Available online at www.sciencedirect.com

ScienceDirect

Computational Statistics & Data Analysis 00 (2023) 1–27

Journal Logo

A novel spatio-temporal clustering algorithm with applications on COVID-19 data from the United States

Soudeep Deb

Decision Sciences Area, Indian Institute of Management Bangalore, Bannerghatta Main Road, Bengaluru, KA 560076, India

Sayar Karmakar*

Department of Statistics, University of Florida, 230 Newell Drive, Gainesville, FL 32611, USA

Abstract

A new clustering algorithm for spatio-temporal data is developed. The proposed method leverages a weighted combination of a spatial haversine distance matrix and a spectral-density based temporal distance matrix between the locations. Concepts of partition around medoids algorithm and the gap statistic are utilized to develop the algorithm and to determine the optimal number of clusters. Such a non-parametric algorithm is novel as it incorporates both spatial and temporal distances of the units and it can work for time-series of possibly different lengths. Theoretical guarantee of consistency of the proposed method is provided. An elaborate simulation study is also given to demonstrate the efficacy of the algorithm. As an interesting real life application, the proposed algorithm is implemented to analyze the spatio-temporal dynamics of the time series of coronavirus (COVID-19) incidence rates observed at county-level in the United States of America. The results are demonstrated on datasets of different sizes: the entire country, the Midwest region and the state of California. Special emphasis is given on the last two cases to display how the clustering results offer interesting insights into the epidemic progression in these areas. Particularly, it sheds light on whether state-mandated restrictions impacted the entire state similarly or if there are interesting local behaviors in terms of the COVID-19 spread.

Keywords: Clustering algorithm, Coronavirus, Gap statistic, PAM, Spatio-temporal, Spectral density

1. Introduction

1.1. A motivating example

The recent outbreak of the novel coronavirus (hereafter denoted as COVID-19) has affected millions of lives in one way or the other. After a year of rampage, death, extreme measures, and a humongous effort that produced vaccines, the numbers are still rising with the virus possibly mutating to stronger variants. The patterns in which the numbers are growing are widely varying based on geographical locations; not only because of epidemiological factors, but also due to different government policies and in general human mobility in response to that. Although vaccination has definitely left its mark on bringing down the numbers, newer variants are shown to escape the protection quite a bit and the risk of this contagious disease is far from over. It is naturally of paramount importance to understand how

*Corresponding author: Sayar Karmakar, Department of Statistics, University of Florida, 230 Newell Drive, Gainesville, FL 32611, USA. Email: sayarkarmakar@ufl.edu, Phone: +1 352-273-1839.

Email addresses: soudeep@iimb.ac.in (Soudeep Deb), sayarkarmakar@ufl.edu (Sayar Karmakar)

the disease has been spreading across space and time. To that end, this article is aimed to comprehensively study the spread of COVID-19 since its inception to identify clusters of locations that can explain different spatial and temporal variations otherwise not understandable from the raw data. We demonstrate how our proposed algorithm is able to render valuable insights about the spatial and temporal closeness of the COVID-19 spread. In particular, we show that certain state boundaries get recreated by obtained cluster boundaries which says that our technique can somewhat extract impact of locally adapted mandates.

We consider county-level data from the contiguous United States of America (USA) and observe the daily incidence rate of COVID-19 from the very beginning of the pandemic. Recall that, in epidemiology, incidence is defined as the proportion of new cases in the population. Our primary goal is to understand the similarity pattern in the individual time-series for all the counties. We undertake a clustering method to do so instead of any model based approach. Our motivations emanate from the practical considerations of how a contagious disease might propagate, what sort of data are readily available and some observatory validation from the early days of the pandemic. USA is an extremely vast land with wide diversity in terms of living conditions, population density, demographics, weather etc. All such external events can influence the number of cases a particular region experiences. Furthermore, these geographical sub-regions do not behave in an independent way. There is a significant spatial correlation among the time-series of COVID-19 progression no matter at what granular level the data is (see Figure 2 in Section 2). It is caused by specific state-mandated restrictions as well as human mobility across the counties in neighboring states. Thus, in the proposed algorithm, we ensure that the spatial closeness of two locations has an effect irrespective of whether they are from the same state. It facilitates us to identify if the counties in both sides of the state borders are experiencing similar COVID-19 spread. Subsequently, specific rules can be strongly enforced in the borders. We further investigate if an entire state gets divided into small number of clusters. An affirmative answer to these for most of the states, which is in fact what we obtain, means that overall temporal diversity throughout a state is not quite varied. This can have important implications for state governments in terms of adopting new strategies for different sub-regions within a state. Our algorithm also provides valuable insights about the clusters around major cities. We shall see the case of Chicago in more detail in our application. It helps us understand the effect of a big city in the spread of the disease.

To the best of our knowledge, no one has yet attempted to address the similarity or dissimilarity between the space and time dimensions for COVID-19 data and how that can affect meaningful clustering. A succinct overview of existing literature is provided next in this regard.

1.2. Relevant literature

With the availability of a wide array of relevant data across the world and with the urgent need to understand the dynamics, there has been a myriad of statistical analyses with COVID-19 data. While a majority of these analyses fit various parametric models in a bid to forecast the numbers in coming days, some other works focused on understanding how the numbers are propagating. We skip these as they are not relevant to the focus of our article. In the parlance of model-based spatio-temporal analysis of COVID-19 progression, the works by [1, 2, 3, 4] are worth mention. On the other hand, [5] explored a scan-type statistics to understand the clusters in space and time for New York City. Their focus was on how a particular social gathering leads to a spread and no particular methodological novelty was proposed therein. [6] studied locational Hoover index and its decline over time to explore COVID-19 spread across Brazil. [7] analyzed the effect of COVID-19 on human mobility in the United States of America (USA). They used state-level network analysis and clustering using the mobility data from Multiscale Dynamic Human Mobility Flow Dataset. Spatio-temporal distribution of COVID-19 infection in England between January and June 2020 was provided in [8] using a kernel density-based spatial variation at different timestamps.

Standard exploratory analysis techniques have been used extensively to analyze COVID-19 as well. [9] used Moran Scatter Plots to locate the COVID-19 spread on the Saudi Arabia map for spatial analysis. [10] performed a spatio-temporal hotspot analysis to simulate a real-time surveillance scenario on a hospital network data from Rhode Island, USA. [11] explored the numbers of COVID-19 in India using Voronoi statistics, spatial autocorrelation techniques from Getis-ord G. [12] adopted emerging hotspot analysis and space-time cube models for a dataset from East Java, Indonesia. Identical approach was utilized in [13] and [14] as well. In essence, these analyses use traditional methods such as local Moran's I statistics for a gridded data or emerging hotspot analysis to find clusters in space and time. Some more recent explorations of similar flavor can be found at [15, 16, 17, 18, 19, 20] etc. among others.

Steering our focus to the literature of general spatio-temporal clustering algorithms, we note that they can be broadly categorized into four categories, first of which is focused on the discovery of groups of events that are close

to each other with respect to space, time and other attributes. A special subclass of these methods appear through space-time scan statistics, cf. [21, 22, 23, 24]. This type of event clustering algorithms are commonly used for analyzing spatio-temporal datasets in the geography literature and are popularly referred to as hotspot analysis. [25] and [26] are two interesting examples in this regard. Second category of algorithms deals with geo-referenced time series data and tries to identify groups of objects that are similar to each other in respect of spatial and non-spatial attributes at any given instance of time. The main tool used here is based on kernel density, and the most famous example is ST-DBSCAN by [27]). This was further developed by [28] to offer ST-Optics and by [29] to offer dual constraint clustering approach. Taking a slight detour, [30] introduced the concept and algorithmic framework of fuzzy clustering for geo-referenced time series data. Meanwhile, [31] proposed a clustering method based on the NeuCube spiking neural network architecture for dynamic spatio-temporal brain data. For the third category, namely, trajectory clustering, the aim is to detect groups of objects that have similar movement behaviour along time. [32] developed a clustering technique on the basis of a mixture model for continuous trajectories. [33] presented a family of algorithms to simultaneously align the spatial and temporal shifts of trajectories within each cluster. There is also a brunch of spatial clustering of functional data that could be relevant here as time series in each location can be viewed as a functional data. See [34, 35, 36, 37, 38] etc. among others.

Finally, we turn attention to the methods that rely on model-based clustering for spatio-temporal data. [39] proposed a model-based approach for clustering time-series data in which the cluster representative is expressed by hidden Markov models to estimate transitions between successive positions. [40] considered a Poisson count model with possible cluster structure and performed variable selection to arrive at the final clusters. Towards spatio-temporal clustering, [41] proposed a Bayesian semiparametric mixture model. [42] developed a modified version of the state-space model to study the temporal coherence and thus inducing the clusters. [43, 44] proposed a finite space-time mixture model and allowed spatio-temporally varying mixing weights to achieve their clustering. Other works using finite mixture models can be found in [45, 46] and [47]. The work by [48] is also interesting in this regard. The authors include spatio-temporal random effects using a conditional autoregressive prior and then cluster the spatial unit by a nonparametric prior. Other Bayesian spatio-temporal works can be found in papers of [49, 50, 51, 52, 53, 54] among others. A varying-coefficient model for spatial data was brought to a spatio-temporal setting to obtain time-varying clusters in [55]. Recasting clustering as a high-dimensional problem, [24] used some regularized methods alongside quasi-Poisson regression and scan statistic. For some related methods in this aspect, the reader is referred to the earlier works of [56] and [57]. Some authors also used tree-based or hierarchical clustering approach on spatio-temporal data such as [58, 59, 60].

1.3. Our contributions

It is critical to differentiate what spatio-temporal clustering means for most of the existing research versus what it means for this article. In existing literature, authors have primarily explored the concepts of hotspots that show an increased volume/intensity in a particular region and time-horizon. On the contrary, with the above setup in mind, our focus remains to find which counties (replaceable by any geographical units) show similar progression over time and are spatially close as well. Such disintegration of dimensions can also be linked to the study of separability that exists in the spatio-temporal literature. Currently available studies only try to understand how the joint covariance matrix decomposes into individual covariances for spatio-temporal data. Here, rather than trying to understand whether the two dimensions can be assumed to act on the data 'independently', we create a bridge between these two distances and optimally choose weights for both the dimensions that provides us the best possible clustering information. Finally, our approach is completely non-parametric here and free of rather restrictive assumptions regarding spatial and temporal effects.

Methodologically, we first show that ACF or correlation-based similarity cannot lead to interpretable clusters. They appear somewhat abruptly without any discernible spatial continuity. Keeping this in mind, we combine the spatial distance and the temporal distance between two locations, where the former is calculated using the haversine formula and the latter is computed using the distance between spectral density estimates which reflect the covariance structures of the time series. Next, we use this bridged distance to obtain meaningful clusters. The theoretical consistency of the proposed algorithm is proved. For practical applications, we utilize the concepts of gap statistic for choosing the optimum number of clusters. We show in detail how the proposed distance metric and the clustering algorithm offer flexibility, objectivity, and better interpretability to understand the dynamics of such a contagious disease.

The rest of the paper is organized as follows. Section 2 presents some exploratory analysis of the data. In Section 3, we discuss the methods and provide a consistency guarantee. Broadly speaking, it consists of three steps. We first define a weighted distance metric following the aforementioned idea. For a given number of clusters, we next propose an algorithm to obtain the clusters with proper guarantees. Finally, we describe how to optimize the weight and the number of clusters using a variant of gap statistic. Next, we conduct an elaborate simulation study to show the efficacy of our methods and the summaries are provided in Section 4. Finally, a county-based analysis of the COVID-19 cases across entire USA is presented in Section 5. Focusing on specific regions, we show how the proposed method can provide reasonable interpretations. For completeness of the paper, the theoretical proofs, some additional results of the COVID-19 analysis, and a second application are collated in the supplementary file.

2. Exploratory analysis of COVID-19 data

The previous section provides a comprehensive review of existing spatio-temporal clustering algorithms. It is to be noted that these methods usually address data that are observed at irregular time-points. These help the practitioners understand when some spatio-temporal hotspots arise so that they can strategize accordingly. However, our motivation, method and goals are different from this kind of traditional event-based clustering. We consider each location as an object, investigate its data for a given time-interval and classify these time series into clusters using a notion of distance that takes into account two aspects. First, it should reflect if the series for two locations (say, s and s') are similar according to their time dependence structures. And second, it evaluates how close s and s' are in terms of their longitude and latitude, so as to understand if the similarity in the time series data are structural or just by chance. Both of these measures of distance have their own merit in the context of our main research problem. Throughout the last year, we have seen how the congested big cities suffered from alarming level of outbreaks while the spread was milder in areas that are more rural. If the series corresponding to some big cities are found to have similar temporal dependence, it would bolster the hypothesis that all such cities have encountered similar extent of outbreaks. However, as policies in a large country like the USA often come from the state government, the geographical distance of these counties cannot be completely ignored and that necessitates the use of spatial closeness in the algorithm.

The above can be better understood by looking at some pertinent results. Throughout this work, we focus on the incidence rate (of COVID-19) which helps us in identifying the similarity of the spread of the disease across different locations. Our data is collected from the repository maintained by [61]. It includes the information for all counties in the contiguous USA and for all days from 22nd January, 2020 to 31st March, 2021. There are few counties (0.8% of the total) with no COVID-19 cases or without relevant information. We remove these counties from our main analysis. That leaves us with 3091 counties in total, each with a time series of length 435.

Our aim is to cluster the counties based on their respective time-series of possibly different lengths. [62], [63] are two other notable works of similar spirit, which categorize the time series for various locations in the context of COVID-19 analysis. In the former, a fuzzy clustering is proposed where some external covariates are taken into consideration to arrive at the optimal clustering. Similar covariate-assisted clustering of COVID-19 spread, albeit of a different spirit, are presented in [64] and [65] as well. They worked with data from Finland and China respectively, using Moran's I index. Note, as much as one can possibly argue for potentially including other covariate information coming from each spatial location, we cannot ignore that bringing in additional covariates would bring the problem of carefully choosing the important factors and then poses a challenging task of choosing the right model. Instead we prefer to use a more non-parametric method that depends solely on the easily available time-series observations from these locations.

In [63], using the data from the early days (roughly 3 months) the authors provided clusters based on the similarity of autocorrelation functions (ACF) of the actual time-series and utilized the popular dynamic time-warping (DTW) approach in this regard. The authors claim that they have seen one dominant time-trend across the globe. While their analysis was able to identify countries with similar progression of the disease, it has a serious drawback when it comes to spatio-temporal data. To explain it better, in the left panel of Figure 1, we display the results of the DTW-based clustering algorithm on the ACF of all the counties across the USA. We use only 10 clusters in this illustration, and it is evident that the results show no spatial continuity. Naturally, the results are barely useful from a practical standpoint. In fact, even for a smaller dataset like that of California, the clustering assignments do not depict any spatial pattern. See the right panel of Figure 1 for reference.

