Dependent Modeling of Temporal Sequences of Random Partitions

Garritt L. Page*
Department of Statistics, Brigham Young University
BCAM - Basque Center of Applied Mathematics, Bilbao, Spain,
Fernando A. Quintana
Departamento de Estadística, Pontificia Universidad Católica de Chile
Millennium Nucleus Center for the Discovery of Structures in Complex Data,
David B. Dahl
Department of Statistics, Brigham Young University

August 3, 2021

Abstract

We consider modeling a dependent sequence of random partitions. It is well-known in Bayesian nonparametrics that a random measure of discrete type induces a distribution over random partitions. The community has therefore assumed that the best approach to obtain a dependent sequence of random partitions is through modeling dependent random measures. We argue that this approach is problematic and show that the random partition model induced by dependent Bayesian nonparametric priors exhibits counter-intuitive dependence among partitions even though the dependence for the sequence of random probability measures is intuitive. Because of this, we suggest directly modeling the sequence of random partitions when clustering is of principal interest. To this end, we develop a class of dependent random partition models that explicitly models dependence in a sequence of partitions. We derive conditional and marginal properties of the joint partition model and devise computational strategies when employing the method in Bayesian modeling. In the case of temporal dependence, we demonstrate through simulation how the methodology produces partitions that evolve gently and naturally over time. We further illustrate the utility of the method by applying it to an environmental data set that exhibits spatio-temporal dependence.

Keywords: correlated partitions; hierarchical Bayes modeling; Bayesian nonparametrics; spatio-temporal clustering.

*The first author gratefully acknowledges support from the Basque Government through the BERC 2018-2021 program, by the Spanish Ministry of Science, Innovation and Universities through BCAM Severo Ochoa accreditation SEV-2017-0718. The second author is supported by the grant FONDECYT 1180034 and by ANID - Millennium Science Initiative Program - NCN17_059.

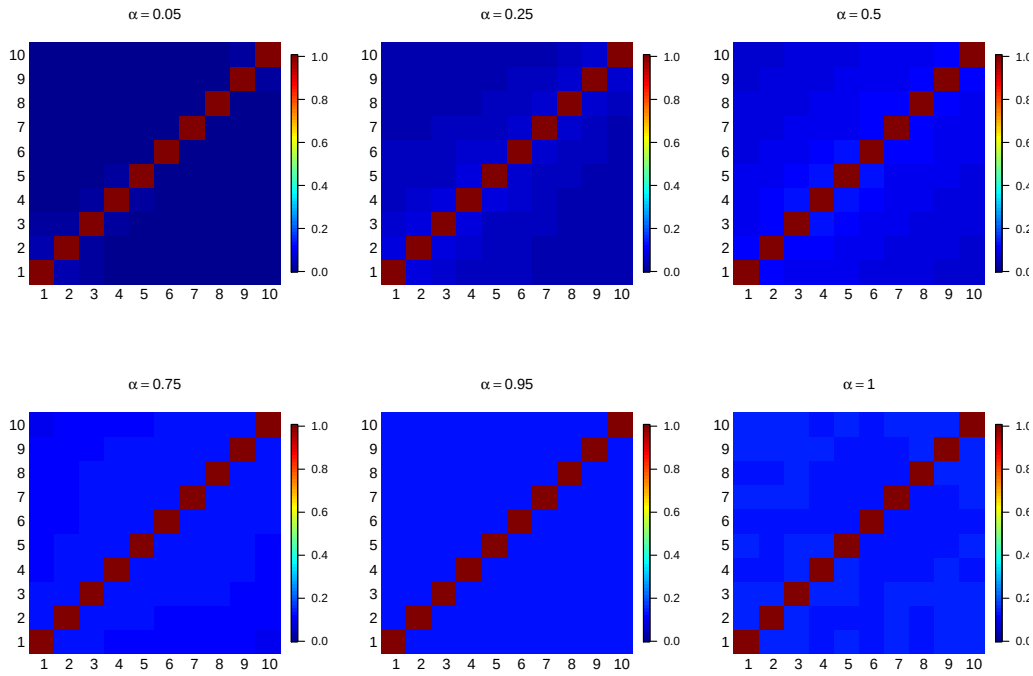

Figure 1: For various values of the temporal dependence parameter α, these plots show the lagged ARI values using the method of Caron et al. (2017) based on concentration parameter $M = 0.5$, discount parameter set to zero, and 10,000 Monte Carlo samples.

1 Introduction

We introduce a method to directly model dependence in a sequence of random partitions. Our approach is motivated by the practical problem of defining a prior distribution to model a sequence of random partitions that potentially exhibits substantial concordance (Übereinstimmung) over time (e.g., gently evolving clusterings over time). Traditionally, dependencies in random partitions (i.e., the clustering of units) have been obtained as a by-product of dependent random measures in Bayesian nonparametric (BNP) methods. We argue, however, that when a sequence of partitions *is* the inferential object of interest, then the sequence of partitions should be modeled *directly* rather than relying on *induced* random partition models, such as those implied by temporally dependent BNP models. But first, we review the literature on dependent BNP methods.

A non-exhaustive list of Bayesian nonparametric methods that temporally correlate a

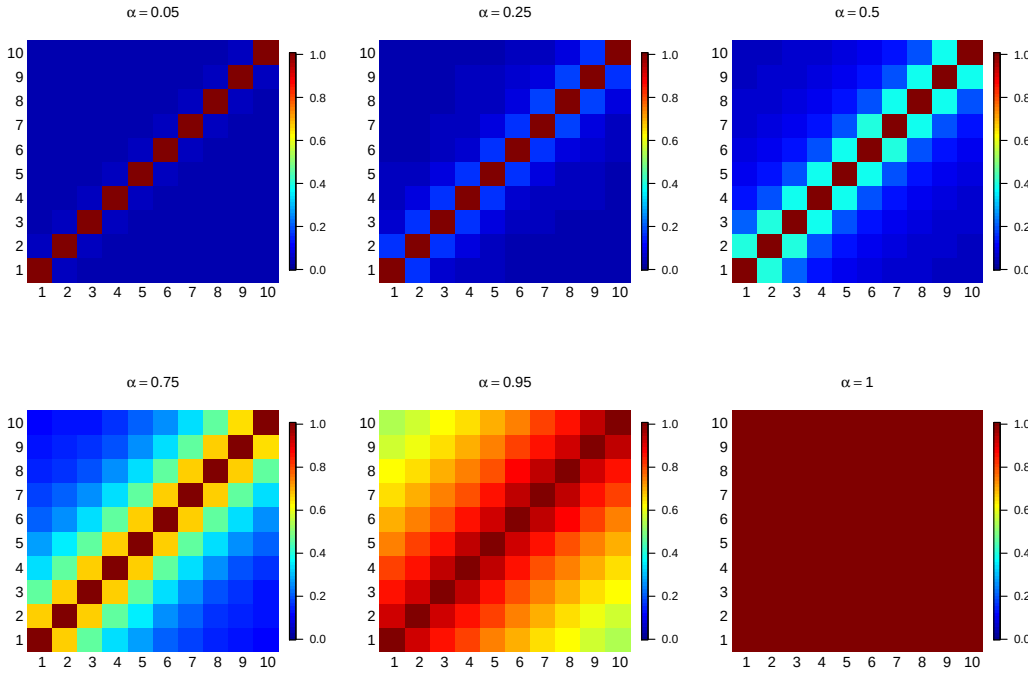

Figure 2: Lagged ARI values based on 10,000 Monte Carlo samples using the method developed in this paper. Partitions show natural and intuitive temporal dependence as lagged time increases and as the temporal dependence parameter α increases.

sequence of random probability measures include Nieto-Barajas et al. (2012), Antoniano-Villalobos and Walker (2016), Gutiérrez et al. (2016), Jo et al. (2017), Kalli and Griffin (2018), DeYoreo and Kottas (2018), and De Iorio et al. (2019). A common aspect of all these methods is that temporal dependence is accommodated in the sequence of random measures by way of the atoms or weights of the stick-breaking representation (Sethuraman, 1994). An alternative approach to producing a sequence of temporally correlated random probability measures can be found in Caron et al. (2007) and Caron et al. (2017). Their construction is based on a generalised Pólya urn scheme where dependencies between distributions that evolve over time are induced by urn-like operations on counts and the parameters to which they are associated. A key insight associated with all mentioned approaches, however, is that the induced random partitions only exhibit weak dependence even when a sequence of random probability measures is highly correlated. To illustrate this point, we conducted a small Monte Carlo simulation where an induced sequence of partitions was generated with

10 time points and 20 units using the method of Caron et al. (2017). To measure similarity of partitions at different time points, we use a time-lagged adjusted Rand index (ARI) (Hubert and Arabie 1985). Figure 1 shows these values averaged over 10,000 Monte Carlo samples. Notice that as the temporal dependence parameter (α) increases, the partitions from time period t to $t+1$ only become slightly more similar, such that the dependence between partitions is, at best, only weak. Further, the dependence is not temporally intuitive as it does not decay as a function of lagged time. In contrast, compare the dependence structure in Figure 1 to that in Figure 2, which contains average lagged ARI values between time lagged partitions generated using the method developed in this paper. Notice that unlike the induced random partitions generated using the method in Caron et al. (2017), the sequence of partitions generated using our approach displays intuitive temporal dependence. That is, as the time lag increases, similarity between partitions decreases and as the temporal dependence parameter α increases, the partitions become dissimilar over time at a slower rate.

The counter-intuitive behavior displayed in Figure 1 is not unique to the approach of Caron et al. (2017). As noted by Wade et al. (2014), the same type of behavior is present when using a linear dependent Dirichlet process mixture model. In fact, all BNP methods that model a sequence of discrete random probability measures will induce a random partition model with similarly weak correlation behavior. This behavior is analogous to trying to induce dependence among random variables from distributions with correlated parameters. There is no guarantee that correlated parameters would produce strong correlations among the random variables themselves.

Our approach is to consider the sequence of partitions as the parameter of principal interest and develop a method that models it directly. This will provide more control over how "smoothly" partitions evolve over time. Perhaps the work closest to ours, in the sense of explicitly modeling a sequence of partitions, can be found in Zanini et al. (2019). Their modeling approach for a temporally-referenced sequence of partitions can be applied to only two time points and differs from ours in that they do not focus on smooth evolution of partitions over time.

The remainder of the article is organized as follows. In Section 2 we present the proposed approach for a sequence of dependent random partitions, discuss its main properties, and suitable computational strategies for inference based on posterior simulation. Section 3 contains the results from three simulation studies that further explore aspects of the model. Section 4 describes an environmental data application and some concluding remarks are provided in Section 5. An accompanying Supplementary Materials file collects the proofs of results stated below, provides details on posterior simulation algorithms, and contains further simulation and data analysis results.

2 Joint Model for a Sequence of Partitions

Before detailing our method, we introduce some general notation. Let $i = 1, \ldots, m$ denote the m experimental units at time t for $t = 1, \ldots, T$. Let $\rho_t = \{S_{1t}, \ldots, S_{k_t t}\}$ denote a partition of the m experimental units at time $t = 1, \ldots, T$ into k_t clusters. An alternative notation is based on m cluster labels at time t denoted by $\boldsymbol{c}_t = \{c_{1t}, \ldots, c_{mt}\}$ where $c_{it} = j$ implies that $i \in S_{jt}$. Finally, any quantity with a "$\star$" superscript will be cluster-specific. For example, we will use $\mu^{\star}_{jt}$ to denote the mean of cluster j at time t so that $\mu_{it} = \mu^{\star}_{c_{it}t}$.

2.1 Temporal Modeling for Sequences of Partitions

Introducing temporal dependence in a collection of partitions requires formulating a joint probability model for $(\rho_1, \ldots, \rho_T)$. Generically, we will denote this joint model with $\Pr(\rho_t, \ldots, \rho_T)$. Temporal dependence among the ρ_t's implies that the cluster configuration in ρ_t could be impacted by that found in $\rho_{t-1}, \rho_{t-2}, \ldots, \rho_1$. However, we assume that the probability model for the sequence of partitions has a first-order Markovian structure. That is, the conditional distribution of ρ_t given $\rho_{t-1}, \rho_{t-2}, \ldots, \rho_1$ only depends on ρ_{t-1}. Thus, we construct $\Pr(\rho_t, \ldots, \rho_T)$ as

$$\Pr(\rho_1, \ldots, \rho_T) = \Pr(\rho_T \mid \rho_{T-1}) \Pr(\rho_{T-1} \mid \rho_{T-2}) \cdots \Pr(\rho_2 \mid \rho_1) \Pr(\rho_1). \tag{1}$$

Here $\Pr(\rho_1)$ is an exchangeable partition probability function (EPPF) that describes how the m experimental units at time period 1 are grouped into k_1 distinct groups with fre-

quencies $n_{11}, \ldots, n_{1k_1}$. One characteristic of an EPPF that will prove useful in what follows is sample size consistency, or what De Blasi et al. (2015) refer to as the *addition rule*. This property dictates that marginalizing the last of $m+1$ elements leads to the same model as if we only had m elements. A commonly encountered EPPF is that induced by a Dirichlet process (DP). This particular EPPF is sometimes referred to as a Chinese restaurant process (CRP) which can be seen as a special case from the family of product partition models (PPM). For more details, see De Blasi et al. (2015). Because we employ the EPPF of the CRP in what follows, we provide its form here

$$\Pr(\rho \mid M) = \frac{M^k}{\prod_{i=1}^{n}(M+i-1)} \prod_{i=1}^{k}(|S_i|-1)!, \tag{2}$$

where k is the number of clusters in ρ and M is a concentration parameter controlling the number of clusters. We will denote this random partition distribution as $CRP(M)$.

Although conceptually straightforward, (1) is silent regarding how ρ_{t-1} influences the form of ρ_t. To make this explicit, we introduce an auxiliary variable that guides the similarity between ρ_t and ρ_{t-1}. Note that if two partitions are highly dependent, then the cluster configurations between them will change very little and as a result only a few of the m experimental units will change cluster assignment. Conversely, two partitions that exhibit low dependence will likely be comprised of very different cluster configurations. The auxiliary variable we introduce identifies which of the experimental units at time $t-1$ will be considered for possible cluster reallocation at time t. Specifically, let γ_{it} denote the following

$$\gamma_{it} = \begin{cases} 1 & \text{if unit } i \text{ is } \textit{not} \text{ reallocated when moving from time } t-1 \text{ to } t \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

for $i = 1, \ldots, m$. Notice that when $\gamma_{it} = 0$, item i is subject to reallocation at time t, but still may, by random assignment, end up in the same cluster at time t as it was at time $t-1$. By construction, we set $\gamma_{i1} = 0$ for all i, i.e., all experimental units are allocated to clusters during the first time period. We then assume that $\gamma_{it} \overset{ind}{\sim} Ber(\alpha_t)$. Note that each of the $\alpha_t \in [0,1]$ acts as a temporal dependence parameter. Specifically, we will interpret $\alpha_t = 1$ as implying that $\rho_t = \rho_{t-1}$ with probability 1. Conversely, when $\alpha_t = 0$, then ρ_t is

independent of ρ_{t-1}. Further, when α_t is constant for all t, the degree of dependence among partitions is constant over time, whereas general values for α_t provide for varying degrees of dependence and more flexible partition patterns over time. For notational convenience, we introduce $\boldsymbol{\gamma}_t = (\gamma_{1t}, \gamma_{2t}, \ldots, \gamma_{mt})$ which is an m-tuple comprised of zeros and ones. The augmented joint model changes (1) to

$$\Pr(\boldsymbol{\gamma}_1, \rho_1, \ldots, \boldsymbol{\gamma}_T, \rho_T) = \Pr(\rho_T \mid \boldsymbol{\gamma}_T, \rho_{T-1})\Pr(\boldsymbol{\gamma}_T)\times \\ \Pr(\rho_{T-1} \mid \boldsymbol{\gamma}_{T-1}, \rho_{T-2})\Pr(\boldsymbol{\gamma}_{T-1}) \cdots \Pr(\rho_2 \mid \boldsymbol{\gamma}_2, \rho_1)\Pr(\boldsymbol{\gamma}_2)\Pr(\rho_1). \quad (4)$$

We describe $\Pr(\rho_t \mid \boldsymbol{\gamma}_t, \rho_{t-1})$ shortly, but first provide a definition.

Definition 1. *We say that partitions ρ_{t-1} and ρ_t are compatible with respect to $\boldsymbol{\gamma}_t$, if ρ_t may be obtained from ρ_{t-1} by reallocating items as indicated by $\boldsymbol{\gamma}_t$, i.e., those items i such that $\gamma_{ti} = 0$ for $i = 1, \ldots, m$. Note that the compatibility relation is an equivalence relation.*

There is a simple way to check if ρ_{t-1} is compatible with ρ_t with respect to $\boldsymbol{\gamma}_t$. Let $\mathfrak{R}_t = \{i : \gamma_{it} = 1\}$ be the collection of units that remain fixed when moving from time $t-1$ to time t, and $\mathfrak{R}_t^C = \{i : \gamma_{it} = 0\}$ is the collection of units that do not. Next denote with $\rho_t^{\mathfrak{R}_t}$ the "reduced" partition at time t that remains after removing all items in $\mathfrak{R}_t^C$ from the subsets of ρ_t. Similarly, let $\rho_{t-1}^{\mathfrak{R}_t}$ be the reduced partition at time $t-1$ based on $\boldsymbol{\gamma}_t$. Then ρ_{t-1} and ρ_t are compatible with respect to $\boldsymbol{\gamma}_t$ if and only if $\rho_{t-1}^{\mathfrak{R}_t} = \rho_t^{\mathfrak{R}_t}$.

Now, to further characterize $\Pr(\rho_t \mid \boldsymbol{\gamma}_t, \rho_{t-1})$, let P denote the set of all partitions of m units and let $P_{C_t} = \{\rho_t \in P : \rho_{t-1}^{\mathfrak{R}_t} = \rho_t^{\mathfrak{R}_t}\}$ be the collection of partitions at time t that are compatible with ρ_{t-1} based on $\boldsymbol{\gamma}_t$. Then, by construction, $\Pr(\rho_t \mid \boldsymbol{\gamma}_t, \rho_{t-1})$ is a random partition distribution whose support is P_{C_t} so that

$$\Pr(\rho_t = \lambda \mid \boldsymbol{\gamma}_t, \rho_{t-1}) = \frac{\Pr(\rho_t = \lambda)\mathrm{I}[\lambda \in P_{C_t}]}{\sum_{\lambda'} \Pr(\rho_t = \lambda')\mathrm{I}[\lambda' \in P_{C_t}]},$$

where $\Pr(\rho_t = \lambda)$ is the EPPF at the first time point evaluated at λ. Here, and in what follows, $\mathrm{I}[A]$ denotes an indicator function with $\mathrm{I}[A] = 1$ if statement A is true, and 0 otherwise.

It would be appealing if marginally each of the ρ_t follow the assumed EPPF for ρ_1, so that the joint probability model for partitions would be stationary. The following propo-

sition establishes this result, which is a consequence of the fact that conditioning on $\boldsymbol{\gamma}_t$ provides a "reduced" EPPF.

