New Report

hopefully written better for you ☺

Here now we deal with understanding the DRPM package. It seems that there is just one function for fitting a model: drpm_fit.
The problem is understanding the meaning of the input parameters, from the paper.

y	An m x T numeric matrix containing the response values measured over time.
s_coords	Two-column matrix containing spatial locations (i.e., longitude and lattitude).
M	Parameter related to Dirichlet process scale or dispersion parameter.
initial_partition	The initial partition that is illicited from an expert.
starting_alpha	Initial value for alpha. If alpha_0 = TRUE (see below), then alpha is fixed to this value.
unit_specific_alpha	Logical that indicates if a unique alpha is to be included for each unit.
time_specific_alpha	Logical that indicates if a unique alpha is to be included for each time point.
alpha_0	Logical where TRUE indicates alpha_t = starting_alpha (i.e., alpha is not updated)
eta1_0	Logical where TRUE indicates eta1 = 0 (i.e., observations are conditionally independent).
phi1_0	Logical where TRUE indicates phi1 = 0 (i.e., iid model for atoms)
modelPriors	Vector containing the following hierarchical model prior values in order

- m0 - mean of phi0
- s20 - variance of phi0
- A - upper bound on sigma
- At - upper bound on tau
- Al - upper bound on lambda
- be - scale prior for xi

```
drpm_fit(y,
         s_coords=NULL,
         M=1,
         initial_partition = NULL,
         starting_alpha = 0.5,
         unit_specific_alpha = FALSE,
         time_specific_alpha = FALSE,
         alpha_0=FALSE,
         eta1_0=FALSE,
         phi1_0=FALSE,
         modelPriors=c(0,100^2,1,1,1,1),
         alphaPriors=rbind(c(1,1)),
         simpleModel = 0,
         theta_tau2 =c(0, 2),
         SpatialCohesion=4,
         cParms=c(0, 1, 2, 1),
         mh=c(0.5, 1, 0.1, 0.1, 0.1),
         verbose=FALSE,
         draws=1100,burn=100,thin=1)
```

next page for better explanation ☺

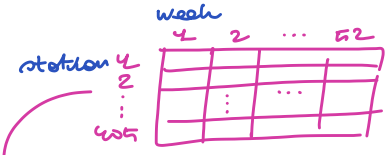
alphaPriors	Matrix containing beta prior values for the alpha parameters. The columns correspond to the 1st and 2nd shape beta shape parameters. The number of rows corresponds to the number of units. If no initial partition is supplied then only the first row is used.
simpleModel	If 1 then the simple model in Sally's simulation is employed.
theta_tau2	Only used if simpleModel=1. This is a two-dimensional vector with the first entry corresponding to the value that is fixed to theta and the second to tau2
SpatialCohesion	Scalar indicating what cohesion to use if spatial coordinates are provided

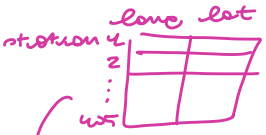
- 3 - auxiliary similarity
- 4 - double dipper similarity

cParms	Vector containing spatial cohesion parameter values (see below for more details)

- mu0 - center of NNIG
- k0 - scale parameter of gaussian part of NNIG
- v0 - degrees of freedom for inverse-gamma part of NNIG
- L0 - scale parameter for inverse-gamma part of NNIG

mh	Vector of gaussian standard deviations for metropolis updates of sigma2, tau, lambda, eta1, phi1.
draws	Number of MCMC samples to collect
burn	Number of the MCMC samples discarded in the burn-in phase of the sampler
thin	The amount of thinning desired for the chain
verbose	Logical indicating if MCMC progression should be print to screen





y	An m x T numeric matrix containing the response values measured over time.
s_coords	Two-column matrix containing spatial locations (i.e., longitude and lattitude).
M	Parameter related to Dirichlet process scale or dispersion parameter.
initial_partition	The initial partition that is illicited from an expert.
starting_alpha	Initial value for alpha. If alpha_0 = TRUE (see below), then alpha is fixed to this value.
unit_specific_alpha	Logical that indicates if a unique alpha is to be included for each unit.
time_specific_alpha	Logical that indicates if a unique alpha is to be included for each time point.
alpha_0	Logical where TRUE indicates alpha_t = starting_alpha (i.e., alpha is not updated)
eta1_0	Logical where TRUE indicates eta1 = 0 (i.e., observations are conditionally independent).
phi1_0	Logical where TRUE indicates phi1 = 0 (i.e., iid model for atoms)
modelPriors	Vector containing the following hierarchical model prior values in order

- m0 - mean of phi0
- s20 - variance of phi0
- A - upper bound on sigma
- At - upper bound on tau
- Al - upper bound on lambda
- be - scale prior for xi

```
drpm_fit(y,
         s_coords=NULL,
         M=1,
         initial_partition = NULL,
         starting_alpha = 0.5,
         unit_specific_alpha = FALSE,
         time_specific_alpha = FALSE,
         alpha_0=FALSE,
         eta1_0=FALSE,
         phi1_0=FALSE,
         modelPriors=c(0,100^2,1,1,1,1),
```

```
alphaPriors=rbind(c(1,1)),
simpleModel = 0,
theta_tau2 =c(0, 2),
SpatialCohesion=4,
cParms=c(0, 1, 2, 1),
mh=c(0.5, 1, 0.1, 0.1, 0.1),
verbose=FALSE,
draws=1100,burn=100,thin=1)
```

alphaPriors	Matrix containing beta prior values for the alpha parameters. The columns correspond to the 1st and 2nd shape beta shape parameters. The number of rows corresponds to the number of units. If no initial partition is supplied then only the first row is used.
simpleModel	If 1 then the simple model in Sally's simulation is employed.
theta_tau2	Only used if simpleModel=1. This is a two-dimensional vector with the first entry corresponding to the value that is fixed to theta and the second to tau2
SpatialCohesion	Scalar indicating what cohesion to use if spatial coordinates are provided • 3 - auxiliary similarity • 4 - double dipper similarity
cParms	Vector containing spatial cohesion parameter values (see below for more details) • mu0 - center of NNIG • k0 - scale parameter of gaussian part of NNIG • v0 - degrees of freedom for inverse-gamma part of NNIG • L0 - scale parameter for inverse-gamma part of NNIG
mh	Vector of gaussian standard deviations for metropolis updates of sigma2, tau, lambda, eta1, phi1.
draws	Number of MCMC samples to collect
burn	Number of the MCMC samples discarded in the burn-in phase of the sampler
thin	The amount of thinning desired for the chain
verbose	Logical indicating if MCMC progression should be print to screen

$$C_3(S_h, \boldsymbol{s}^\star_h) = M \times \Gamma(|S_h|) \times \int \prod_{i \in S_h} q(\boldsymbol{s}_i|\boldsymbol{\xi}_h) q(\boldsymbol{\xi}_h) d\boldsymbol{\xi}_h. \tag{3.4}$$