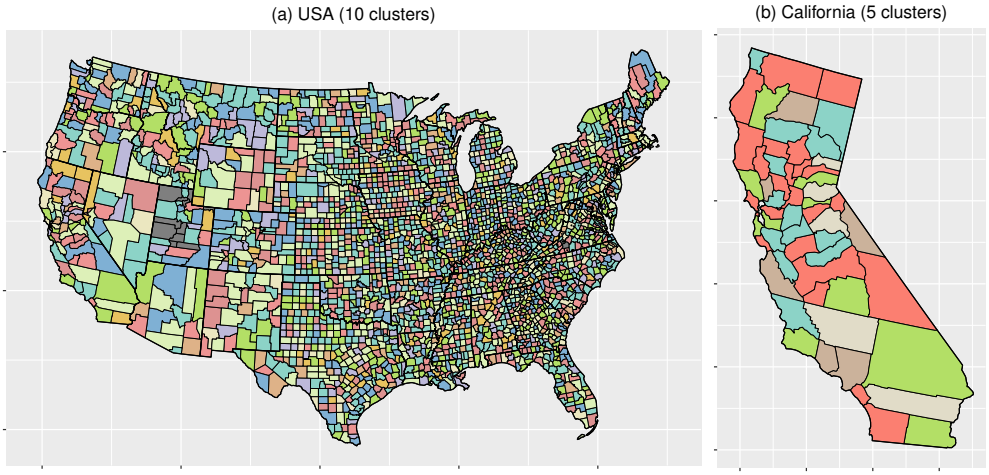

Figure 1. Clustering results based on the autocorrelation function of the time series of COVID-19 incidence rate. Panel (a) shows the results for the entire country using 10 clusters, while panel (b) shows the results for only one state, with 5 clusters.

The above clearly suggests that popular ways of clustering spatio-temporal data are not necessarily adept in providing meaningful interpretation. Especially for an epidemiological dataset like COVID-19, it is essential to amalgamate the spatial closeness with the time series properties. To further support this, we compute the Moran's I index ([66]) which quantifies the extent of spatial correlation. In Figure 2, the value of the index for different time-points is presented. The significance of the index is denoted using a black circle, and we see that the spatial correlation is in fact significant for almost all time-points. This motivates us to bring in the spatial proximity as a regularizing term in our clustering algorithm which primarily utilizes the covariance structure of the time series. This approach would ensure some spatial smoothness in the clustering assignments. In Section 5, we shall observe that with this injection of spatial information one can detect proper clusters which also lead to a meaningful interpretation of how the disease has spread.

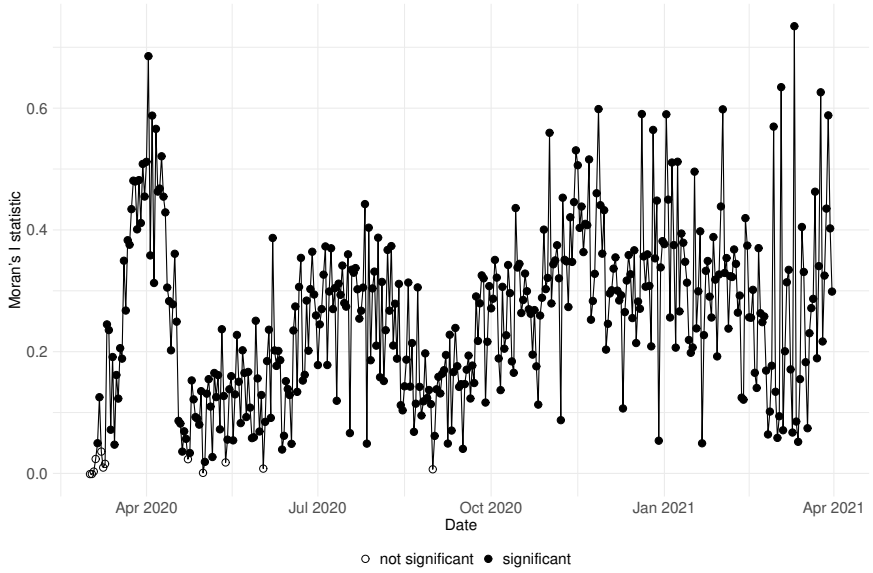

Figure 2. Moran's I index for the entire USA at every time-point. A black circle indicates a significant spatial correlation at 0.01 level of significance.

3. Methodology

Throughout this paper, the set of locations is denoted by $\mathcal{S} = \{s_1, \ldots, s_n\}$ and the set of time-points is denoted by $\Gamma = \{1, \ldots, T\}$. We reiterate that, although Γ is assumed to be regular, our proposed methods can be easily adapted if the time points are irregularly spaced across locations. Let $Y(s_i, t)$ denote the observation, suitably centered to have mean zero, for location s_i and at time t. We use $\boldsymbol{Y}_i$ for the time series $(Y(s_i, t))_{1 \leqslant t \leqslant T}$. For each time series in the study, we shall consider a very general class of stationary time series which are functions of independent and identically distributed (iid) random variables, cf. [67].

Assumption 1. *Let ε_j, $j \in \mathbb{Z}$, be iid random variables. Then, every time series $\boldsymbol{X}$ considered in this study satisfies the following:*

$$X_i = \eta(\varepsilon_{i-s}; s \in \mathbb{Z}), \qquad i \in \mathbb{Z}, \tag{3.1}$$

where η is a measurable function such that X_i is well-defined. Following [67], we define the functional dependence as follows

$$\delta_{j,p} = \|X_i - X_{i,\{j\}}\|_p \tag{3.2}$$

where $X_{i,\{j\}} = \eta_i(\varepsilon_i, \cdots \varepsilon_{i-j+1}, \varepsilon'_{i-j}, \varepsilon_{i-j-1}, \cdots)$, with $\{\varepsilon'_s : s \in \mathbb{Z}\}$ is an independent copy of the process $\{\varepsilon_s : s \in \mathbb{Z}\}$. This functional dependence measure allows us to replace conditions on parametric models for a large number of stationary series under a singular framework. Additionally, we assume $\mathbb{E}(X_i) = 0$, $X_i \in \mathcal{L}_4$, and $\Theta_{0,4} = \sum_{k \geqslant 0} \delta_{k,4}$ to be finite.

Call $\gamma_j = \mathbb{E}(X_i X_{i+j})$, we define the corresponding spectral density (assumed to be finite throughout the paper) for $0 \leqslant \theta < 2\pi$ as $f(\theta) = \sum_{j \in \mathbb{Z}} \gamma_j e^{\iota j \theta}$. In the following discussions, we shall use $f(\boldsymbol{Y}_i, \theta)$ to denote the true spectral density of the underlying DGP for $\boldsymbol{Y}_i$. We recall that the most common way to estimate the spectral density is to use the periodogram which, for $\boldsymbol{Y}_i$, is defined as follows:

$$I_T(\boldsymbol{Y}_i, \theta) = \frac{1}{T} \sum_{j=1}^{T} \sum_{k=1}^{T} Y(s_i, j) Y(s_i, k) \exp\{\iota(j-k)\theta\}. \tag{3.3}$$

Further, in light of the fact that the periodogram is not consistent and is a wildly fluctuating estimate of the spectrum, it is necessary to apply the idea of smoothing which enables us to obtain a stable estimate. To that end, as an estimate of $f(\boldsymbol{Y}_i, \theta)$, we use the smoothed periodogram

$$\hat{f}(\boldsymbol{Y}_i, \theta) = \int I_T(\boldsymbol{Y}_i, \lambda) K((\lambda - \theta)/b_T)\, d\lambda, \tag{3.4}$$

where $K(\cdot)$ is a kernel function and b_T is a bandwidth sequence satisfying the following assumption.

Assumption 2. *In the spectral density estimate given by eq. (3.4), the kernel function is considered to be even and defined on $[-1, 1]$ such that $\int K(\lambda)\, d\lambda = 1$, $\int K^2(\lambda)\, d\lambda < \infty$. On the other hand, the bandwidth sequence satisfies $b_T \to 0$, $T b_T \to \infty$ as $T \to \infty$.*

A plethora of literature have discussed at length that the estimates are not too sensitive to the choice of the kernel. For application purposes, we shall be using the Daniell kernel with $b_T = T^{-1/5}$ throughout this paper. [68] is an excellent reading for a detailed discussion on smoothed periodogram estimates and related theories. It should be pointed out that the above choice of the bandwidth is the mean-squared error (MSE) optimal and a popular choice in related literature. Since it provides good results in our practical applications, we do not adopt other techniques of bandwidth optimization. However, one may also consider the concepts of risk minimization, cross-validation etc. to decide on the bandwidth (see, for example, [69] and [70]).

3.1. Spatio-temporal distance matrix

For any clustering algorithm, a necessary and crucial step is to define an appropriate measure that can reflect how similar two observations are. As we are dealing with spatio-temporal data, we propose to cluster the objects

based on a convex combination of the spatial distance matrix and the temporal distance matrix. The spatial distance matrix, hereafter denoted as $\Delta_{\mathcal{S}}$, is defined in the usual way where we compute the euclidean norm between the spatial coordinates of the locations. In particular, if $l(s_i)$ denotes the 2-dimensional vector of latitude and longitude for the ith location and $\|\cdot\|_2$ denotes the euclidean norm, then the (i, j)th element of $\Delta_{\mathcal{S}}$ is

$$\Delta_{\mathcal{S}}(i, j) = \left\| l(s_i) - l(s_j) \right\|_2 . \tag{3.5}$$

For computing the distance between two time-series, we focus on the difference between their covariance structures, and that motivates us to use the spectral densities. To quantify the aforementioned distance, we compute the $\mathcal{L}_2$ distance between the above spectral density estimates. Let $\hat{\Delta}_\Gamma$ denote the pairwise distance matrix in terms of the spectral density estimates for all the locations, i.e. the (i, j)th element of $\hat{\Delta}_\Gamma$ is given by

$$\hat{\Delta}_\Gamma(i, j) = \left[\int_0^{2\pi} \left\| \hat{f}(\boldsymbol{Y}_i, \theta) - \hat{f}(\boldsymbol{Y}_j, \theta) \right\|^2 \, d\theta \right]^{1/2} . \tag{3.6}$$

Here, $\hat{\Delta}_\Gamma$ serves as an estimate of the true spectral density distance matrix (Δ_Γ) given by

$$\Delta_\Gamma(i, j) = \left[\int_0^{2\pi} \left\| f(\boldsymbol{Y}_i, \theta) - f(\boldsymbol{Y}_j, \theta) \right\|^2 \, d\theta \right]^{1/2} \tag{3.7}$$

and as is natural, we use $\hat{\Delta}_\Gamma$ in place of the unknown Δ_Γ.

One should note that the above expression is not dependent on the set Γ. We keep the subscript for notational consistency. Furthermore, note that $\hat{\Delta}_\Gamma$ is a random variable whereas $\Delta_{\mathcal{S}}$ is deterministic for a given set of locations. They capture respectively the temporal closeness and the spatial closeness of the data. We leverage it to define the following convex combination of $\hat{\Delta}_\Gamma$ and $\Delta_{\mathcal{S}}$ as the spatio-temporal distance matrix to be used in the clustering algorithm:

$$\hat{\Delta}_{\mathcal{S},\Gamma}(\alpha) = \alpha \left(\frac{\hat{\Delta}_\Gamma}{\|\hat{\Delta}_\Gamma\|_F} \right) + (1 - \alpha) \left(\frac{\Delta_{\mathcal{S}}}{\|\Delta_{\mathcal{S}}\|_F} \right). \tag{3.8}$$

Here, $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. We point out that α represents the importance of the spectral density based distance matrix. In particular, $\alpha = 0$ leads to the standard spatial clustering based on the euclidean distance between the (longitude, latitude) of the locations whereas $\alpha = 1$ corresponds to purely temporal clustering of the data.

3.2. Clustering algorithm for known number of classes

In this section, we present the spatio-temporal clustering algorithm assuming that the true number of classes (say k) is known. We employ a k-medoid clustering using the distance function in eq. (3.8). This is chosen over a more conventional k-means procedure since the k-medoid algorithm always chooses one of the existing point as cluster centers. From an application point of view, it makes sense to designate the central location as an origin for the temporal patterns observed in other members of the same cluster. Naturally, it allows added interpretation with respect to the research question we consider. Key steps of the algorithm for the uninitiated are discussed below to make this exposition more complete.

From the implementation perspective, we rely on the partitioning around medoid (PAM) technique which uses a greedy search. Although it may not find the optimum solution, it is faster than an exhaustive search. The PAM algorithm starts by selecting k of the n data points as the medoids in a *greedy* algorithm. With any particular initialization, it associates each data point to its closest medoid. For each medoid m, and for each non-medoid data point r, it then considers the swap of m and r, and computes the cost change. Such cost reduction happens with the best possible m and r and when no such swapping is possible, the algorithm terminates.

Note that the run-time complexity of the original PAM algorithm per iteration of three is $O(k(n-k)^2)$, by only computing the change in cost. A naive implementation that recomputes the entire cost function every time will be in $O(n^2k^2)$. This run-time can be further reduced to $O(n^2)$ by splitting the cost change into three parts such that

computations can be shared or avoided. See [71] for in-depth reading on this. We use the PAM as a key tool and propose the following algorithm.

Algorithm 1: Spatio-temporal clustering algorithm when the true number of classes is known

Input: Set of locations $\mathcal{S} = \{s_1, \ldots, s_n\}$, and for each location, the corresponding time-series $\boldsymbol{Y}_i$, each of length T. True number of classes k and the value of the mixing parameter α.
Output: Clustering assignment for each location.

- **Step 1:** Using the coordinates of s_i's, calculate the spatial distance matrix by eq. (3.5). Using $\boldsymbol{Y}_i$'s, calculate the temporal distance matrix by eq. (3.6).
- **Step 2:** For given α, use eq. (3.8) to calculate the spatio-temporal distance matrix $\hat{\Delta}_{\mathcal{S},\Gamma}(\alpha)$.
- **Step 3:** Implement the partitioning around medoid (PAM) algorithm for the distance matrix $\hat{\Delta}_{\mathcal{S},\Gamma}(\alpha)$ calculated above and obtain the clustering assignment.

In addition to the aforementioned points, another vital merit of the k-medoid clustering algorithm in a standard setting is that it is a consistent method, in the sense that under some mild regularity conditions, the algorithm returns the true clustering assignments with probability 1. We aim to provide similar results for Algorithm 1. To that end, we make the following assumption.

Assumption 3. *Two locations $s_i, s_j \in \mathcal{S}$ are in the same class if and only if $\boldsymbol{Y}_i$ and $\boldsymbol{Y}_j$ are generated from the same DGP.*

Theorem 1. *Suppose that the set of locations $\mathcal{S}$ is a subset of a region with bounded area, and $c_1, \ldots, c_k \in \mathcal{S}$ are the centers of the k (fixed) clusters in the data. Any randomly chosen location $s \in \mathcal{S}$ is assumed to be included in the cluster centered at c_j with probability $\pi(\delta(s, c_j))$, where $\pi : [0, \infty) \to [0, 1]$ is a decreasing function and $\delta(s, c_j)$ is the spatial distance between the two locations. Suppose that $\mathcal{G}$ is the true clustering assignments and $\mathcal{H}(\alpha)$ is the clustering obtained from Algorithm 1 for $\alpha \in [0, 1]$. Then, under Assumption 3, there exists α_0 such that for all $\alpha \in (\alpha_0, 1]$, as $n \to \infty$ and $T \to \infty$, $P(\mathcal{H}(\alpha) = \mathcal{G}) \to 1$.*

The proof is relegated to Section 7. In the above theorem, the value of α_0 depends on the relative magnitude of the spatial closeness and the temporal closeness of the clusters. Let $\delta_{\mathcal{S}}$ be the minimum pairwise distance (scaled by the norm) between the cluster centers. Then, a higher value of $\delta_{\mathcal{S}}$ ensures that a lower value of α_0 would guarantee consistency of the proposed clustering algorithm. Moreover, one may relax Assumption 3 as well. In fact, the following result can be derived as an immediate corollary.