Proposition 1. *Let $\rho_1 \sim EPPF$ and $\boldsymbol{\gamma}_1 = \mathbf{0}$. If a joint model for $\rho_1 \ldots, \rho_T$ is constructed as described above by introducing $\boldsymbol{\gamma}_t$ for $t = 2, \ldots, T$, then we have that marginally $\rho_1, \ldots, \rho_T$ are identically distributed with law coming from the EPPF used to model ρ_1. Specifically, letting $\rho_{-t} = (\rho_1, \ldots, \rho_{t-1}, \rho_{t+1}, \ldots, \rho_T)$ and $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_T)$, we have that for all $\lambda \in P$,*

$$\Pr(\rho_t = \lambda) = \sum_{\rho_{-t} \in P^{\otimes m}} \sum_{\boldsymbol{\gamma} \in \Gamma^{\otimes m}} \Pr(\boldsymbol{\gamma}_1, \rho_1, \ldots, \rho_t = \lambda, \ldots, \boldsymbol{\gamma}_T, \rho_T) = \Pr(\rho_1 = \lambda),$$

where $P^{\otimes m} = P \times P \times \ldots \times P$, P a collection of all partitions of m units, $\Gamma^{\otimes m} = \Gamma \times \Gamma \times \ldots \times \Gamma$, and Γ a collection of all possible binary vectors of size m.

Proof. See supplementary material. □

In what follows, we will use $tRPM(\boldsymbol{\alpha}, M)$ to denote our temporal random partition model (4) parameterized by $\alpha_1, \ldots, \alpha_T$ and the EPPF in (2). We briefly mention that introducing γ_{it} is similar in spirit to the approach taken by Caron et al. (2007, 2017). However, they use $\boldsymbol{\gamma}_t$ to identify a partial partition at time t that informs how *all* the observational units will be reallocated at time $t + 1$. While this difference may seem inconsequential at first glance, it has drastic ramifications on the type of dependence that exists among the actual sequence of partitions. This is illustrated in Figures 1 and 2 provided in the Introduction. The sequence of partitions used to create Figure 2 were generated using the $tRPM(\boldsymbol{\alpha}, M)$ with $M = 0.5$. As mentioned in the Introduction, when the main emphasis is on modeling a "smoothly" evolving sequence of random partitions, the temporal dependence displayed in Figure 2 is much more natural than that found in Figure 1.

2.2 Dependence in Partitions

We now further explore how our method models dependence across partitions. To do this, we analyze closeness between partitions ρ_1 and ρ_2 by way of co-clustering of cluster labels

$(c_{11}, \ldots, c_{m1})$ and $(c_{12}, \ldots, c_{m2})$, respectively. We base our exploration on the Rand index which is defined a

$$R(\rho_1, \rho_2) = \frac{a+b}{\binom{m}{2}},$$

where a is the number of pairs (i, j) with $i, j \in [m] = \{1, \ldots, m\}$ that simultaneously co-cluster in ρ_1 and ρ_2 and b is the number of such pairs that simultaneously do not co-cluster. Writing $\varphi_{ij} = P(c_{i1} = c_{j1}, c_{i2} = c_{j2}) + P(c_{i1} \neq c_{j1}, c_{i2} \neq c_{j2})$, we note that

$$E\left[R(\rho_1, \rho_2)\right] = \binom{m}{2}^{-1} \sum_{1 \leq i < j \leq m} \varphi_{ij}.$$

To provide context to the co-clustering probabilities of cluster labels based on $tRPM(\alpha, M)$, we consider the model proposed in Caron et al. (2007). In their approach and assuming $\rho_1 \sim CRP(M)$, each $i \in [m]$ is randomly removed from the partition with probability $1-\alpha$, and ρ_2 is formed by running an extra $CRP(M)$ process, but starting from an urn that has weights given by the normalized cluster sizes left from the removal process. See details in Caron et al. (2007). We denote partitions that follow the model of Caron et al. (2007) as $\rho_1, \rho_2 \sim CAR(\alpha, M)$. For both our method and $CAR(\alpha, M)$, the case that $\alpha = 0$ leads to $\rho_1, \rho_2 \overset{iid}{\sim} CRP(M)$, while the largest degree of dependence between ρ_1 and ρ_2 is achieved when $\alpha = 1$. The following proposition characterizes the co-clustering probabilities under our method and $CAR(\alpha, M)$.

Proposition 2. *Let $m = T = 2$, so that $E\left[R(\rho_1, \rho_2)\right] = \varphi_{12}$.*

(a) If $\rho_1, \rho_2 \sim tRPM(\alpha, M)$, where to simplify notation we write $\alpha \equiv \alpha_2$, then

$$\varphi_{12} = \alpha^2 + \frac{(1+M^2)}{(1+M)^2}(1-\alpha)^2.$$

(b) If $\rho_1, \rho_2 \sim CAR(\alpha, M)$ then

$$\varphi_{12} = \left[\frac{6+3M+4M^2+M^3)}{(M+1)(M+2)(M+3))}\right]\alpha^2 + \frac{(1+M^2)}{(1+M)^2}\left(1-\alpha^2\right).$$

Proof. See supplementary material. □

An interesting consequence of Proposition 2 is that we can compute the expected value of the Rand index in the case $\rho_1, \rho_2 \overset{iid}{\sim} CRP(M)$.

Corollary 1. *If* $\rho_1, \rho_2 \overset{iid}{\sim} CRP(M)$ *then for any* $m \geq 2$,

$$E\left[R(\rho_1, \rho_2)\right] = \frac{(1+M^2)}{(1+M)^2}.$$

Proof. The result follows immediately by noting that the i.i.d. case coincides with $tRPM(0, M)$ and that by exchangeability and independence, $\varphi_{ij} = \varphi_{12}$ for all $1 \leq i < j \leq m$. □

The result from Proposition 2 (a) shows that, under the $tRPM(\alpha, M)$ model, $\lim_{\alpha \to 0^+} \varphi_{12} = \frac{(1+M^2)}{(1+M)^2}$, i.e., it agrees with $E\left[R(\rho_1, \rho_2)\right]$ under the i.i.d. case. The same holds as $\alpha \to 0^+$ under the $CAR(\alpha, M)$ model. Furthermore, for the $tRPM(\alpha, M)$, we get the appealing result that $\lim_{\alpha \to 1^+} \varphi_{12} = 1$, but the same limit under the $CAR(\alpha, M)$ is a number strictly less than 1 for any $M > 0$. This reveals that the closeness between partitions under the proposed $tRPM(\alpha, M)$, as measured by the φ_{12} quantity, can attain its maximum value of 1, which simply corresponds to the case where none of the units is relocated. The same cannot hold for the $CAR(\alpha, M)$ model because partitions are linked through a latent mechanism rather than directly as in the proposed model. Finally, we conjecture that similar results can be obtained for $m > 2$ but calculations become more involved. The result from Corollary 1 is nevertheless valid for any $m \geq 2$.

2.3 Toy Example to Illustrate Conditional Model

To build intuition regarding the transition from ρ_{t-1} to ρ_t, consider the conditional probabilities in equation (4) and the very simple scenario of $m = 3$ and $T = 2$. We have that

$$\Pr(\rho_2 \mid \rho_1) = \sum_{\boldsymbol{\gamma}_2 \in \Gamma} \Pr(\rho_2 \mid \boldsymbol{\gamma}_2, \rho_1) \Pr(\boldsymbol{\gamma}_2),$$

where again, Γ is the collection of all possible binary 3-tuples and operate under $\rho_1 \sim CRP(M)$. The conditional probabilities are provided in Table 1, where we set $M = 1$ for simplicity. From Table 1, notice that $\Pr(\rho_2 \mid \rho_1)$ is a reweighted CRP and that, as $\alpha \to 0$, partition probabilities correspond to those from the original CRP and, as $\alpha \to 1$, $\Pr(\rho_2 = \rho_1) \to 1$. Further, notice that partitions associated with ρ_2 that are more similar to ρ_1 are given larger weight relative to a CRP. For example, given $\rho_1 = \{\{1, 2, 3\}\}$ then

Table 1: Partition probabilities from the conditional distribution $\Pr(\rho_2 \mid \rho_1)$ using a CRP EPPF

(c_1, c_2, c_3)	$\Pr(\rho_1)$	$\Pr(\rho_2 \mid \rho_1 = a)$	$\Pr(\rho_2 \mid \rho_1 = b)$	$\Pr(\rho_2 \mid \rho_1 = c)$	$\Pr(\rho_2 \mid \rho_1 = d)$	$\Pr(\rho_2 \mid \rho_1 = e)$
$a = (1,1,1)$	$\frac{2}{6}$	$\frac{2}{6}[1 + 3\alpha^2 - \alpha^3]$	$\frac{2}{6}[1 - \alpha^2]$	$\frac{2}{6}[1 - \alpha^2]$	$\frac{2}{6}[1 - \alpha^2]$	$\frac{2}{6}[1 - 3\alpha^2 + 2\alpha^3]$
$b = (1,1,2)$	$\frac{1}{6}$	$\frac{1}{6}[1 - \alpha^2]$	$\frac{1}{6}[1 + 3\alpha^2 + 2\alpha^3]$	$\frac{1}{6}[1 - \alpha^2]$	$\frac{1}{6}[1 - \alpha^2]$	$\frac{1}{6}[1 + \alpha^2 - 2\alpha^3]$
$c = (1,2,1)$	$\frac{1}{6}$	$\frac{1}{6}[1 - \alpha^2]$	$\frac{1}{6}[1 - \alpha^2]$	$\frac{1}{6}[1 + 3\alpha^2 + 2\alpha^3]$	$\frac{1}{6}[1 - \alpha^2]$	$\frac{1}{6}[1 + \alpha^2 - 2\alpha^3]$
$d = (1,2,2)$	$\frac{1}{6}$	$\frac{1}{6}[1 - \alpha^2]$	$\frac{1}{6}[1 - \alpha^2]$	$\frac{1}{6}[1 - \alpha^2]$	$\frac{1}{6}[1 + 3\alpha^2 + 2\alpha^3]$	$\frac{1}{6}[1 + \alpha^2 - 2\alpha^3]$
$e = (1,2,3)$	$\frac{1}{6}$	$\frac{1}{6}[1 - 3\alpha^2 + 2\alpha^3]$	$\frac{1}{6}[1 + \alpha^2 - 2\alpha^3]$	$\frac{1}{6}[1 + \alpha^2 - 2\alpha^3]$	$\frac{1}{6}[1 + \alpha^2 - 2\alpha^3]$	$\frac{1}{6}[1 + 3\alpha^2 + 2\alpha^3]$

$\rho_2 = \{\{1,2\},\{3\}\}$ has higher probability than $\rho_2 = \{\{1\},\{2\},\{3\}\}$ for any $\alpha > 0$ but have equal probability in a CRP. From this toy example we see that the conditional co-clustering probabilities display dependencies in line with the desire to have partitions evolve gently over time.

2.4 Hierarchical Data Model

Once a partition model is specified, there is tremendous flexibility regarding how to model time (global or cluster-specific) at different levels of a hierarchical model (at the data level, process level, or both). Since we are interested to see how including time in the partition model impacts clustering and model fits, in the simulations of Section 3, we consider a hierarchical model where time only appears in the partition model. In particular, using cluster label notation, we will employ the following hierarchical model

$$
\begin{aligned}
Y_{it} \mid \boldsymbol{\mu}_t^\star, \boldsymbol{\sigma}_t^{2\star}, \boldsymbol{c}_t &\overset{ind}{\sim} N(\mu^\star_{c_{it}t}, \sigma^{2\star}_{c_{it}t}),\ i = 1, \ldots, m \text{ and } t = 1, \ldots, T, \\
(\mu^\star_{jt}, \sigma^\star_{jt}) \mid \theta_t, \tau_t^2 &\overset{ind}{\sim} N(\theta_t, \tau_t^2) \times UN(0, A_\sigma),\ j = 1, \ldots, k_t, \\
(\theta_t, \tau_t) &\overset{iid}{\sim} N(\phi_0, \lambda^2) \times UN(0, A_\tau),\ t = 1, \ldots, T, \\
(\phi_0, \lambda) &\sim N(m_0, s_0^2) \times UN(0, A_\lambda), \\
\{\boldsymbol{c}_t, \ldots, \boldsymbol{c}_T\} &\sim tRPM(\boldsymbol{\alpha}, M),\ \text{with } \alpha_t \overset{iid}{\sim} Beta(a_\alpha, b_\alpha),
\end{aligned}
\tag{5}
$$

where Y_{it} denotes the response measured on the ith unit at time t, UN denotes a uniform distribution and A_σ, A_τ, A_λ, m_0, s_0^2, a_α, b_α, M are user-supplied hyper-parameters. The remaining assumptions (e.g., independence across clusters and exchangeability within each cluster) are commonly employed.

2.5 Computation

As the posterior distribution implied by model (5) is not of a known form, we build an algorithm to sample from it. The construction of $\Pr(\rho_1, \ldots, \rho_T)$ naturally leads one to consider a Gibbs sampler. In the Gibbs sampler, $\boldsymbol{\gamma}_t$ will need to be updated in addition to ρ_t (by way of $\boldsymbol{c}_t$). But the Markovian assumption reduces some of the cost as we only need to consider ρ_{t-1} and ρ_{t+1} when updating ρ_t. Even though each update of ρ_t and $\boldsymbol{\gamma}_t$ for $t = 1, \ldots, T$ needs to be checked for compatibility, it is fairly straightforward to adapt standard algorithms, e.g. Algorithm 8 of Neal (2000), with care to make sure that only experimental units with $\gamma_{it} = 0$ are considered when updating c_{it}. Here we provide a general sketch for updating c_{it} and γ_{it} within an MCMC algorithm, with much more detail provided in Section **??** the online supplementary material

The MCMC algorithm we employ depends on deriving the complete conditionals for ρ_t and γ_t. A key result needed to derive them is provided in the following proposition.

Proposition 3. *Based on the construction of a joint probability model as described in Section 2.1, we have*

$$\Pr(\rho_t \mid \boldsymbol{\gamma}_t, \rho_{t-1}) = \begin{cases} \Pr(\rho_t)/\Pr(\rho_t^{\mathfrak{R}_t}) & \text{if } \rho_{t-1}^{\mathfrak{R}_t} = \rho_t^{\mathfrak{R}_t} \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

Proof. See the supplementary material. □

When updating γ_{it} in a Gibbs sampler, one can think of removing γ_{it} from $\boldsymbol{\gamma}_t$, and then reinsert it as either a 0 or 1. To this end let $\mathfrak{R}_t^{(-i)} = \mathfrak{R}_t \setminus \{i\}$ and $\mathfrak{R}_t^{(+i)} = \mathfrak{R}_t^{(-i)} \cup \{i\}$ and let $\boldsymbol{\gamma}_{t,+i}$ denote the $\boldsymbol{\gamma}_t$ vector with the ith entry set to 1. Then the full conditional for $\gamma_{it} = 1$, denoted by $\Pr(\gamma_{it} = 1 \mid -)$, is

$$\begin{aligned} \Pr(\gamma_{it} = 1 \mid -) &\propto \Pr(\rho_t \mid \boldsymbol{\gamma}_{t,+i}, \rho_{t-1}) \Pr(\boldsymbol{\gamma}_{t,+i}) \mathrm{I}[\rho_{t-1}^{\mathfrak{R}_t^{(+i)}} = \rho_t^{\mathfrak{R}_t^{(+i)}}], \\ &\propto \frac{\Pr(\rho_t)}{\Pr(\rho_t^{\mathfrak{R}_t^{(+i)}})} \alpha_t^{\gamma_{it}} \mathrm{I}[\rho_{t-1}^{\mathfrak{R}_t^{(+i)}} = \rho_t^{\mathfrak{R}_t^{(+i)}}], \end{aligned}$$

which results in

$$\Pr(\gamma_{it} = 1 \mid -) = \frac{\alpha_t \Pr(\rho_t^{\mathfrak{R}_t^{(-i)}})}{\alpha_t \Pr(\rho_t^{\mathfrak{R}_t^{(-i)}}) + (1 - \alpha_t) \Pr(\rho_t^{\mathfrak{R}_t^{(+i)}})} \mathrm{I}[\rho_{t-1}^{\mathfrak{R}_t^{(+i)}} = \rho_t^{\mathfrak{R}_t^{(+i)}}]. \tag{7}$$

For a given EPPF that has a closed form (e.g., CRP), it is straightforward to compute $\Pr(\rho_t^{\mathfrak{R}_t^{(-i)}})$ and $\Pr(\rho_t^{\mathfrak{R}_t^{(+i)}})$. If, however, the EPPF does not have a closed form, then note that (7) can be re-expressed as

$$\Pr(\gamma_{it} = 1 \mid -) = \frac{\alpha_t}{\alpha_t + (1-\alpha_t)\Pr(\rho_t^{\mathfrak{R}_t^{(+i)}})/\Pr(\rho_t^{\mathfrak{R}_t^{(-i)}})} \mathrm{I}[\rho_{t-1}^{\mathfrak{R}_t^{(+i)}} = \rho_t^{\mathfrak{R}_t^{(+i)}}]. \tag{8}$$

The quantity $\Pr(\rho_t^{\mathfrak{R}_t^{(+i)}})/\Pr(\rho_t^{\mathfrak{R}_t^{(-i)}})$ is a commonly encountered expression in MCMC methods that employ Neal's Algorithm 8 (Neal, 2000). Those same methods can be employed to calculate the desired probabilities. See Section **??** of the online supplementary material for more detail.

When updating c_{it} note that, within the MCMC algorithm, only those c_{it} for which $\gamma_{it} = 0$ are updated. Thus $\rho_{t-1}^{\mathfrak{R}_t} = \rho_t^{\mathfrak{R}_t}$ by construction. As a result, only compatibility between ρ_t and ρ_{t+1} (i.e., $\rho_t^{\mathfrak{R}_{t+1}} = \rho_{t+1}^{\mathfrak{R}_{t+1}}$) needs to be checked when updating c_{it}. Now letting $\Pr(c_{it} = h) = \Pr(c_{1t}, \ldots, c_{it} = h, \ldots, c_{mt})$ and denoting the partition based on $\{c_{1t}, \ldots, c_{it} = h, \ldots, c_{mt}\}$ as $\rho_{t:c_{it}=h} = \{S_{1t}^{-i}, \ldots, S_{ht}^{-i} \cup \{i\}, \ldots, S_{k_t^{-i}t}^{-i}\}$ where S_{jt}^{-i} denotes the jth cluster at time t with the ith unit removed (note it is possible that $S_{jt}^{-i} = S_{jt}$), the full conditional multinomial probability for c_{it} is

$$\Pr(c_{it} = h \mid -) \propto \begin{cases} N(Y_{it} \mid \mu^\star_{c_{it}=h,t}, \sigma^{2\star}_{c_{it}=h,t}) \Pr(c_{it} = h) \mathrm{I}[\rho_{t:c_{it}=h}^{\mathfrak{R}_{t+1}} = \rho_{t+1}^{\mathfrak{R}^{t+1}}] & \text{for } h = 1, \ldots, k_t^{-i}, \\ N(Y_{it} \mid \mu^\star_{new_h,t}, \sigma^{2\star}_{new_h,t}) \Pr(c_{it} = h) \mathrm{I}[\rho_{t:c_{it}=h}^{\mathfrak{R}_{t+1}} = \rho_{t+1}^{\mathfrak{R}^{t+1}}] & \text{for } h = k_t^{-i} + 1, \end{cases}$$

where $\mu^\star_{new_h,t}$ along with $\sigma^{2\star}_{new_h,t}$ are auxiliary parameters drawn from the prior as in Neal (2000)'s Algorithm 8 (with one auxiliary parameter) and k_t^{-i} is the number of clusters at time t when the ith unit has been removed. Further $N(\cdot \mid m, s^2)$ denotes a normal density with mean m and variance s^2. Given ρ_t and $\boldsymbol{\gamma}_t$, the full conditionals of the remaining parameters in model (5) follow standard techniques. A sample can be drawn from the posterior distribution implied by model (5) by iterating through the complete conditionals for $\boldsymbol{\gamma}_t$ and ρ_t and those of other model parameters. See Section **??** of the online supplementary material for more detail.