In Bayesian modeling $\int \prod_{i \in S_h} q(\boldsymbol{s}_i|\boldsymbol{\xi}_h) q(\boldsymbol{\xi}_h) d\boldsymbol{\xi}_h$ is often called the marginal likelihood or prior predictive distribution and is used to measure the similarity among the locations belonging to cluster h. Therefore, C_3 favors partitioned location vectors ($\boldsymbol{s}^\star$) that produce large marginal likelihood values. To simplify evaluating C_3 and retain coherence across sample sizes, $q(\boldsymbol{s}|\boldsymbol{\xi})$ and $q(\boldsymbol{\xi})$ are specified to form a conjugate probability model. We emphasize however that we are not assuming the $\boldsymbol{s}_i$'s to be random, we are simply employing the conjugate model as a means to measure spatial proximity and encourage co-clustering of locations that are near each other. Both areal and point referenced data can be considered when C_3 is employed, all that is required is specifying appropriate $q(\boldsymbol{s}|\boldsymbol{\xi})$ and $q(\boldsymbol{\xi})$. For example, if point referenced data are available, a conjugate Gaussian/Gaussian-Inverse-Wishart model would be appropriate. In this case $\boldsymbol{\xi} = (\boldsymbol{m}, \boldsymbol{V})$ would denote a mean and covariance, $q(\boldsymbol{s}|\boldsymbol{\xi}) = N(\boldsymbol{s}|\boldsymbol{m}, \boldsymbol{V})$ a bivariate Gaussian density and $q(\boldsymbol{\xi}) = NIW(\boldsymbol{m}, \boldsymbol{V}|\boldsymbol{\mu}_0, \kappa_0, \nu_0, \boldsymbol{\Lambda}_0)$ a bivariate Normal-Inverse-Wishart density. For areal data a conjugate multinomial/Dirichlet model could be utilized. In what follows we focus on point reference case and will occasionally refer to C_3 as the auxiliary cohesion. Finally, as in the previous two cohesions, $M \times \Gamma(|S_h|)$ is included to avoid creating many singleton clusters.

The fourth and final cohesion that we consider is similar to what Quintana et al. (In press) call a "double dipper" cohesion. It has the same form as C_3, but instead of employing a prior predictive conjugate model, a posterior predictive conjugate model is used. Therefore C_4 has the following form

$$C_4(S_h, \boldsymbol{s}^\star_h) = M \times \Gamma(|S_h|) \times \int \prod_{i \in S_h} q(\boldsymbol{s}_i|\boldsymbol{\xi}_h) q(\boldsymbol{\xi}_h|\boldsymbol{s}^\star_h) d\boldsymbol{\xi}_h. \tag{3.5}$$

Since the posterior predictive is typically more peaked than the prior predictive, C_4 puts more weight on partitions that are local. Once again both areal and point referenced data are possible, but in what follows we focus on point-referenced and use the following conjugate model: $N_2(\boldsymbol{s}_i|\boldsymbol{m}_h, \boldsymbol{V}_h) NIW(\boldsymbol{m}_h, \boldsymbol{V}_h|\boldsymbol{s}^\star_h)$.

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

$$\Pr(\rho_t|v_0, M) \propto \prod_{j=1}^{k_t} c(S_{jt}|M) g(\boldsymbol{s}^\star_{jt}|v_0). \tag{11}$$

Here, $c(\cdot|M) \geq 0$ is called the cohesion and is a set function that produces cluster weights a priori. We consider the cohesion $c(S_{jt}|M) = M \times (|S_{jt}| - 1)!$ as it has connections with the CRP making this version of the sPPM a type of spatially re-weighted CRP. The similarity function $g(\cdot|v_0)$ is a set function parametrized by v_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

$$g(\boldsymbol{s}^\star_{jt}|v_0) = \int \prod_{i \in S_{jt}} N(\boldsymbol{s}_i|\boldsymbol{m}, \boldsymbol{V}) NIW(\boldsymbol{m}, \boldsymbol{V}|\boldsymbol{0}, 1, v_0, \boldsymbol{I})\, d\boldsymbol{m}\, d\boldsymbol{V}, \tag{12}$$

JOURNAL OF COMPUTATIONAL AND GRAPHICAL STATISTICS
2022, VOL. 31, NO. 2, 614–627
https://doi.org/10.1080/10618600.2021.1987255

Taylor & Francis
Taylor & Francis Group

Dependent Modeling of Temporal Sequences of Random Partitions

Garritt L. Page[a,b], Fernando A. Quintana[c], and David B. Dahl[a]

[a]Department of Statistics, Brigham Young University, Provo, UT; [b]BCAM—Basque Center of Applied Mathematics, Bilbao, Spain; [c]Departamento de Estadística, Pontificia Universidad Católica de Chile, Millennium Nucleus Center for the Discovery of Structures in Complex Data, Santiago, Chile

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 dataset that exhibits spatio-temporal dependence. Supplemental files for this article are available online.

ARTICLE HISTORY
Received July 2020
Revised September 2021

KEYWORDS
Bayesian nonparametrics;
Correlated partitions;
Hierarchical Bayes modeling;
Spatio-temporal clustering

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 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 nonexhaustive list of BNP methods that temporally correlate a sequence of random probability measures include Nieto-Barajas et al. (2012), Antoniano-Villalobos and Walker (2016), Gutiérrez, Mena, and Ruggiero (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, Davy, and Doucet (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 article. 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,

CONTACT Garritt L. Page page@stat.byu.edu Department of Statistics, Brigham Young University, Provo, UT and BCAM - Basque Center of Applied Mathematics, Bilbao, Spain.

Supplementary materials for this article are available online. Please go to www.tandfonline.com/r/JCGS.

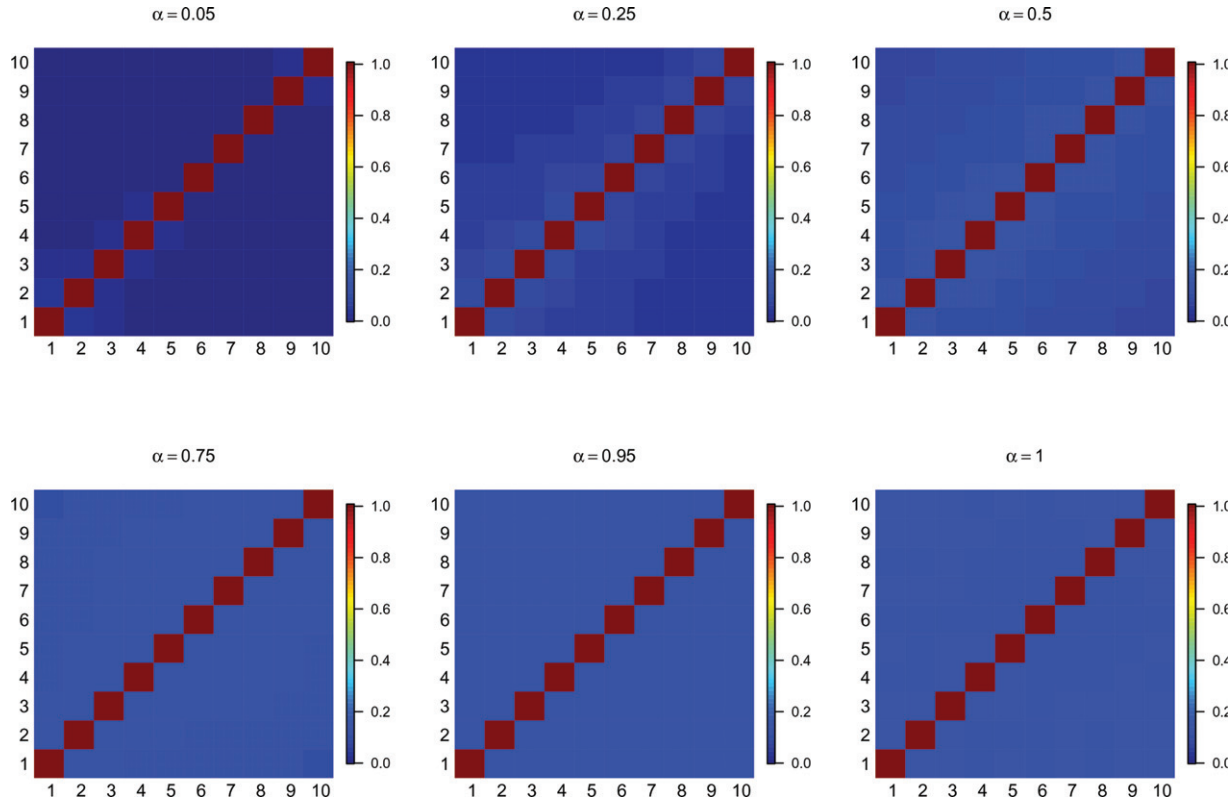


Figure 1. For various values of the temporal dependence parameter α, these plots show the lagged ARI values using the method of Caron et al. (2017) based on concentration parameter $M = 0.5$, discount parameter set to zero, and 10,000 Monte Carlo samples.