Remark 1. *Let $C_1, \ldots, C_k \subset \mathcal{S}$ be the k clusters. Suppose that the true DGPs for locations $s_1, \ldots, s_n$ are respectively $\mathcal{D}_1, \ldots, \mathcal{D}_n$, with the corresponding true spectral density functions being $f_1, \ldots, f_n$. Then, the conclusion of Theorem 1 holds if the true minimum inter-cluster temporal distance is strictly greater than the true maximum intra-cluster temporal distance, i.e. if*

$$\max_{(i,j,r):s_i,s_j\in C_r} \int_0^{2\pi} \left\|f_i(\theta) - f_j(\theta)\right\|^2 \, d\theta < \min_{(i,j,r,k):s_i\in C_r, s_j\in C_k, r\neq k} \int_0^{2\pi} \left\|f_i(\theta) - f_j(\theta)\right\|^2 \, d\theta.$$

The above results imply that the algorithm is consistent for all high values of α (for example, it is always consistent for $\alpha = 1$). It is however necessary to point out that the convergence to the true clustering assignments requires less amount of data if the consistency is attained for a lower value of α, i.e. if the true cluster centers are well-separable. Keeping that in view, from a practical perspective, we consider all possible $0 < \alpha < 1$ while implementing our algorithm. Albeit one can use some criteria to optimize over α, we intentionally do not do that and the reasons are two-fold. First, different choices of this parameter helps in the interpretability of the results from different aspects. As α is changed from 0 to 1, i.e. from purely spatial clustering to purely temporal clustering, one can see the change in the

class assignments and that can provide valuable insights. Second, we emphasize that the two notions of distance used in our method are fundamentally different. In fact, the purely spatial one is deterministic while the temporal distance is random as it depends on the observed time-series in these locations. It is difficult to choose one weighing scheme while combining two distances that could produce uniformly better results than others for any type of clustering assignment. Finally, since the spatial distances are deterministic, one can also think of adding the spatial component in the distance as a regularizing term that guarantees some spatial closeness. Thus, the freedom to choose α allows the user to flexibly decide on the amount of constraints depending on particular applications.

3.3. Optimum number of clusters through gap statistic

In the previous section, we provided a consistent algorithm when the true number of classes is known. However, that may not happen in a real-life application. A major challenge in such problems is to design an effective criteria of choosing the optimal number of clusters. Here we describe an adaptive way which leverages the concept of gap statistic originally developed by [72] and later modified by [73].

The gap statistic makes use of the resampling technique and associated distribution. To elaborate, let $\bar{W}_\ell$ be a suitably chosen measure that evaluates the closeness of the clusters if the locations are split into ℓ clusters. Then, the estimate of the optimal number of clusters is the value of ℓ for which $\log(\bar{W}_\ell)$ falls the farthest below the reference curve determined by the resampling distribution. Mathematically, for a possibly varying ℓ, consider the clustering of n objects into the clusters $C = \{C_1, \ldots, C_\ell\}$ with $n_m = |C_m|$. Then the optimum number of clusters is defined as

$$\hat{k}_{Gap} = \underset{\ell}{\operatorname{argmin}} \left\{ \mathbb{E}_*(\log(\bar{W}_\ell)) - \log \bar{W}_\ell \right\}, \tag{3.9}$$

where $\mathbb{E}_*$ is the expectation under the resampling distribution and

$$\bar{W}_\ell = \sum_{m=1}^{\ell} \frac{1}{2n_m(n_m-1)} \sum_{i,j \in C_m} \hat{\Delta}_{\mathcal{S},\Gamma}(\alpha)^2_{ij}. \tag{3.10}$$

Note that, [73] used a double differenced value of the weighted version of the gap statistic for picking the optimum value but we just look at the weighted gap statistic since it provides reasonably good results in our simulation studies. Another key point is to decide on the possible number of clusters for which the optimization step needs to be implemented. In all our applications, we would do the minimization in eq. (3.9) for $\ell \in \mathcal{K} = \{\sqrt{n}/2, \cdots, 3\sqrt{n}/2\}$. The pseudo-code of the algorithm is now formally presented in Algorithm 2.

3.4. Evaluation of the clustering results

In order to quantify the effectiveness of the proposed algorithm, we calculate different measures of similarity. On one hand, if the ground truth is known, these measures can be used to detect how accurate our proposed algorithm is. We employ that in the simulation studies presented in the following section. On the other hand, the same measures can help us in understanding the similarity between different clustering results obtained for different values of α. This technique is helpful to give an idea about how closely the purely temporal information align with the purely spatial one and whether any particular value of α provides some interesting phenomena. We shall discuss these aspects in more detail in the real life applications provided in Section 5.

In what follows, let $\mathcal{G} = \{G_1, \ldots, G_k\}$ and $\mathcal{H} = \{H_1, \ldots, H_l\}$ be two different clustering assignments for n objects. Let TP (respectively, TN) denote the number of pairs of objects which are in the same subset (respectively, in different subsets) in both $\mathcal{G}$ and $\mathcal{H}$. Similarly, let FP be the number of pairs of objects that are in different subsets in $\mathcal{G}$ but in the same subset in $\mathcal{H}$; and let FN be the remaining number of pairs. Clearly, $TP + FP + TN + FN = \binom{n}{2}$.

The first measure of similarity we use in this study is called symmetric purity (SP). This is a symmetrized version of the usual purity index (see [74] for related discussions), which is a well-known external evaluation criterion. Note that the purity index is defined for a clustering result $\mathcal{H}$ against a ground truth $\mathcal{G}$ as $RP(\mathcal{H}, \mathcal{G}) = \sum_{i=1}^{k} \max_j \left|G_i \cap H_j\right| / n$. Quite evidently, this measure is not symmetric and it does not make sense when the ground truth is unknown, which happens in all real-life applications. In view of that, to avoid knowledge of ground truth, we

Algorithm 2: Spatio-temporal clustering algorithm when the true number of classes is unknown

Input: Set of locations $\mathcal{S} = \{s_1, \ldots, s_n\}$, and for each location, the corresponding time-series $\boldsymbol{Y}_i$, each of length T. Value of the mixing parameter α.

Output: Number of optimum clusters $k_0(\alpha)$, and the corresponding clustering assignment for each location.

- **Step 1:** Using the coordinates of s_i's, calculate the spatial distance matrix by eq. (3.5). Using $\boldsymbol{Y}_i$'s, calculate the temporal distance matrix by eq. (3.6).
- **Step 2:** For given α, use eq. (3.8) to calculate the spatio-temporal distance matrix $\hat{\Delta}_{\mathcal{S},\Gamma}(\alpha)$.
- **Step 3:** Choose a candidate set of values for the number of clusters (k). For practical purposes we suggest $\mathcal{K} = \{\sqrt{n}/2, \cdots, 3\sqrt{n}/2\}$.
- **Step 4: for** $k \in \mathcal{K}$
  - assume that the true number of classes is k,
  - obtain the clustering assignment for k classes following Step 3 of Algorithm 1 using the distance matrix $\hat{\Delta}_{\mathcal{S},\Gamma}(\alpha)$ calculated in Step 2 above,
  - compute $\bar{W}_k$ as defined in eq. (3.10) and denote it as $\bar{W}_{k,\alpha}$.

  end
- **Step 5:** Find $k_0(\alpha) = \text{argmax}_{k\in\mathcal{K}} \bar{W}_{k,\alpha}$, and return it along with the corresponding clustering output obtained in Step 4 above.

define a symmetric measure SP for two different clustering assignments $\mathcal{G}$ and $\mathcal{H}$ as follows:

$$\text{SP}(\mathcal{H}, \mathcal{G}) = \frac{RP(\mathcal{H}, \mathcal{G}) + RP(\mathcal{G}, \mathcal{H})}{2}. \tag{3.11}$$

Next, as the second measure of similarity in this study, we use the F-measure (FM). Define

$$\text{FM}(\mathcal{H}, \mathcal{G}) = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}, \tag{3.12}$$

where $\beta \geqslant 0$ is a fixed parameter and $P = TP/(TP + FP), R = TP/(TP + FN)$ indicate the precision and the recall, respectively.

It is worth mentioning that both precision and recall in themselves are evaluation criteria. However, they only address specific aspects of the clustering assignments and are not adequate. In particular, the precision penalizes only the pairs counted in FP whereas recall penalizes only the ones in FN. Furthermore, another popular and similar criteria in this regard is the Rand index which is defined as $(TP + TN)/\binom{n}{2}$. In this case, all pairs from FP and FN are penalized, but they are equally weighted, which can be an undesirable criteria in applications. The F-measure addresses all of these concerns and is therefore a better evaluation method. The parameter β corresponds to the weight one would want to put on recall. In all our applications below, we shall use $\beta = 1$, which is commonly known as the $F1$-measure. The reader is further referred to [75] for detailed discussions on these evaluation methods.

The above two metrics will be used in both simulation studies and in the real-life applications. Additionally, we shall consider two other measures for assessing the accuracy of a clustering algorithm in the simulation framework where the truth is known. One of them is a loss function originally proposed by [76]. The principle of this metric relies on a general loss function to evaluate the disagreements in all possible pairs of observations between the estimated clustering and the ground truth. In particular, this function penalizes (possibly using different weights) the two types of clustering errors, i.e. the pairs constituting FP and FN in the earlier notations, in a quadratic manner.

In our calculations, for simplicity, we use equal weights but interested readers are referred to [77] for a more general discussion. Using identical notations as before, if $\mathcal{G} = \{G_1, \ldots, G_k\}$ is the ground truth and $\mathcal{H} = \{H_1, \ldots, H_l\}$ is the estimated clustering, with $\left|G_i \cap H_j\right|$ being denoted as n_{ij}, then the Binder loss is calculated as

$$\mathrm{BL}(\mathcal{H}, \mathcal{G}) = \frac{1}{2}\left[\sum_{i=1}^{k}\left(\sum_{j=1}^{l} n_{ij}\right)^2 + \sum_{i=1}^{l}\left(\sum_{i=1}^{k} n_{ij}\right)^2 - 2\sum_{i=1}^{k}\sum_{j=1}^{l} n_{ij}^2\right]. \tag{3.13}$$

It is clear that a higher value of the BL corresponds to higher discordance between the two partitions and thus, a lower value is desired. Finally, as the last measure of comparison for clustering algorithms, we consider the entropy, in the same spirit as [78]. We take the negative of the usual entropy function and therefore, a higher value is desired. One must note that this measure can be construed in the same vein as the purity index. Mathematically, we define the entropy of $\mathcal{H}$ with respect to $\mathcal{G}$ as

$$\mathrm{ENT}(\mathcal{H}, \mathcal{G}) = \frac{1}{n \log_2 k} \sum_{j=1}^{l}\sum_{i=1}^{k} n_{ij} \log_2\left(\frac{n_{ij}}{\sum_{j=1}^{l} n_{ij}}\right). \tag{3.14}$$

4. Simulation study

The objective of the simulation study in this paper is two-fold. First, we evaluate the accuracy of the clustering algorithm assuming that the true number of classes is known. Second, we aim to identify how correctly the gap statistic based procedure can estimate the true number of clusters. All codes and data used for the simulation studies and real-data analysis will me made available publicly in a GitHub repository maintained by the first author.

We consider six different setups in this study. In the first one (hereafter called DGP1), n locations are generated randomly from a uniform distribution on a rectangle in the xy-plane. These locations are put into k clusters based on their (x, y) coordinates, following a k-medoid algorithm. Then, for every location within a cluster, time-series data of length T are simulated from a common auto-regressive moving average (ARMA) process of order $(2, 2)$. Thus, there are k number of different processes in total. We randomly generate the ARMA parameters to define and distinguish between these processes. Note that this setup is considering a deterministic way to assign the locations into different clusters. The second setup (hereafter called DGP2) follows the conditions from Theorem 1 with a specific structure of $\pi(\cdot)$. Here, after generating n locations in a similar way as above, we randomly pick k of them to be the cluster centers. Then, for every other location, it is assigned to one of these clusters (say, the jth one) with a probability proportional to $\exp(-d_j)$, where d_j denotes the distance of the location from the jth cluster center. Once the clusters are formed, the time series data are generated using the same procedure as in DGP1.

Next, in the third setup (hereafter denoted as DGP3), we consider a more general scenario. The clusters, in this case, are formed in the same fashion as in DGP2. Then, k different ARMA$(2, 2)$ processes $\xi_1, \ldots, \xi_k$ are considered for the k classes. Now, for the location s_i in the jth cluster centered at c_j, the corresponding time series data is simulated using an additive combination of a realization from ξ_j, a realization from a sequence of iid zero-mean normal random variables with standard deviation $\left\|l(s_i) - l(c_j)\right\|_2$ and a scalar multiple of the realized series for the cluster center. The multiplication factor we take for the last term is of the form $\exp(-\rho\left\|l(s_i) - l(c_j)\right\|_2)$. Observe that, in this way, the data for the locations within a cluster are not simulated from the same process but from different processes which are close among themselves. Further, the term mentioned above imposes an exponentially decaying dependence with the cluster center, which is a common assumption in many spatio-temporal studies. DGP3 would allow us to evaluate the accuracy of the proposed algorithm under the setup of Remark 1. The fourth DGP used in this study is the standard susceptible-infected-recovered (SIR) model, commonly used in epidemiological applications. For implementation, we rely on the basic SIR model outlined in [79]. In this case, first the clusters are created in the same manner as in DGP1. Subsequently, for every cluster, same SIR model parameters are used in generating the data for the infected population. These parameters are randomly chosen from a viable set. Once we simulate the incidence data, we add a small random perturbation to make the data more realistic.

Finally, as the last two settings, we consider two extreme scenarios. In DGP5, we assume only a spatial clustering framework, where the locations are put into clusters based on a k-medoid algorithm, but the time series data for each

location is simulated from different ARMA processes. In stark contrast to this, DGP6 works with the assumption that the data for different locations within the same cluster are indeed generated from the same process as in DGP1, but the locations are assigned to the clusters randomly. We reiterate that DGP5 presumes no temporal similarity between the clusters whereas DGP6 entails no spatial similarity for different clusters.

We use different combinations of n, k, T to understand the efficacy of our method across various scenarios. Before presenting those results, in Figure 3, we show examples of simulated clusters (for $n = 100, k = 5$) along with the $\mathcal{L}_2$ distance (scaled to a value between 0 and 1) between true spectral densities of the processes corresponding to the cluster centers. It is evident that in DGP1 (and also in DGP4 and DGP5), every point belongs to the cluster whose center is closest to it, whereas in DGP2 and DGP3, there are instances where nearby locations belong to different clusters. In DGP6, no discernible pattern can be observed in the assignment of the clusters as the locations are randomly assigned to the groups. On the other hand, from the plots on the second and fourth columns of the figure, we can see that different classes are considerably different (darker colour indicates higher deviation) in all cases except for DGP5. It suggests that the clusters in these five cases are well separated in the temporal sense. For the setup with only spatial clustering, as expected, we observe that the time series corresponding to different locations within a cluster do not conform to any particular pattern. In fact, one may infer that the clusters in this setting are not well-separated in the temporal sense. Therefore, this setting is an example of deviation from the assumptions under which we developed the theoretical guarantees of our method.