In our experience the MCMC algorithm described here and in Section **??** of the online supplementary material generally behaves well with regards to mixing and convergence. However, applications where $\alpha \approx 1$ can negatively affect the performance of the algorithm.

Having a parameter close to the boundary of its support commonly produces computational issues. It is possible to mitigate this by selecting a prior for α that keeps it from its boundary. Alternatively, a specialized algorithm will be needed to accommodate the boundary effect.

3 Simulation Studies

In this section we detail three simulation studies that illustrate different aspects of our modeling approach. In Section **??** of the online supplemental material, we provide additional simulation results and details regarding a fourth simulation study that considers the performance of our method when the response exhibits spatio-temporal dependence.

3.1 Simulation 1: Temporal Dependence in Estimated Partitions

The purpose of the first simulation is to study the accuracy of partition estimates (i.e., $\hat{\rho}_t$) and how much they change over time. (For a discussion of how $\hat{\rho}_t$ is obtained, see below.) In addition, we explore accuracy in estimating $\mu_{it} = \mu^{\star}_{c_{it}t}$ and α_t. To this end, we considered model (5) as a data generating mechanism to create one hundred datasets with fifty observations at five time points. We used $tRPM(\boldsymbol{\alpha}, M)$ with $\alpha_t = \alpha$ for all t and $M = 1$. We generate synthetic datasets under $\alpha \in \{0, 0.1, 0.25, 0.5, 0.75, 0.9, 0.999\}$. For all i and t, we set $\sigma^{2\star}_{c_{it}t} = \sigma^2 = 1$, $\tau^2 = 25$, and $\theta_t = 0$.

To each synthetic data set we fit model (5) using the MCMC algorithm detailed in Section 2.5 by collecting 10,000 iterates and discarding the first 5,000 as burn-in and thinning by 5 (resulting in 1,000 MCMC samples). As prior parameters we used $A_\sigma = 5$, $A_\tau = 10$, $A_\lambda = 10$, $m_0 = 0$, $S_0^2 = 100$, $a_\alpha = b_\alpha = M = 1$. For simplicity we set $\alpha_t = \alpha$ for all t. All partition point estimates were estimated using the method in the `salso` R package (Dahl et al. 2020) with the binder loss function (Binder 1978). To measure similarity between partitions, we employed the adjusted Rand index (Rand 1971; Hubert and Arabie 1985) and we used WAIC (Gelman et al. 2014) to measure model fit.

Table 2 displays the lagged 1 and 4 adjusted Rand index (ARI) as a function of α.

Table 2: Adjusted Rand index when comparing $\hat{\rho}_1$ to $\hat{\rho}_2$ and $\hat{\rho}_1$ to $\hat{\rho}_5$. Note that $ARI(\cdot, \cdot)$ denotes the adjusted Rand index as a function of two partitions. Coverage rates for α and μ_{it} and model fit metrics for $tRPM(\alpha, M)$ and $CRP(M)$. These values are averaged over the 100 generated data sets. The values in parenthesis are Monte Carlo standard errors. Note that smaller values of WAIC indicate better fit.

			Coverage		WAIC	
	$ARI(\hat{\rho}_1, \hat{\rho}_2)$	$ARI(\hat{\rho}_1, \hat{\rho}_5)$	α	μ_{it}	tRPM	CRP
$\alpha = 0.0$	0.05 (0.02)	0.01 (0.01)	0.00 (0.00)	0.93 (0.01)	896	892
$\alpha = 0.1$	0.04 (0.01)	0.00 (0.01)	0.96 (0.02)	0.92 (0.01)	910	907
$\alpha = 0.25$	0.19 (0.03)	0.02 (0.02)	0.90 (0.03)	0.91 (0.01)	890	891
$\alpha = 0.5$	0.44 (0.02)	0.04 (0.01)	0.82 (0.04)	0.91 (0.01)	881	903
$\alpha = 0.75$	0.67 (0.02)	0.23 (0.02)	0.89 (0.03)	0.92 (0.01)	822	893
$\alpha = 0.9$	0.87 (0.01)	0.55 (0.02)	0.90 (0.03)	0.90 (0.01)	816	896
$\alpha = 0.9999$	0.97 (0.01)	0.93 (0.01)	0.58 (0.05)	0.90 (0.01)	795	888

As expected, for both lags, the ARI increases as α increases. Also as expected, lagged 4 ARI increases less as a function of α compared to the lagged 1 ARI. Note that on average the lagged 1 ARI for $\alpha \in \{0.1, 0.25\}$ is smaller than that for $\alpha = 0$. This is because the variability associated with lagged 1 ARI when $\alpha = 0$ is much larger than when $\alpha > 0$, producing a few lagged ARI values that are large. The median of the lagged ARI values increase as a function of α monotonically.

To study the ability to recover μ_{it} and α, 95% credible intervals for each were computed and coverage was estimated. Results are provided in Table 2. Notice that coverage for α is low when the true α is at or near the boundary (e.g., $\alpha \in \{0, 0.9999\}$) which is to be expected. The coverage associated with μ_{it} is close to the nominal rate regardless of the value of α. Therefore, temporal dependence in the partition model does not adversely impact the ability to estimate individual means.

Lastly, to compare model fit when using $tRPM(\alpha, M)$ as the RPM in model (5) relative to $\rho_t \overset{iid}{\sim} CRP(M)$, we calculated the WAIC for each data set when fitting model (5) under both RPMs. Results are provided in Table 2 where each entry is an average WAIC value over all 100 datasets. Notice that, when the independent partitions were used to generate data (i.e., $\alpha = 0$), modeling partitions independently produces slightly better model fit as would be expected. But even if relatively weak temporal dependence exists among

the sequence of partitions, there are gains in modeling the sequence of partitions with $tRPM(\alpha, M)$, with gains becoming substantial as α increases.

The upshot from this simulation study is that lagged partition estimates when employing $tRPM(\alpha, M)$ display intuitive behavior in that similarity between partition estimates decreases as lag increases. In addition, employing the $tRPM(\alpha, M)$ partition model does not negatively impact parameter estimation and produces improved model fits when dependence is present in the sequence of partitions and a minimal cost in model fit when it is not.

3.2 Simulation 2: Induced Correlation at the Response Level

A potential benefit of developing a joint model for partitions is the ability to accommodate temporal dependence that may exist between Y_{it} and Y_{it+1}. To study this, we conducted a small Monte Carlo simulation study that is comprised of sampling repeatedly from the $tRPM(\alpha, M)$ using the computational approach of Section 2.5. Once the partition is generated, the temporal dependence among the $\boldsymbol{Y}_i$ depends on specific model choices for $\mu^\star_{jt}$. Here we use $\mu^\star_{jt} \sim N(\phi_1 \mu^\star_{jt-1}, \tau^2(1-\phi_1^2))$ for $t > 2$, $j = 1, \ldots, k_t$, and $|\phi_1| \leq 1$. For $t = 1$ we use $\mu^\star_{j1} \sim N(0, \tau^2)$ and if $k_{t+1} > k_t$ new $\mu^\star_{jt+1}$ values are drawn from $N(0, \tau^2)$. Now setting $m = 25$, $T = 10$, $\tau = 10$, and $\sigma = 1$, 100 datasets were generated for $\phi_1 \in \{0, 0.25, 0.5, 075, 0.9, 1\}$. For each data set generated, the lagged auto-correlations among $\boldsymbol{Y}_i$ were computed for $i = 1, \ldots, m$. The results found in Figure 3 are the lagged auto-correlations averaged over the m units for $\alpha \in \{0, 0.25, 0.5, 0.75, 0.9\}$.

As can be seen in Figure 3, when partitions are independent (i.e., $\alpha = 0$), no correlation propagates to the data level. The same can be said if atoms are *iid* (i.e., $\phi_1 = 0$). As the temporal dependence among $\mu^\star_{jt}$ increases (i.e., ϕ_1 increases), there is stronger temporal dependence among $Y_{i1}, \ldots, Y_{iT}$. Notice further that this dependence persists longer in time as α increases as one would expect.

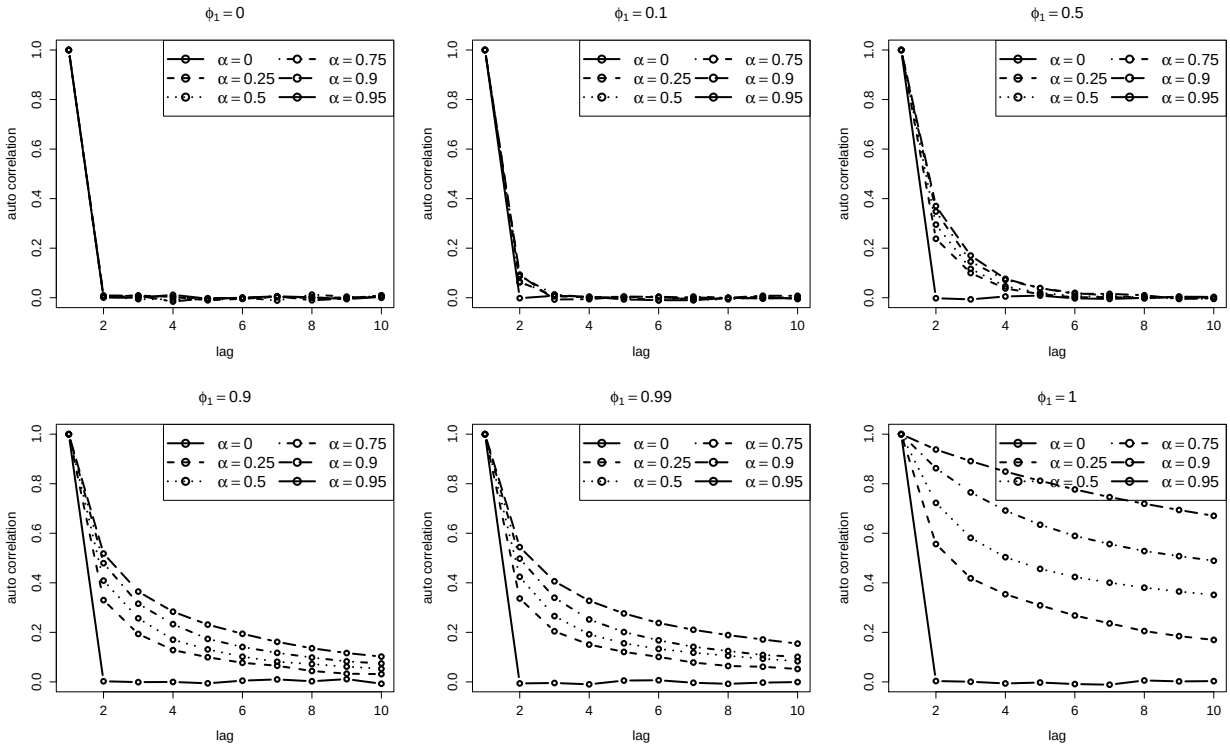

Figure 3: Lagged auto-correlations among the $(Y_{i1}, \ldots, Y_{iT})$ when modeling $\mu^{\star}_{jt}$ with an AR(1) type structure.

3.3 Simulation 3: AR(1)-type synthetic data

In our final simulation experiment, we consider data generated from an AR(1) process. To create synthetic datasets for the ith unit, we employ the following as a data generating mechanism

$$Y_{it} = \mu_{c_{it}} + \omega Y_{it-1} + \epsilon_{it}, \text{ for } i = 1, \ldots, m, \text{ and } t = 1, \ldots, T,$$

where $|\omega| < 1$ and $\epsilon_{it} \sim N(0, v^2)$. We consider synthetic datasets with four clusters so that $c_{it} \in \{1, 2, 3, 4\}$ corresponding to $\mu \in \{-2, 0, 2, 4\}$. The four clusters are formed by dividing the $m = 100$ units into equal groups of 25. At time points $t = 2, \ldots, T$, four units from each cluster are shifted to other clusters in a systematic way so that clusters change over time. An example of the type of data this procedures creates can be seen in Figure **??** of the supplementary material. Data are generated using the function `arima.sim` in `R` (R Core Team 2020) under $\omega \in \{0, 0.1, 0.25, 0.5, 0.75, 0.9\}$, $v^2 \in \{0.5^2, 1^2\}$, and $T \in \{5, 10\}$. A

total of 100 datasets are created under each scenario (totaling 28) and to each we fit the following competing models.

1. A weighted dependent Dirichlet process (wddp) described in Quintana et al. (2020) and chapter 4.4.4 of Müller et al. (2015). This model incorporates time in the weights of a DDP. As such, we fit this model to a concatenated version of the data (Y_i, t_i) for $i = 1, \ldots, Tm$. Specific details of this procedure are provided in Section **??** of the online supplementary material.

2. A linear dependent Dirichlet process (lddp) described in Quintana et al. (2020) and chapter 4.4.2 of Müller et al. (2015). This model incorporates time in the atoms of a DDP. As in the wddp, to this model we concatenated version of the data (Y_i, t_i) for $i = 1, \ldots, Tm$. Specific details of this procedure are also provided in Section **??** of the online supplementary material.

3. A Griffiths-Milne dependent Dirichlet process (gmddp) mixture. This model was fit using the `BNPmix` package in `R` (Corradin et al., 2020). See Corradin et al. (2021) for specific model details.

4. A temporally independent $CRP(M)$ model (ind_crp). This procedure corresponds to $\alpha = 0$ and is a special case of our approach and that proposed in Caron et al. (2007). The exact details of this model are also provided in Section **??** of the online supplementary material.

5. A temporally static $CRP(M)$ model (static_crp). This procedure corresponds to $\alpha = 1$ and is also a special case of the model detailed in Caron et al. (2007). Like the wddp and lddp, this procedure is fit to a concatenated version of the data (Y_i, for $i = 1, \ldots, Tm$. See Section **??** of the online supplementary material for more details.

Results from this simulation study are presented in Figure 4. The left plot in the figure displays the WAIC model fit metric for each procedure averaged over the 100 synthetic data sets, while the right plot displays the ARI value. The ARI values were produced by calculating ARI for each MCMC sample from the posterior distribution of ρ_t. This was

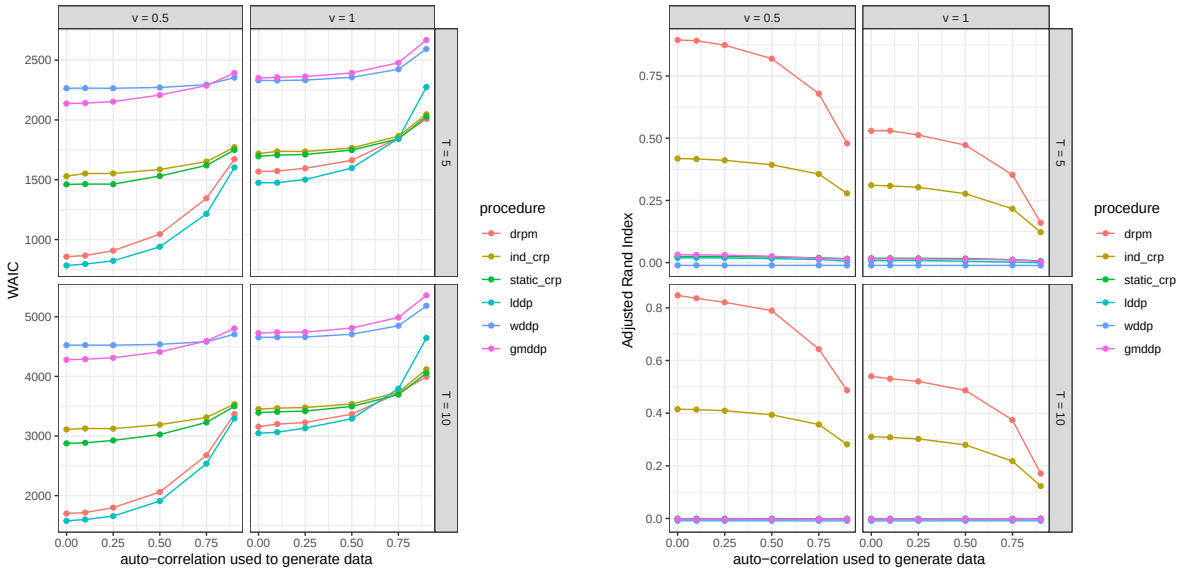

Figure 4: Results from simulation study with data that contains AR(1)-type temporal correlation. The left plot corresponds to results for the model fit metric WAIC and the right plot displays results associated with ARI.

done separately for each $t = 1, \ldots, T$ and then averaged across time and MCMC samples. From the left plot, our method (crpm) is superior to all other methods in terms of the model fit metric WAIC, save for the lddp method. Against the lddp method, the drpm produces better fits when the data are generated with high auto-correlation and larger data noise. This is to be expected as the drpm only incorporates temporal information in the prior on partitions while the lddp includes this information in the atoms (i.e., likelihood). That said, in terms of partition estimation, the drpm easily outperforms all other competitors in terms of ARI. The fact that drpm outperformed wddp and lddp in terms of ARI is not surprising as the later methods incorporate time differently compared to the drpm. However, the drpm and gmddp methods both treat time similarly, but the former from a partition perspective while the latter from a random probability measure perspective. This highlights the fact that when partitions are of interest, modeling them directly provides benefit.

4 Application

In this section we apply our method to a real-world data set coming from the field of environmental science. A second application in education is provided in Section **??** of the online supplementary material. As mentioned previously, once a partition model is specified, there is quite a bit of flexibility regarding how (or if) temporal dependence is incorporated in other parts of a hierarchical model. To illustrate this, we incorporate temporal dependence in three places of the hierarchical model we construct.