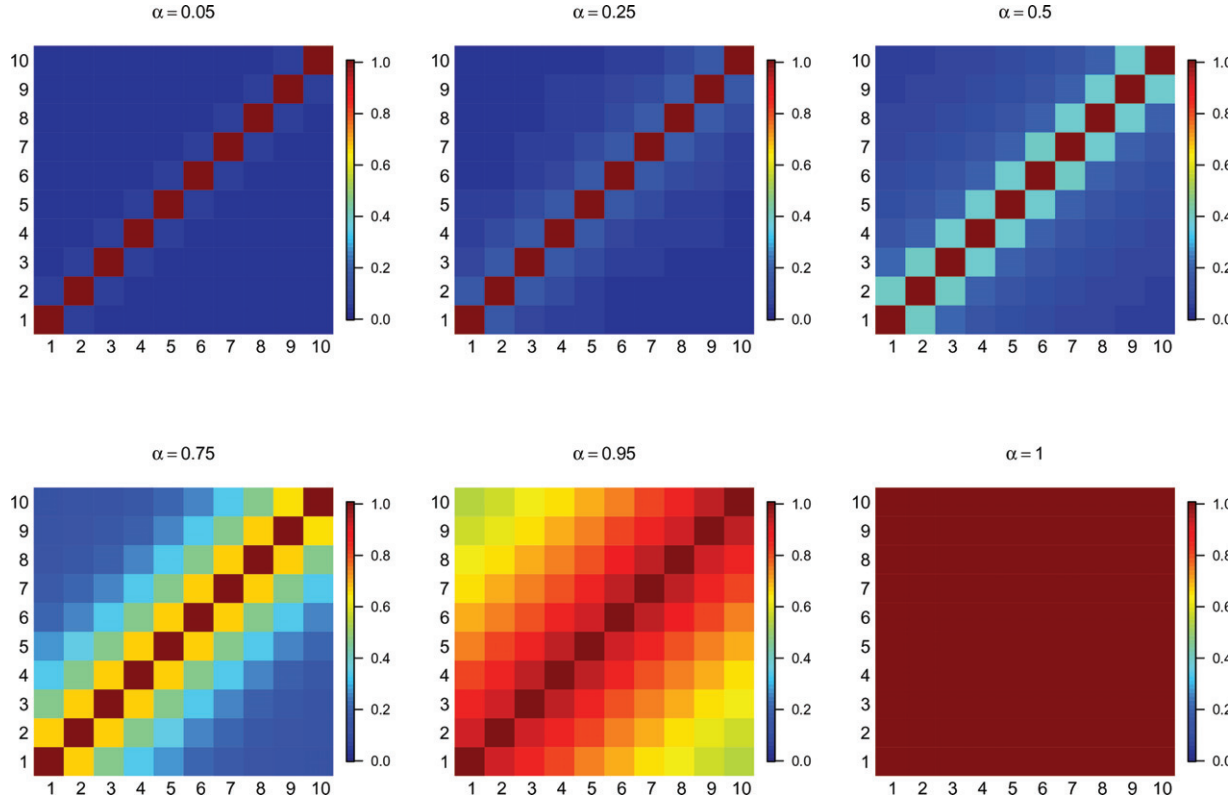


Figure 2. Lagged ARI values based on 10,000 Monte Carlo samples using the method developed in this paper. Partitions show natural and intuitive temporal dependence as lagged time increases and as the temporal dependence parameter α increases.

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, Walker, and Petrone (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

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

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 frequencies $n_{11}, \ldots, n_{1k_1}$. 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

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

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

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

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, that is, all experimental units are allocated to clusters during the first time period. We then assume that $\gamma_{it} \overset{\text{ind}}{\sim} \text{Ber}(\alpha_t)$. 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

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

We describe $\Pr(\rho_t|\boldsymbol{\gamma}_t, \rho_{t-1})$ 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 $\boldsymbol{\gamma}_t$, that is, those items i such that $\gamma_{it} = 0$ 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 $\Pr(\rho_t|\boldsymbol{\gamma}_t, \rho_{t-1})$, 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, $\Pr(\rho_t|\boldsymbol{\gamma}_t, \rho_{t-1})$ is a random partition distribution whose support is P_{C_t} so that

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

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 proposition 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 $\lambda \in P$,

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

where $P^{\otimes m} = P \times P \times \cdots \times P$, P a collection of all partitions of m units, $\Gamma^{\otimes m} = \Gamma \times \Gamma \times \cdots \times \Gamma$, 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, Davy, and Doucet (2007); Caron et al. (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 as

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

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(α, M), we also consider the model proposed in Caron, Davy, and Doucet (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, Davy, and Doucet (2007). We denote partitions that follow the model of Caron, Davy, and Doucet (2007) as $\rho_1, \rho_2 \sim CAR(\alpha, M)$. For both our method and CAR(α, M), the case that $\alpha = 0$ leads to $\rho_1, \rho_2 \overset{\text{iid}}{\sim}$ 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(α, 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(α, 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(α, M) then

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

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 $\rho_1, \rho_2 \overset{\text{iid}}{\sim}$ CRP(M).

Corollary 1. If $\rho_1, \rho_2 \overset{\text{iid}}{\sim} \text{CRP}(M)$ 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(α, M) model, $\lim_{\alpha \to 0^+} \varphi_{12} = \frac{(1+M^2)}{(1+M)^2}$, that is, it agrees with $E\left[R(\rho_1, \rho_2)\right]$ under the iid case. The same holds as $\alpha \to 0^+$ under the CAR(α, M) model. Furthermore, for the tRPM(α, M), we get the appealing result that $\lim_{\alpha \to 1^+} \varphi_{12} = 1$, but the same limit under the CAR(α, M) is a number strictly less than 1 for any $M > 0$. This reveals that the closeness between partitions under the proposed tRPM(α, 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(α, 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

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

where again, Γ is the collection of all possible binary 3-tuples and operate under $\rho_1 \sim \text{CRP}(M)$. The conditional probabilities are provided in Table 1, where we set $M = 1$ for simplicity. From Table 1, notice that $\Pr(\rho_2|\rho_1)$ 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 $\rho_2 = \{\{1,2\},\{3\}\}$ 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

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

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 hyperparameters. 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, for example, 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 B 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

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

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

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

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 $\mathfrak{R}_t^{(-i)} = \mathfrak{R}_t \backslash \{i\}$ 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 $\Pr(\gamma_{it} = 1|-)$, is

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

which results in

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

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

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

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 B.1 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 provided in (9) where

$$\Pr(c_{it} = h|-) \propto \begin{cases} N(Y_{it}|\mu^\star_{c_{it}=h,t}, \sigma^{2\star}_{c_{it}=h,t})\Pr(c_{it} = h)\mathrm{I}[\rho_{t:c_{it}=h}^{\mathfrak{R}_{t+1}} = \rho_{t+1}^{\mathfrak{R}^{t+1}}] \text{ for } h = 1, \ldots, k_t^{-i}, \\ N(Y_{it}|\mu^\star_{new_h,t}, \sigma^{2\star}_{new_h,t})\Pr(c_{it} = h)\mathrm{I}[\rho_{t:c_{it}=h}^{\mathfrak{R}_{t+1}} = \rho_{t+1}^{\mathfrak{R}^{t+1}}] \text{ for } h = k_t^{-i} + 1, \end{cases} \tag{9}$$

where $\mu^\star_{\text{new}_h,t}$ along with $\sigma^{2\star}_{\text{new}_h,t}$ 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 $N(\cdot|m, s^2)$ 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 B.2 of the online supplementary material for more detail.