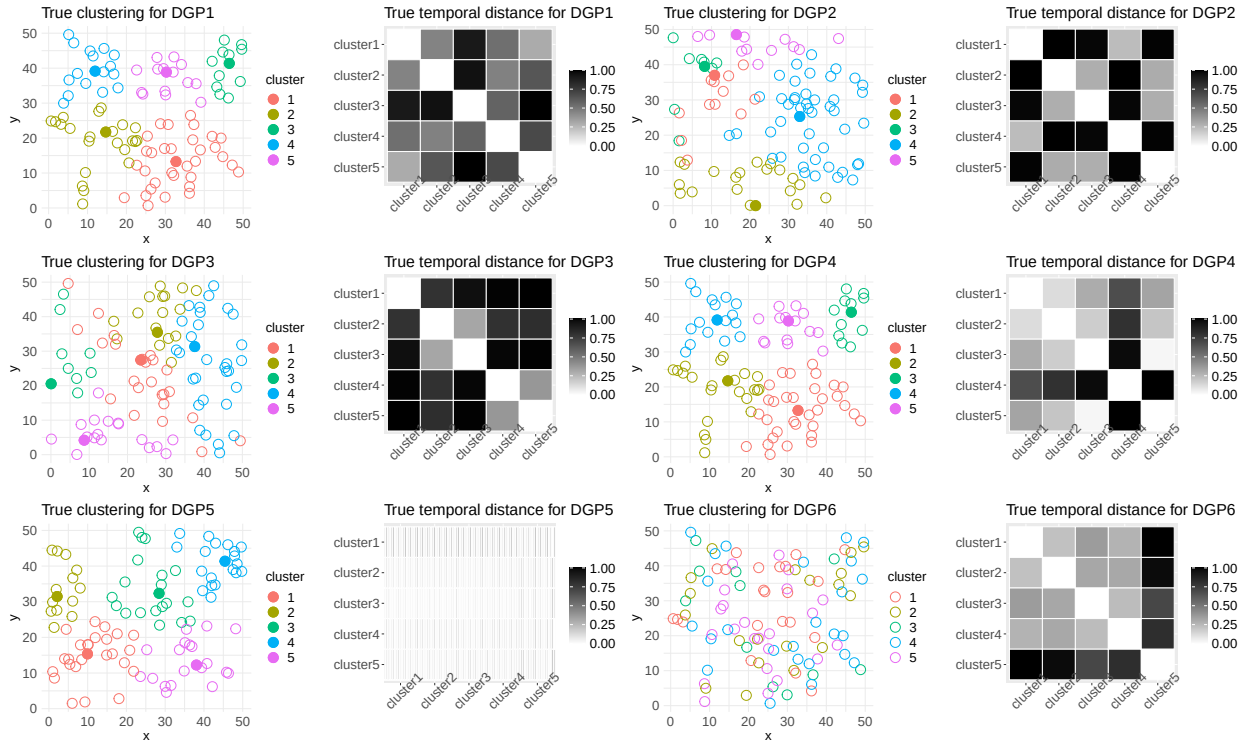

Figure 3. Examples of the six setups for $n = 100$, $k = 5$. Plots on the first and third columns show the true clusters (filled circles indicate the centers wherever appropriate) and the plots on the other columns show the $\mathcal{L}_2$ distances (scaled to values between 0 and 1) between the true spectral densities of the processes corresponding to the clusters.

Next, we present the accuracy of the proposed algorithm for various scenarios, with $n \in \{100, 500\}$, $k \in \{5, 10, 20\}$ and $T \in \{100, 500, 1000\}$. For every combination of these parameters, data are generated under the aforementioned setups and Algorithm 1 is implemented. The SP and FM values for the output, against the ground truth, are then computed. This is done for $\alpha \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$ to understand how the result varies with the choice of the mixing parameter. For every scenario, we run the experiment 100 times and calculate the mean of the SP and FM values over all these simulations. Results on the SP values are displayed in Figure 4. The interpretations

and the findings on the FM values are similar. That plot is provided in the supplementary file.

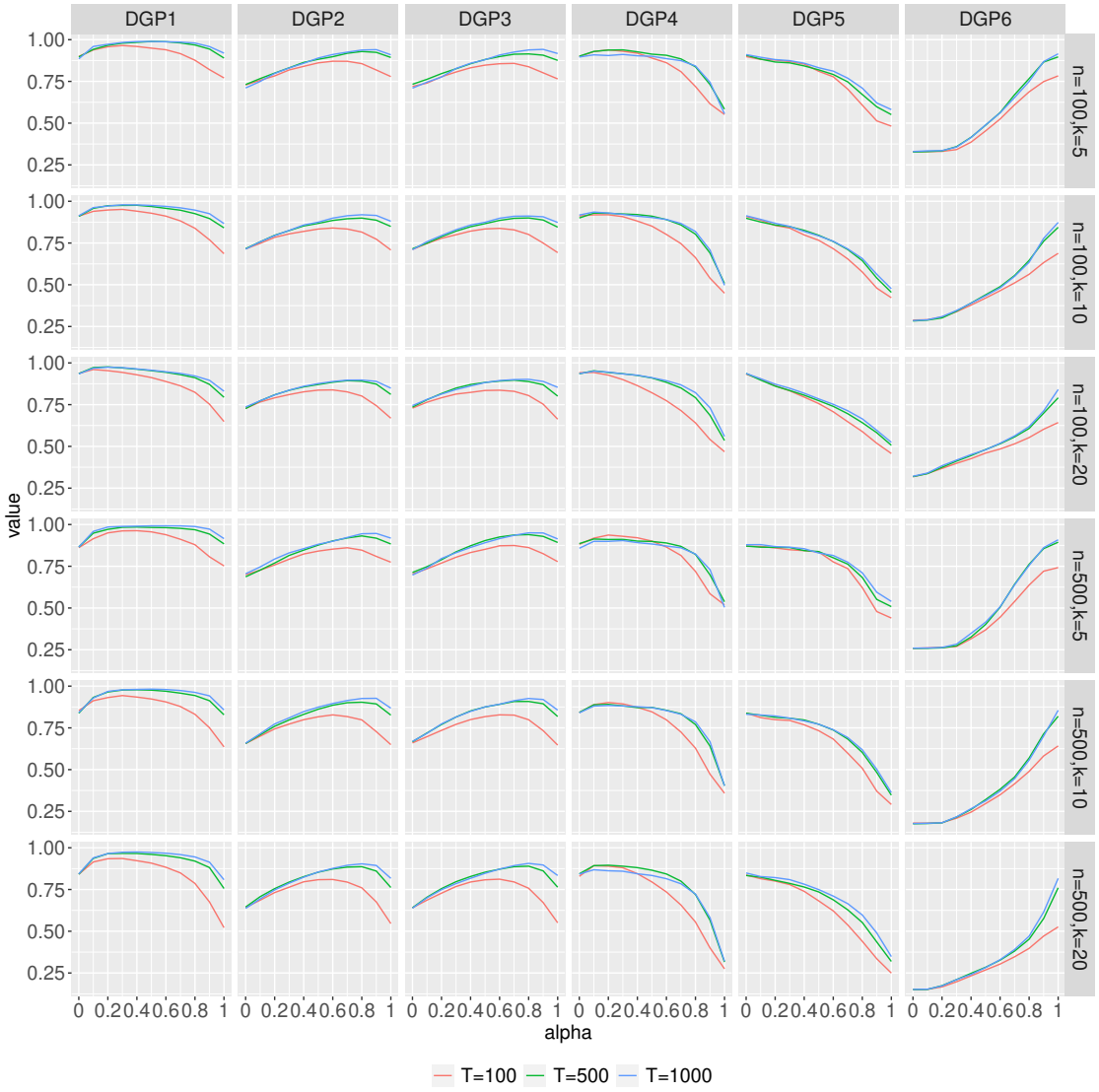

Figure 4. Symmetric purity values (mean taken over many simulations) for different values n, k, T. Plots for different DGPs are presented in different columns.

In the plots (Figure 4) corresponding to the first four data generating processes, we see that the symmetric purity is the highest for some α usually between 0 and 1. It clearly establishes that combining the spatial distance and the temporal distance in an appropriate way results in superior performance. In fact, the maximum SP values in all cases are above 0.9, which shows great accuracy in finding out the true clusters. Next, recall that DGP1 and DGP4 assign the clusters in a deterministic way based on the locations whereas DGP2 and DGP3 do that in a random manner. That is reflected in the SP values, as purely spatial clustering (which corresponds to $\alpha = 0$) in DGP2 and DGP3 perform poorly. However, in these setups, purely temporal clustering records higher accuracy. Furthermore, this accuracy increases considerably as T is increased from 100. In fact, when k is 5 or 10, purely temporal clustering for $T = 1000$ records SP values of 0.85 or more. There is a minor drop in the accuracy when the average number of locations per cluster, i.e. n/k, decreases. Turning attention to the plots in the last two columns of Figure 4, we observe distinct patterns. In the only spatial clustering setup, understandably, the proposed method fares worse as the value of α

increases. Since there is no temporal closeness, we observe that the purely temporal clustering algorithm can only register a symmetric purity value of approximately 0.25. An identical phenomena in the different direction is noticed for DGP6, where the data do not have any spatial closeness. In this case, SP values of around 0.8 ascertain that the purely temporal clustering works well to capture the similarity in the time series observations.

Next, to understand the efficacy of the proposed method, we aim to compare its accuracy in identifying the true clusters, with respect to two other benchmark methods. First of the competing approaches is chosen based on the work of [80] who proposed a Ward-like hierarchical clustering algorithm including spatial or geographical constraints. In principle, this method appears to be analogous to our technique where a mixing parameter (call it κ) between 0 and 1 is used to combine two different distance matrices. A bigger value of κ puts more importance on the geographical constraints whereas a smaller value considers the data to be more important. This method will be abbreviated as CWHC below. For appropriate comparison, we shall use $\kappa = 0.1, 0.5, 0.9$ to assess the method's accuracy under different combinations. As the second competing method in this article, we implement the previously discussed ST-DBSCAN approach, which is the most popular algorithm in the extant literature of space-time clustering. Now, to choose the setting of this comparison study, we work with a more realistic spatial set where the locations are chosen as the true coordinates of the counties in a particular region of the USA. Namely, we work with 83 counties in the states of New York and New Jersey, which constitute a standard federal region. The length of the time series for each county is also kept fixed at 500, which is comparable to our real dataset. Then, data are simulated from the six DGPs and the three methods are implemented. Both CWHC and our method are run with three different mixing parameters. Each setup is run 100 times and the average performances are computed based on the four measures. We report the SP values and the BL values in Tables 1 and 2, respectively. The other two tables can be found in the supplementary material.

Table 1. Averages symmetric purity index (taken over all simulations) recorded by different competing methods in different settings. The set of locations is the counties in New York and New Jersey, and T is fixed at 500.

		Proposed method			CWHC			ST-DBSCAN
DGP	Classes	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 0.9$	$\kappa = 0.1$	$\kappa = 0.5$	$\kappa = 0.9$	
DGP1	$k = 5$	0.944	**0.978**	0.884	0.738	0.863	0.838	0.768
	$k = 10$	**0.970**	0.969	0.834	0.722	0.834	0.852	0.701
	$k = 20$	**0.973**	0.950	0.792	0.708	0.804	0.897	0.637
DGP2	$k = 5$	0.776	**0.884**	0.878	0.730	0.750	0.730	0.780
	$k = 10$	0.764	**0.871**	0.849	0.747	0.763	0.725	0.721
	$k = 20$	0.789	**0.882**	0.813	0.730	0.785	0.749	0.644
DGP3	$k = 5$	0.757	**0.881**	0.877	0.743	0.741	0.725	0.790
	$k = 10$	0.767	**0.881**	0.847	0.718	0.757	0.733	0.722
	$k = 20$	0.791	**0.883**	0.813	0.730	0.787	0.758	0.645
DGP4	$k = 5$	0.831	0.858	0.624	0.798	**0.873**	0.828	0.715
	$k = 10$	0.852	0.861	0.523	0.698	**0.868**	0.841	0.698
	$k = 20$	0.818	**0.832**	0.598	0.667	0.817	0.795	0.726
DGP5	$k = 5$	**0.900**	0.817	0.577	0.788	0.873	0.854	0.612
	$k = 10$	**0.931**	0.792	0.474	0.745	0.856	0.920	0.558
	$k = 20$	0.896	0.791	0.540	0.699	0.796	**0.906**	0.532
DGP6	$k = 5$	0.337	0.494	**0.891**	0.529	0.347	0.341	0.773
	$k = 10$	0.311	0.455	**0.841**	0.563	0.399	0.313	0.700
	$k = 20$	0.396	0.519	**0.806**	0.589	0.508	0.367	0.639

The results corresponding to our method are mostly in line with the earlier conclusions. We can see that the proposed approach outperforms the other two methods in almost all cases. The values are most impressive under DGP1, where the clusters are assigned based on the locations in a deterministic way and the time series data within a cluster are generated from the same process. In DGP2 and DGP3, there is a drop in the accuracy of our method, likely due to the randomness in assigning the clusters. Interestingly, CWHC and ST-DBSCAN do not suffer from this problem, thereby indicating that these two methods are robust to the true underlying process, albeit the accuracy is

Table 2. Averages Binder loss (taken over all simulations) recorded by different competing methods in different settings. The set of locations is the counties in New York and New Jersey, and T is fixed at 500.