As part of preliminary exploratory data analysis (not shown), we examined serial dependence for each experimental unit (monitoring station), and concluded that they all exhibited a particular type of temporal dependence. Because of this, we introduce a unit-specific temporal dependence parameter $|\eta_{1i}| \leq 1$ and model observations from a single unit over time $(Y_{1i}, \ldots, Y_{iT})$ with an AR(1) structure. In addition, motivated by a desire for parsimony, we employed a Laplace prior for η_{1i}. Finally, to permit the temporal dependence in the partition model to propagate through the hierarchical model, we assume an AR(1) structure for the θ_t's. The full hierarchical model is detailed in (9).

$$
\begin{aligned}
Y_{it} \mid Y_{it-1}, \boldsymbol{\mu}_t^\star, \boldsymbol{\sigma}_t^{2\star}, \boldsymbol{\eta}, \boldsymbol{c}_t &\overset{ind}{\sim} N(\mu^\star_{c_{it}t} + \eta_{1i} Y_{it-1}, \sigma^{2\star}_{c_{it}t}(1-\eta_{1i}^2)), \\
Y_{i1} &\overset{ind}{\sim} N(\mu^\star_{c_{i1}1}, \sigma^{2\star}_{c_{i1}1}), \\
\xi_i = \text{Logit}(0.5(\eta_{1i}+1)) &\overset{iid}{\sim} Laplace(a,b), \\
(\mu^\star_{jt}, \sigma^\star_{jt}) &\overset{ind}{\sim} N(\theta_t, \tau_t^2) \times UN(0, A_\sigma), \\
\theta_t \mid \theta_{t-1} &\overset{ind}{\sim} N((1-\phi_1)\phi_0 + \phi_1\theta_{t-1}, \lambda^2(1-\phi_1^2)), \\
(\theta_1, \tau_t) &\sim N(\phi_0, \lambda^2) \times UN(0, A_\tau), \\
(\phi_0, \phi_1, \lambda) &\sim N(m_0, s_0^2) \times UN(-1,1) \times UN(0, A_\lambda), \\
\{\boldsymbol{c}_t, \ldots, \boldsymbol{c}_T\} &\sim tRPM(\boldsymbol{\alpha}, M), \text{ with } \alpha_t \overset{iid}{\sim} Beta(a_\alpha, b_\alpha),
\end{aligned}
\tag{9}
$$

where all Roman letters correspond to parameters that are user supplied. There are a number of special cases embedded in our hierarchical model. For example, $\eta_{i1} = 0$ for all i results in conditionally independent observations. Further, $\phi_1 = 0$ results in independent atoms and $\alpha_t = 0$ for all t in independent partitions over time. The model (5) used in the simulation studies is a special case of (9), obtained by setting $\phi_1 = 0$ and $\eta_{i1} = 0$ for all i.

4.1 Rural Background PM_{10} Data Application

The rural background PM_{10} data is taken from the European air quality database. These data are comprised of the daily measurements of particulate matter with a diameter less than 10 μm from rural background stations in Germany and are publicly available in the `gstat` package (Gräler et al. 2016) found on CRAN in `R` (R Core Team 2020). We focus on average monthly PM_{10} measures from the year 2005 (i.e., $T = 12$). Of the 69 stations, 9 were removed because of missing values.

We fit the hierarchical model (9) to these data and consider all the possible special cases (i.e., $\eta_{1i} = 0$ or not, $\phi_1 = 0$ or not, $\alpha_t = 0$ or not). This resulted in 8 total models that were fit by collecting 1,000 MCMC iterates after discarding the first 10,000 as burn-in and thinning by 40 (i.e., 50,000 total MCMC samples were collected). Running the algorithm for 50,000 samples on a laptop with 16GB of RAM took between 1 and 2.5 minutes. We use the following prior values: $A_\sigma = 10$, $A_\tau = A_\lambda = 5$, $m_0 = 0$, $s^2 = 100$, $a = 0$, $b = 1$, and $a_\alpha = b_\alpha = 2$. The prior for α_t was specified to encourage α from approaching 1. The WAIC and log pseudo marginal likelihood (LPML) for each model are presented in Table 3. To improve the computational stability of the LPML, for each model fit, the MCMC iterates that correspond to 0.5% of the smallest likelihood values were not included in the calculation of LPML.

First we note that among all the model fits, employing a variant of $tRPM(\boldsymbol{\alpha}, M)$ (i.e., rows with "Yes" in the "In Partition" column) improves model fit. The best performing model in terms of WAIC includes temporal dependence in the partition model only, while that for LPML includes temporal dependence in the partition model and in the likelihood.

Now focusing on partition inference, we provide Figure 5. This figure displays the lagged ARI values for each of the 8 models. Notice that when partitions are modeled independently (first or third rows of Figure 5) then partitions evolve over time quite erratically in the sense that the cluster configuration can change dramatically from one time point to the next. However, when employing $tRPM(\boldsymbol{\alpha}, M)$ (second row of Figure 5) the partitions seemed to evolve much more "smoothly" as there is less drastic changes in cluster configuration. In fact the model that produces the best model fit metrics (right most plot of second row)

Table 3: PM_{10} data: Results of model fitting. The bold font identifies best model fits in terms of LPML and WAIC. Higher values for LPML indicate better fit while lower values for WAIC indicate better fit.

Temporal Dependence In				
Likelihood (η_{1i})	Atoms (ϕ_1)	Partition (α_t)	LPML	WAIC
No	No	No	-1814	3683
No	No	Yes	-1656	**3031**
No	Yes	No	-1752	3539
No	Yes	Yes	-1644	3271
Yes	No	No	-1704	3554
Yes	No	Yes	**-1578**	3186
Yes	Yes	No	-1706	3544
Yes	Yes	Yes	-1586	3153

seems to produce partitions that change quite gently over time as desired.

Finally, we provide Figure 6 as a means to visualize how estimated partitions based on our joint partition model evolve over time relative to modeling partitions with an *iid* model. Each plot in Figure 6 displays the estimated partition at each time point. Each color represents a cluster and each number corresponds to a specific measuring station. The plots illustrate the sequential nature of cluster forming with the first cluster always containing the first measuring station, the next cluster is formed by the first station not included in the first cluster and so on. The plots in the right column correspond to using $tRPM(\boldsymbol{\alpha}, M)$ to jointly model partitions while those on the right employ $\rho_t \overset{iid}{\sim} CRP(M)$. It is evident from Figure 6 that from one time point to the next that partitions based on our construction evolve much more gently over time. This more closely mimics how PM_{10} measurements would evolve as a function of time compared to the *iid* model.

4.2 Extensions to the Joint Partition Model

Based on our generic joint model construction, it is straightforward to incorporate other information in the partition model such as covariates. For example, in the data application of 4.1 each monitoring station's spatial coordinates were recorded. Incorporating spatial dependence in our analysis of the PM_{10} data can be easily accommodated via the EPPF

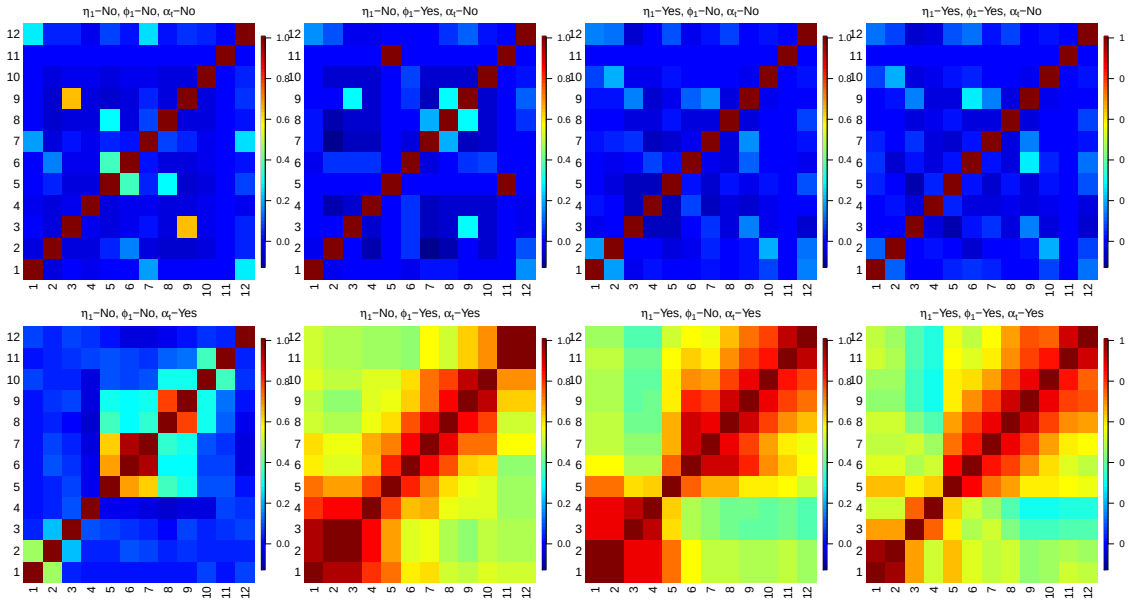

Figure 5: PM_{10} data. Each figure is a summary of the lagged ARI values corresponding to the 8 models in Table 3. At each time point the partition was estimated using the `salso` function in the `salso` R package (Dahl et al. 2020) based on binder loss.

in our construction. This would result in spatially informed clusters that evolve over time. If the spatially referenced EPPF preserves sample size consistency, then Proposition 1 still holds. One such EPPF is part of the spatial product partition model (sPPM) class developed in Page and Quintana (2016). To illustrate the ease of incorporating space in our model construction, here we model the PM_{10} data using model (9) but rather than use the $tRPM(\boldsymbol{\alpha}, M)$ to model the sequence of partitions, we use a version of our dependent partition model that employs the sPPM.

In order to introduce the sPPM, let $\boldsymbol{s}_i$ denote the spatial coordinates of the ith item (note that these coordinates do not change over time) and let $\boldsymbol{s}^{\star}_{jt}$ be the subset of spatial coordinates that belong to the jth cluster at time t. Then we express the EPPF of the tth partition with the following product form

$$\Pr(\rho_t \mid \nu_0, M) \propto \prod_{j=1}^{k_t} c(S_{jt} \mid M) g(\boldsymbol{s}^{\star}_{jt} \mid \nu_0). \tag{10}$$

Here $c(\cdot \mid M) \geq 0$ is called the cohesion and is a set function that produces cluster weights *a priori*. We consider the cohesion $c(S_{jt} \mid M) = M \times (|S_{jt}| - 1)!$ as it has connections with the CRP making this version of the sPPM a type of spatially re-weighted CRP. The similarity

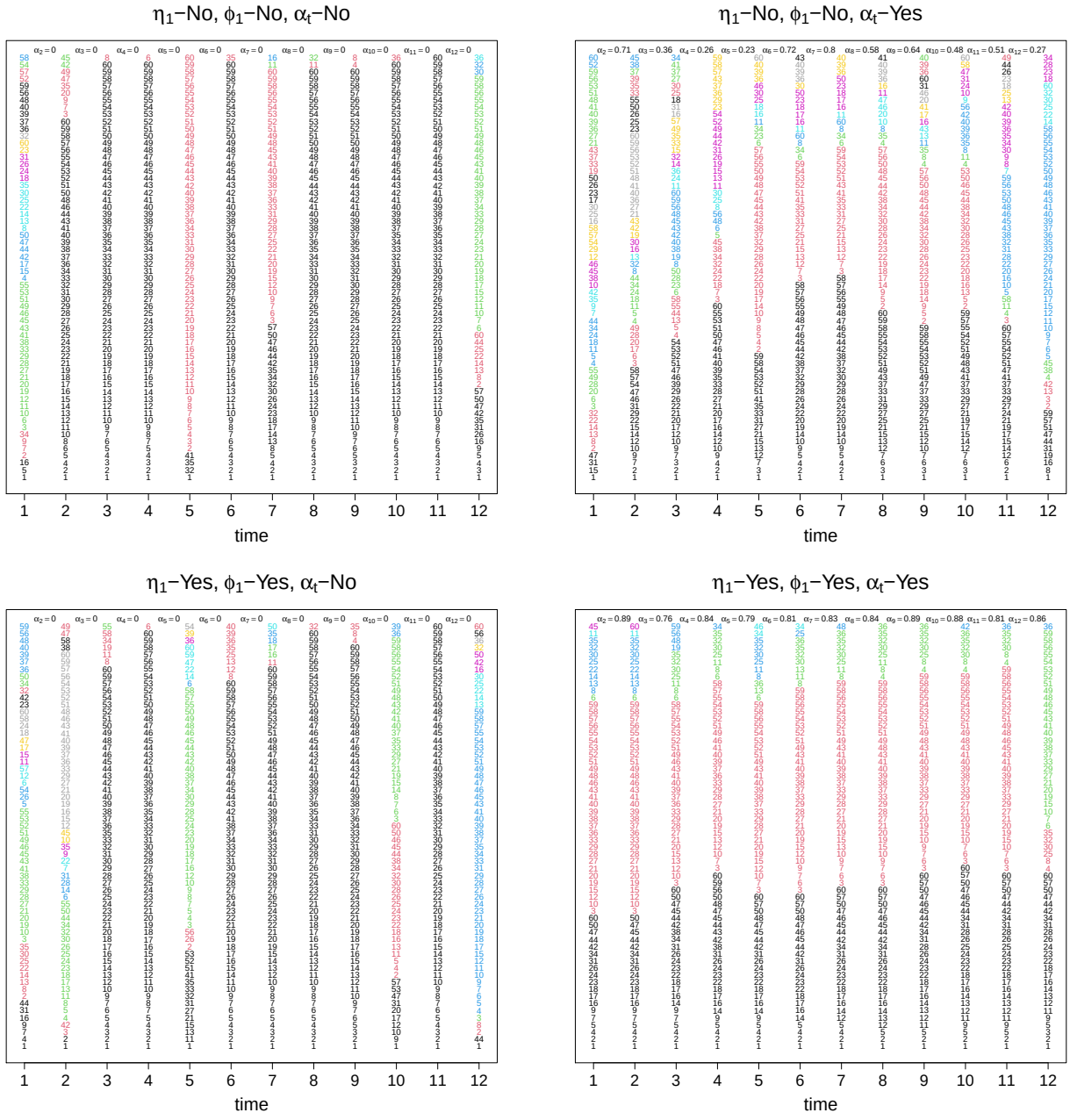

Figure 6: Each plot displays the estimated partition at each time point. Plots in the left column are based on $\rho_t \overset{iid}{\sim} CRP(M)$ while those in the right column are based on $(\rho_1, \ldots, \rho_T) \sim tRPM(\boldsymbol{\alpha}, M)$. Clusters are highlighted by color and each number corresponds to a specific monitoring station.

function $g(\cdot \mid \nu_0)$ is a set function parametrized by ν_0 that measures the compactness of the spatial coordinates in $\boldsymbol{s}^{\star}_{jt}$ producing higher values if the spatial coordinates in $\boldsymbol{s}^{\star}_{jt}$ are close to each other. Not all similarity functions preserve sample size consistency so to ensure this, after standardizing spatial locations, we employ

$$g(\boldsymbol{s}^{\star}_{jt} \mid \nu_0) = \int \prod_{i \in S_{jt}} N(\boldsymbol{s}_i \mid \boldsymbol{m}, \boldsymbol{V}) NIW(\boldsymbol{m}, \boldsymbol{V} \mid \boldsymbol{0}, 1, \nu_0, \boldsymbol{I})\, d\boldsymbol{m}\, d\boldsymbol{V}, \tag{11}$$

where $N(\cdot \mid \boldsymbol{m}, \boldsymbol{V})$ denotes a bivariate normal density and $NIW(\cdot, \cdot \mid \boldsymbol{0}, 1, \nu_0, \boldsymbol{I})$ a normal-inverse-Wishart density with mean $\boldsymbol{0}$, scale equal to 1, inverse scale matrix equal to $\boldsymbol{I}$, and ν_0 being the user-supplied degrees of freedom. Note that larger values of ν_0 increase spatial influence on partition probabilities. For more details on why this formulation preserves sample size consistency, see Müller et al. (2011) and Quintana et al. (2018). For more information regarding the impact of ν_0 on product form of the partition model, see Page and Quintana (2016, 2018). We will use $stRPM(\boldsymbol{\alpha}, \nu_0, M)$ to denote our spatio-temporal random partition model (4) parameterized by $\alpha_1, \ldots, \alpha_T$ and EPPF detailed in (10) and (11).

As in the previous section, we fit model (9) to the PM_{10} data but replacing $tRPM(\boldsymbol{\alpha}, M)$ with the $stRPM(\boldsymbol{\alpha}, \nu_0, M)$. Also as before, we consider all the possible special cases of the model (i.e., $\eta_{1i} = 0$ or not, $\phi_1 = 0$ or not, $\alpha_t = 0$ or not). This resulted in 8 total models that were fit by collecting 1,000 MCMC iterates after discarding the first 10,000 as burn-in and thinning by 40. Fitting the eight models based on the $stRPM(\boldsymbol{\alpha}, \nu_0, M)$ took between 20 and 57 minutes. The prior values employed were the same as before with the addition of setting $\nu_0 = 5$ which places in the partition prior moderate weight on spatial locations.

Incorporating space in the partition model seems to provide benefit in terms of model fit as the LPML and WAIC values associated with the model that includes space in the partition model and temporal dependence in all levels of the model fits best in terms of LPML with -1487 compared to -1586 listed in Table 3 and temporal dependence in partition and atoms fits best with regards to WAIC with 3140 compared to 3271 listed in Table 3. Additionally, Figure 7 displays the lagged ARI values for the 8 models that include space. As in Figure 5 when there is no temporal dependence in the partition model, then the estimated partitions exhibit no temporal dependence. However, when time and space are

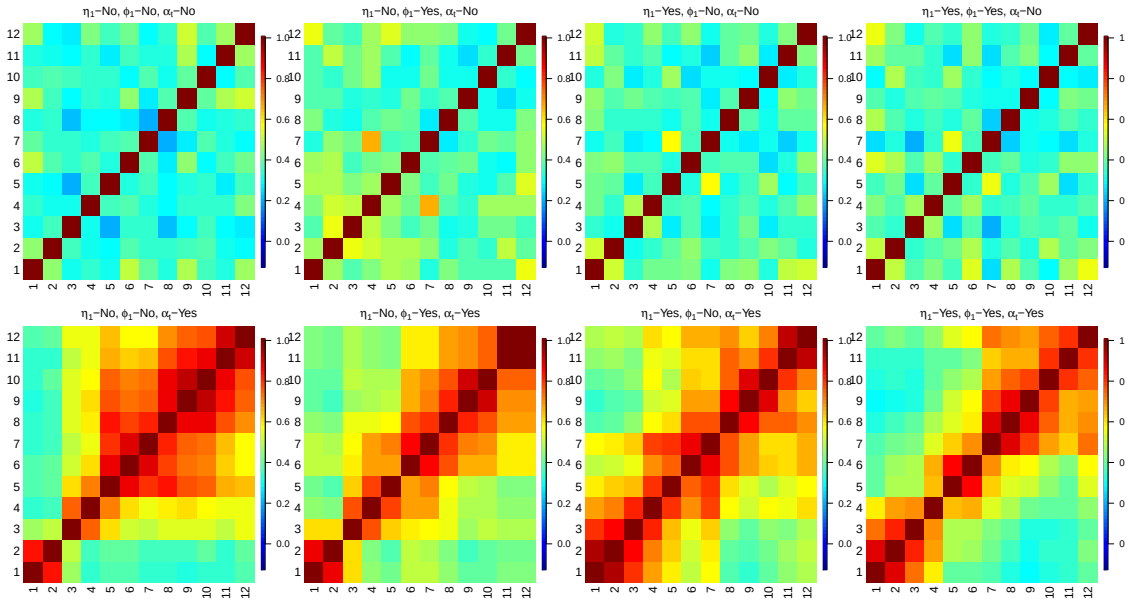

Figure 7: PM_{10} data. Each figure is a summary of the lagged ARI values corresponding to the 8 models that are detailed in Table 3 except that space is also included in the dependent random partition model. At each time point the partition was estimated using the `salso` function in the `salso` R package (Dahl et al. 2020) based on binder loss.

included then there is clear dependence between partitions as a function of time. Further, the dependence between partitions seems to decay faster with space and time included in the partition model compared to just time.