In our experience the MCMC algorithm described here and in Section B 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 C 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 100 datasets with 50 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 5000 as burn-in and thinning by 5 (resulting in 1000 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, Johnson, and Müller 2020) with the binder loss function (Binder 1978). To measure similarity between partitions, we employed the ARI (Rand 1971; Hubert and Arabie 1985) and we used WAIC (Gelman, Hwang, and Vehtari 2014) to measure model fit.

Table 2 displays the lagged 1 and 4 ARI as a function of α. 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

Table 2. ARI when comparing $\hat{\rho}_1$ to $\hat{\rho}_2$ and $\hat{\rho}_1$ to $\hat{\rho}_5$.

			Coverage		WAIC	
	$\text{ARI}(\hat{\rho}_1, \hat{\rho}_2)$	$\text{ARI}(\hat{\rho}_1, \hat{\rho}_5)$	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

NOTES: $\text{ARI}(\cdot, \cdot)$ denotes the adjusted Rand index as a function of two partitions. Coverage rates for α and μ_{it} and model fit metrics for $\text{tRPM}(\alpha, M)$ and $\text{CRP}(M)$. These values are averaged over the 100 generated data sets. The values in parenthesis are Monte Carlo standard errors. Smaller values of WAIC indicate better fit.

1 ARI for $\alpha = 0.1$ 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. Since ARI isn't necessarily a monotone function of α, care must be taken when interpreting α as a dependence parameter based on ARI.

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 $\text{tRPM}(\alpha, M)$ as the RPM in model (5) relative to $\rho_t \overset{\text{iid}}{\sim} \text{CRP}(M)$, 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 $\text{tRPM}(\alpha, M)$, with gains becoming substantial as α increases.

The upshot from this simulation study is that lagged partition estimates when employing $\text{tRPM}(\alpha, M)$ display intuitive behavior in that similarity between partition estimates decreases as lag increases. In addition, employing the $\text{tRPM}(\alpha, M)$ 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 $\text{tRPM}(\alpha, M)$ 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.

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 S.1 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 C.2 of the supplementary material.
2. A linear dependent Dirichlet process (lddp) described in Quintana et al. (2020) and chapter 4.4.2 of Müller et al.

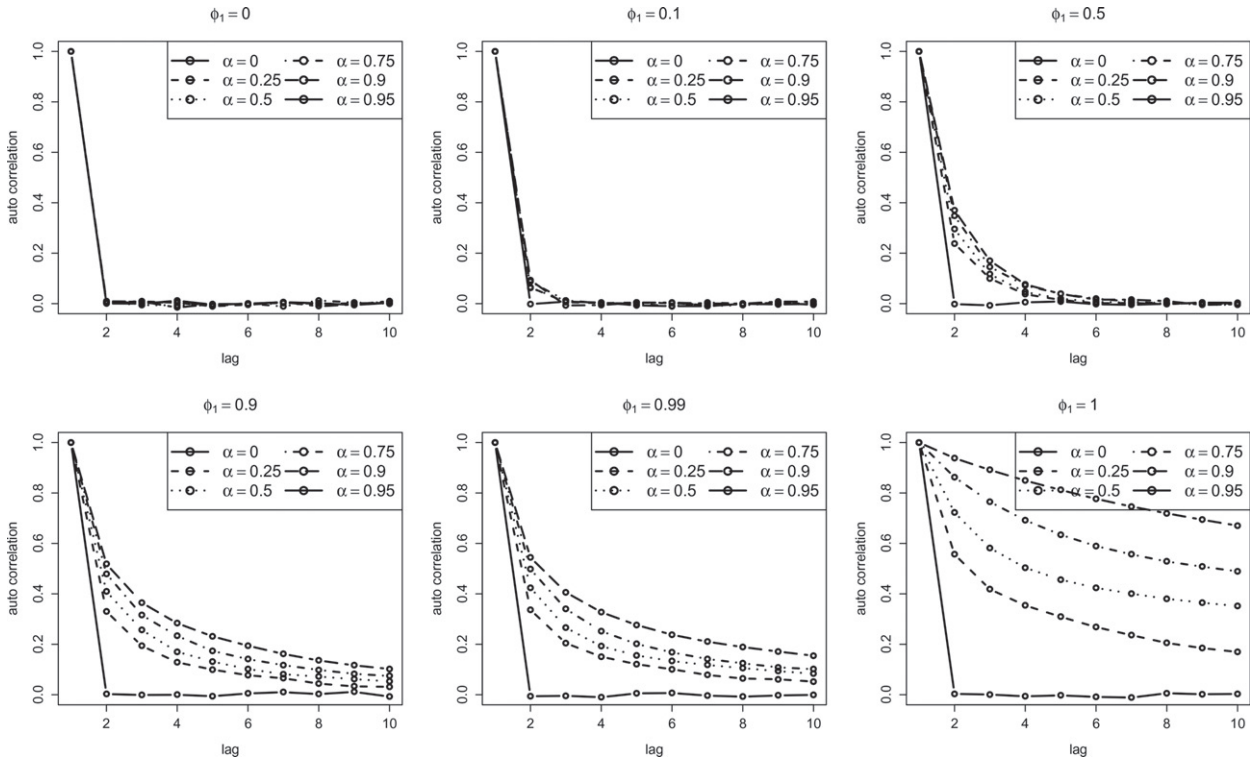


Figure 3. Lagged auto-correlations among the $(Y_{i1}, \ldots, Y_{iT})$ when modeling $\mu_{jt}^{\star}$ with an AR(1) type structure.

(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 C.2 of the supplementary material.

3. A Griffiths-Milne dependent Dirichlet process (gmddp) mixture. This model was fit using the `BNPmix` package in R (Corradin, Canale, and Nipoti 2020). See Corradin, Canale, and Nipoti (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, Davy, and Doucet (2007). The exact details of this model are also provided in Section C.2 of the 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, Davy, and Doucet (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 C.2 of the 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 datasets, 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 done separately for each $t = 1, \ldots, T$ and then averaged across time and MCMC samples. From the left plot, our method (drpm) 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 D.1 of the 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

