Supplement of
Conditional partial exchangeability: a probabilistic framework for multi-view clustering

B. Franzolini, M. De Iorio, J. Eriksson

S1 Preliminaries on exchangeable partitions and partial exchangeability

This section contains some preliminaries from existing literature on exchangeable partition models and partial exchangeability, which are relevant to the framework developed in the main paper.

Exchangeable partition models are a fundamental framework in Bayesian nonparametrics, particularly in clustering and mixture modeling, where the primary goal is often to infer underlying latent structures from observed data (see, for instance, Ghosal and Van der Vaart, 2017, Chapter 14.1). Such models are developed under the assumption of exchangeability of the sequence of random variables $(X_1, \dots, X_n)$, where X_i represents the ith observation in a sample of size n. The exchangeability assumption ensures that the joint distribution of the sequence remains invariant under any permutation of the indices. That is, for any permutation σ of the first n natural numbers, we have

$$(X_1, \dots, X_n) \stackrel{d}{=} (X_{\sigma(1)}, \dots, X_{\sigma(n)}),$$

where $\stackrel{d}{=}$ denotes equality in distribution. The standard extension of this assumption arises when the observed sample is considered as a finite subset of an infinite population, denoted by $X = (X_1, X_2, \dots)$. In this case, the assumption of exchangeability is naturally extended to the entire infinite sequence, requiring that any finite subset, regardless of its size, remains exchangeable.

The assumption of exchangeability for an infinite sequence of observations leads to the renowned de Finetti's representation theorem (De Finetti, 1937), which states that the law of any infinite and exchangeable sequence can be expressed as a mixture of independent and identically distributed (i.i.d.) random variables. Specifically, for any n and measurable $A_1, \dots, A_n$,

$$\mathbb{P}[X_1 \in A_1, \dots, X_n \in A_n] = \int \prod_{i=1}^{n} \tilde{p}(A_i)\, Q(\mathrm{d}\tilde{p}),$$

where $\tilde{p}$ is a random probability measure, and Q serves as a prior distribution on $\tilde{p}$.

Under these assumptions, exchangeable partitions play a fundamental role in nonparametric modeling strategies to determine the specification of Q, i.e., the distribution of $\tilde{p}$. In fact, many popular nonparametric priors Q (e.g., Ferguson, 1973; Pitman and Yor, 1997) almost surely select discrete random probability measures, i.e.,

$$\tilde{p} \stackrel{a.s.}{=} \sum_{h \geq 1} w_h \delta_{\theta_h},$$

where δ_x denotes a Dirac measure at x, and the weights satisfy $\sum_{h\geq 1} w_h = 1$. Extensions to continuous distributions can be achieved through mixture models, where the almost surely discrete random probability measure serves as a mixing measure for a continuous density kernel k (Ferguson, 1983; Lo, 1984), leading to

$$X_i \mid \tilde{p} \overset{iid}{\sim} \int k(x,\theta)\tilde{p}(\mathrm{d}\theta) \tag{1}$$

or equivalently

$$X_i \mid \xi_i \overset{ind}{\sim} k(x,\xi_i), \quad \xi_i \overset{iid}{\sim} \tilde{p}.$$

The discreteness of $\tilde{p}$, whether used directly to model the data distribution or as a mixing measure, induces ties among observations $(X_1,\ldots,X_n)$ or among latent parameters $(\xi_1,\ldots,\xi_n)$, respectively, thereby defining a random partition of $[n]=\{1,\ldots,n\}$. The random partition arises imposing that $\xi_i = \xi_j$ if and only if i and j belong to the same set in the partition. For instance, the Dirichlet process mixture (DPM) model, obtained by imposing a Dirichlet process prior on $\tilde{p}$ in (1), can be equivalently represented in terms of the induced partition structure. Let ρ denote the random partition and $\boldsymbol{c} = (c_1,\ldots,c_n)$ be a vector of subject-specific allocation variables encoding ρ, i.e. $c_i \in [n]$ and $c_i = c_j$ if and only if i and j belong to the same set in the partition. The DPM model can then be expressed as

$$X_i \mid \theta_{c_i} \overset{ind}{\sim} k(x,\theta_{c_i}), \quad \theta_h \overset{iid}{\sim} P_0, \quad \rho \sim \mathcal{L}_\alpha,$$