Finally, we provide Figure 8 which displays the estimated spatially referenced partitions at each time point based on the model that achieved the best fit (space in the partition model and temporal dependence in all levels of the model). The size of each point in the figure is proportional to the PM_{10} measured at the particular station and each depicts a cluster. To make connections with Figure 6 each monitoring station is labeled with the same number as before. Notice that there are clear similarities from one time point to the next for most months. That said, there are two time periods for which changes in the PM_{10} are more drastic relative to the previous time period (e.g., August to September). In these months the estimated α_t is quite a bit smaller and as a result, the estimated partitions are more different.

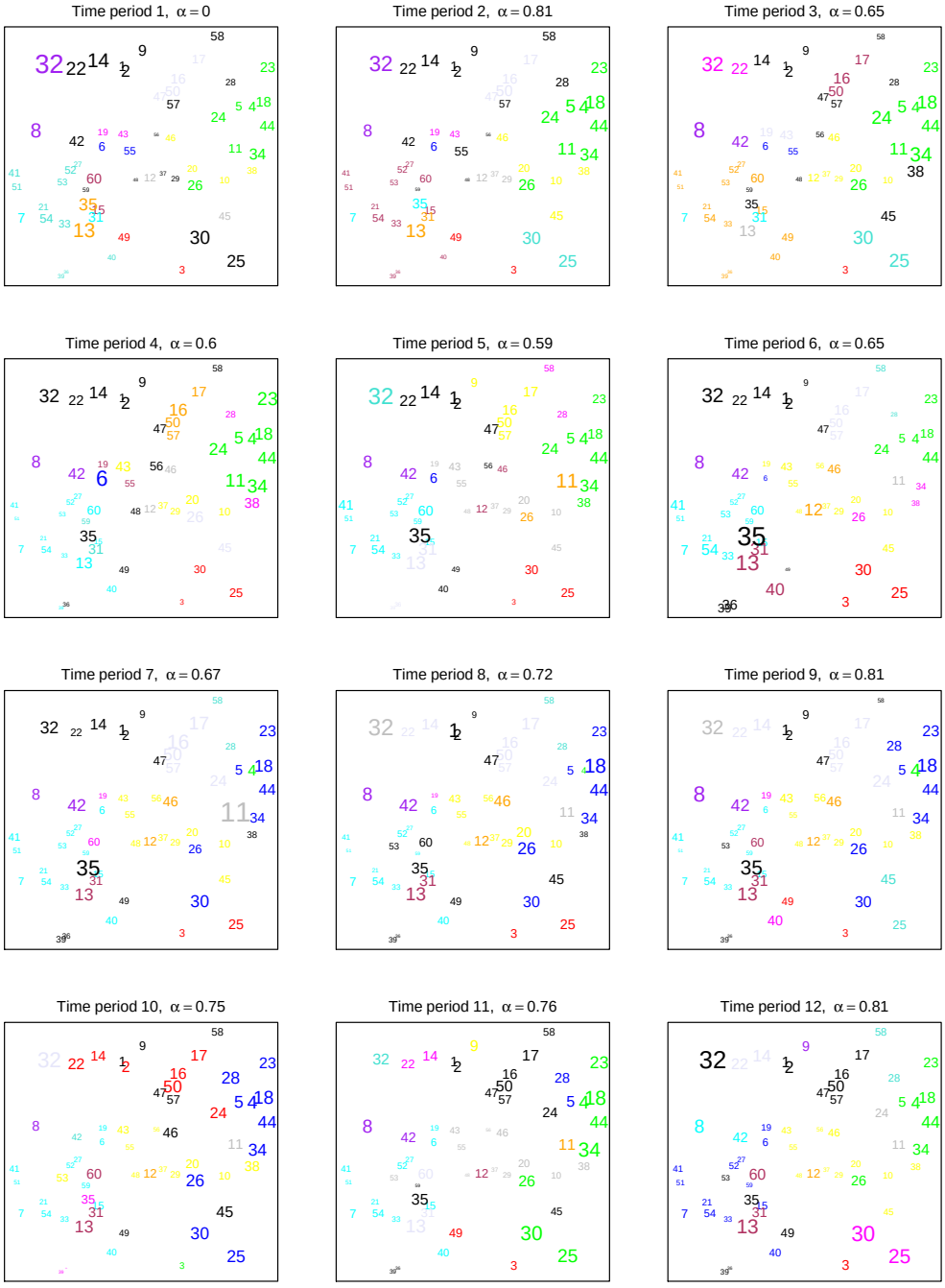

Figure 8: PM_{10} data. Graphical display for spatially-referenced estimated partitions for each time point based on the model that produced best fit (space in the partition model and temporal dependence in all levels of the model). The size of a point is proportional to the PM_{10} measured at that station. Clusters are identified by color. Each monitoring station is labeled by the same numbers as in Figure 6. At each time point the partition was estimated using the `salso` function in the `salso` R package (Dahl et al. 2020) based on binder loss.

5 Conclusions

We developed a joint probability model for a sequence of partitions that explicitly considers temporal dependence among the partitions. Further, we showed that our methodology is capable of accommodating partitions that evolve slowly over time in that the adjusted Rand index between estimated partitions decays as the lag in time increases. We also showed that if partitions are indeed independent over time, then employing our joint partition prior regardless results in a minimal cost in terms of model fit.

The predictive nature of the temporal prior on a sequence of random partitions we have presented has a first-order Markovian structure. Various extensions can be considered, such as adding higher order dependence across time or dependence in baseline covariates. All of these cases would build on our constructive definition, as extra refinements of the basic idea of carrying smooth transitions on time (or time and space). Lastly, the Markovian structure could be exploited to carry out predictive inference as well.

SUPPLEMENTARY MATERIAL

Supplementary Material: Online supplementary material file that contains proofs of propositions, computation details, additional simulation, and application results.

R-package: The R-package `drpm` contains the function `drpm` that fits all models described in the article.

References

Antoniano-Villalobos, I. and Walker, S. G. (2016), "A nonparametric model for stationary time series," *J. Time Series Anal.*, 37, 126–142.

Binder, D. A. (1978), "Bayesian Cluster Analysis," *Biometrika*, 65, 31–38.

Caron, F., Davy, M., and Doucet, A. (2007), "Generalized Polya Urn for Time-varying Dirichlet Process Mixtures," in *Proceedings of the Twenty-Third Conference on Uncer-*

tainty in Artificial Intelligence, UAI'07, Arlington, Virginia, United States: AUAI Press, URL http://dl.acm.org/citation.cfm?id=3020488.3020493.

Caron, F., Neiswanger, W., Wood, F., Doucet, A., and Davy, M. (2017), "Generalized Pólya Urn for Time-Varying Pitman-Yor Processes," *Journal of Machine Learning Research*, 18, 1–32, URL http://jmlr.org/papers/v18/10-231.html.

Corradin, R., Canale, A., and Nipoti, B. (2020), *BNPmix: Bayesian Nonparametric Mixture Models*, URL https://CRAN.R-project.org/package=BNPmix. R package version 0.2.6.

— (2021), "BNPmix: an R package for Bayesian nonparametric modelling via Pitman-Yor mixtures," *Journal of Statistical Software*, to appear.

Dahl, D. B., Johnson, D. J., and Müller, P. (2020), *salso: Search Algorithms and Loss Functions for Bayesian Clustering*, URL https://CRAN.R-project.org/package=salso. R package version 0.2.5.

De Blasi, P., Favaro, S., Lijoi, A., Mena, R. H., Prünster, I., and Ruggiero, M. (2015), "Are Gibbs-Type Priors the Most Natural Generalization of the Dirichlet Process?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37, 212–229.

De Iorio, M., Favaro, S., Guglielmi, A., and Ye, L. (2019), "Bayesian nonparametric temporal dynamic clustering via autoregressive Dirichlet priors," ArXiv:1910.10443.

DeYoreo, M. and Kottas, A. (2018), "Modeling for dynamic ordinal regression relationships: an application to estimating maturity of rockfish in California," *Journal of the American Statistical Association*, 113, 68–80, URL https://doi.org/10.1080/01621459.2017.1328357.

Gelman, A., Hwang, J., and Vehtari, A. (2014), "Understanding predictive information criteria for Bayesian models," *Statistics and Computing*, 24, 997–1016.

Gräler, B., Pebesma, E., and Heuvelink, G. (2016), "Spatio-Temporal Interpolation using gstat," *The R Journal*, 8, 204–218, URL https://journal.r-project.org/archive/2016-1/na-pebesma-heuvelink.pdf.

Gutiérrez, L., Mena, R. H., and Ruggiero, M. (2016), "A time dependent Bayesian nonparametric model for air quality analysis," *Computational Statistics & Data Analysis*, 95, 161 – 175.

Hubert, L. and Arabie, P. (1985), "Comparing Partitions," *Journal of Classification*, 2, 193–218.

Jo, S., Lee, J., Müller, P., Quintana, F. A., and Trippa, L. (2017), "Dependent Species Sampling Models for Spatial Density Estimation," *Bayesian Analysis*, 12, 379–406, URL https://doi.org/10.1214/16-BA1006.

Kalli, M. and Griffin, J. E. (2018), "Bayesian nonparametric vector autoregressive models," *Journal of Econometrics*, 203, 267–282.

Müller, P., Quintana, F., and Rosner, G. L. (2011), "A Product Partition Model With Regression on Covariates," *Journal of Computational and Graphical Statistics*, 20, 260–277.

Müller, P., Quintana, F. A., Jara, A., and Hanson, T. (Editors) (2015), *Bayesian Nonparametric Data Analysis*, Switzerland: Springer International Publishing, 1 edition.

Neal, R. M. (2000), "Markov Chain Sampling Methods for Dirichlet Process Mixture Models," *Journal of Computational and Graphical Statistics*, 9, 249–265.

Nieto-Barajas, L. E., Müller, P., Ji, Y., Lu, Y., and Mills, G. B. (2012), "A Time-Series DDP for Functional Proteomics Profiles," *Biometrics*, 68, 859–868.

Page, G. L. and Quintana, F. A. (2016), "Spatial Product Partition Models," *Bayesian Analysis*, 11, 265–298.

— (2018), "Calibrating Covariate Informed Product Partition Models," *Statistics and Computing*, 28, 1009–1031, URL https://doi.org/10.1007/s11222-017-9777-z.

Quintana, F. A., Loschi, R. H., and Page, G. L. (2018), *Bayesian Product Partition Models*, Wiley StatsRef: Statistics Reference Online, 1–15, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118445112.stat08123.

Quintana, F. A., Müller, P., Jara, A., and MacEachern, S. N. (2020), "The Dependent Dirichlet Process and Related Models," ArXiv:2007.06129v1.

R Core Team (2020), *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, URL https://www.R-project.org/.

Rand, W. M. (1971), "Objective Criteria for the Evaluation of Clustering Methods," *Journal of the American Statistical Association*, 66, 846–850.

Sethuraman, J. (1994), "A constructive definition of Dirichlet priors," *Statistica Sinica*, 4, 639–650.

Wade, S., Walker, S. G., and Petrone, S. (2014), "A Predictive Study of Dirichlet Process Mixture Models for Curve Fitting," *Scandinavian Journal of Statistics*, 41, 580–605.

Zanini, C. T. P., Müller, P., Ji, Y., and Quintana, F. A. (2019), "A Bayesian Random Partition Model for Sequential Refinement and Coagulation," *Biometrics*, 75, 988–999.

Supplementary Material: Dependent Modeling of Temporal Sequences of Random Partitions

Garritt L. Page
Brigham Young University, Provo, USA
BCAM - Basque Center of Applied Mathematics, Bilbao, Spain
and
Fernando A. Quintana *
Pontificia Universidad Católica de Chile, Santiago, Chile
and
David B. Dahl
Brigham Young University, Provo, USA.

August 3, 2021

This document contains supplementary material to the paper "Dependent Modeling of Temporal Sequences of Random Partitions".

A Proofs of Propositions

In this section of the supplementary material we provide proofs of the two propositions described in the main article

A.1 Proof of Proposition 1

Proof. For clarity, here we introduce notation that highlights the dependence of partitions on sample size. For example, $\rho_{t,m} = (S_{1,t}, \ldots, S_{k_t(m),t})$ and $[m] = \{1, \ldots, m\}$. By

*Partially supported by grant FONDECYT 1180034 and by Iniciativa Científica Milenio - Minecon Núcleo Milenio MiDaS

assumption $\Pr(\rho_{1,m})$ is specified by means of an EPPF which we now construct. Denote $\mathbb{N}^* = \cup_{k=0}^{\infty}\mathbb{N}^k$, and identify any $\boldsymbol{n} = (n_1, \ldots, n_k) \in \mathbb{N}^*$ with the infinite sequence $(n_1, \ldots, n_k, 0, 0, \ldots)$. Given $\boldsymbol{n} \in \mathbb{N}^*$, let $k(\boldsymbol{n})$ denote the number of non-zero entries in $\boldsymbol{n}$ and denote by $\boldsymbol{n}^{j+}$ the result of incrementing $\boldsymbol{n}$'s jth component (i.e., n_j) by 1, with $1 \leq j \leq k(\boldsymbol{n}) + 1$. An EPPF is then any function $r : \mathbb{N}^* \longrightarrow [0, 1]$ that is symmetric in its arguments and where

$$r(1) = 1 \qquad \text{and} \qquad r(\boldsymbol{n}) = \sum_{j=1}^{k(\boldsymbol{n})+1} r(\boldsymbol{n}^{j+}) \qquad \text{for all } \boldsymbol{n} \in \mathbb{N}^*. \tag{S.1}$$

Condition (S.1) implies that a EPPF is sample size consistent, i.e., marginalizing the $(n+1)$st element leads to the model for n elements. The EPPF also implies exchangeability of configurations in the sense that a EPPF is invariant under permutations of the elements that keep the cluster sizes unaltered. We also note that any valid EPPF defines a predictive rule of the form

$$r_j(\boldsymbol{n}) = \frac{r(\boldsymbol{n}^{j+})}{r(\boldsymbol{n})}, \qquad \text{for } 1 \leq j \leq k(\boldsymbol{n}) + 1, \tag{S.2}$$

where it is assumed that $r(\boldsymbol{n}) > 0$ and $r_j(\boldsymbol{n})$ represents the probability of a new element joining the jth already existing cluster, for $1 \leq j \leq k(\boldsymbol{n})$, or starting a new one (the $k(\boldsymbol{n}) + 1$). The one-step rule (S.2) can also be extended to predictions of two or more elements by simply iterating the one-step rule as many times as needed. Now, given an EPPF r, we have that

$$\Pr(\rho_{1,m} = (S_{1,1}, \ldots, S_{k_1(m),1})) = r(n_{1,1}, \ldots, n_{k_1(m),1}). \tag{S.3}$$

To prove the result, it suffices to show that it holds for $\rho_{2,m}$ and then by induction the result holds generally. Denote by $[\Gamma] = \{i \in \{1, \ldots, m\} : \gamma_{i2} = 0\}$ the (random) set of elements removed from $\rho_{1,m}$. Then, $\rho_{1,m}^{-N_{02}}$ is a partition of the elements of $[m] - [\Gamma]$ (where as before $N_{02} = \sum_{j=1}^{m} I[\gamma_{j2} = 0]$). By exchangeability and the fact that an EPPF is sample

size consistent, we have that for any partition $S_1^-, \ldots, S_{k([m]-[\Gamma])}^-$ of $[m] - [\Gamma]$:

$$\begin{aligned}\Pr(\rho_{2,m}^{-N_{02}} = (S_1^-, \ldots, S_{k([m]-[\Gamma])}^-) \mid [\Gamma]) &= \Pr(\rho_{1,m}^{-N_{02}} = (S_1^-, \ldots, S_{k([m]-[\Gamma])}^-) \mid [\Gamma]) \\ &= r(|S_1^-|, \ldots, |S_{k([m]-[\Gamma])}^-|),\end{aligned}$$

where $|S_j|$ is the number of elements in S_j. In addition, and again by exchangeability and sample size consistency, the predictive rule starting from $[m] - [\Gamma]$ (or from any subset of $[m]$ for that matter) depends only on the sizes of the subsets in that partition. Thus, conditioning on all reallocation configurations and initial partition after subject removal we have:

$$\begin{aligned}\Pr(\rho_{2,m} = (S_1, \ldots, S_k)) &= \sum_{[\Gamma]} \sum_{\rho_{2,m}^{-N_{02}}} \Pr(\rho_{2,m} = (S_1, \ldots, S_k) \mid [\Gamma],\, \rho_{2,m}^{-N_{02}}) \times \\ &\qquad \Pr(\rho_{2,m}^{-N_{02}} \mid [\Gamma]) \Pr([\Gamma]), \\ &= \sum_{[\Gamma]} \sum_{\rho_{1,m}^{-N_{02}}} \Pr(\rho_{1,m} = (S_1, \ldots, S_k) \mid [\Gamma],\, \rho_{1,m}^{-N_{02}}) \times \\ &\qquad \Pr(\rho_{1,m}^{-N_{02}} \mid [\Gamma]) \Pr([\Gamma]), \\ &= \Pr(\rho_{1,m} = (S_1, \ldots, S_k)),\end{aligned}$$

where the second to last equality follows from the constructive description given earlier and the properties of the EPPF. The result then follows. □

A.2 Proof of Proposition 2

Proof. The proof proceeds by direct calculations.

(a) Let $\gamma_i = 1$ if unit $i \in [m]$ is not relocated. Note that $\gamma_1, \gamma_2 \overset{iid}{\sim} Ber(\alpha)$. By definition, $P(c_{11} = c_{21}) = \frac{1}{M+1}$ and $P(c_{11} \neq c_{21}) = \frac{M}{M+1}$. By conditioning on γ_1, γ_2 and c_{11}, c_{21}

we get

$$P\left(c_{12}=c_{22}\mid c_{11}=c_{21},(\gamma_1,\gamma_2)\right)=\begin{cases}1 & \text{if } (\gamma_1,\gamma_2)=(1,1)\\ \frac{1}{M+1} & \text{otherwise.}\end{cases}$$

It then follows that

$$\begin{aligned}P\left(c_{12}=c_{22},c_{11}=c_{21}\right) &= \sum_{\gamma_1,\gamma_2} P\left(c_{12}=c_{22}\mid c_{11}=c_{21},(\gamma_1,\gamma_2)\right)P\left(c_{11}=c_{21}\right)P(\gamma_1)P(\gamma_2)\\ &= \frac{\alpha^2}{M+1}+\frac{(1-\alpha^2)}{(M+1)^2} \qquad \text{(S.4)}\end{aligned}$$

Similarly,

$$P\left(c_{12}\neq c_{22}\mid c_{11}\neq c_{21},(\gamma_1,\gamma_2)\right)=\begin{cases}1 & \text{if } (\gamma_1,\gamma_2)=(1,1)\\ \frac{M}{M+1} & \text{otherwise,}\end{cases}$$

and proceeding as before, we easily get

$$\begin{aligned}P\left(c_{12}\neq c_{22},c_{11}\neq c_{21}\right) &= \sum_{\gamma_1,\gamma_2} P\left(c_{12}\neq c_{22}\mid c_{11}\neq c_{21},(\gamma_1,\gamma_2)\right)P\left(c_{11}\neq c_{21}\right)P(\gamma_1)P(\gamma_2)\\ &= \frac{M\alpha^2}{M+1}+\left(\frac{M}{M+1}\right)^2\left(1-\alpha^2\right) \qquad \text{(S.5)}\end{aligned}$$

The result now easily follows by summing (S.4) and (S.5).