		Proposed method			CWHC			ST-DBSCAN
DGP	Classes	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 0.9$	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 0.9$	
DGP1	$k = 5$	0.039	**0.020**	0.132	0.262	0.100	0.115	0.643
	$k = 10$	**0.012**	0.014	0.126	0.211	0.077	0.054	0.711
	$k = 20$	**0.006**	0.013	0.086	0.126	0.059	0.021	0.751
DGP2	$k = 5$	0.171	**0.103**	0.140	0.293	0.195	0.205	0.642
	$k = 10$	0.095	**0.065**	0.127	0.222	0.117	0.109	0.687
	$k = 20$	0.045	**0.032**	0.090	0.125	0.069	0.051	0.746
DGP3	$k = 5$	0.181	**0.107**	0.138	0.289	0.198	0.204	0.607
	$k = 10$	0.094	**0.058**	0.122	0.214	0.115	0.106	0.724
	$k = 20$	0.045	**0.032**	0.093	0.120	0.066	0.050	0.741
DGP4	$k = 5$	0.137	0.113	0.275	0.137	**0.108**	0.133	0.664
	$k = 10$	**0.046**	0.050	0.177	0.091	0.047	0.057	0.754
	$k = 20$	**0.034**	0.041	0.122	0.064	0.040	0.041	0.822
DGP5	$k = 5$	**0.061**	0.123	0.467	0.248	0.096	0.110	0.665
	$k = 10$	0.035	0.087	0.257	0.197	0.071	**0.032**	0.722
	$k = 20$	0.025	0.055	0.138	0.124	0.061	**0.023**	0.762
DGP6	$k = 5$	0.314	0.278	**0.116**	0.380	0.319	0.319	0.673
	$k = 10$	0.177	0.175	**0.133**	0.275	0.202	0.183	0.723
	$k = 20$	0.094	0.100	**0.088**	0.149	0.115	0.092	0.747

much lower than the proposed technique. In case of the SIR model, we observe that all approaches record comparable performances in all simulations. From the results of $\alpha = 0.9$ in our method and $\kappa = 0.1$ in CWHC, it appears that the data generated from the SIR models in our study do not show profound between-cluster temporal distance and the sample sizes used here are not sufficient enough to pick the true clusters properly. Furthermore, the results for different metrics demonstrate that CWHC records better performance if the number of clusters is small, but our method tends to work better if k is larger. Meanwhile, in DGP6, one can ascertain that our proposed approach captures the temporal closeness in a much more profound way than the competitors. Even though the setting includes only temporal clusters, the SP and BL values for $\alpha = 0.9$ reflect that the accuracy is commendable. It is also imperative to note that CWHC is a better approach for only spatial clustering, especially when number of classes is bigger. This can be supported by looking at both symmetric purity and Binder loss.

Continuing with the above results, we take a look at the best values of α for different data generating processes. In DGP1 and DGP4, a value closer to 0 turns out to be the best whereas in DGP2 and DGP3, a value closer to 1 is the most effective. This can be easily explained from the way the clusters are assigned in these setups. Based on these results, we hypothesize that a lower value of α would work out the best if the extent of spatial dependence is considerably more than the temporal dependence. In the opposite scenario, higher value of α should be preferred. These results are further substantiated by the findings in case of DGP5 and DGP6. However, without the knowledge of the underlying data generating process, it is impossible to determine whether spatial or temporal dependence is more significant. To that end, the concept of gap statistic presented in Section 3.3 can be utilized. Below, we present simulation results to establish the usefulness of the gap statistic in determining the optimum number of clusters as well as the most useful choice of α. We shall restrict ourselves to the first three DGPs since the results for DGP4 mostly match that of DGP1 and the other two extreme cases depict distinguishing pattern as illustrated above.

For the interest of space, we discuss the results for a particular scenario ($n = 100$, $k = 10$, $T = 500$), which is in fact in line with the real-life applications presented in Section 5. Results from the other scenarios largely match the above. For every DGP in this exercise, data are generated multiple times and we run our algorithm using the gap statistic to find out the optimum choice of k. The accuracy measures are next computed based on the clustering assignments using that optimum value of k. In all calculations of the gap statistic, 300 bootstrapped samples are used. Our experiments show that around this value, the results become stable. Also as mentioned above, our findings

at Figure 4 reveal that the best solution is attained for α strictly between 0 and 1 if the data is generated from a space-time process; whereas for purely spatial or purely temporal clustering frameworks, the optimum solutions are expected at the edge cases of $\alpha = 0$ and $\alpha = 1$. Moreover, the gap statistic does not have any meaning for purely spatial clustering. In this light, we restrict ourselves to $\alpha \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. In Figure 5, in the top panel, boxplots of the optimum values of k chosen by the gap statistic for different values of α are displayed, while in the bottom panel, we present the boxplots of the SP values for the optimum cluster assignments.

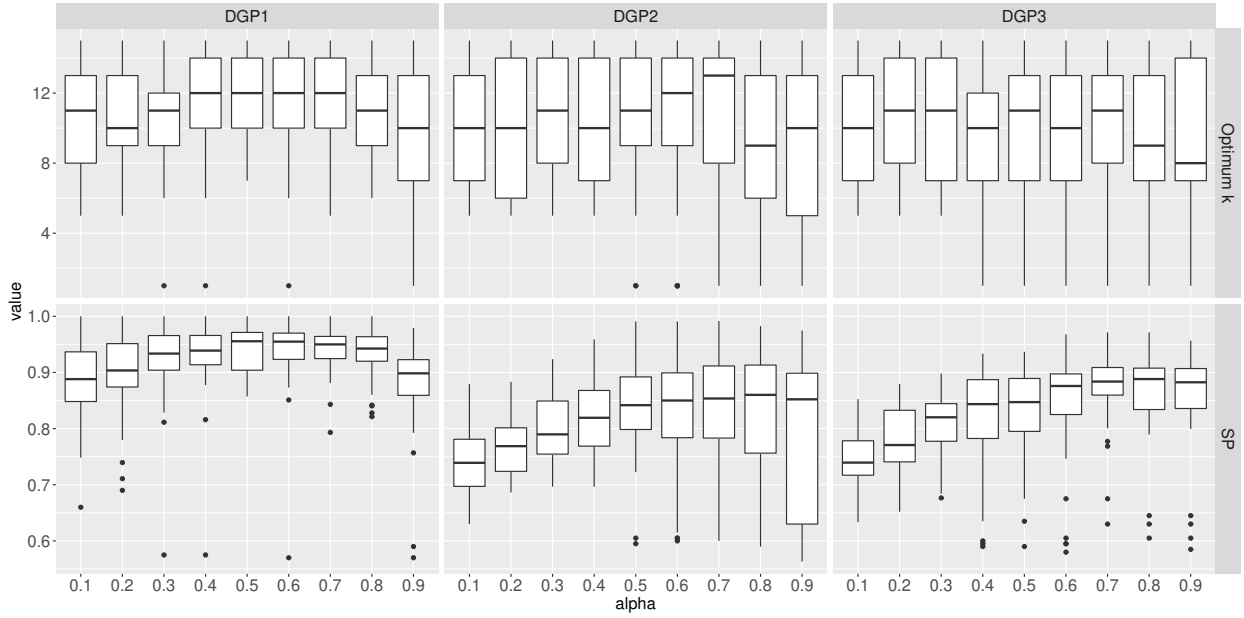

Figure 5. Boxplots of the optimum cluster number (top panel) and the symmetric purity values (bottom panel) for many simulations under the setup $n = 100$, $k = 10$, $T = 500$. Plots for different DGPs are presented in different columns.

Barring some outliers, we see that the optimum number of clusters always remains within 5 and 14. For DGP1, the ranges are most concise. For DGP3 too, the values are usually above 7 whereas for DGP2, the ranges are slightly bigger. Furthermore, notice that the results for $0.6 \leqslant \alpha \leqslant 0.8$ are most stable. The same thing can be seen in the SP values as well. Here, the highest accuracy tends to be achieved for similar values of α, which demonstrates that a convex combination where slightly more weight is enforced on the temporal closeness is the most appropriate. The level of accuracy in these cases is usually above 0.8. We hypothesize that the accuracy would increase further if T is increased. An optimum value of α is expected to be closer to 1 in those cases. It is also worth mentioning that, akin to the earlier finding, experimentation showed that the accuracy increases significantly if n/k increases.

To conclude this section, we provide the average accuracy, along with average values of the optimum choices of α and k, over all simulations. Table 3 supports our earlier observation that for all DGPs, the best choice of α is slightly above 0.5. It is also evident that, on average, the best choice of k is close to the truth, although the method of gap statistic has a tendency to overestimate it. However, that does not cause a significant loss in accuracy.

Table 3. Averages (taken over all simulations) of the optimum choices of the mixing parameter (α), cluster number (k) and the corresponding accuracy measures.

DGP	Optimum α	Optimum k	SP	FM
DGP1	0.756	11.1	0.916	0.83
DGP2	0.689	11.4	0.832	0.664
DGP3	0.762	11.1	0.873	0.733

5. Spatio-temporal clustering of COVID-19 data

We recall that the data used in this paper includes incidence rate of COVID-19 of 3091 counties from 435 days. As an initial exploration, we look into the stationarity of the time series for each location using the augmented Dickey-Fuller (ADF) test. We consider the first type ADF test which utilizes a model with no drift and linear trend term with respect to time, and implement it for a maximum lag of 5. The R package aTSA by [81] is used in this regard. Overall, we find that the incidence series for around 79.8% of the counties display stationary patterns which provides acceptable support towards our theoretical assumption of stationarity. Non-stationary behavior is only more common in some smaller states around the New England region in the north-east side of the country. Now, considering that the data encompasses a huge region with a diverse range of longitude and latitude, the extent of spatial dependence is expected to behave very differently in different parts of the country. Furthermore, various COVID-19 regulations have been enforced by state and local governments and can have different effects on the incidence rates. Considering these, we find it prudent to apply our algorithm on localized datasets. Not only it provides more meaningful understanding of the problem, but it also alleviates the computational burden to handle 3091 counties together in the optimization step via gap statistic. However, before delving deeper into those results, in Figure 6, we look at the results for the entire country if all the locations are divided into 10 clusters. Recall that the ACF-based approach failed to provide any meaningful result for this problem (refer to Figure 1 in Section 2). Contrarily, our proposed algorithm, by virtue of combining spatial and temporal closeness appropriately, identifies interesting patterns across the whole country.

In the middle panel of Figure 6, when the spatial distance and the temporal distance are assigned equal weights, we see that the entire north-east region falls within the same cluster, thereby indicating that the disease spread has been consistent across this region. Right below that, New York, Pennsylvania and New Jersey observe different clustering assignments within the state. One can relate it to the fact that this region has more international connections and are extremely heterogeneous in terms of ethnicity or socio-economic status. Quite interestingly, a similar variation in the clusters is observed in the state of Illinois as well (red and blue cluster in the middle), especially around Chicago which is another big city with similar properties. We also observe that for most states, the counties in the central parts are in the same cluster while some variations are noted in the boundaries. It clearly resonates with the idea that state laws and regulations have played a critical role in determining the COVID-19 progression.

We further point out that even when we choose $\alpha = 0.9$ (bottom panel of Figure 6), the clusters appear more connected than what we saw in the ACF based clustering. In fact, the results are almost identical as in the previous case ($\alpha = 0.5$) in many regions such as New England; Texas, Louisiana and surrounding areas (south); Washington, Oregon and California (west) etc. On the other hand, states like New York, Illinois, Florida etc. are found to have considerably more variation within themselves. Overall, we see that the countrywide results are more interpretable and they establish the usefulness of the proposed algorithm. To better understand the COVID-19 progression in the country, we next turn attention to the results obtained for data at a more local scale.

In what follows, we work with the ten standard federal regions suggested by the Office of Management and Budget in 1969 (OMB Circular A-105). It should be pointed out that this regional division is followed by the Environmental Protection Agency and the Department of Health, Education, and Welfare; and we consider it relevant in the study of COVID-19 as well. The regions, along with the total number of counties in each of them, are described in Table 4.

Following Section 3.3, we apply the gap statistic separately on the ten regions to find optimal number of clusters, and look at the optimal number of clusters in each. Algorithm 2 is run for $\alpha \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. In Table 5, optimum numbers of clusters in each region for different α are presented. Corresponding values of the gap statistic are included in the table as well. It is imperative to point out that these values can also be used to optimize over α. Additionally, we compute the similarity of the clustering assignment for different α with respect to the purely spatial clustering (which corresponds to the case of $\alpha = 0$), by using the symmetric purity index defined earlier. The columns with heading SP(0) in the table display these values. A value closer to 1 indicates greater similarity to the purely spatial case. We note that under the optimal choices, the symmetric purity values are usually around 0.5 to 0.6, which signify that the temporal closeness impact the clustering results more than the spatial proximity. Even in the situation of $\alpha = 0.1$, the outcomes are similar to the spatial clustering only for regions II and VIII, and not for the other regions. Thus, we may argue that the most optimal clustering outputs are more affected by the temporal distances, while also maintaining the spatial closeness between the counties.

A key observation from table 5 is that the optimum numbers of clusters vary considerably with α. If we look at the best cases, i.e. where the gap statistic is maximized, we observe that barring two regions, the optimum number

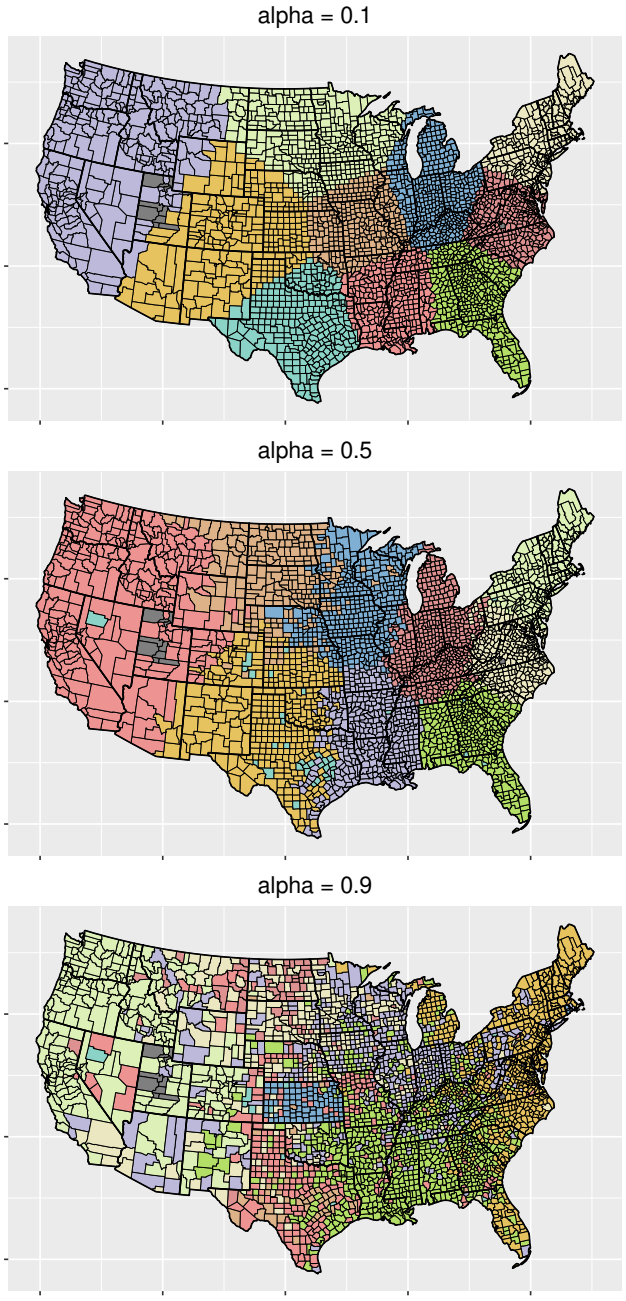

Figure 6. Clustering results (for 10 clusters) for different mixing parameters (α) for the COVID-19 incidence data from the USA.

of clusters is around 10 only. In region IV (8 states, 736 counties), the algorithm returns 27 clusters whereas in region VI (5 states, 503 counties), as many as 36 clusters are found. In both cases, the optimal value of α is obtained to be 0.9. Considering that α represents the contribution of the temporal closeness in the clustering algorithm, we conjecture that possibly the temporal variation in the incidence series for these regions has been more significant and more diverse. A detailed investigation of separate clusters in each case can be an interesting follow-up study to assert this and to obtain additional insights about every region. Moreover, upon further inspection we find out that the gap objective function assumes values close to each other for different k and that could also lead to somewhat abruptly big

Table 4. Ten standard federal regions of the USA.