where P_0 and α denote the base measure and concentration parameter of the Dirichlet process, respectively, and $\mathcal{L}_\alpha$ represents the law of the partition induced by the Dirichlet process, referred to as exchangeable partition probability function (EPPF), which equals

$$\mathcal{L}_\alpha(\rho) = \frac{\alpha^K \prod_{h=1}^K (n_h-1)!}{\prod_{j=1}^n (\alpha+j-1)},$$

where K is the number of sets in ρ, and n_h denotes the number of elements assigned to the hth set in ρ (according to some arbitrary labeling of the sets in ρ). Note that, since the EPPF depends only on the cardinalities of the sets in ρ and it is symmetric as a function of the cardinalities $(n_1,\ldots,n_K)$, the latent random partition ρ inherits the exchangeability assumption imposed on the law of the observable $(X_1,\ldots,X_n)$. Permuting the partitioned elements does not change the probability of a given partition, formally

$$\mathcal{L}_\alpha(\,\{A_1,\ldots,A_K\}\,) = \mathcal{L}_\alpha(\sigma(\{A_1,\ldots,A_K\})),$$

e.g., $n=3$, $K=2$, and $\sigma=(1,3)$

$$\mathcal{L}_\alpha(\,\{\,\{1\},\{2,3\}\,\}\,) = \mathcal{L}_\alpha(\,\{\,\{1,2\},\{3\}\,\}\,).$$

This approach can be generalized to various partition laws, all satisfying exchangeability assumptions (Pitman, 1996), providing a flexible framework for clustering and mixture modeling.

However, while standard exchangeable partition models offer mathematical tractability, they impose strong assumptions that may not always align with real-world data, particularly when dependencies or structural constraints exist among observations. This limitation has motivated the development of dependent nonparametric priors, allowing for controlled deviations from full exchangeability. In contrast to full exchangeability, partial exchangeability provides a more flexible alternative by relaxing the assumption that all observations

are interchangeable. In many practical applications, data points are grouped into subsets where exchangeability holds within each subset but not necessarily across them. A number d of sequences of random variables $X_1 = (X_{1,1}, X_{1,2}, \ldots), \ldots, X_d = (X_{d,1}, X_{d,2}, \ldots)$ are said to be *partially exchangeable* if their joint distribution is invariant under permutations that only rearrange elements within each sequence. More formally, for any finite samples sizes $n_1, \ldots, n_d$ and any permutations $\sigma_1, \ldots, \sigma_d$, we have

$$((X_{1,i})_{i=1}^{n_1}, \ldots, (X_{d,i})_{i=1}^{n_d}) \overset{d}{=} \left((X_{1,\sigma_1(i)})_{i=1}^{n_1}, \ldots, (X_{d,\sigma_d(i)})_{i=1}^{n_d}\right).$$

This framework naturally arises in hierarchical or spatial settings, where dependencies exist but are not global. This leads to models such as the hierarchical Dirichlet process (Teh et al., 2006), nested Dirichlet process (Rodríguez et al., 2008), and dependent Dirichlet process (MacEachern, 2000), which introduce structured dependencies while preserving local exchangeability within groups. Importantly, partial exchangeability does not imply that the d sequences follow marginally the same distribution.

Nonetheless, in this work, we show that neither exchangeability nor partial exchangeability are appropriate for estimating multiple latent partitions of the same objects, as they fail to adequately capture heterogeneity or homogeneity in the data structure.

S2 Finite exchangeability of telescopic clustering

Theorem 2 in the main paper trivially implies that the collection $(X_{1i}, X_{2i})_{i\geq 1}$ is exchangeable in i and, consequently, that any finite subsequence of size n extracted from it is also exchangeable. In this section, we provide an alternative proof of exchangeability for any finite sequence of length n that does not rely on the representation of the infinite collection $(X_{1i}, X_{2i})_{i\geq 1}$ given in Theorem 2.