(b) As before, denote by $\gamma_i=1$ if unit $i\in[m]$ is *not* removed from the partition. By conditioning on γ_1,γ_2 and c_{11},c_{21} we get

$$P\left(c_{12}=c_{22}\mid c_{11}=c_{21},(\gamma_1,\gamma_2)\right)=\begin{cases}\frac{6+M}{(M+2)(M+3)} & \text{if } (\gamma_1,\gamma_2)=(1,1)\\ \frac{1}{M+1} & \text{otherwise,}\end{cases}$$

and proceeding as earlier,

$$P\left(c_{12}=c_{22},c_{11}=c_{21}\right)=\frac{(6+M)\alpha^2}{(M+1)(M+2)(M+3)}+\frac{(1-\alpha^2)}{(M+1)^2} \tag{S.6}$$

Also,

$$P\left(c_{12}\neq c_{22}\mid c_{11}\neq c_{21},(\gamma_1,\gamma_2)\right)=\begin{cases}\frac{M(M+4)+2}{(M+2)(M+3)} & \text{if } (\gamma_1,\gamma_2)=(1,1)\\ \frac{M}{M+1} & \text{otherwise,}\end{cases}$$

from which

$$P\left(c_{12}\neq c_{22},c_{11}\neq c_{21}\right)=\left(\frac{M(M+4)+2}{(M+2)(M+3)}\right)\left(\frac{M}{M+1}\right)\alpha^2+\left(\frac{M}{M+1}\right)^2\left(1-\alpha^2\right). \tag{S.7}$$

The result now easily follows by summing (S.6) and (S.7). □

A.3 Proof of Proposition 3

Proof. Let $P_{C_t}=\{\rho_t\in P:\rho_{t-1}^{\mathfrak{R}_t}=\rho_t^{\mathfrak{R}^t}\}$ denote the collection of all partitions of the elements of $[m]$ at time t that are compatible with ρ_{t-1} based on $\boldsymbol{\gamma}_t$. Then by construction, $\Pr(\rho_t|\boldsymbol{\gamma}_t,\rho_{t-1})$ is a random partition distribution whose support is P_{C_t} so that

$$\Pr(\rho_t=\lambda|\boldsymbol{\gamma}_t,\rho_{t-1})=\frac{\Pr(\rho_t=\lambda)I[\lambda\in P_{C_t}]}{\sum_{\lambda'}\Pr(\rho_t=\lambda')I[\lambda'\in P_{C_t}]}.$$

It only remains to show that $\sum_{\lambda\in P_{C_t}}\Pr(\rho_t=\lambda)=\Pr(\rho_t^{\mathfrak{R}_t})$ which is more easily seen employing cluster label notation. Let $c_{\gamma_t}=\{c_{it}:\gamma_{it}=0\}$. By iteratively invoking the

sample size consistency property we have that

$$\begin{aligned}\Pr(\rho_t^{\mathfrak{R}_t}) &= \sum_{c_{\gamma_t}} \Pr(\rho_t = \{c_{1t}, \ldots, c_{mt}\}) \\ &= \sum_{\lambda \in P_{C_t}} \Pr(\rho_t = \lambda),\end{aligned}$$

where the last equality holds since summing over c_{γ_t} is based only on cluster labels that are not fixed from time point $t-1$ to t which results in summing over all possible compatible partitions (i.e., $\lambda \in P_{C_t}$). □

B Details Associated With the MCMC Algorithm

Here we provide much more detail associated with the MCMC scheme. We place emphasis on the updating steps for $\boldsymbol{\gamma}_{it}$ and ρ_t as the other steps are straightforward once these parameters have been updated. That said, pseudocode of the entire algorithm is provided in Algorithm 1. A key component of the MCMC algorithm is to check the compatibility between ρ_{t-1} and ρ_t, and between ρ_t and ρ_{t+1} when updating $\boldsymbol{\gamma}_t$ and ρ_t. This is equivalent to ensuring that $\rho_{t-1}^{\mathfrak{R}_t} = \rho_t^{\mathfrak{R}_t}$ and $\rho_t^{\mathfrak{R}_{t+1}} = \rho_{t+1}^{\mathfrak{R}_{t+1}}$. We describe the process of updating each of the c_{it} and γ_{it} sequentially in time so that the entire vector $\boldsymbol{\gamma}_t$ is updated first and then $\boldsymbol{c}_t$.

B.1 Updating γ_{it}

First note that $\boldsymbol{\gamma}_t$ only connects ρ_{t-1} to ρ_t so that when updating γ_{it} only compatibility between ρ_{t-1} and ρ_t needs to be checked (i.e., ρ_t remains compatible with ρ_{t+1} due to γ_{t+1} by construction). We detail updating γ_{it} in an MCMC algorithm based on its full

conditional found in (**??**) of the main paper and which we provide here for sake of clarity

$$\Pr(\gamma_{it} = 1 \mid -) = \frac{\alpha_t}{\alpha_t + (1-\alpha_t)\Pr(\rho_t^{\mathfrak{R}_t^{(+i)}})/\Pr(\rho_t^{\mathfrak{R}_t^{(-i)}})} \mathrm{I}[\rho_{t-1}^{\mathfrak{R}_t^{(+i)}} = \rho_t^{\mathfrak{R}_t^{(+i)}}]. \tag{S.8}$$

The appeal of this form of the full conditional compared to that found in (**??**) of the main paper is that the EPPF used to compute $\Pr(\rho_t^{\mathfrak{R}_t^{(+i)}})$ and $\Pr(\rho_t^{\mathfrak{R}_t^{(-i)}})$ need not have a tractable normalizing constant. That said, an exchangeable sequence of cluster labels $(c_{1t}, \ldots c_{mt})$ is necessary. Now let $\gamma_{it}^{(d)}$ and $\rho_t^{(d)}$ be the value of γ_{it} and ρ_t at the dth MCMC iterate. Note that there are four scenarios to consider when moving from $\gamma_{it}^{(d-1)}$ to $\gamma_{it}^{(d)}$. They are

1. $\gamma_{it}^{(d-1)} = 1 \to \gamma_{it}^{(d)} = 0$ (For this move $\rho_{t-1}^{(d)\mathfrak{R}_t^{(-i)}} = \rho_t^{(d-1)\mathfrak{R}_t^{(-i)}}$ continues to hold),
2. $\gamma_{it}^{(d-1)} = 1 \to \gamma_{it}^{(d)} = 1$ (For this move $\rho_{t-1}^{(d)\mathfrak{R}_t^{(+i)}} = \rho_t^{(d-1)\mathfrak{R}_t^{(+i)}}$ continues to hold),
3. $\gamma_{it}^{(d-1)} = 0 \to \gamma_{it}^{(d)} = 0$ (For this move $\rho_{t-1}^{(d)\mathfrak{R}_t^{(-i)}} = \rho_t^{(d-1)\mathfrak{R}_t^{(-i)}}$ continues to hold), and
4. $\gamma_{it}^{(d-1)} = 0 \to \gamma_{it}^{(d)} = 1$ (For this move $\rho_{t-1}^{(d)\mathfrak{R}_t^{(+i)}} = \rho_t^{(d-1)\mathfrak{R}_t^{(+i)}}$ needs to be verified).

Thus compatibility needs to be checked only for (4).

As expected, calculating the ratio $\Pr(\rho_t^{\mathfrak{R}_t^{(+i)}})/\Pr(\rho_t^{\mathfrak{R}_t^{(-i)}})$ in (S.8) is the most challenging part of computing $\Pr(\gamma_{it} = 1 \mid -)$. However, it can be straightforwardly calculated using exchangeability and ideas from Neal (2000). Under the assumption of exchangeability, which permits allocating the ith unit by treating it as if it were the last unit, we have that

$$\frac{\Pr(\rho_t^{\mathfrak{R}_t^{(+i)}})}{\Pr(\rho_t^{\mathfrak{R}_t^{(-i)}})} = \frac{\Pr(c_{it}|\rho_t^{\mathfrak{R}_t^{(-i)}})\Pr(\rho_t^{\mathfrak{R}_t^{(-i)}})}{\Pr(\rho_t^{\mathfrak{R}_t^{(-i)}})} = \Pr(c_{it}|\rho_t^{\mathfrak{R}_t^{(-i)}}). \tag{S.9}$$

Note that the probability in (S.9) is a standard calculation, used in Neal's Algorithm 8, for each c_{it}. When the EPPF does not have a tractable normalizing constant, one may compute the unnormalized probability of allocation to each of the k_t existing clusters and to a new singleton cluster and then normalize to obtain (S.1). Of course, here we know the

value of c_{it} from $\rho_t^{(d-1)}$ and, in constraints to Neal's Algorithm 8, this computation is done for the sake of computing the full conditional distribution of γ_{it}. Once (S.9) is calculated, computing (S.8) and updating γ_{it} is straightforward.

B.2 Updating ρ_t Using Cluster Labels

First note that only those c_{it} that correspond to $\gamma_{it} = 0$ are updated. As a result, the compatibility between ρ_{t-1} and ρ_t is preserved and so only the compatibility between ρ_t and ρ_{t+1} needs to be checked when updating any of the c_{it}. Recall that the full conditional of c_{it} corresponding to $\gamma_{it} = 0$ is

$$\Pr(c_{it} = h \mid -) \propto \begin{cases} N(Y_{it} \mid \mu^{\star}_{c_{it}=h,t}, \sigma^{2\star}_{c_{it}=h,t}) \Pr(c_{it} = h) \mathrm{I}[\rho_t^{h\mathfrak{R}_{t+1}} = \rho_{t+1}^{\mathfrak{R}_{t+1}}] & \text{for } h = 1, \ldots, k_t^{-i}, \\ N(Y_{it} \mid \mu^{\star}_{new_h,t}, \sigma^{2\star}_{new_h,t}) \Pr(c_{it} = h) \mathrm{I}[\rho_t^{h\mathfrak{R}_{t+1}} = \rho_{t+1}^{\mathfrak{R}_{t+1}}] & \text{for } h = k_t^{-i} + 1, \end{cases} \tag{S.10}$$

where $\Pr(c_{it} = h) = \Pr(c_{1t}, \ldots, c_{it} = h, \ldots, c_{mt})$, and k_t^{-i} is the number of clusters at time t when the ith unit has been removed. The partition constructed from $\{c_{1t}, \ldots, c_{it} = h, \ldots, c_{mt}\}$ is denoted as $\rho_t^h = \{S_{1t}^{-i}, \ldots, S_{ht}^{-i} \cup \{i\}, \ldots, S_{k_t^{-i}t}^{-i}\}$ where S_{jt}^{-i} denotes the jth cluster at time t with the ith unit removed. Note that it is possible that $S_{jt}^{-i} = S_{jt}$. Further, abusing notation, for $h = k_t^{-i} + 1$ we have $\rho_t^h = \{S_{1t}^{-i}, \ldots, S_{ht}^{-i}, \ldots, S_{k_t^{-i}t}^{-i}, \{i\}\}$. Additionally, $\mu^{\star}_{new_h,t}$ and $\sigma^{2\star}_{new_h,t}$ are auxiliary parameters drawn from the prior as in Neal (2000)'s Algorithm 8 (with one auxiliary parameter). Then based on a spatial product partition model for ρ_t, for $h = 1, \ldots, k_t^{-i}$ the $\Pr(c_{it} = h)$ becomes

$$\Pr(c_{it} = h) = Pr(\rho_t^h) \propto M\Gamma(|S_{ht}^{-i} \cup \{i\}|) g(\boldsymbol{s}_{ht}^{-i\star} \cup \boldsymbol{s}_i | \nu_0) \prod_{j \neq h}^{k_t^{-i}} M\Gamma(|S_{jt}^{-i}|) g(\boldsymbol{s}_{jt}^{-i\star} | \nu_0), \tag{S.11}$$

while for $h = k_t^{-i} + 1$

$$\Pr(c_{it} = h) = Pr(\rho_t^h) \propto M\Gamma(|\{i\}|)g(\boldsymbol{s}_i|\nu_0)\prod_{j=h}^{k_t^{-i}} M\Gamma(|S_{jt}^{-i}|)g(\boldsymbol{s}_{jt}^{-i\star}|\nu_0), \tag{S.12}$$

where $g(\cdot)$ is the auxiliary similarity function detailed in Page & Quintana (2016) and $\boldsymbol{s}_{jt}^{-i\star} = \{\boldsymbol{s}_{i'} : i' \in S_{jt}^{-i}\}$ are the spatial coordinates from units that belong to the jth cluster at time t. Updating c_{it} can be carried out by evaluating (S.10) based on (S.11) or (S.12) for each $h = 1, \ldots, k_t^{-i} + 1$ and then normalizing.

Once each of the $\boldsymbol{c}_t$ and $\boldsymbol{\gamma}_t$ are updated the MCMC algorithm is completed by cycling through remaining model and latent parameters found in model (**??**) and updating them on an individual basis using well known approaches. In order to visualize all the moving parts of the MCMC algorithm we provide some pseudocode in Algorithm 1. For sake of simplicity, Algorithm 1 describes an MCMC procedure that can be employed to sample from the joint posterior distribution based on model (**??**).

C Simulation Studies

In this section we provide more details associated with Simulation 1, the competitors included in Simulation 3, and provide results from additional simulations similar to that described in the Section 3.3 of the main document. We then provide details associated with a simulation study that includes spatial information.

C.1 Simulation 1: Continued

Table S.1 contains the adjusted Rand index values between the estimated partitions and that which was used to generate data. Interestingly, as α increases, the ARI values also tend to increase.

Algorithm 1 : Pseudocode for the MCMC algorithm for model (**??**) of main article. Let T be the number of time points, m the number of units at each time point, and D the number of MCMC iterations.

for $d = 1, \ldots, D$ **do**
  for $t = 1, \ldots, T$ **do** ▷ For each t, update the entire $\boldsymbol{\gamma}_t$ vector first and then $\boldsymbol{c}_t$
    for $i = 1, \ldots, m$ **do**
      Set $\gamma_{i1}^{(d)} = 0$.
      if $t > 1$ **then**
        - Update γ_{it} based on procedure described in Section B.1. That is,
        if $\gamma_{it}^{(d-1)} = 1$ **then**
          Move to $\gamma_{it}^{(d)}$ using Bernoulli probability in (S.8). Compatibility holds by construction.
        if $\gamma_{it}^{(d-1)} = 0$ **then**
          Move to $\gamma_{it}^{(d)}$ using Bernoulli probability in (S.8). If $\gamma_{it}^{(d)} = 0$, then compatibility holds by construction. If $\gamma_{it}^{(d)} = 1$, the compatibility needs to be checked. If $\rho_{t-1}^{(d)\mathfrak{R}_t^{(+i)}} \neq \rho_t^{(d-1)\mathfrak{R}_t^{(+i)}}$, then set $\gamma_{it}^{(d)} = 0$.
    for $i = 1, \ldots, m$ **do**
      - Update c_{it} based on procedure described in Section B.2.
      for $h = 1, \ldots, k_t^{-i}$ **do**
        Compute the unnormalized multinomial probability in (S.10) based on (S.11).
        if $\rho_t^{h\mathfrak{R}_{t+1}} \neq \rho_{t+1}^{\mathfrak{R}_{t+1}}$ **then**
          Set unnormalized multinomial probability to zero.
      for $h = k_t^{-i} + 1$ **do**
        Compute the unnormalized multinomial probability in (S.10) based on (S.12).
      Sample c_{it} using the normalized $k_t^{-i} + 1$ multinomial probabilities.
    for $j = 1, \ldots, K^{(d)}$ **do** ▷ $K^{(d)}$ = number of clusters at the dth iteration.
      - Update $\mu_{jt}^{\star}$ based on Gaussian full conditional derived using well known arguments.
      - Update $\sigma_{jt}^{2\star}$ using a random walk Metropolis step.
    - Update θ_t based on Gaussian full conditional derived using well known arguments.
    - Update α_t based on beta full conditional derived using well known arguments .
  - Update τ^2 using a random walk Metropolis step.
  - Update ϕ_0 based on Gaussian full conditional derived using well known arguments.
  - Update λ^2 using a random walk Metropolis step.

Table S.1: Adjusted Rand index when comparing $\hat{\rho}_t$ to the true ρ_t for $t = 1, \ldots, 5$. Note that $ARI(\cdot, \cdot)$ denotes the adjusted Rand index as a function of two partitions. These values are averaged over the 100 generated data sets. The values in parenthesis are Monte Carlo standard errors.

	$ARI(\hat{\rho}_1, \rho_1)$	$ARI(\hat{\rho}_2, \rho_2)$	$ARI(\hat{\rho}_3, \rho_3)$	$ARI(\hat{\rho}_4, \rho_4)$	$ARI(\hat{\rho}_5, \rho_5)$
$\alpha = 0.0$	0.58 (0.03)	0.63 (0.03)	0.58 (0.03)	0.54 (0.03)	0.56 (0.03)
$\alpha = 0.1$	0.63 (0.03)	0.56 (0.03)	0.55 (0.03)	0.62 (0.03)	0.57 (0.03)
$\alpha = 0.25$	0.52 (0.03)	0.57 (0.03)	0.55 (0.03)	0.63 (0.03)	0.62 (0.03)
$\alpha = 0.5$	0.60 (0.03)	0.70 (0.03)	0.69 (0.03)	0.66 (0.02)	0.59 (0.03)
$\alpha = 0.75$	0.78 (0.02)	0.77 (0.02)	0.82 (0.02)	0.80 (0.02)	0.75 (0.02)
$\alpha = 0.9$	0.83 (0.02)	0.86 (0.02)	0.87 (0.02)	0.84 (0.02)	0.76 (0.02)
$\alpha = 0.9999$	0.92 (0.01)	0.92 (0.01)	0.92 (0.01)	0.92 (0.01)	0.92 (0.01)

C.2 Simulation 3: Continued

As referenced in the main article, Figure S.1 provides an example of the type of data that is generated in the simulation of Section **??** in the main article. To each of the 100 data sets generated, we fit our method and four other procedures. We know provide specific details of the competing methods.