Region	States	Counties
Region I	Connecticut, Maine, Massachusetts, New Hampshire, Rhode Island, Vermont	65
Region II	New Jersey, New York	83
Region III	Delaware, District of Columbia, Maryland, Pennsylvania, Virginia, West Virginia	279
Region IV	Alabama, Florida, Georgia, Kentucky, Mississippi, North Carolina, South Carolina, Tennessee	736
Region V	Illinois, Indiana, Michigan, Minnesota, Ohio, Wisconsin	524
Region VI	Arkansas, Louisiana, New Mexico, Oklahoma, Texas	503
Region VII	Iowa, Kansas, Missouri, Nebraska	412
Region VIII	Colorado, Montana, North Dakota, South Dakota, Utah, Wyoming	280
Region IX	Arizona, California, Nevada	90
Region X	Idaho, Oregon, Washington	119

Table 5. Optimal number of clusters (k) for the ten regions and corresponding values of gap statistic for different values of α. Maximum values of gap statistic and corresponding k are shown in bold. SP(0) indicates the similarity with purely spatial clustering, computed via the symmetric purity index.

	Region I			Region II			Region III			Region IV			Region V		
α	k	Gap	SP(0)	k	Gap	SP(0)	k	Gap	SP(0)	k	Gap	SP(0)	k	Gap	SP(0)
0.1	8	0.048	0.585	6	0.078	0.934	18	0.123	0.592	37	0.096	0.677	33	0.071	0.644
0.2	8	0.078	0.585	5	0.182	0.762	13	0.221	0.576	14	0.130	0.632	26	0.144	0.675
0.3	8	0.090	0.585	14	0.209	0.709	8	0.354	0.678	42	0.056	0.720	22	0.189	0.703
0.4	8	0.073	0.585	12	0.241	0.733	23	0.408	0.564	42	0.157	0.719	18	0.179	0.641
0.5	15	0.048	0.569	12	0.315	0.696	27	0.494	0.545	41	0.762	0.591	**14**	**0.221**	**0.601**
0.6	14	0.090	0.554	12	0.208	0.692	24	0.569	0.563	30	0.539	0.676	15	0.109	0.538
0.7	15	0.413	0.562	9	0.335	0.573	9	0.767	0.676	39	0.907	0.722	27	0.081	0.485
0.8	13	0.425	0.569	7	0.417	0.539	**11**	**0.833**	**0.646**	33	1.026	0.684	23	0.058	0.461
0.9	**11**	**0.517**	**0.600**	**7**	**0.438**	**0.523**	16	0.734	0.582	**27**	**1.372**	**0.624**	27	0.097	0.398
	Region VI			Region VII			Region VIII			Region IX			Region X		
α	k	Gap	SP(0)	k	Gap	SP(0)	k	Gap	SP(0)	k	Gap	SP(0)	k	Gap	SP(0)
0.1	30	0.109	0.726	32	0.158	0.681	14	0.093	0.904	15	0.157	0.691	17	0.199	0.685
0.2	18	0.183	0.681	20	0.262	0.732	27	0.330	0.735	8	0.321	0.928	12	0.223	0.779
0.3	20	0.224	0.669	17	0.359	0.685	26	0.523	0.757	6	0.541	0.746	7	0.332	0.839
0.4	19	0.224	0.679	14	0.418	0.674	18	0.636	0.791	9	0.669	0.903	17	0.411	0.760
0.5	11	0.230	0.633	11	0.522	0.671	18	0.831	0.744	9	0.698	0.881	18	0.640	0.776
0.6	12	0.253	0.613	**10**	**0.612**	**0.667**	11	0.975	0.734	4	0.815	0.691	14	0.710	0.796
0.7	14	0.153	0.564	32	0.488	0.613	11	1.066	0.700	5	1.236	0.727	14	0.735	0.771
0.8	36	0.446	0.570	10	0.442	0.601	9	1.338	0.636	5	1.217	0.729	11	1.503	0.672
0.9	**36**	**0.456**	**0.563**	31	0.356	0.524	**9**	**1.485**	**0.594**	**4**	**1.427**	**0.787**	**11**	**1.507**	**0.597**

maximizer sometimes. In the other eight regions, on the contrary, lower number of clusters represents less temporal variation across the region. A particularly interesting example is region IX where 90 counties are split into only four clusters. In fact, it is found that only two clusters cover most of the counties in this region. Another intriguing case is that of region V, where the best value of the gap statistic is obtained at $\alpha = 0.5$, thereby reflecting that both spatial and temporal closeness are equally important in this case. We shall look into this region in more detail in the following subsection. We also point out that the flexibility of the proposed algorithm facilitates its use in different granularity, for instance, to identify the clusters within a single state. This is demonstrated at length in Section 5.2 using only the data from California.

5.1. Results for the Midwest (Region V)

For the Midwest region, the best value of α is obtained as 0.5. Correspondingly, 524 counties are divided into only 14 clusters. Below, in Figure 7, we take a detailed look at the clustering outputs for all cases in this region. Similar plots of all other regions are included in the supplement for conciseness of the paper.

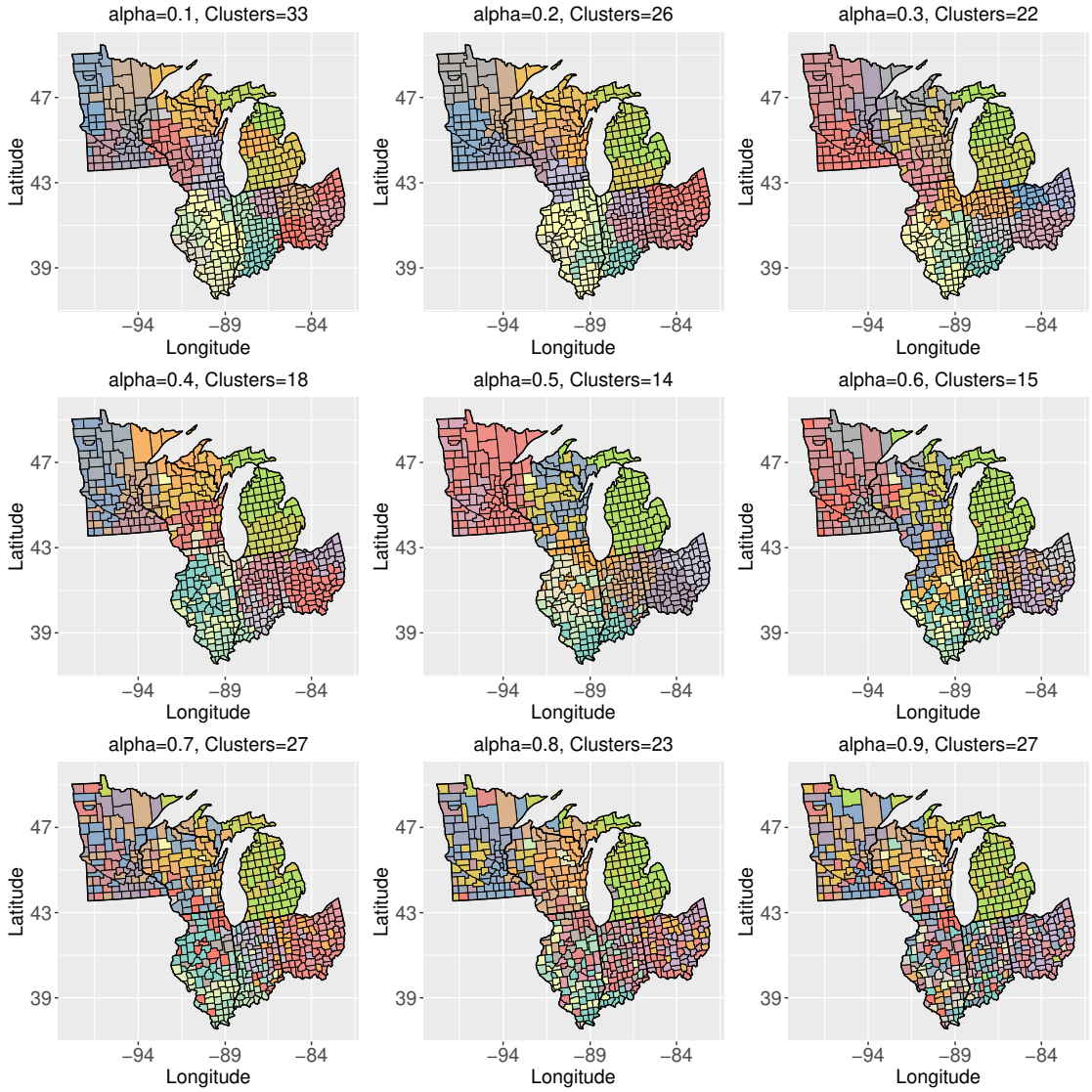

Figure 7. Clustering results for the optimum choices of the cluster number corresponding to different mixing parameters (α) for region V in the COVID-19 incidence data.

It is evident that for lower α, the clusters are a lot more connected than what happens for higher values of α. Looking at the optimum case of $\alpha = 0.5$, we can identify a few interesting patterns. For example, the state of Michigan turns out to be mostly covered in green color, indicating a single cluster across the state. In fact, the green cluster almost marks the state border, which indicates that the counties in Michigan display identical pattern and that the temporal pattern is different from the rest of the region. This phenomena can be attributed to the fact that Michigan was one of the few states heavily impacted by the pandemic in the early days. It subsequently compelled the government to impose strict preventative measures which eventually brought down the incidence rates dramatically (see the report from [82]).

On the other hand, we note that there is a very small localized cluster, marked in darker yellow, around Chicago (situated at the border of Illinois, Indiana and Michigan). It is the biggest city in this entire region and naturally it attracts people from neighboring counties as well as from other states. Therefore, the authorities needed to plan differently and as a consequence, stricter and more specific rules have been adopted for the city, some of which have not been applied to other parts of the state of Illinois. The report by [83] is relevant in this regard. We hypothesize that these measures have caused different temporal progressions for Chicago and surrounding areas which subsequently resulted in the small cluster alluded to above.

We also point out that, akin to Michigan, Minnesota (leftmost state in the graph) and Ohio (rightmost-bottom state in the graph) constitute of primarily two clusters. It implies that the spread of the pandemic has seen less variation across different counties in these states. In comparison, the other three states of Wisconsin, Illinois and Indiana observe more variations in the incidence series. These insights can be valuable in building suitable statewide or local policies to curtail the effects of the pandemic.

5.2. Results for California

As a last piece of analysis in this section, we steer our focus to a state-level analysis. Such an analysis can help in understanding the progression of the pandemic in different counties within a state. As an illustration, in Table 6 and Figure 8, we show the results for California, the most populated state in the country.

Table 6. Optimum choices of cluster number and the corresponding gap statistic for the COVID-19 incidence data in California, for different values of the mixing parameter α. Maximum values of the gap statistic and the corresponding k are shown in bold.

α	k	Gap
0.1	12	0.338
0.2	8	0.294
0.3	8	0.323
0.4	5	0.462
0.5	3	0.867
0.6	3	0.914
0.7	**3**	**0.939**
0.8	3	0.891
0.9	3	0.931

Observe that for all $\alpha \geqslant 0.5$, the optimum number of clusters is obtained as 3, which is striking as it points to very less variation across the whole state. The best value of the gap statistic is obtained for $\alpha = 0.7$. Looking at this case in the figure, we further find out that the northern counties primarily display a similar pattern while the southern counties are in another cluster altogether. Only the Lassen county is in the third cluster. It is one of the less populated areas in California. Reports (e.g. [84]) reveal that it was one of the least affected counties in the whole state in the early days of the pandemic. However, later in the time horizon, more than 20% people got affected in this county. This different trajectory of the incidence is potentially responsible for our method to find it as a singleton cluster. Finally, these results demonstrate that albeit California has not seen hugely different effects of the pandemic, specific policies can be adopted differently in the northern part than in the southern part.

6. Conclusion and discussion

In this paper, we propose a novel clustering method for different spatial units based on their similarity of temporal dynamics. Our motivations are drawn from analyzing COVID-19 progression in the USA based on the respective time-series obtained in all the counties from the contiguous USA. We first delineate how we differ from the traditional spatio-temporal clustering. In particular, we are hereby trying to cluster the spatial locations instead of obtaining clusters in space and time continuum. In the context of COVID-19 data in US, such clusters are particularly meaningful since different policies regarding lockdown and other restrictions are usually more locally adopted instead of being federally mandated. We have learnt over the past two years that outbreaks causing from new variants do not happen

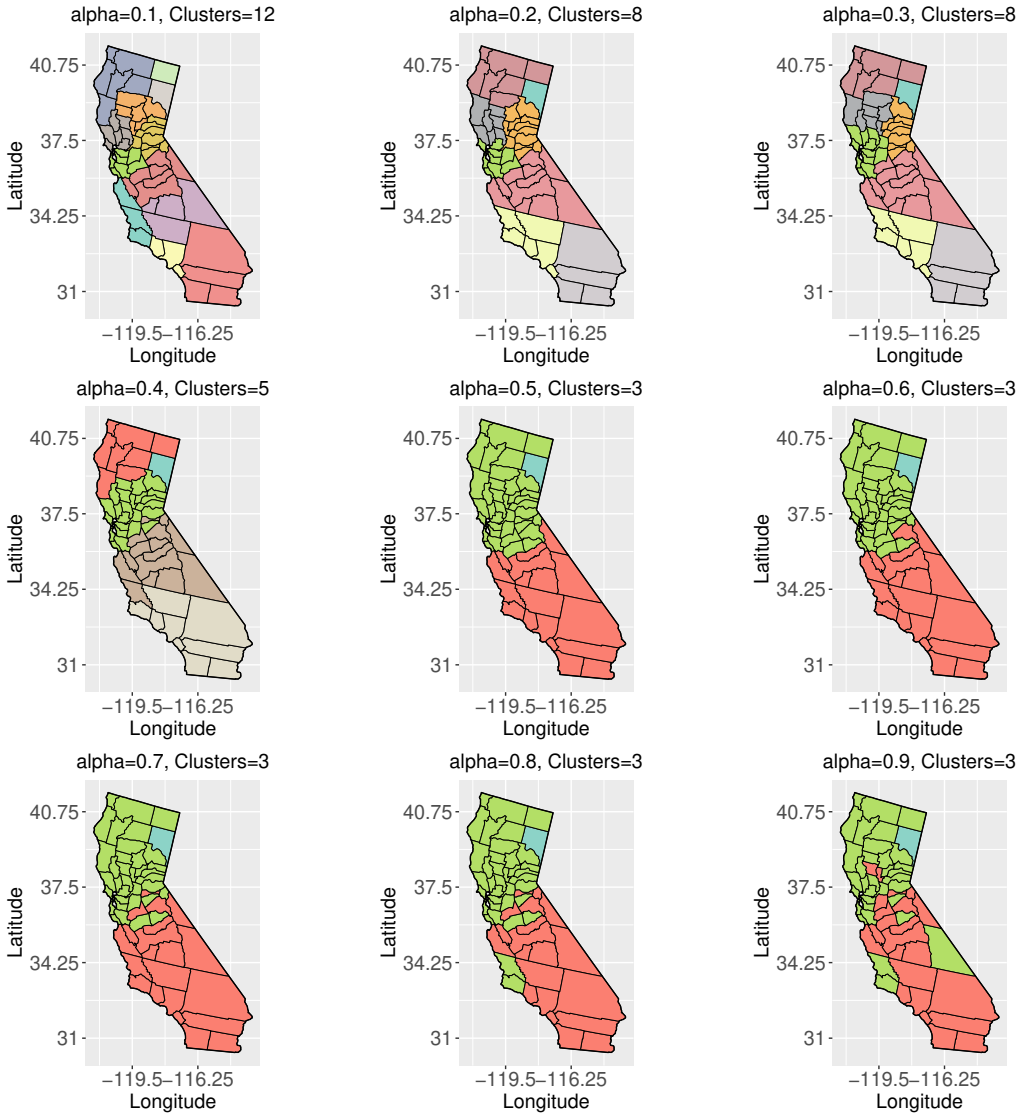

Figure 8. Clustering results for the optimum choices of the cluster number corresponding to different mixing parameters (α) for the COVID-19 incidence data in California.

simultaneously everywhere, so potentially if we find two locations in the same cluster with very different current numbers, there might be reason to believe that eventually they will follow similar trajectory and strategies can be adapted based on that.