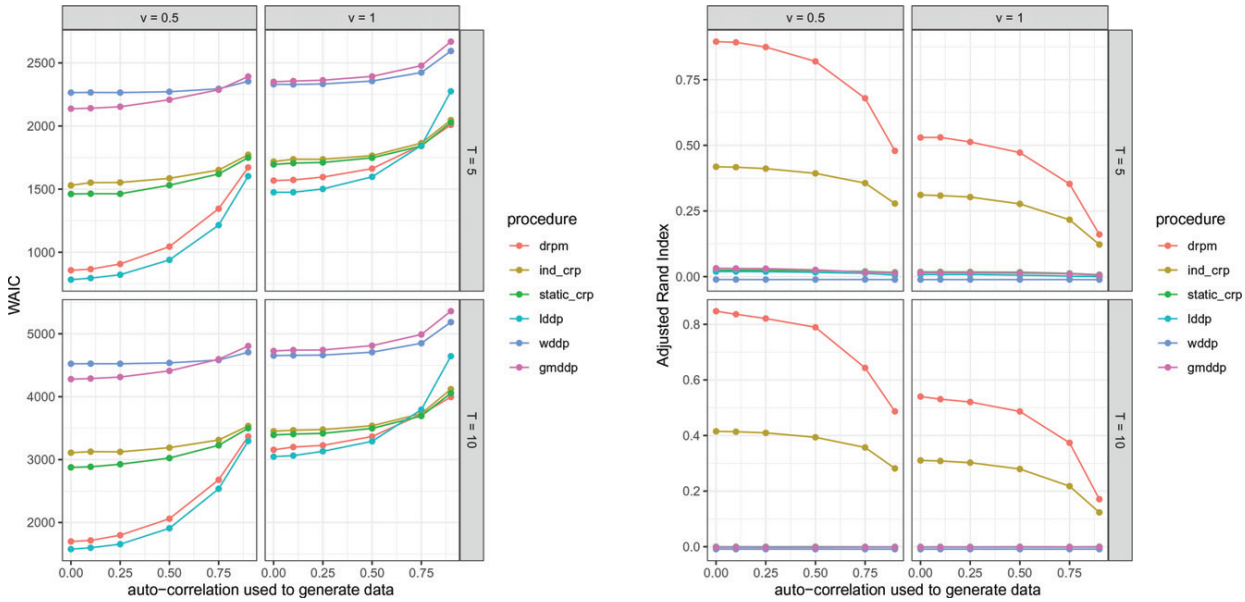


Figure 4. Results from simulation study with data that contains AR(1)-type temporal correlation. The left plot corresponds to results for the model fit metric WAIC and the right plot displays results associated with ARI.

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 Equation (10).

$$
\begin{aligned}
Y_{it}|Y_{it-1}, \boldsymbol{\mu}_t^\star, \boldsymbol{\sigma}_t^{2\star}, \boldsymbol{\eta}, \boldsymbol{c}_t &\overset{\text{ind}}{\sim} N(\mu^\star_{c_{it}t} + \eta_{1i} Y_{it-1}, \sigma^{2\star}_{c_{it}t}(1-\eta_{1i}^2)),\\
Y_{i1} &\overset{\text{ind}}{\sim} N(\mu^\star_{c_{i1}1}, \sigma^{2\star}_{c_{i1}1}),\\
\xi_i = \text{Logit}(0.5(\eta_{1i}+1)) &\overset{\text{iid}}{\sim} \text{Laplace}(a,b),\\
(\mu^\star_{jt}, \sigma^\star_{jt}) &\overset{\text{ind}}{\sim} N(\theta_t, \tau_t^2) \times \text{UN}(0, A_\sigma),\\
\theta_t|\theta_{t-1} &\overset{\text{ind}}{\sim} N((1-\phi_1)\phi_0 + \phi_1\theta_{t-1}, \lambda^2(1-\phi_1^2)),\\
(\theta_1, \tau_t) &\sim N(\phi_0, \lambda^2) \times \text{UN}(0, A_\tau),\\
(\phi_0, \phi_1, \lambda) &\sim N(m_0, s_0^2) \times \text{UN}(-1,1) \times \text{UN}(0, A_\lambda),\\
\{\boldsymbol{c}_t, \ldots, \boldsymbol{c}_T\} &\sim \text{tRPM}(\boldsymbol{\alpha}, M), \quad \text{with } \alpha_t \overset{\text{iid}}{\sim} \text{Beta}(a_\alpha, b_\alpha),
\end{aligned}
\tag{10}
$$

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 (10), obtained by setting $\phi_1 = 0$ and $\eta_{i1} = 0$ for all i.

4.1. Rural Background PM_{10} Data Application

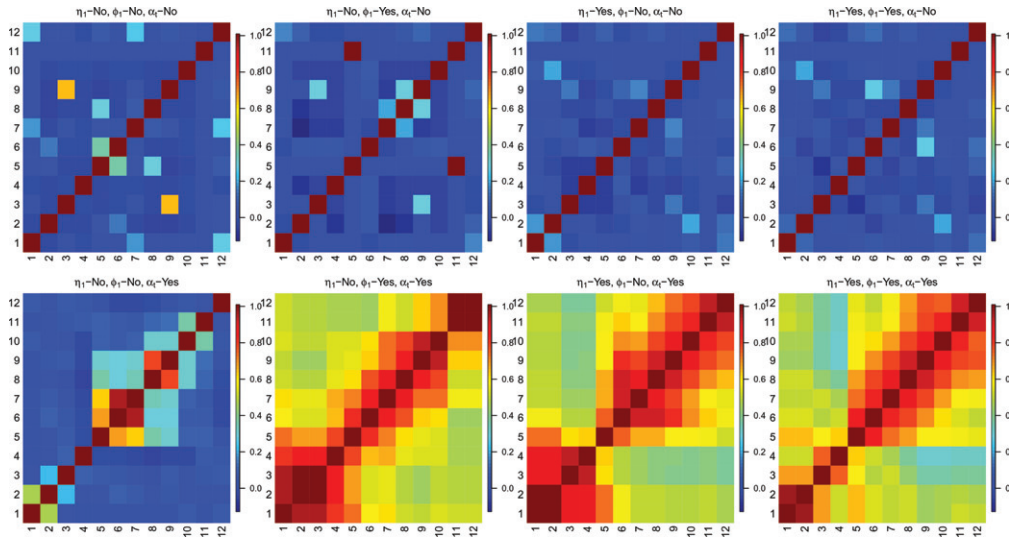
The rural background PM_{10} data is taken from the European air quality database. These data are comprised of the daily measurements of particulate matter with a diameter less than 10 μm from rural background stations in Germany and are publicly available in the `gstat` package (Gräler, Pebesma, and Heuvelink 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.

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

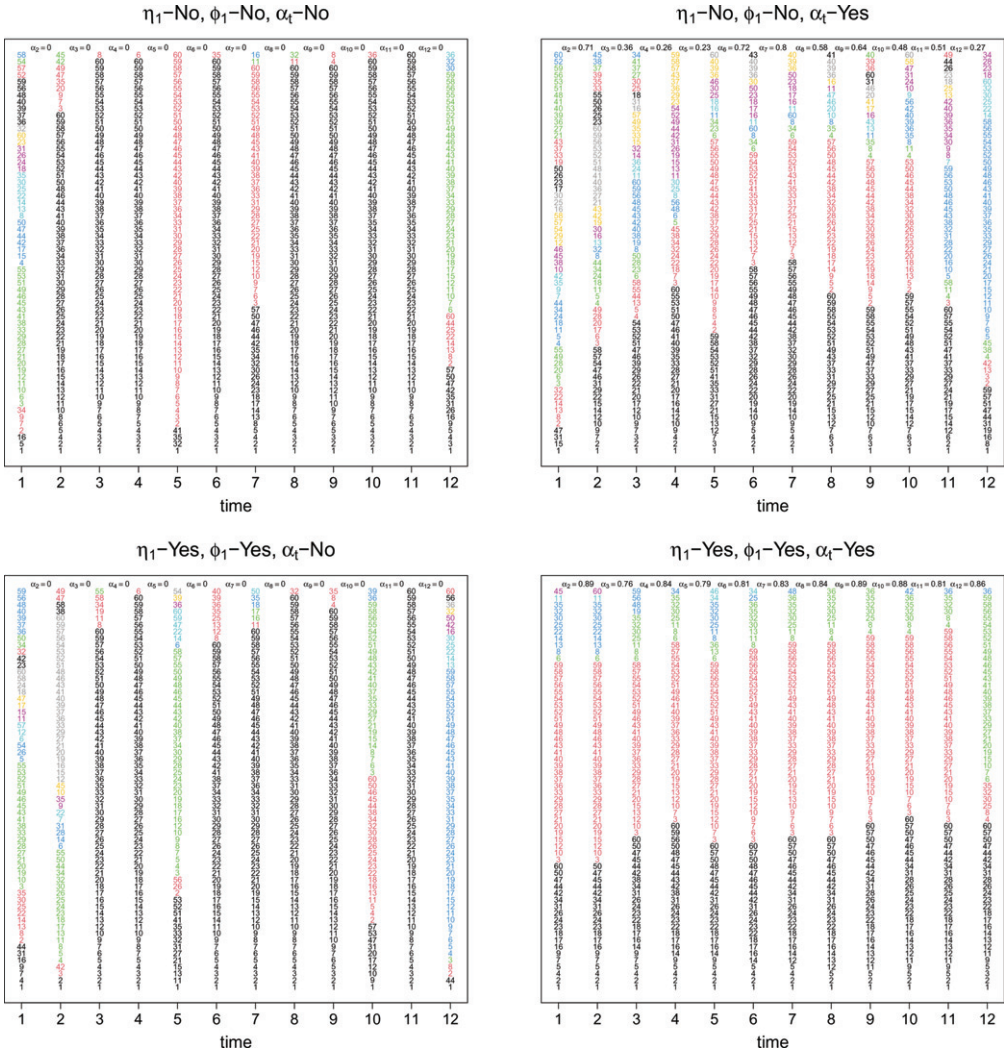
We fit the hierarchical model (10) to these data and consider all the possible special cases (i.e., $\eta_{1i} = 0$ or not, $\phi_1 = 0$ or not, $\alpha_t = 0$ or not). This resulted in 8 total models that were fit by collecting 1000 MCMC iterates after discarding the first 10,000 as burn-in and thinning 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 "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.