Assume that $(X_{1i}, X_{2i})_{i=1}^n$ is a finite sample distributed according to a telescopic clustering model. By marginal exchangeability of $(X_{1i})_{i=1}^n$, we have that, for any measurable $A \subset \mathbb{X}_1$ and any $\sigma \in \mathcal{P}(n)$, with $\mathcal{P}(n)$ set of permutations of n elements,

$$\mathbb{P}\left[(X_{1i})_{i=1}^n \in A\right] = \mathbb{P}\left[(X_{\sigma(i)}^{(1)})_{i=1}^n \in A\right] \tag{2}$$

Moreover, since the marginal model at layer 1 admits the equivalent representation

$$\begin{aligned} (X_{1i}, \theta_i) \mid \tilde{p}_1 &\overset{ind}{\sim} k_1(X_{1i}, \theta_i) \times \tilde{p}_1(\mathrm{d}\theta_i) \qquad \text{for } i = 1, \ldots, n \\ \tilde{p}_1 &\sim P_1 \end{aligned}$$

we have that $(X_{1i}, \theta_i)_{i=1}^n$ is exchangeable, i.e.,

$$\mathbb{P}\left[(X_{1i}, \theta_i)_{i=1}^n \in A \times B\right] = \mathbb{P}\left[(X_{\sigma(i)}^{(1)}, \theta_{\sigma(i)})_{i=1}^n \in A \times B\right] \tag{3}$$

for any measurable $B \subset \Theta$, and, therefore, by (2) and (3)

$$\mathbb{P}\left[(\theta_i)_{i=1}^n \in B \mid (X_{1i})_{i=1}^n \in A\right] = \mathbb{P}\left[(\theta_{\sigma(i)})_{i=1}^n \in B \mid (X_{\sigma(i)}^{(1)})_{i=1}^n \in A\right]$$

Moreover, we note that the partition ρ_1 is a deterministic function of the latent parameters $(\theta_1, \ldots, \theta_n)$ thus its posterior law has to preserve the same invariance of the posterior of the latent parameters, i.e., for any $\sigma \in \mathcal{P}(n)$,

$$\mathbb{P}\left[\rho_1 = \rho_1 \mid (X_{1i})_{i=1}^n \in A\right] = \mathbb{P}\left[\rho_1 = \sigma(\rho_1) \mid (X_{\sigma(i)}^{(1)})_{i=1}^n \in A\right] \tag{4}$$

where $\sigma(\rho_1)$ is the partition obtained applying the permutation σ to the elements in the clusters identified by ρ_1.

Consider now the second layer and a measurable rectangle $C = \bigotimes_{i=1}^{n} C_i$, note that

$$\mathbb{P}\left[(X_{2i})_{i=1}^{n} \in C \mid (X_{1i})_{i=1}^{n} \in A)\right]$$

equals

$$\sum_{\rho_1 \in \Pi(n)} \left\{\mathbb{P}\left[(X_{2i})_{i=1}^{n} \in C \mid \rho_1 = \rho_1\right] \mathbb{P}\left[\rho_1 = \rho_1 \mid (X_{1i})_{i=1}^{n} \in A\right]\right\} \tag{5}$$

where $\Pi(n)$ is the set of partitions of n elements and $\mathbb{P}[(X_{2i})_{i=1}^{n} \in C \mid \rho_1 = \rho_1]$ is

$$\begin{aligned}
&\int_{\mathcal{P}_{\mathbb{X}_2}^{M}} \prod_{m \in \boldsymbol{m}} \prod_{i:c_{1i}=m} \int_{\Theta_2} \int_{C_i} k_2(x,\theta)\mathrm{d}x\, \tilde{p}_{2m}(\mathrm{d}\theta) P_2(\mathrm{d}\tilde{p}_{21} \ldots \mathrm{d}p_{2M}) \\
&= \int_{\mathcal{P}_{\mathbb{X}_2}^{M}} \prod_{m \in \boldsymbol{m}} \prod_{i:c_{\sigma(i)}=m} \int_{\Theta_2} \int_{C_{\sigma(i)}} k_2(x,\theta)\mathrm{d}x\, \tilde{p}_{2m}(\mathrm{d}\theta) P_2(\mathrm{d}\tilde{p}_{21} \ldots \mathrm{d}p_{2M}) \\
&= \mathbb{P}((X_{2\sigma(i)})_{i=1}^{n} \in C \mid \rho_1 = \sigma(\rho_1))
\end{aligned} \tag{6}$$

where $\mathcal{P}_{\mathbb{X}_2}$ is the space of all probability measures on $\mathbb{X}_2$ and the mixing or de Finetti measure P_2 is a probability measure on $\mathcal{P}_{\mathbb{X}_2}^{M}$.