While the traditional space-time hotspots or clusters can shed light on if any particular gathering created a surge in the number of cases, our approach is more holistic. It looks at the entire time series of two locations and identifies how similar they are. Such similarity pursuit has been tried in the literature using conventional approaches such as ACF, but we show that they do not really give meaningful results as it can produce abrupt clusters. This motivated us to propose a clustering method that can ensure somewhat optimal spatial continuity. We define a distance between two locations by a weighted average of their harvesine distance and the distance between the spectral density of the respective time-series observed and further use this to obtain k-means clusters. We also employ a gap statistic type criteria to determine the optimal number of clusters. Note that, the amount of spatial continuity injected by this method or the optimal weight for this bridged distance is chosen from possible grid values between 0 to 1 as

each such value can provide different insights. However, if needed, this can also be chosen objectively through a constrained gap-like criterion or some form of cross-validation which might be less computationally expensive. We mention this below as one direction of a future research persuasion. We evaluate our method on various simulated datasets and show that the retrieved clusters have a significant agreement with the true ones. Subsequently, detailed discussion of the COVID-19 clusters, especially related to the Midwest and the state of California, are provided. It is observed that the clusters provide meaningful socio-economic interpretation. An obvious natural question arises as how deterministic is this study in delineating specific policy-related effect on the time-series observed in different location. Note that, we wanted to keep our approach here non-parametric and thus we avoided bringing in additional covariate. An immediate future work along this line could be whether the clustering structures remain intact when additional covariates, possibly including some policy interventions are brought into picture.

As concluding remarks, we now outline some other future works that follow naturally given our exposition in this paper. Note that we have implemented the optimization algorithm on region-wise data from the USA instead of the entire country taken together. The reasons are two-fold. First, as we mentioned above, since it is a huge country with more local pattern than global, it does not make sense to bring all spatial units together for finding potential number of clusters. The second reason is more technical. Determining the optimal weight and the appropriate number of clusters through a resample based method like gap statistics poses huge computational burden. As [72] and others developing the literature of gap statistic suggests, a bootstrap sample of at least 300 makes the analysis robust. With over 3000 locations, each having a sample time series of around 500 time points, examining $\sqrt{3000} \approx 55$ candidate values for the optimum number of clusters for 300 resamples is extremely time-consuming. In order to circumvent this problem, one can potentially adopt a divide-and-conquer technique, and that provides an interesting future direction to this work. On a related note, the bootstrap steps outlined in Section 3.3 computes the means under the resampling distribution. For large number of locations, one can then use a large number of parallel processors, compute a small number of resamples in each and combine the means from each such processor to arrive at the grand average. Moreover, one can choose to do a constrained optimization of the gap-statistic objective function to reduce the range of potential optimal number of clusters and thus save more on the computation.

Another important point, as observed from our simulation study, is that we typically obtain 11 clusters where the true number of clusters is 10. Such an overestimation phenomenon is not new when the gap statistic is used in its original form and has been discussed in earlier works. One of the possible remedies could be using the DD-weighted gap statistic proposed by [73]. We did not need to pursue this direction, as the gap-objective values for $k = 10$ and $k = 11$ were usually very close and even with an additional cluster, the accuracy measures were close to 1. We believe, however, that it is possible to slightly tweak the related objective functions by means of adding suitable penalty that can objectively handle this overestimation when needed.

As a final remark, we note that the COVID-19 data is not point-level but we use the proposed technique for this aggregate areal data following similar works [85, 86]. It is thus crucial to point out that although COVID-19 progression analysis was the main motivation behind this work, our methodology can extend to a large number of applications that deal with spatio-temporal datasets. One interesting example from the field of environmental research is included in the supplementary file. The results strengthen the point that combining spatial and temporal distance in the way we propose leads to valuable insights in spatio-temporal clustering problems. There is a recent surge of large publicly-available geospatial data sources with accurate timestamps in many applications such as crime analysis, traffic flow, climate variables, etc. Obtaining clusters of spatial locations based on their temporal dynamics can lead to better strategic planning and resource allocations.

Acknowledgements

We are thankful to the associate editor and two anonymous referees for providing important feedback which has helped us improve the quality of this paper. The research of the second author is partially supported by NSF DMS 2124222.

References

[1] I. A. Adekunle, S. A. Tella, K. O. Oyesiku, I. O. Oseni, Spatio-temporal analysis of meteorological factors in abating the spread of COVID-19 in Africa, Heliyon 6 (8) (2020) e04749.

[2] X. Chen, R. Quan, A spatiotemporal analysis of urban resilience to the COVID-19 pandemic in the Yangtze River Delta, Natural Hazards 106 (1) (2021) 829–854.
[3] S. Rawat, S. Deb, A spatio-temporal statistical model to analyze COVID-19 spread in the USA, Journal of Applied Statistics (2021) 1–20.
[4] B. Sartorius, A. Lawson, R. Pullan, Modelling and predicting the spatio-temporal spread of COVID-19, associated deaths and impact of key risk factors in England, Scientific reports 11 (1) (2021) 1–11.
[5] S. K. Greene, E. R. Peterson, D. Balan, L. Jones, G. M. Culp, A. D. Fine, M. Kulldorff, Detecting Emerging COVID-19 Community Outbreaks at High Spatiotemporal Resolution-New York City, June-July 2020, medRxiv.
[6] M. C. Castro, S. Kim, L. Barberia, A. F. Ribeiro, S. Gurzenda, K. B. Ribeiro, E. Abbott, J. Blossom, B. Rache, B. H. Singer, Spatiotemporal pattern of COVID-19 spread in Brazil, Science 372 (6544) (2021) 821–826.
[7] S. Wang, K. Wei, L. Lin, W. Li, Spatial-temporal Analysis of COVID-19's Impact on Human Mobility: the Case of the United States, arXiv preprint arXiv:2010.03707.
[8] R. Elson, T. M. Davies, I. R. Lake, R. Vivancos, P. B. Blomquist, A. Charlett, G. Dabrera, The spatio-temporal distribution of COVID-19 infection in England between January and June 2020, Epidemiology & Infection 149.
[9] A. S. Almobarak, H. R. Almohammadi, S. A. Aboalnaser, L. Syed, Spatio-Temporal Analysis of the Spread COVID-19 in Saudi Arabia, in: 2020 13th International Conference on Developments in eSystems Engineering (DeSE), IEEE, 2020, pp. 341–346.
[10] E. K. Mylona, F. Shehadeh, M. Kalligeros, G. Benitez, P. A. Chan, E. Mylonakis, Real-Time Spatiotemporal Analysis of Microepidemics of Influenza and COVID-19 Based on Hospital Network Data: Colocalization of Neighborhood-Level Hotspots, American journal of public health 110 (12) (2020) 1817–1824.
[11] G. S. Bhunia, S. Roy, P. K. Shit, Spatio-temporal analysis of COVID-19 in India–a geostatistical approach, Spatial Information Research (2021) 1–12.
[12] P. Purwanto, S. Utaya, B. Handoyo, S. Bachri, I. S. Astuti, K. S. B. Utomo, Y. E. Aldianto, Spatiotemporal analysis of COVID-19 spread with emerging hotspot analysis and space–time cube models in East Java, Indonesia, ISPRS International Journal of Geo-Information 10 (3) (2021) 133.
[13] C. Mo, D. Tan, T. Mai, C. Bei, J. Qin, W. Pang, Z. Zhang, An analysis of spatiotemporal pattern for COIVD-19 in China based on space-time cube, Journal of medical virology 92 (9) (2020) 1587–1595.
[14] L. Ye, L. Hu, Spatiotemporal distribution and trend of COVID-19 in the Yangtze River Delta region of the People's Republic of China, Geospatial Health 15 (1).
[15] Y. L. Cheong, S. Mohd Ghazali, M. K. Che Ibrahim, C. C. Kee, N. H. Md Iderus, B. Singh Gill, C. H. Florence Lee, K. H. Lim, et al., Assessing the spatiotemporal spread patterns of the COVID-19 pandemic in Malaysia, Frontiers in Public Health (2022) 301.
[16] Y. Choi, A. Ladoy, D. De Ridder, D. Jacot, S. Vuilleumier, C. Bertelli, I. Guessous, T. Pillonel, S. Joost, G. Greub, Detection of SARS-CoV-2 infection clusters: The useful combination of spatiotemporal clustering and genomic analyses, Frontiers in Public Health 10 (ARTICLE) (2022) 1016169.
[17] T. Choudhury, R. Arunachalam, A. Khanna, E. Jasinska, V. Bolshev, V. Panchenko, Z. Leonowicz, A Social Network Analysis Approach to COVID-19 Community Detection Techniques, International Journal of Environmental Research and Public Health 19 (7) (2022) 3791.
[18] L. Jie, L. Xin, H. Guihua, Z. Tinghui, Information Visualization Technology and Mathematical Statistics Applied in the Research on Spatiotemporal Clustering and Changing of COVID-19, in: 2022 IEEE 2nd International Conference on Data Science and Computer Application (ICDSCA), IEEE, 2022, pp. 667–672.
[19] L. Liu, Q. Meng, K. Qin, G. Xu, Spatio-temporal variations of the covid-19 epidemic in mexico, in: 2022 29th International Conference on Geoinformatics, IEEE, 2022, pp. 1–8.
[20] M. Midoun, A. Amrani-Midoun, Analysis of Spatiotemporal Pattern for COVID-19 in Algeria Using Space-Time-Cubes, International Review on Modelling and Simulations (2022) 27–35.
[21] M. Kulldorff, A spatial scan statistic, Communications in Statistics-Theory and methods 26 (6) (1997) 1481–1496.
[22] B. Anbaroğlu, T. Cheng, B. Heydecker, Non-recurrent traffic congestion detection on heterogeneous urban road networks, Transportmetrica A: Transport Science 11 (9) (2015) 754–771.
[23] S. Hudjimartsu, T. Djatna, A. Ambarwari, et al., Spatial temporal clustering for hotspot using kulldorff scan statistic method (KSS): A case in Riau Province, in: IOP Conference Series: Earth and Environmental Science, Vol. 54, IOP Publishing, 2017, p. 012056.
[24] M. E. Kamenetsky, J. Lee, J. Zhu, R. E. Gangnon, Regularized spatial and spatio-temporal cluster detection, Spatial and Spatio-temporal Epidemiology 41 (2022) 100462.
[25] S. Kisilevich, F. Mansmann, M. Nanni, S. Rinzivillo, Spatio-temporal clustering, in: Data mining and knowledge discovery handbook, Springer, 2009, pp. 855–874.
[26] F. Di Martino, W. Pedrycz, S. Sessa, Spatiotemporal extended fuzzy C-means clustering algorithm for hotspots detection and prediction, Fuzzy Sets and Systems 340 (2018) 109–126.
[27] D. Birant, A. Kut, ST-DBSCAN: An algorithm for clustering spatial–temporal data, Data & knowledge engineering 60 (1) (2007) 208–221.
[28] K. Agrawal, S. Garg, S. Sharma, P. Patel, Development and validation of OPTICS based spatio-temporal clustering technique, Information Sciences 369 (2016) 388–401.
[29] J. Liu, C. Xue, Y. He, Q. Dong, F. Kong, Y. Hong, Dual-constraint spatiotemporal clustering approach for exploring marine anomaly patterns using remote sensing products, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11 (11) (2018) 3963–3976.
[30] H. Izakian, W. Pedrycz, I. Jamal, Clustering spatiotemporal data: An augmented fuzzy c-means, IEEE transactions on fuzzy systems 21 (5) (2012) 855–868.
[31] M. G. Doborjeh, N. Kasabov, Dynamic 3D clustering of spatio-temporal brain data in the NeuCube spiking neural network architecture on a case study of fMRI data, in: International Conference on Neural Information Processing, Springer, 2015, pp. 191–198.
[32] S. Gaffney, P. Smyth, Trajectory clustering with mixtures of regression models, in: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, 1999, pp. 63–72.
[33] D. Chudova, S. Gaffney, E. Mjolsness, P. Smyth, Translation-invariant mixture models for curve clustering, in: Proceedings of the ninth ACM