check on these code how we get these plots

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, Johnson, and Müller 2020) based on binder loss.



also these may be but we should use our own test which is better

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

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 third row 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

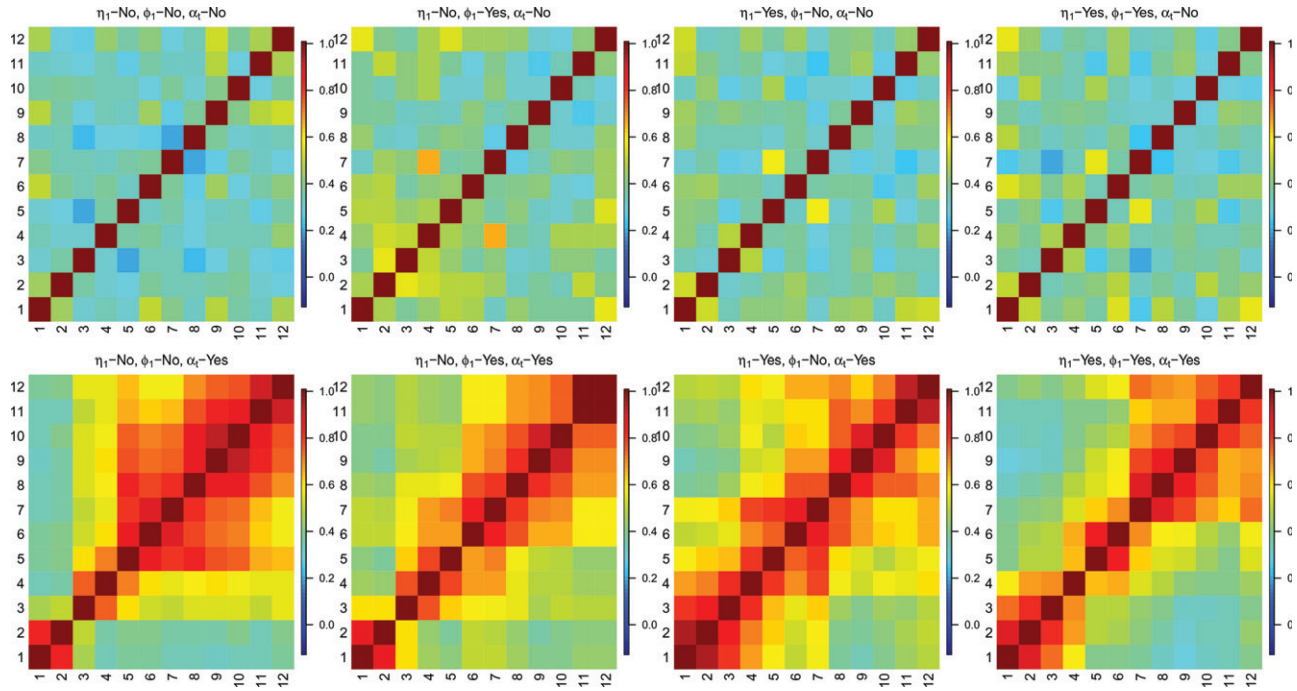


Figure 7. PM_{10} data. Each figure is a summary of the lagged ARI values corresponding to the 8 models that are detailed in Table 3 except that space is also included in the dependent random partition model. At each time point the partition was estimated using the `salso` function in the `salso` R package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

employing tRPM($\boldsymbol{\alpha}, M$) (second row of Figure 5) the partitions seem 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) 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 left employ $\rho_t \overset{\text{iid}}{\sim} \text{CRP}(M)$. 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 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 (10) 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

$$\Pr(\rho_t|\nu_0, M) \propto \prod_{j=1}^{k_t} c(S_{jt}|M) g(\boldsymbol{s}^{\star}_{jt}|\nu_0). \tag{11}$$

Here, $c(\cdot|M) \geq 0$ is called the cohesion and is a set function that produces cluster weights a priori. We consider the cohesion $c(S_{jt}|M) = M \times (|S_{jt}| - 1)!$ as it has connections with the CRP making this version of the sPPM a type of spatially re-weighted CRP. The similarity function $g(\cdot|\nu_0)$ 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

$$g(\boldsymbol{s}^{\star}_{jt}|\nu_0) = \int \prod_{i \in S_{jt}} N(\boldsymbol{s}_i|\boldsymbol{m}, \boldsymbol{V}) NIW(\boldsymbol{m}, \boldsymbol{V}|\boldsymbol{0}, 1, \nu_0, \boldsymbol{I})\, d\boldsymbol{m}\, d\boldsymbol{V}, \tag{12}$$