The extension of the result in (6) to any measurable set C can be obtained thanks to Dynkin's π-λ theorem, recalling that rectangles are a generating π-system of the Borel product σ-algebra and that the set of measurable C for which (6) holds true is easily proved to be a λ-system.

Putting together (5) with (4) and (6), for any measurable A and C, we get

$$\mathbb{P}\left[(X_{2i})_{i=1}^{n} \in C \mid (X_{1i})_{i=1}^{n} \in A\right] = \mathbb{P}\left[(X_{2\sigma(i)})_{i=1}^{n} \in C \mid (X_{\sigma(i)}^{(1)})_{i=1}^{n} \in A\right]$$

Finally, consider the joint prior predictive distribution of the whole finite matrix $(X_{1i}, X_{2i})_{i=1}^{n}$ obtained as

$$\mathbb{P}\left[(X_{1i})_{i=1}^{n} \in A\right] \mathbb{P}\left[(X_{2i})_{i=1}^{n} \in C \mid (X_{1i})_{i=1}^{n} \in A\right]$$

and plug-in the previous results to obtain invariance with respect to any permutation σ.

S3 Sampling schemes for generic telescopic clustering models

For simplicity of exposition, the algorithms for the general class of telescopic clustering models are in this section presented referring to the Markovian graphical structure in Figure 2 in Section 3.3, whose special cases include telescopic clustering with two layers. Algorithms for different graph structures can be obtained analogously. As an example of this, see the sampling strategy derived for the t-HDP in Section S4.1 below which is suitable for any polytree structure of dependence across layers.

S3.1 Marginal MCMC

In this section, both the underlying random probabilities and the cluster-specific parameters are marginalized out. The sampling of the partitions is then performed based on the exchangeable partition probability function (EPPF) of the first layer and the conditional

partial exchangeable partition probability functions (pEPPF) of the subsequent layers. The algorithms' output is a posterior sample of the clustering configuration only. The marginal MCMC's core structure is in Algorithm 1. Algorithm 1 requires to sample from the full conditional of the partition ρ_t, for $t = 1, \dots, T$. To derive the full conditional, we recall that $\boldsymbol{X}_t$ are the observations at layer t, c_{ti} is the allocation variable for the ith subject referring to the partition at layer t. We denote with $\boldsymbol{c}_t$ the collection of all allocation variables identifying the partition ρ_t, i.e., $\boldsymbol{c}_t = (c_{ti} : i = 1, \dots, n)$ and with $\boldsymbol{c}_t^{-i}$ the vector where the ith entry as been removed, i.e., $\boldsymbol{c}_t^{-i} = (c_{tj} : j \in [n] \setminus \{i\})$.

Algorithm 1 General model - Marginal algorithm core structure

Input: Data matrix $(X_{ti}, t = 1, \dots, T)_{i=1}^n$
Output: posterior distribution of $(\rho_t, t = 1, \dots, T)$

Sample ρ_1 from its full conditional proportional to

$$\mathbb{P}(\rho_1)\mathbb{P}(\boldsymbol{X}_1 \mid \rho_1)\mathbb{P}(\rho_2 \mid \rho_1)$$

for t in 2:$(T-1)$ **do**
Sample ρ_t from its full conditional proportional to

$$\mathbb{P}(\rho_t \mid \rho_{t-1})\mathbb{P}(\boldsymbol{X}_t \mid \rho_t)\mathbb{P}(\rho_{t+1} \mid \rho_t)$$