1. weighted DDP (wddp): A complete description of this model can be found in Section 4 of Quintana et al. (2020) (and Müller et al. 2015). We only provide pertinent details here. Let $\boldsymbol{z}_i = (Y_i, t_i)$ be the response and time pair for $i = 1, \ldots, mT$. The wddp models $\boldsymbol{z}_i$ with a Dirichlet process mixture model (DPM) and then derives the conditional model $(Y_i|t_i)$. In hierarchical form (including cluster labels) the model is

$$
\begin{aligned}
\boldsymbol{z}_i|\boldsymbol{\mu}^\star, \boldsymbol{\Sigma}^\star, c_i &\overset{ind}{\sim} N_2(\boldsymbol{\mu}^\star_{c_i}, \boldsymbol{\Sigma}^\star_{c_i}), \quad i = 1, \ldots, mT \\
\boldsymbol{\mu}^\star_j &\overset{iid}{\sim} N_2(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0), \quad j = 1, \ldots, K \\
\boldsymbol{\Sigma}^\star_j &\overset{iid}{\sim} \text{Inverse-Wishart}_2(\nu, \psi\boldsymbol{I}), \quad j = 1, \ldots, K \\
\boldsymbol{\mu}_0 &\overset{iid}{\sim} N_2(\boldsymbol{m}, s^2\boldsymbol{I}), \\
\boldsymbol{\Sigma}_0 &\overset{iid}{\sim} \text{Inverse-Wishart}_2(\nu_0, \psi_0\boldsymbol{I}), \\
\Pr(c_i = j) = \pi_j \quad &\text{where } \pi_j = V_j \textstyle\prod_{\ell<j}(1 - V_\ell), \\
V_\ell &\overset{iid}{\sim} \text{Beta}(1, M).
\end{aligned}
$$

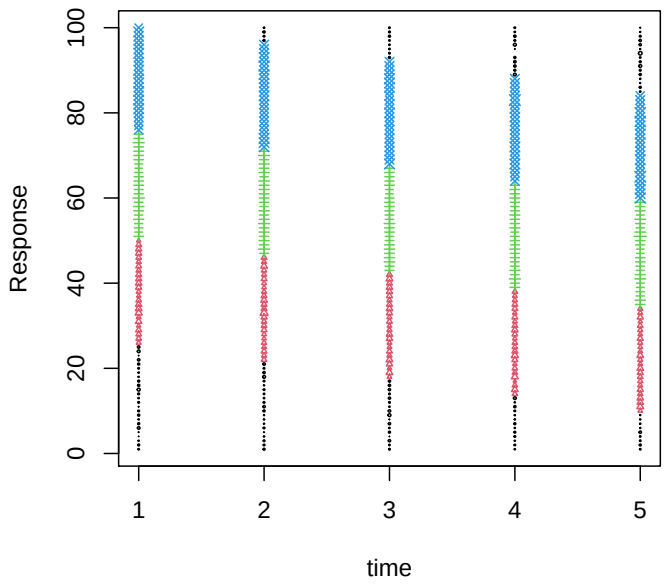

Figure S.1: One realization of a synthetic data set from simulation study in Section **??**. Each color corresponds to a cluster and the size of plotted symbol is proportional to the value of point being plotted.

We set $\boldsymbol{m} = 0\boldsymbol{j}$, $s^2 = 25$, $\nu = \nu_0 = 4$, $\psi = \psi_0 = 1$, $K = 30$, and $M = 1$. The model induces a weight-dependent mixture model of regressions

$$f(y_i|t_i) = \sum_{j=1}^{K} w_j(t_i) N(y_i|\beta_{0j}^\star + \beta_{1j}^\star t_i, \sigma_j^{2\star}),$$

where

$$w_j(t_i) = \frac{\pi_j N(t_i|\mu_{2j}^\star, \Sigma_{22j}^\star)}{\sum_{\ell=1}^{K} \pi_\ell N(t_i|\mu_{2\ell}^\star, \Sigma_{22\ell}^\star)}, \quad j = 1, \ldots, K,$$

and $\beta_{0j}^\star = \mu_{1j}^\star - \frac{\Sigma_{12j}^\star}{\Sigma_{22j}^\star}\mu_{2j}^\star$, $\beta_{1h} = \frac{\Sigma_{12j}^\star}{\Sigma_{22j}^\star}$, and $\sigma_j^{2\star} = \Sigma_{11j}^\star - \frac{\Sigma_{12j}^\star \Sigma_{21j}^\star}{\Sigma_{22j}^\star}$. Note that time is include in the weights of the the conditional model which is employed to calculate LPML and WAIC. A blocked Gibbs sampler was employed to sample from the posterior where $V_K = 1$ to ensure that $\sum_{j=1}^{K} \pi_j = 1$.

2. linear DDP (lddp): A complete description of this model is provided in chapter 4.4.2 of Müller et al. (2015) (see also Quintana et al. 2020). We only provide pertinent details here. As with the wddp model, for the lddp we consider (Y_i, t_i) for $i = 1, \ldots, mT$. Time is incorporated in the atoms of a Dirichlet process (DP) so that the jth atom is expressed as $\sum_\ell^q B_\ell(t, \boldsymbol{\xi})\beta_{j\ell}$ where $B_\ell(t, \boldsymbol{\xi})$ denotes the ℓ-th B-spline basis function evaluated at t for knots $\boldsymbol{\xi}$. Letting $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_q)$ and $\boldsymbol{B}(t_i, \boldsymbol{\xi})$ the q-dimensional B-spline basis vector for unit i and after introducing cluster labels, the lddp model can be expressed hierarchically as

$$\begin{aligned}
Y_i|\boldsymbol{\beta}^\star, \sigma^{2\star}, c_i &\overset{ind}{\sim} N(\boldsymbol{B}'(t_i, \boldsymbol{\xi})\boldsymbol{\beta}_{c_i}^\star, \sigma_{c_i}^{2\star}), \quad i = 1, \ldots, mT, \\
\boldsymbol{\beta}_j^\star &\overset{iid}{\sim} N_q(\boldsymbol{\beta}_0, \boldsymbol{\Sigma}_0), \quad j = 1, \ldots, k, \\
\sigma_j^{2\star} &\sim \text{Inverse-Gamma}(a, b), \quad j = 1, \ldots, k, \\
\boldsymbol{\beta}_0 &\overset{iid}{\sim} N_2(\boldsymbol{m}, s^2\boldsymbol{I}), \\
\boldsymbol{\Sigma}_0 &\overset{iid}{\sim} \text{Inverse-Wishart}_2(\nu_0, \psi_0\boldsymbol{I}), \\
\{c_1, \ldots, c_{mT}\} &\sim CRP(M).
\end{aligned}$$

We set $\boldsymbol{m} = 0\boldsymbol{j}$, $s^2 = 25$, $\nu_0 = q + 2$, $\psi_0 = 10$, $a = b = 1$, and $M = 1$. Neal's Algorithm 8 (Neal 2000) was employed to sample from the posterior distribution.

3. Griffiths-Milne dependent Dirichlet process (gmddp) mixture. This is carried out using `DDPdensity` in the `BNPmix` package found in `R`. The function considers partially exchangeable data (Lijoi et al., 2014) such that exchangeability is assumed within each group and the vector of random probability measures at each time point are modeled jointly as a vector of GM-DDP.

4. A temporally independent $CRP(M)$ model (ind_crp): This model is a special case of Caron et al. (2007)'s and our model. Specifically, α is set to 0. For this procedure, the following model was fit separately for each time period.

$$
\begin{aligned}
Y_i \mid \boldsymbol{\mu}^\star, \boldsymbol{\sigma}^{2\star}, c_i &\overset{ind}{\sim} N(\mu^\star_{c_i}, \sigma^{2\star}_{c_i}),\ i = 1, \ldots, m, \\
(\mu^\star_j, \sigma^\star_j) \mid \theta, \tau^2 &\overset{ind}{\sim} N(\theta, \tau^2) \times UN(0, A_\sigma),\ j = 1, \ldots, k, \\
(\theta, \tau) &\overset{iid}{\sim} N(m_0, s_0^2) \times UN(0, A_\tau).
\end{aligned}
$$

   We set $m_0 = 0$, $s_0^2 = 10^2$, $A_\sigma = 0.5sd(Yvec)$, and $A_\tau = 100$. Neal's Algorithm 8 (Neal 2000) was used to sample from the posterior distribution.

5. A temporally static $CRP(M)$ model (static_crp): This procedure is a special case of Caron et al. (2007) ($\alpha = 1$) and is fit to a concatenated version of the data Y_i, for $i = 1, \ldots, mT$. Specifically, the following model was fit

$$
\begin{aligned}
Y_i \mid \boldsymbol{\mu}^\star, \boldsymbol{\sigma}^{2\star}, c_i &\overset{ind}{\sim} N(\mu^\star_{c_i}, \sigma^{2\star}_{c_i}),\ i = 1, \ldots, mT, \\
(\mu^\star_j, \sigma^\star_j) \mid \theta, \tau^2 &\overset{ind}{\sim} N(\theta, \tau^2) \times UN(0, A_\sigma),\ j = 1, \ldots, k, \\
(\theta, \tau) &\overset{iid}{\sim} N(m_0, s_0^2) \times UN(0, A_\tau).
\end{aligned}
$$

   We set $m_0 = 0$, $s_0^2 = 10^2$, $A_\sigma = 0.5sd(Yvec)$, and $A_\tau = 100$. Neal's Algorithm 8 (Neal 2000) was used to sample from the posterior distribution.

C.2.1 Results from Additional Synthetic Data

In addition to the synthetic data generated in Simulation 3 of the main document, we also generated data as described in the following two scenarios.

Scenario 1: For the ith unit, we employ the following as a data generating mechanism

$$y_{it} = \omega y_{it-1} + \epsilon_{it}, \text{ for } i = 1, \ldots, m, \text{ and } t = 1, \ldots, T,$$

where $|\omega| < 1$ and $\epsilon_{it} \sim N(0, v^2)$. For this scenario measurements are are correlated across time, but independent between the m units. Data were generated with $m = 100$ and using the following levels of factors of interest

- $\omega \in \{0, 0.1, 0.25, 0.5, 0.75, 0.9\}$
- $v^2 \in \{0.5^2, 1^2\}$
- $T \in \{5, 10\}$

Notice that for this scenario, there is no "true" partition and so we are interested only in comparing the model fit of our approach to that of the five competitors.

Scenario 2: For the ith unit, we employ the following as a data generating mechanism

$$y_{it} = \omega_{c_i} y_{it-1} + \epsilon_{it}, \text{ for } i = 1, \ldots, m, \text{ and } t = 1, \ldots, T,$$

where as before $c_{it} \in \{1, 2, 3, 4\}$ with $\omega \in \{-0.75, -0.25, 0.25, 0.75\}$ and $\epsilon_{it} \sim N(0, v^2)$. As in the previous scenarios measurements are correlated across time, but independent between the m units. Data were generated with $m = 100$ and using the following levels of factors of interest

- $v^2 \in \{0.5^2, 1^2\}$
- $T \in \{5, 10\}$

In this scenario, there is a "true" partition but our approach, nor the competitors, are parametrized in such a way as to detect it. Indeed, our method models temporal

dependence only through the partition (i.e, there is no temporal correlation parameter in the likelihood). That said, we still compare partition recovery by way of the adjusted Rand index (ARI) in addition to model fit.

In both scenarios the function `arima.sim` in `R` (R Core Team 2020) is used to generate the 100 replicate data sets. We collect 1,000 MCMC iterates after discarding the first 25,000 as burn-in and thinning by 25 (i.e., 50,000 total MCMC draws were collected). The prior parameters that we used are $A_\sigma = 0.5sd(vec(Y))$, $A_\tau = 100$, $A_\lambda = 100$, $m_0 = 0$, $s2_0 = 100$, $a_\alpha = 1$, $b_\alpha = 1$. WAIC is used to compare each of the procedures in terms of model fit and ARI to compare ability of estimating the true partition structure. Results are found in Figures S.2 - S.4. From Figure S.2 we see that our approach produces the smallest WAIC for all factors considered Scenarios 1's data generating schemes. Thus, our approach tends to fit the data best.

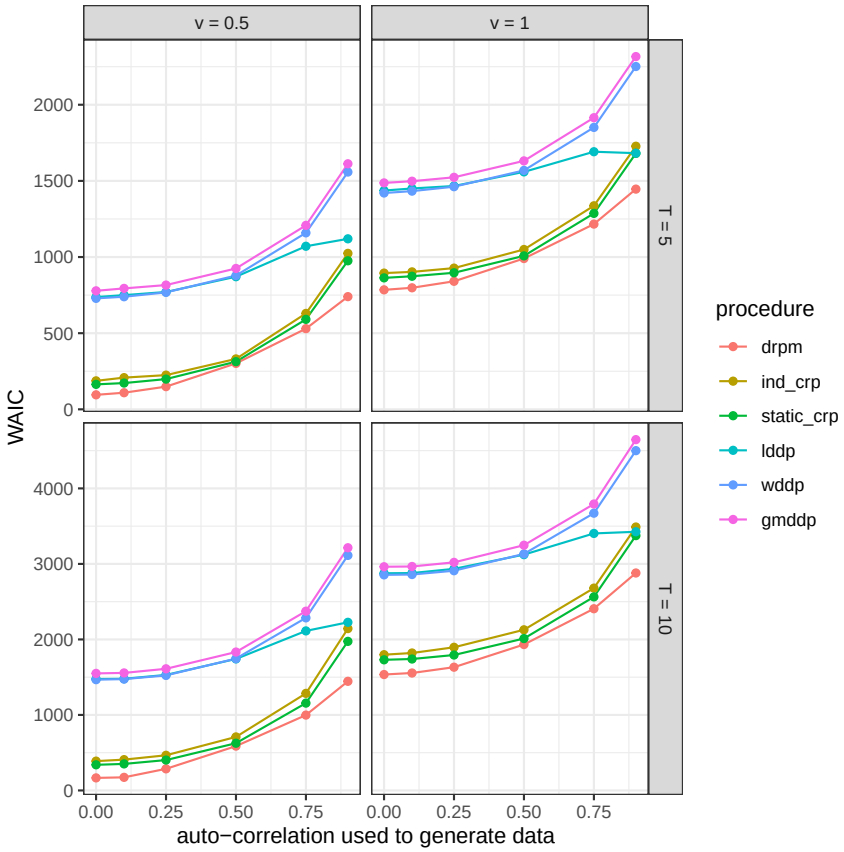

Figure S.2: Results for WAIC from the first data generating scenario.

For Scenario 2, static_crp is quite competitive to our approach and produces similar WAIC values. Apart from that, our approach does better than the other competitors. From Figure S.4 our approach does much better at recovering the partition compared to other procedures. That said, we the ARI values are still quite small (which was expected).

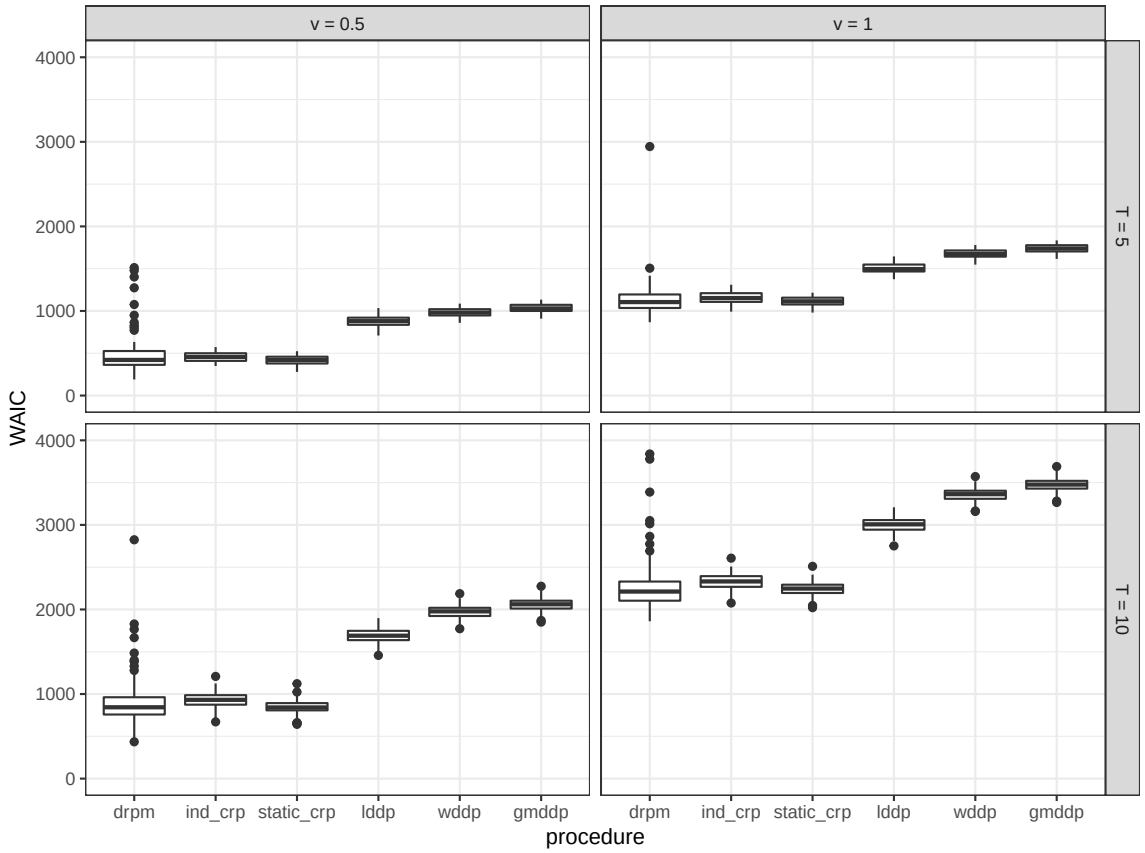

Figure S.3: Results for WAIC from the fourth data generating scenario.