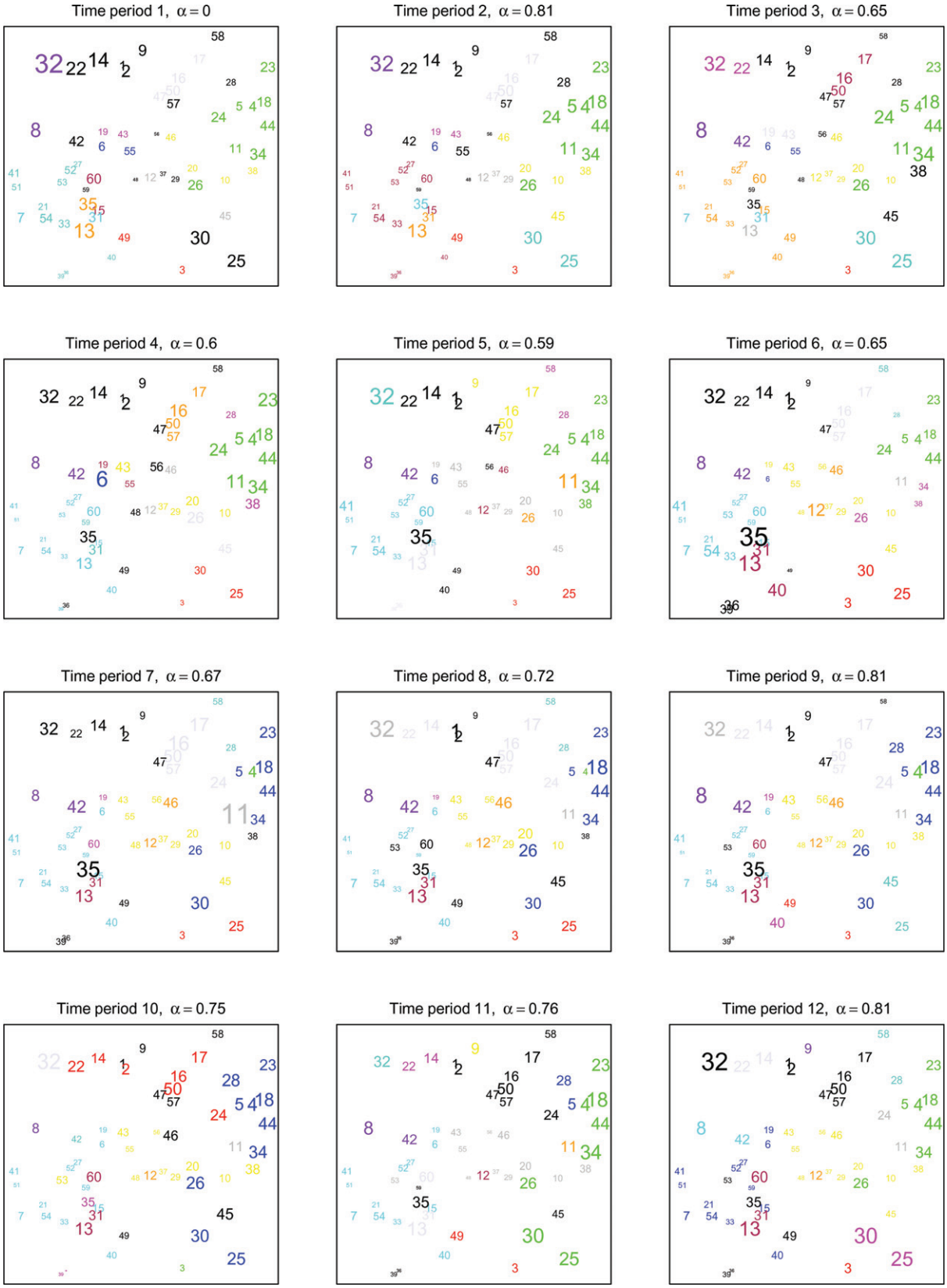


Figure 8. PM_{10} data. Graphical display for spatially referenced estimated partitions for each time point based on the model that produced best fit (space in the partition model and temporal dependence in all levels of the model). The size of a point is proportional to the PM_{10} measured at that 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, Johnson, and Müller 2020) based on binder loss.

where $N(\cdot|\boldsymbol{m}, \boldsymbol{V})$ denotes a bivariate normal density and $NIW(\cdot, \cdot|\mathbf{0}, 1, \nu_0, \boldsymbol{I})$ a normal-inverse-Wishart density with mean $\mathbf{0}$, 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, Quintana, and Rosner (2011) and Quintana, Loschi, and Page (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 Equations (11) and (12).

As in the previous section, we fit model (10) 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 1000 MCMC iterates after discarding the first 10,000 as burn-in and thinning by 40. Fitting each of 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 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 color depicts. 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.

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 ARI 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.

Acknowledgments

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.

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," *Journal of Time Series Analysis*, 37, 126–142. [614]

Binder, D. A. (1978), "Bayesian Cluster Analysis," *Biometrika*, 65, 31–38. [619]

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 Uncertainty in Artificial Intelligence* (UAI'07), Arlington, VA: AUAI Press. Available at: *http://dl.acm.org/citation.cfm?id=3020488.3020493*. [614,617,621]

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. Available at: *http://jmlr.org/papers/v18/10-231.html*. [614,615,617]

Corradin, R., Canale, A., and Nipoti, B. (2020), "BNPmix: Bayesian Nonparametric Mixture Models." Available at: *https://CRAN.R-project.org/package=BNPmix*. R package version 0.2.6. [621]

——— (2021), "BNPmix: An R Package for Bayesian Nonparametric Modelling Via Pitman–Yor Mixtures," *Journal of Statistical Software*. [621]

Dahl, D. B., Johnson, D. J., and Müller, P. (2020), *salso: Search Algorithms and Loss Functions for Bayesian Clustering*. Available at: *https://CRAN.R-project.org/package=salso*. R package version 0.2.5. [619,623,624,625]

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. [616]

De Iorio, M., Favaro, S., Guglielmi, A., and Ye, L. (2019), "Bayesian Nonparametric Temporal Dynamic Clustering Via Autoregressive Dirichlet Priors," arXiv:1910.10443. [614]

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. Available at: *https://doi.org/10.1080/01621459.2017.1328357*. [614]

Gelman, A., Hwang, J., and Vehtari, A. (2014), "Understanding Predictive Information Criteria for Bayesian Models," *Statistics and Computing*, 24, 997–1016. [619]

General idea: we have n distinct locations $z_1, \ldots, z_n$ ($z_i = $ (lat, long) eg) paper 3 page 5

the goal is to define a model for the partitioning of z_i into $k_n(t)$ groups

so we define $\rho_n = \{S_1, \ldots, S_{k_n(t)}\}$ the clusters such that, $i \in S_h \iff$ location z_i is in cluster h.
Or equivalently let $c_1, \ldots, c_n$ where $c_i = h$ iff ecc.

(is my model for us at will become $C(S_h, t)$?)

Then the prior for ρ_n is

$$\pi_{\rho_n}(\rho) = \mathbb{P}(\rho = \rho_n) \propto \prod_{h=1}^{k_n} C(S_h, z_h^*)$$

(S_h will be like $\{1, \ldots, n\}$ subset)

(this is a cohesion function: measures how likely those indices of S_h would be clustered together a priori)

(this makes the clustering process dependent on location)

(the set of locations that go to S_h: like if $S_h = \{1, 3, 9\}$ then $z_h^* = \{z_1, z_3, z_9\}$)

we call this model/law as sPPM (spatial product partition model)

Old Part

Then the idea is to set to each obs y_i for $i = 1 \to n$ a law/likelihood which is cluster dependent

$$Y_i = Y(z_i) \mid \theta, c_i \overset{ind}{\sim} f(\theta^*_{c_i}) \quad \text{for } i = 1 \to n$$ paper 3 page 46

$$\theta^*_h \overset{iid}{\sim} G_0 \quad \text{for } h = 1 \to k_n$$

(we actually know are also $\rho_n(t)$ as $k_n = |\rho_n(t)|$)

$$\{c_i\}_{i=1}^n \sim \text{sPPM}$$

(for example $\theta^*_{c_i} = (\mu_{c_i}, \sigma^2_{c_i})$ and we set $f(\ldots) = \mathcal{N}(\ldots)$)

And we can include covariates with

let $y_i(t)$ the values observed (precious for us)
let $x_i(t)$ the covariate set there
let $\beta^*_1, \ldots, \beta^*_{k_n(t)}$ the cluster specific regression params

(the $\theta^*_{c_i}$ of before)

(wait, how can we have $k_n(t)$? a posteriori, we will see after inference? i.e., how will it work then? eg the $\# k_n$ varies, and not then the βs are time and the more "may be")

then we model

$$Y_i = Y(z_i) \mid x_i(t), c_i(t), \beta^*, \sigma^2 \overset{ind}{\sim} \mathcal{N}(x_i^T \beta^*_{c_i}, \sigma^2)$$

(is all this temporal extension that makes me worry)

Here there are some more examples of models. but note that the rules $\{c_i\}_{i=1}^n \sim$ sPPM, is equivalent to $\rho_n \sim$ sPPM, as for the ρ_n we can translate to get the c_i. paper 3 page 22

$\rho_n = \{S_1, \ldots, S_{k_n}\}$ eg (2 3 7)([illegible])(8)(6 9) $k_n = 4$

```
for i in 1:n
| c_i = which(i in S_j for j=1:k_n)
end
```

Ok, now, now we can understand better the model of our (in. files) most relevant paper, the paper 4. paper 4 pag 9

Here they also provide the time extension! at first glance

As they set $w = 1 \to m$ the units, ok, then model the partition as $\rho_t = \{S_1(t), \dots, S_{k_t}^{(t)}\}$ ⇒ NICE or again $c_t = (c_1(t), \dots, c_m(t))$ as before.

