC'17)*. IEEE, Los Alamitos, CA, 1–8.

Andrew Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory* 13, 2 (1967), 260–269.

Junxiang Wang, Yuyang Gao, Andreas Züfle, Jingyuan Yang, and Liang Zhao. 2018b. Incomplete label uncertainty estimation for petition victory prediction with dynamic features. In *Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM'18)*. IEEE, Los Alamitos, CA, 537–546.

Pengyang Wang, Yanjie Fu, Jiawei Zhang, Pengfei Wang, Yu Zheng, and Charu Aggarwal. 2018a. You are how you drive: Peer and temporal-aware representation learning for driving behavior analysis. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, 2457–2466.

Pengyang Wang, Xiaolin Li, Yu Zheng, Charu Aggarwal, and Yanjie Fu. 2019. Spatiotemporal representation learning for driving behavior analysis: A joint perspective of peer and temporal dependencies. *IEEE Transactions on Knowledge and Data Engineering*. Early Access. August 14, 2019.

Xing Wang, Jessica Lin, Pavel Senin, Tim Oates, Sunil Gandhi, Arnold P. Boedihardjo, Crystal Chen, and Susan Frankenstein. 2016. RPM: Representative pattern mining for efficient time series classification. In *Proceedings of the 19th International Conference on Extending Database Technology (EDBT'16)*.

Yimin Xiong and Dit-Yan Yeung. 2004. Time series clustering with ARMA mixtures. *Pattern Recognition* 37, 8 (2004), 1675–1689. DOI: https://doi.org/10.1016/j.patcog.2003.12.018

Lexiang Ye and Eamonn Keogh. 2009. Time series shapelets: A new primitive for data mining. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'09)*. 947–956. DOI: https://doi.org/10.1145/1557019.1557122

T. P. Yuen, H. Wong, and K. F. C. Yiu. 2018. On constrained estimation of graphical time series models. *Computational Statistics & Data Analysis* 124 (2018), 27–52.

L. Zhao, F. Chen, and Y. Ye. 2019. Efficient learning with exponentially-many conjunctive precursors to forecast spatial events. *IEEE Transactions on Knowledge and Data Engineering*. Early Access. April 23, 2019.

Liang Zhao, Olga Gkountouna, and Dieter Pfoser. 2019. Spatial auto-regressive dependency interpretable learning based on spatial topological constraints. *ACM Transactions on Spatial Algorithms and Systems* 5, 3 (2019), 1–28.

Received November 2018; revised April 2020; accepted April 2020

Author Queries

Q1: AU: Please review this article very carefully for clarity.

Q2: AU: Please rephrase sentence for clarity: "Hence, although cross the controlled..."

Q3: AU: Please rephrase sentence for clarity and to create a complete sentence: "For example, because alcohol can increase..."

Q4: AU: Please rephrase for clarity: "...as this will cause some learned..."

Q5: AU: Please confirm phrasing in the legend to Fig. 3: "...minimize the amount cost spent..."

Q6: AU: "Three steps" are mentioned, yet numbers (1) through (4) followed in the original text. Please review and revise as necessary.

Q7: AU: Please note that there is an opening parenthesis before the brace, yet there is no closing parenthesis. Please review.

Q8: AU: Please check clarity carefully in Section 6.2.1.