Sample ρ_T from its full conditional proportional to

$$\mathbb{P}(\rho_T \mid \rho_{T-1})\mathbb{P}(\boldsymbol{X}_T \mid \rho_T)$$

The full conditional of the partition at layer t is

$$\begin{aligned}\mathbb{P}(\rho_t \mid \rho_1, \dots, \rho_{t-1}, \rho_{t+1}, \dots, \rho_T, \boldsymbol{X}_1, \dots, \boldsymbol{X}_T) &\propto \mathbb{P}(\rho_1)\prod_{s=2}^{T}\mathbb{P}(\rho_s \mid \rho_{s-1})\prod_{s=1}^{T}\mathbb{P}(\boldsymbol{X}_s \mid \rho_s) \\ &\propto \mathbb{P}(\rho_t \mid \rho_{t-1})\mathbb{P}(\rho_t \mid \rho_{t+1})\mathbb{P}(\boldsymbol{X}_t \mid \rho_t)\end{aligned}$$

Sampling ρ_t from its full conditional is typically unfeasible since it requires evaluating the pEPPF, i.e., $\mathbb{P}(\rho_t \mid \rho_{t-1})$, for all possible realizations of ρ_t. This problem is not specific of telescopic clustering models. EPPFs and similar probability mass functions describing the law of partitions have always a large support that increases with n accordingly to the Bell number of n, and thus a posteriori is typically unfeasible to sample directly from them. The sampling of the partition in probabilistic clustering models is usually done by sampling each subject-specific allocation variable c_{ti} at a time, conditional on all the others. Following this strategy for telescopic clustering, we have:

$$\begin{aligned}\mathbb{P}(c_{ti} = m \mid \boldsymbol{X}_t, \boldsymbol{c}_t^{-i}, \rho_{t-1}, \rho_{t+1}) &\propto \mathbb{P}(c_{ti} = m, X_{it}, \rho_{t+1}, \mid \boldsymbol{X}_t^{-i}, \boldsymbol{c}_t^{-i}, \rho_{t-1}) \\ &= \mathbb{P}(c_{ti} = m, X_{it} \mid \boldsymbol{X}_t^{-i}, \boldsymbol{c}_t^{-i}, \rho_{t-1})\mathbb{P}(\rho_{t+1} \mid c_{ti} = m, \boldsymbol{X}_t, \boldsymbol{c}_t^{-i}, \rho_{t-1}) \\ &= \mathbb{P}(c_{ti} = m, X_{it} \mid \boldsymbol{X}_t^{-i}, \boldsymbol{c}_t^{-i}, \rho_{t-1})\mathbb{P}(\rho_{t+1} \mid c_{ti} = m, \boldsymbol{c}_t^{-i}) \\ &= \frac{\mathbb{P}(c_{ti} = m, \boldsymbol{c}_t^{-i}, \boldsymbol{X}_t \mid \rho_{t-1})}{\mathbb{P}(\boldsymbol{c}_t^{-i}, \boldsymbol{X}_t^{-i} \mid \rho_{t-1})}\mathbb{P}(\rho_{t+1} \mid c_{ti} = m, \boldsymbol{c}_t^{-i}) \\ &= \frac{\mathbb{P}(c_{ti} = m, \boldsymbol{c}_t^{-i} \mid \rho_{t-1})}{\mathbb{P}(\boldsymbol{c}_t^{-i} \mid \rho_{t-1})}\frac{\mathbb{P}(\boldsymbol{X}_t \mid c_{ti} = m, \boldsymbol{c}_t^{-i})}{\mathbb{P}(\boldsymbol{X}_t^{-i} \mid \boldsymbol{c}_t^{-i})}\mathbb{P}(\rho_{t+1} \mid c_{ti} = m, \boldsymbol{c}_t^{-i})\end{aligned}$$

This means that c_{ti} should be sampled accordingly to

$$p(c_{ti} = m \mid \boldsymbol{c}_t^{-i}, \boldsymbol{c}_{t-1}, \boldsymbol{c}_{t+1}, X^{(t)}) = \\ \mathrm{Past}_{imt}(\boldsymbol{c}_t^{-i}, \boldsymbol{c}_{t-1}) \times \mathrm{Fut}_{imt}(\boldsymbol{c}_t^{-i}, \boldsymbol{c}_{t+1}) \times \mathrm{Lik}_{imt}(\boldsymbol{c}_t^{-i}, \boldsymbol{X}_t)$$