C.3 Simulation 4: Space-Time Data Generation

Here we discuss our final simulation study, where we investigated the performance of our procedure when both space and time are considered. To do so, we created synthetic data sets that contain spatio-temporal structure. Each employs a 15×15 regular grid with spatial locations coming from the unit interval. In addition, either 5 or 10 time points were considered resulting in 1,125 or 2,250 total observations. Response values were generated

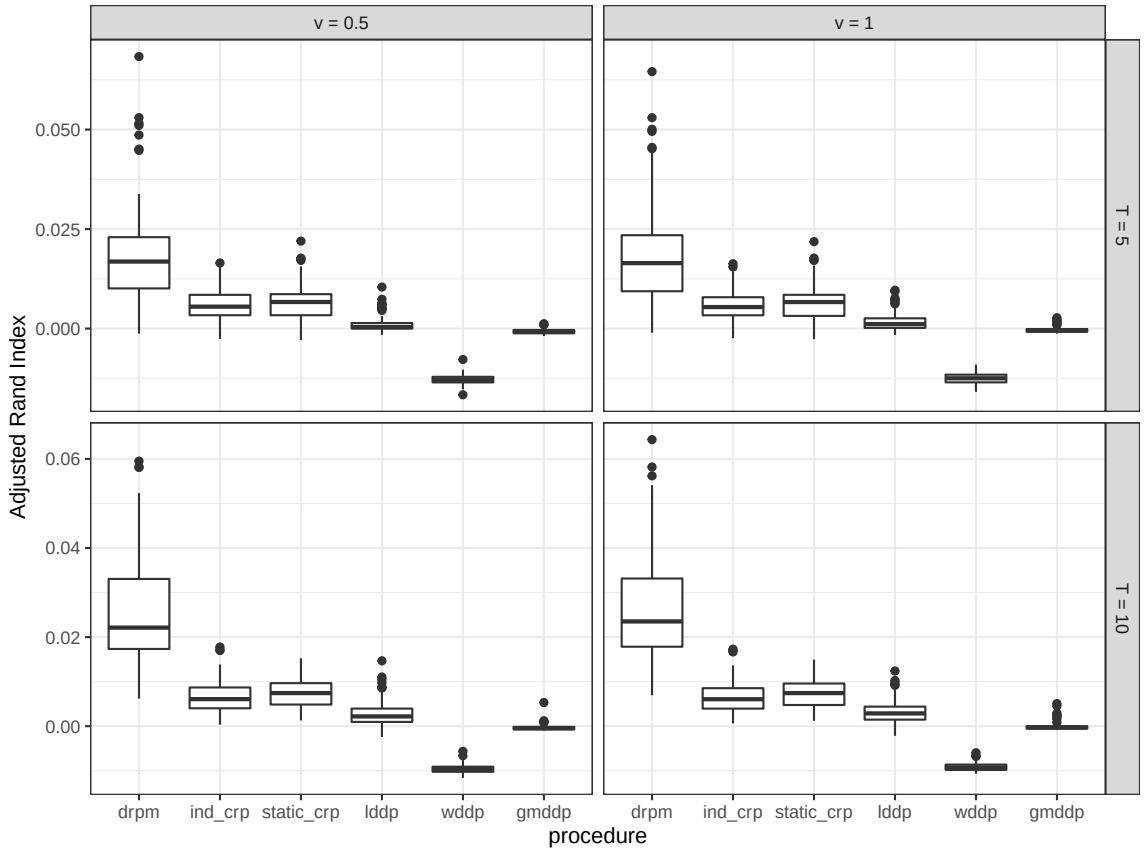

Figure S.4: Results for ARI from the fourth data generating scenario.

in two ways. The first employs a Gaussian process with a separable spatio-temporal exponential covariance function. We set the spatial scale to 0.3, the temporal scale to 2 and the sill to 1.75 (see Padoan & Bevilacqua 2015 for more details). Note that no "true" partition exists for this data generating mechanism. However, we study it to explore performance of our method when spatial structure exists among observations but was not induced through partitioning. The second method of generating response values essentially employs model (**??**) as a data generating mechanism. Spatio-temporal partitions were generated using (**??**) together with conditional cluster label probabilities of Müller et al. (2011, pg. 265) and setting $\alpha_t = \alpha$ for all t with $\alpha \in \{0, 0.5, 0.9\}$ (note that for $\alpha = 0$ no temporal dependence exists among partitions). In the similarity function (**??**) we considered $\nu_0 \in \{2, 20\}$ where $\nu_0 = 2$ corresponds to light weight on spatial proximity and $\nu_0 = 20$ moderate weight. Finally, we set $\tau^2 = 1$ and $\sigma^{2\star}_{c_{it}t} = \sigma^2 = 0.04$ for all i and t resulting in smaller with-in cluster variability relative to between-cluster variability.

To determine the impact that each component of our spatio-temporal partition model has on model fit, we fit the hierarchical model (**??**) to each synthetic data set using a variety of random partition models which are listed below. As a competitor, we consider a linear dependent Dirichlet process (MacEachern 2000, De Iorio et al. 2009), indexing the random probability measure through the mean function of the atoms by space and time. To ensure sufficient flexibility, B-spline basis functions for both spatial coordinates were employed. The details of each model considered are

Model 1: $(\rho_1, \ldots, \rho_T) \sim stRPM(\boldsymbol{\alpha}, \nu_0, M)$

Model 2: $\rho_t \overset{iid}{\sim} sPPM(\nu_0, M)$ for $t = 1, \ldots, T$.

Model 3: $(\rho_1, \ldots, \rho_T) \sim tRPM(\boldsymbol{\alpha}, M)$

Model 4: $\rho_t \overset{iid}{\sim} CRP(M)$ for $t = 1, \ldots, T$.

Model 5: linear dependent Dirichlet process mixture model (DDPM).

Additionally, for each model that employs the sPPM, we considered both $\nu_0 = 2$ (models 1a, 2a) and $\nu_0 = 20$ (models 1b, 2b). For each data generating scenario, 100 data sets

were created and each of the models listed was fit by collecting 1,000 MCMC samples after discarding the first 5,000 as burn-in and thinning by 5 after setting $A_\sigma = 1$ and $A_\tau = 2$. Model fits were compared using WAIC. Results can be found in Figures S.5 and S.6.

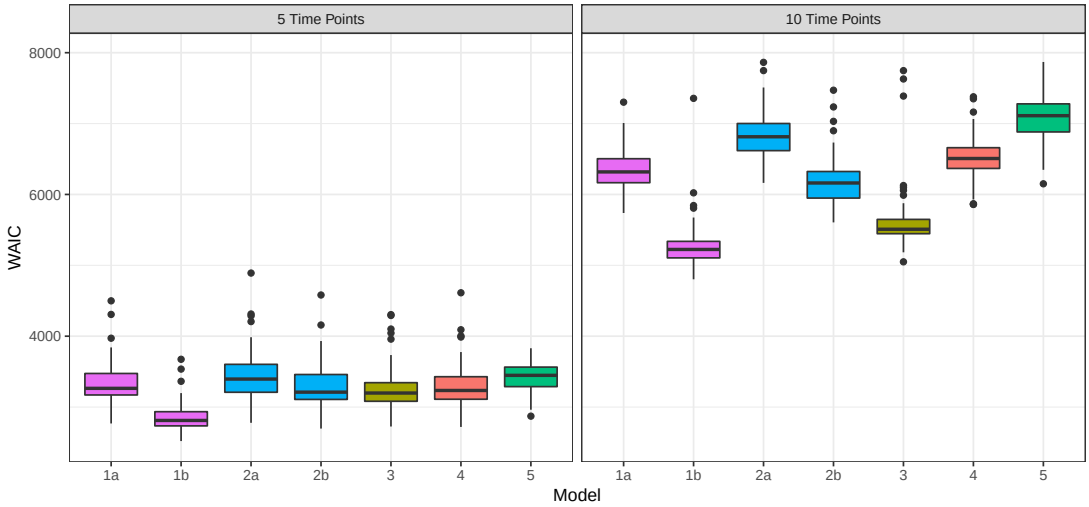

Figure S.5: Results from simulation study when observations were generated using a spatio-temporal Gaussian process. Boxplots display the 100 WAIC values that correspond to model fit for each synthetic data set. Note that smaller WAIC values indicate a better fit.

The primary purpose of Figure S.5 is to compare model fit from the spatio-temporal partition model we develop to that from the linear DDPM (model 5). It appears that all methods are competitive to the linear DDPM, which is particularly true with 10 time points. Thus, our dependent partition model accommodates temporal dependence more efficiently relative to the linear DDPM under this data generating scenario. Note that regardless of the number of time points, model 1b ($stRPM(\alpha, \nu_0, M)$ with $\nu_0 = 20$) appears to perform best. Surprisingly, $tRPM(\alpha, M)$ (model 4) is quite competitive, particularly with 10 time points. The conclusion here is that employing $stRPM(\alpha, \nu_0, M)$ to model partitions appears to accommodate spatio-temporal dependence even if there is no underlying partition structure.

From Figure S.6 we see that when partitions are generated independently, there is very little lost by employing the dependent joint model in terms of model fit (see top left panel for model 3 and 4). However, as spatial and/or temporal structure is introduced in the partition model, there are clear gains in terms of model fit when employing $tRPM(\boldsymbol{\alpha}, M)$ and/or $stRPM(\boldsymbol{\alpha}, \nu_0, M)$. From this simulation it seems that employing the $tRPM(\alpha, M)$

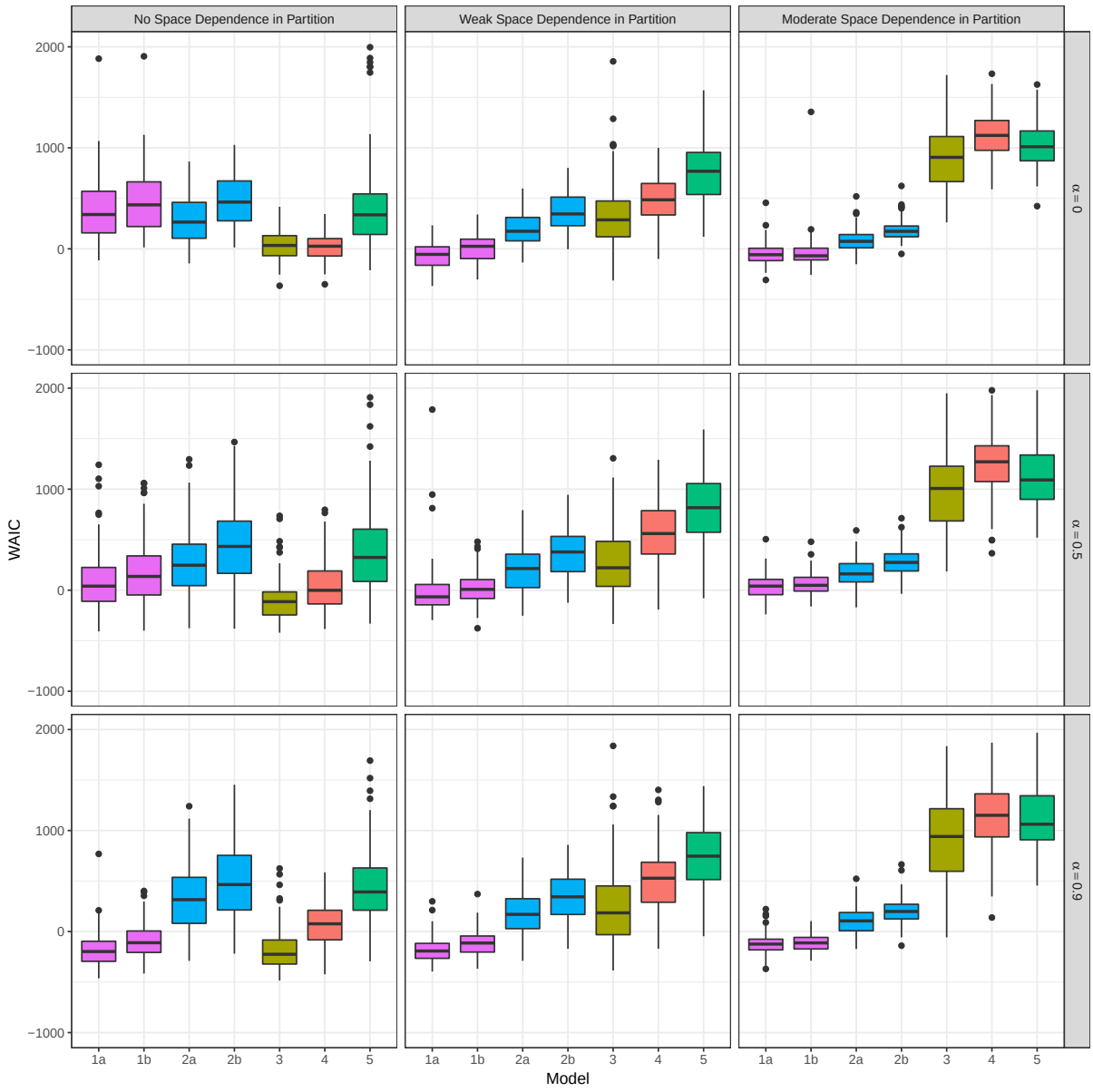

Figure S.6: Results from simulation study for the scenario in which partition structure is included in data generation process. Boxplots display the 100 WAIC values that correspond to model fit for each synthetic data generating scenario. Note that smaller indicates better fit.

regardless of the strength of temporal dependence among partitions is reasonable as there is minimal cost in terms of model fit even when partitions are generated independently. Finally, it appears that $stRPM(\boldsymbol{\alpha}, \nu_0, M)$ performed best.

D Data Applications

In this section, we detail an additional application in the field of education.

D.1 SIMCE Data Application

Incorporating spatio-temporal structure in education studies has been explored (e.g., Cepeda-Cuervo & Núñez-Antón 2013, Fotheringham et al. 2001). In school assessment and effectiveness studies, temporal persistence in school performance is of principal interest. It seems likely that school performance from year-to-year is relatively stable except in circumstances where a school undergoes many changes in personnel (faculty and students) or curriculum from one year to the next. In addition it seems reasonable that geographic location plays a role in school performance, particularly if communities are segregated socio-economically which happens to be the case in metropolitan area of Santiago, Chile. For these reasons we fit model (**??**) to these data as well.

In order to formally assess both national and school level education effectiveness in Chile, the Chilean national learning outcome assessment system (Sistema de Medición de Calidad de la Educación, SIMCE) was created to, among other things, administer standardized tests to education institutions in Chile. Each year a standardized test in mathematics and language is administered to 4th grade students. We were granted access to 7 years of data (2005-2011) where the longitude and latitude of most schools were recorded.

For the SIMCE data in addition to the 16 models fit to the PM_{10} data, we also considered an alternative to employing the sPPM at each time period which is more computationally efficient. The alternative approach models only ρ_1 with an $sPPM(\nu_0, M)$ (equation (**??**) of the main document) and the remaining $T-1$ partitions with an $tRPM(\boldsymbol{\alpha}, M)$ (equation

(**??**) of the main document) with a $CRP(M)$ EPPF. In this formulation, the strength of α_t would be the only mechanism by which the spatial structure found in ρ_1.

We employed the same prior parameter values here as in Section **??** of the main document, except we set $A_\sigma = 15$ and $\nu_0 = 2$ to account for the higher variability present in the SIMCE data. Each of the 24 models were fit to the SIMCE data by collecting 1000 MCMC draws after discarding the first 5000 as burn-in and thinning by 5. The LPML and WAIC results can be found in Table S.2.

Similar to the PM_{10} analysis, the best performing model in terms of WAIC includes spatio-temporal dependence in the partition model, temporal dependence among the atoms, and temporal dependence in the likelihood. The best performing model in terms of LPML assumed atoms are *iid*. Notice further, that generally speaking, incorporating temporal dependence in the model for $(\rho_1, \ldots, \rho_7)$ improves model fit. It appears that there is a cost in model fit associated with employing the $sPPM(\nu_0, M)$ at the first time period and the $CRP(M)$ for subsequent time periods in terms of model fit. However, the cost is not exorbitant relative to extraordinary computation gains (12 hours for model that includes space at time 1 versus 6 days for the model that includes space at each time point). To see how estimated partitions from the two models (that which includes space in the first time point versus that which includes space at each time point) change over time, we provide Figure S.7. Notice that there is a change in dependence from time period 2 and 3 and that the similarity between partitions decays when including space at each time point.

References

Caron, F., Davy, M. & Doucet, A. (2007), Generalized polya urn for time-varying dirichlet process mixtures, *in* 'Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence', UAI'07, AUAI Press, Arlington, Virginia, United States, pp. 33–40.

URL: *http://dl.acm.org/citation.cfm?id=3020488.3020493*

Cepeda-Cuervo, E. & Núñez-Antón, V. (2013), 'Spatial double generalized beta regression

Table S.2: Results of fitting 24 models to the SIMCE data. The bold font identifies the models that produced the best LPML and WAIC values. Higher values for LPML indicate better fit while lower values for WAIC indicate better fit.

			Space					
			No		Yes			
Temporal Dependence In					Each Time		First Time	
Partition	Likelihood	Atoms	LPML	WAIC	LPML	WAIC	LPML	WAIC
No	No	No	-34094	62963	-33543	62416	-34054	62960
No	No	Yes	-34040	62693	-33558	62577	-34044	63043
No	Yes	No	-31214	60087	-30701	60400	-31129	59349
No	Yes	Yes	-31241	59572	-30712	60944	-31115	59686
Yes	No	No	-32457	64835	-30760	61045	-31198	61516
Yes	No	Yes	-31007	61948	-31180	61348	-32690	64578
Yes	Yes	No	-30390	60340	**-29936**	58122	-30573	60132
Yes	Yes	Yes	-30378	60314	-30959	**57834**	-30331	59544

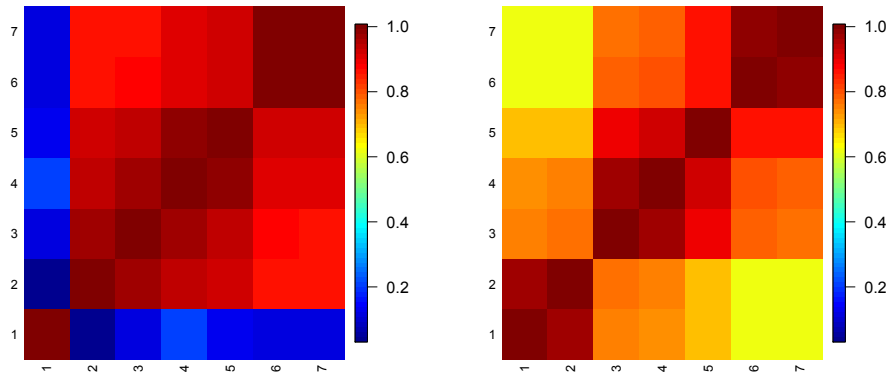

Figure S.7: Each figure is a summary of the lagged *ARI* value corresponding to models that include space in different ways. The left plot corresponds to model that includes space in partition model only at time period 1. The right plot corresponds to model that includes space in partition model at each of the seven time periods.

models: Extensions and application to study quality of education in colombia', *Journal of Educational and Behavioral Statistics* **38**, 604–628.

De Iorio, M., Johnson, W., Müller, P. & Rosner, G. (2009), 'Bayesian nonparametric nonproportional hazards survival modeling', *Biometrics* **65**(3), 762–771.

Fotheringham, A. S., Charlton, M. E. & Brunsdon, C. (2001), 'Spatial variations in school performance: a local analysis using geographically weighted regression', *Geographical & Environmental Modelling* **5**, 43–66.

MacEachern, S. N. (2000), Dependent dirichlet processes, Technical report, Ohio State University.

Müller, P., Quintana, F. A., Jara, A. & Hanson, T., eds (2015), *Bayesian Nonparametric Data Analysis*, 1 edn, Springer International Publishing, Switzerland.

Müller, P., Quintana, F. & Rosner, G. L. (2011), 'A product partition model with regression on covariates', *Journal of Computational and Graphical Statistics* **20**(1), 260–277.

Neal, R. M. (2000), 'Markov chain sampling methods for dirichlet process mixture models', *Journal of Computational and Graphical Statistics* **9**, 249–265.

Padoan, S. A. & Bevilacqua, M. (2015), 'Analysis of random fields using CompRandFld', *Journal of Statistical Software* **63**(9), 1–27.
URL: *http://www.jstatsoft.org/v63/i09/*

Page, G. L. & Quintana, F. A. (2016), 'Spatial product partition models', *Bayesian Analysis* **11**, 265–298.

Quintana, F. A., Müller, P., Jara, A. & MacEachern, S. N. (2020), 'The dependent dirichlet process and related models'. arXiv:2007.06129v1.

R Core Team (2020), *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria.
URL: *https://www.R-project.org/*