SIGKDD international conference on Knowledge discovery and data mining, 2003, pp. 79–88.
[34] X. Liu, M. C. Yang, Simultaneous curve registration and clustering for functional data, Computational Statistics & Data Analysis 53 (4) (2009) 1361–1376.
[35] J. Jacques, C. Preda, Functional data clustering: a survey, Advances in Data Analysis and Classification 8 (2014) 231–255.
[36] A. J. Suarez, S. Ghosal, Bayesian clustering of functional data using local features, Bayesian Analysis 11 (1) (2016) 71–98.
[37] G. Hu, J. Geng, Y. Xue, H. Sang, Bayesian spatial homogeneity pursuit of functional data: an application to the us income distribution, Bayesian Analysis 1 (1) (2022) 1–27.
[38] V. Vandewalle, C. Preda, S. Dabo-Niang, Clustering spatial functional data, Geostatistical Functional Data Analysis (2022) 155–174.
[39] J. Alon, S. Sclaroff, G. Kollios, V. Pavlovic, Discovering clusters in motion time-series data, in: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings., Vol. 1, IEEE, 2003, pp. I–I.
[40] V. Gómez-Rubio, P. Moraga, J. Molitor, B. Rowlingson, DClusterm: model-based detection of disease clusters, Journal of Statistical Software 90 (2019) 1–26.
[41] L. E. Nieto-Barajas, A. Contreras-Cristán, A bayesian nonparametric approach for time series clustering.
[42] F. Finazzi, R. Haggarty, C. Miller, M. Scott, A. Fasso, A comparison of clustering approaches for the study of the temporal coherence of multiple time series, Stochastic environmental research and risk assessment 29 (2015) 463–475.
[43] C. Fernández, P. J. Green, Modelling spatially correlated data via mixtures: a Bayesian approach, Journal of the royal statistical society: series B (Statistical methodology) 64 (4) (2002) 805–826.
[44] L. Paci, F. Finazzi, Dynamic model-based clustering for spatio-temporal data, Statistics and Computing 28 (2018) 359–374.
[45] S. Frühwirth-Schnatter, S. Kaufmann, Model-based clustering of multiple time series, Journal of Business & Economic Statistics 26 (1) (2008) 78–89.
[46] C. Viroli, Model based clustering for three-way data structures, Bayesian Analysis 6 (4) (2011) 573–602.
[47] B. Neelon, A. E. Gelfand, M. L. Miranda, A multivariate spatial mixture model for areal data: examining regional differences in standardized test scores, Journal of the Royal Statistical Society. Series C, Applied statistics 63 (5) (2014) 737.
[48] A. Mozdzen, A. Cremaschi, A. Cadonna, A. Guglielmi, G. Kastner, Bayesian modeling and clustering for spatio-temporal areal data: an application to Italian unemployment, Spatial Statistics 52 (2022) 100715.
[49] T. Goicoa, M. Ugarte, J. Etxeberria, A. Militino, Comparing CAR and P-spline models in spatial disease mapping, Environmental and Ecological Statistics 19 (2012) 573–599.
[50] A. Rushworth, D. Lee, R. Mitchell, A spatio-temporal model for estimating the long-term effects of air pollution on respiratory hospital admissions in Greater London, Spatial and spatio-temporal epidemiology 10 (2014) 29–38.
[51] A. B. Lawson, Hierarchical modeling in spatial epidemiology, Wiley Interdisciplinary Reviews: Computational Statistics 6 (6) (2014) 405–417.
[52] D. Lee, A. Lawson, Quantifying the spatial inequality and temporal trends in maternal smoking rates in Glasgow, The annals of applied statistics 10 (3) (2016) 1427.
[53] G. Napier, D. Lee, C. Robertson, A. Lawson, A bayesian space–time model for clustering areal units based on their disease trends, Biostatistics 20 (4) (2019) 681–697.
[54] V. Nicoletta, A. Guglielmi, A. Ruiz, V. Bélanger, E. Lanzarone, Bayesian spatio-temporal modelling and prediction of areal demands for ambulance services, IMA Journal of Management Mathematics 33 (1) (2022) 101–121.
[55] J. Lee, M. E. Kamenetsky, R. E. Gangnon, J. Zhu, Clustered spatio-temporal varying coefficient regression model, Statistics in medicine 40 (2) (2021) 465–480.
[56] E. Zaghlool, S. ElKaffas, A. Saad, A density-based clustering of spatio-temporal data, in: New Contributions in Information Systems and Technologies, Springer, 2015, pp. 41–50.
[57] D. Zhang, K. Lee, I. Lee, Hierarchical trajectory clustering for spatio-temporal periodic pattern mining, Expert Systems with Applications 92 (2018) 1–11.
[58] Q. Xiang, Q. Wu, Tree-based and optimum cut-based origin-destination flow clustering, ISPRS International Journal of Geo-Information 8 (11) (2019) 477.
[59] L. V. Teixeira, R. M. Assunção, R. H. Loschi, Bayesian space-time partitioning by sampling and pruning spanning trees., J. Mach. Learn. Res. 20 (2019) 85–1.
[60] I. G. N. M. Jaya, H. Folmer, Identifying spatiotemporal clusters by means of agglomerative hierarchical clustering and Bayesian regression analysis with spatiotemporally varying coefficients: methodology and application to dengue disease in Bandung, Indonesia, Geographical Analysis 53 (4) (2021) 767–817.
[61] JHU-CSSE, 2019 Novel Coronavirus COVID-19 (2019-nCoV) Data Repository by Johns Hopkins CSSE (2020).
URL https://github.com/CSSEGISandData/COVID-19
[62] R. Mattera, A weighted approach for spatio-temporal clustering of COVID-19 spread in Italy, Spatial and Spatio-temporal Epidemiology 41 (2022) 100500.
[63] W. Xiang, B. Swallow, Multivariate spatio-temporal analysis of the global COVID-19 pandemic, medRxiv.
[64] M. Siljander, R. Uusitalo, P. Pellikka, S. Isosomppi, O. Vapalahti, Spatiotemporal clustering patterns and sociodemographic determinants of COVID-19 (SARS-CoV-2) infections in Helsinki, Finland, Spatial and spatio-temporal epidemiology 41 (2022) 100493.
[65] S.-Q. Yang, Z.-G. Fang, C.-X. Lv, S.-Y. An, P. Guan, D.-S. Huang, W. Wu, Spatiotemporal cluster analysis of COVID-19 and its relationship with environmental factors at the city level in mainland China, Environmental Science and Pollution Research 29 (9) (2022) 13386–13395.
[66] P. A. Moran, Notes on continuous stochastic phenomena, Biometrika 37 (1/2) (1950) 17–23.
[67] W. B. Wu, Nonlinear system theory: Another look at dependence, Proceedings of the National Academy of Sciences of the United States of America 102 (40) (2005) 14150–14154.
[68] P. Bloomfield, Fourier analysis of time series: an introduction, John Wiley & Sons, 2004.
[69] T. C. Lee, A simple span selector for periodogram smoothing, Biometrika 84 (4) (1997) 965–969.
[70] H. C. Ombao, J. A. Raz, R. L. Strawderman, R. Von Sachs, A simple generalised crossvalidation method of span selection for periodogram

smoothing, Biometrika 88 (4) (2001) 1186–1192.
[71] E. Schubert, P. J. Rousseeuw, Faster k-medoids clustering: improving the PAM, CLARA, and CLARANS algorithms, in: International conference on similarity search and applications, Springer, 2019, pp. 171–187.
[72] R. Tibshirani, G. Walther, T. Hastie, Estimating the number of clusters in a data set via the gap statistic, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63 (2) (2001) 411–423.
[73] M. Yan, K. Ye, Determining the number of clusters using the weighted gap statistic, Biometrics 63 (4) (2007) 1031–1037.
[74] A. Moayedi, R. A. Abbaspour, A. Chehreghan, An evaluation of the efficiency of similarity functions in density-based clustering of spatial trajectories, Annals of GIS 25 (4) (2019) 313–327.
[75] H. Schütze, C. D. Manning, P. Raghavan, Introduction to information retrieval, Vol. 39, Cambridge University Press Cambridge, 2008.
[76] D. A. Binder, Bayesian cluster analysis, Biometrika 65 (1) (1978) 31–38.
[77] D. B. Dahl, D. J. Johnson, P. Müller, Search algorithms and loss functions for Bayesian clustering, Journal of Computational and Graphical Statistics 31 (4) (2022) 1189–1201.
[78] H. Kim, H. Park, Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis, Bioinformatics 23 (12) (2007) 1495–1502.
[79] M. J. Keeling, P. Rohani, Modeling Infectious Diseases in Humans and Animals, Princeton University Press, 2011.
[80] M. Chavent, V. Kuentz-Simonet, A. Labenne, J. Saracco, ClustGeo: an R package for hierarchical clustering with spatial constraints, Computational Statistics 33 (4) (2018) 1799–1822.
[81] D. Qiu, aTSA: Alternative Time Series Analysis, r package version 3.1.2 (2015).
URL https://CRAN.R-project.org/package=aTSA
[82] State of Michigan, MDHHS Epidemic Orders: Gatherings and Face Mask Order, Official website of the State of Michigan.Updated on May 15, 2021.
URL https://www.michigan.gov/coronavirus/resources/orders-and-directives/lists/executive-directives-content/may-6-2021-gatherings-and-face-mask-order
[83] Associated Press, Chicago order: If you travel from a surging COVID-19 state, self-quarantine for 14 days, Daily Herald.
URL https://www.dailyherald.com/news/20200702/chicago-order-if-you-travel-from-a-surging-covid-19-state-self-quarantine-
[84] Los Angeles Times, Tracking the coronavirus in Lassen County, Los Angeles TimesUpdated on May 24, 2021.
URL https://www.latimes.com/projects/california-coronavirus-cases-tracking-outbreak/lassen-county/
[85] X. Jin, B. P. Carlin, S. Banerjee, Generalized hierarchical multivariate CAR models for areal data, Biometrics 61 (4) (2005) 950–961.
[86] J. R. Bradley, S. H. Holan, C. K. Wikle, Multivariate spatio-temporal models for high-dimensional areal data with application to Longitudinal Employer-Household Dynamics, The Annals of Applied Statistics 9 (4) (2015) 1761 – 1791. doi:10.1214/15-AOAS862.
URL https://doi.org/10.1214/15-AOAS862
[87] W. Liu, W. B. Wu, Asymptotics of spectral density estimates, Econometric Theory (2010) 1218–1245doi:10.1017/S026646660999051X.
[88] D. Pollard, Strong consistency of k-means clustering, The Annals of Statistics (1981) 135–140.
[89] H. Jiang, E. Arias-Castro, On the Consistency of Metric and Non-Metric K-Medoids, in: International Conference on Artificial Intelligence and Statistics, PMLR, 2021, pp. 2485–2493.
[90] G. Brown, Some new applications of Riesz products, in: Miniconference on Probability and Analysis, Australian National University, Mathematical Sciences Institute, 1992, pp. 1–13.

7. Proofs

Proof of Theorem 1. In what follows, for each α, we consider the population analogue of the spatio-temporal distance matrix defined in eq. (3.8) by replacing $\hat{f}$ in the definition of $\hat{\Delta}_\Gamma$ with the true spectral density f. To keep notational consistency, let this be denoted as

$$\Delta_{\mathcal{S},\Gamma}(\alpha) = \alpha\left(\frac{\Delta_\Gamma}{\|\Delta_\Gamma\|_F}\right) + (1-\alpha)\left(\frac{\Delta_\mathcal{S}}{\|\Delta_\mathcal{S}\|_F}\right). \tag{7.1}$$

The consistency of the proposed clustering algorithm is proved in three steps. First, we show that the spatio-temporal distance matrix defined in eq. (3.8) converges to the population analogue defined in eq. (7.1) as $T \to \infty$. Next, for $n \to \infty$, using the population version of the distance matrix, it can be argued that the true cluster centers are obtained through the PAM algorithm with probability 1. Finally, we prove that any location is assigned to the true class with probability 1 if α is greater than some $\alpha_0 \in (0,1)$.

To prove the first step, we need the following lemma.

Lemma 1. *Suppose that two time series $(X_{1,t})_{1\leqslant t\leqslant T}$ and $(X_{2,t})_{1\leqslant t\leqslant T}$ are generated from two DGPs with true spectral densities $f_1(\theta)$ and $f_2(\theta)$, respectively. Let $\hat{f}(\boldsymbol{X}_1,\theta)$ and $\hat{f}(\boldsymbol{X}_2,\theta)$ be the corresponding spectral density estimates following eq. (3.4). Then, as $T \to \infty$,*

$$\int_0^{2\pi} \left\|\hat{f}(\boldsymbol{X}_1,\theta) - \hat{f}(\boldsymbol{X}_2,\theta)\right\|^2 \, d\theta \xrightarrow{P} \int_0^{2\pi} \|f_1(\theta) - f_2(\theta)\|^2 \, d\theta. \tag{7.2}$$

Proof. From Theorem 1 of [87], under Assumption 1, we can say that $\sup_\theta \|\hat{f}(\boldsymbol{X}_1, \theta) - f_1(\theta)\| \xrightarrow{P} 0$, as $T \to \infty$. We omit the details for brevity. Thus we can obtain that for all $\theta \in (0, 2\pi)$,

$$\left\|\hat{f}(\boldsymbol{X}_1, \theta) - \hat{f}(\boldsymbol{X}_2, \theta)\right\| \xrightarrow{P} \|f_1(\theta) - f_2(\theta)\|. \tag{7.3}$$

It is also easy to argue that $\|\hat{f}(\boldsymbol{X}_1, \theta) - \hat{f}(\boldsymbol{X}_2, \theta)\|$ is bounded. Then, applying the continuous mapping theorem and the dominated convergence theorem yield the required result. □

Using the above lemma, straightforward calculations imply that for all (i, j) and for all $\alpha \in [0, 1]$,

$$\hat{\Delta}_{\mathcal{S},\Gamma}(\alpha)(i, j) \xrightarrow{P} \Delta_{\mathcal{S},\Gamma}(\alpha)(i, j). \tag{7.4}$$

Since k is fixed, the above further implies that $\hat{\Delta}_{\mathcal{S},\Gamma}(\alpha) \xrightarrow{P} \Delta_{\mathcal{S},\Gamma}(\alpha)$. Now, focus on the population analogue $\Delta_{\mathcal{S},\Gamma}(\alpha)$. It is easy to verify that this distance is a metric for any α. Thus, if a k-medoid algorithm is run on this distance matrix, the consistency in the sense of obtaining the true cluster centers is achieved by the arguments presented in [88] and [89]. An application of continuous mapping theorem in view of eq. (7.4) then ensures that as $n \to \infty$ and $T \to \infty$, the PAM algorithm recovers the true cluster centers $c_1, \ldots, c_k$ with probability 1.

For the final step of the proof, note that the proposed algorithm would assign any randomly chosen location $s \in \mathcal{S}$ to the cluster centered at c_j if and only if the empirical distance (computed according to eq. (3.8)) between s and c_j is less than the same between s and c_i for $i \neq j$. Equivalently, as $T \to \infty$, following Lemma 1, we can say that s is assigned to the cluster centered at c_j if and only if (by minor abuse of notation)

$$j = \underset{1 \leqslant r \leqslant k}{\operatorname{argmin}} \left\{ \left(\frac{\alpha}{\|\Delta_\Gamma\|_F} \right) \Delta_\Gamma(s, c_r) + \left(\frac{1-\alpha}{\|\Delta_\mathcal{S}\|_F} \right) \Delta_\mathcal{S}(s, c_r) \right\}. \tag{7.5}$$

Now, under Assumption 3, c_j corresponds to the true class of s if and only if $\Delta_\Gamma(s, c_j) = 0$. Further, under the given assumptions, both $\|\Delta_\Gamma\|_F$ and $\|\Delta_\mathcal{S}\|_F$ are of order $O_P(n)$. It can be shown using the assumption of the region to have bounded area, the properties of $\pi(\cdot)$, and applying the strong law of large numbers for dependent random variables (Proposition 1 from [90]). A direct implication of this result, in view of the fact that $\Delta_\Gamma(s, c_j) = 0$ and $\Delta_\Gamma(s, c_r) > 0$ for all $r \neq j$, is that there exists α_0 such that for all $\alpha \in (\alpha_0, 1]$,

$$\left(\frac{1-\alpha}{\|\Delta_\mathcal{S}\|_F} \right) \Delta_\mathcal{S}(s, c_j) < \min_{1 \leqslant r \leqslant k,\, r \neq j} \left\{ \left(\frac{\alpha}{\|\Delta_\Gamma\|_F} \right) \Delta_\Gamma(s, c_r) + \left(\frac{1-\alpha}{\|\Delta_\mathcal{S}\|_F} \right) \Delta_\mathcal{S}(s, c_r) \right\}. \tag{7.6}$$

Combining the above three steps of the proof, we can now conclude that as $n \to \infty, T \to \infty$, for $\alpha \in (\alpha_0, 1]$, the proposed clustering algorithm puts all locations in the correct cluster with probability approaching 1, and that completes the proof. □