So we want to study the P of getting a certain collection of partitions. So, assuming a MC structure, we get our target to be as

$$p(\rho_1, \dots, \rho_T) = P(\rho_1(t) = \rho_1, \dots) = p(\rho_T | \rho_{T-1}) \cdot \ldots \cdot p(\rho_2 | \rho_1) \cdot p(\rho_1)$$

(markov prop)

Then they use as prior for the partition structure this

$$\text{“}Pr(\rho|M)\text{”} = p_{\rho_m|M}(\rho|M) = P(\rho_m = \rho | M) = \frac{M^k}{\prod_{i=1}^{m} (M+i-1)} \prod_{j=1}^{k} (|S_j|-1)!$$

(the constant normalization const) (the $\Gamma(|S|)$ told in the previous paper)

we call this law CRP(M). But that law misses how we want to model the connection among ρ_{t-1} and ρ_t.

we $\rho \sim CRP(M)$

So we introduce

$$\gamma_{it} = \begin{cases} 1 & \text{unit } i \text{ stays in the same cluster from } t-1 \text{ to } t \\ 0 & \text{otherwise} \end{cases}$$

we set this as given a $\gamma_{it} \overset{ind}{\sim} Ber(\alpha_t)$ — $\alpha_t = 0: \rho_t \perp \rho_{t-1}$; $\alpha_t = 1: \rho_t = \rho_{t-1}$

Then for better writing, assuming we introduce, to resume $\gamma_t = (\gamma_1(t), \dots, \gamma_m(t))$ where $\gamma_i(t) \in \{0,1\}$, to say which values we get from the $Ber(\alpha_t)$ for the γ_{it} we introduced above.

We call this model $tRPM(\alpha, M)$, and that is what eventually we will find in the model, about c_t, we ... on the ρ_t.

$$\begin{aligned}
Y_{it} | \mu_t^\star, \sigma_t^{2\star}, c_t &\overset{\text{ind}}{\sim} N(\mu^\star_{c_{it}t}, \sigma^{2\star}_{c_{it}t}),\ i = 1, \dots, m \text{ and } t = 1, \dots, T, \\
(\mu^\star_{jt}, \sigma^\star_{jt}) | \theta_t, \tau_t^2 &\overset{\text{ind}}{\sim} N(\theta_t, \tau_t^2) \times \mathrm{UN}(0, A_\sigma),\ j = 1, \dots, k_t, \\
(\theta_t, \tau_t) &\overset{\text{iid}}{\sim} N(\phi_0, \lambda^2) \times \mathrm{UN}(0, A_\tau),\ t = 1, \dots, T, \\
(\phi_0, \lambda) &\sim N(m_0, s_0^2) \times \mathrm{UN}(0, A_\lambda), \\
\checkmark\ \{c_t, \dots, c_T\} &\sim \mathrm{tRPM}(\boldsymbol{\alpha}, M), \text{ with } \alpha_t \overset{\text{iid}}{\sim} \mathrm{Beta}(a_\alpha, b_\alpha),
\end{aligned} \qquad (5)$$

The effect of this γ_t on the law of ρ is just that it reduces the support, i.e. now

$$P(\rho_t = \lambda | \gamma_t, \rho_{t-1}) = \begin{cases} 0 & \text{if } \rho_{t-1} \text{ and } \rho_t \text{ are not compatible wrt } \gamma_t \\ \dots & \text{other} \end{cases}$$

⋮ now it gets more complex, less clear

Actually, the paper continues and becomes non parametric as it talks about getting laws from a Dirichlet Process (DP), which models as a "prior that outputs distributions".

So model is too complex.
So we can use we can switch to our own models.

Anyway, here there was a data generation mechanism so we could use it to make up our own more models.

$$y(s_i) = y_i = \mu^*_{c_i}(s_i) + x_i^T \beta_{c_i} + \theta(s_i) + x_i^T \beta$$

spatial effect cluster + global specific

covariate effect cluster + global specific

paper 3 pag 23

$$y_i(t) \mid y_i(t-1), \dots, s_t \overset{\perp}{\sim} N\left(\mu^*_{c_{it}}(t) + w_i y_i(t-1), \dots\right)$$

to introduce an AR(1) effect

also by removing just $\overset{\perp}{\sim}$ (we have $\overset{iid}{\sim}$) we can see all these models as mixture models

$y_i(t) \mid \underline{w}_t, f_e \overset{iid}{\sim} f_y(y) = \sum_{e=1}^{k(t)} w_e(t) \cdot f_e(y)$ where $w_e(t) = P(c_i(t) = e \mid \underline{w})$

OR

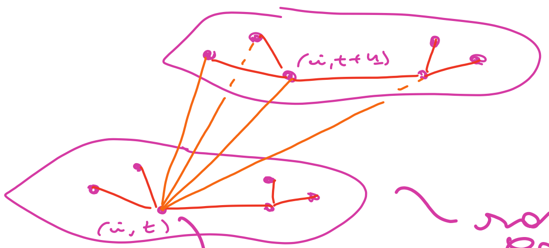
$y_i(t) \mid s_t, \dots \overset{iid}{\sim} \dots$ where $s_t \sim \dots$

paper 4 pag 40

interpretative TA 4

Model about the graph structure.

paper 9

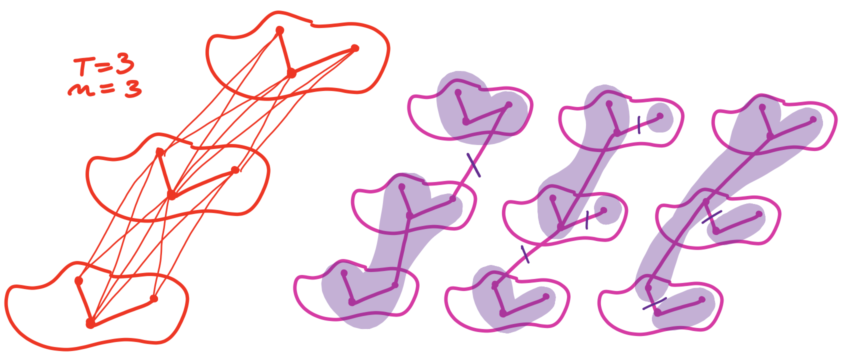


We start creating a graph for space and time: a repeated replica of the map, with also time connections.

~ same map repeated for T times

each point has response target and covariates, $(y_{i,t}, x_{i,t})$

assume for now $y_{tj} \mid x_{tj} \overset{ind}{\sim} f(y_{tj} \mid x_{tj})$ $\; t = 1 \to T, \; j = 1 \to m$



Here on paper 4 we can use covariates in the part of the ρ prior. But for me I think it is not enough, as the prior is one, but covariates vary over time.

paper 4