where

$$\mathrm{Past}_{imt}(\boldsymbol{c}_t^{-i}, \boldsymbol{c}_{t-1}) = \begin{cases} \frac{\mathbb{P}(c_{ti}=m, \boldsymbol{c}_t^{-i})}{\mathbb{P}(\boldsymbol{c}_t^{-i})} & \text{for } t = 1 \\ \frac{\mathbb{P}(c_{ti}=m, \boldsymbol{c}_t^{-i} \mid \rho_{t-1})}{\mathbb{P}(\boldsymbol{c}_t^{-i} \mid \rho_{t-1})} & \text{for } t = 2, \dots, T \end{cases}$$

$$\mathrm{Fut}_{imt} = \begin{cases} \mathbb{P}(\rho_{t+1} \mid c_{ti} = m, \boldsymbol{c}_t^{-i}) & \text{for } t = 1, \dots, T-1 \\ 1 & \text{for } T = 1 \end{cases}$$

and

$$\mathrm{Lik}_{imt} = \begin{cases} \dfrac{\int k_t(x_i^{(t)}, \theta) \prod\limits_{\substack{j: c_{jt}=m \\ j \neq i}} k_t(x_j^{(t)}, \theta) \mathrm{d}P_\theta(\theta)}{\int \prod\limits_{\substack{j: c_{jt}=m \\ j \neq i}} k_t(x_j^{(t)}, \theta) \mathrm{d}P_\theta(\theta)} & \text{if } m \in \boldsymbol{c}_t^{-i} \\ \int k_t(x_i^{(t)}, \theta) \mathrm{d}P_\theta(\theta) & \text{otherwise} \end{cases}$$

Thus, the complexity and the mixing performance of this strategy largely depend on two aspects. The first is how fast the cluster-specific marginal likelihood

$$\int_{\Theta_t} \prod_{j: c_{tj}=m} k_t(x_{tj}, \theta) \mathrm{d}P_\theta(\theta)$$

can be computed. In this regard, the best scenario is when the kernel and the base measure are conjugate so that typically a closed-form expression for the marginal likelihood is available.

The second important aspect is how fast the ratio Past_{imt} and the factor Fut_{imt} can be computed. These both depend on the specific model chosen and may require the use of auxiliary random variables to be computed. For instance, when we devise a marginal algorithm for the t-HDP, Past_{imt} can be simplified by introducing the auxiliary variables referring to the labels of the tables in the restaurant franchise metaphor (see, Teh et al., 2006, for more details). Nonetheless, computing Fut_{imt} still requires evaluating the pEPPF $\mathbb{P}(\rho_{t+1} \mid \rho_t)$ for different configurations of ρ_t. This problem is specific to telescopic clustering models and not encountered in classical Bayesian mixture models. When it comes to Fut_{imt}, the introduction of latent variables to simplify this computation may not always be a viable strategy. In the model of Page et al. (2022), which is a specific case of telescopic clustering models, Fut_{imt} can be simplified to be an indicator function, thanks to some binary latent variables. However, for instance, with the t-HDP, introducing the table labels of the subsequent layer slows the mixing of the chain of parent nodes to unfeasible levels.

Whenever a specific telescopic clustering model is affected by this problem, there exist at least two possible solutions: the first is to derive a block marginal Gibbs sampler, that does not require the evaluation of Fut_{imt} and the second is to employ a conditional algorithm. They are described in the next two sections. Note that the next two sections provide core algorithms that are feasible only for a limited number of layers, nonetheless, the conditional one can be easily adapted to any number of layers and any structure of the CPE dependence as shown later in Section S4.2.

S3.2 Block Marginal Gibbs sampling for two layers

When the number of layers is small, e.g., $T = 2$, a marginal sampling scheme can be devised accordingly to Algorithm 2. Contrary to Algorithm 1, each allocation variable is sampled by integrating out the allocation variables of descendant/future layers of the same subject, resulting in a block structure where each subject is allocated to all layers conditional on the other subjects' allocation.

However, since the allocation variable at descendant layers is integrated out, each layer is sampled from a distribution that depends also on observations at subsequent layers. Such marginalization is only feasible for a limited number of layers since it increases the computational time per iteration proportionally to the number of descendant layers.

Algorithm 2 General model - Block Marginal algorithm core structure

Input: Data matrix $(X_{1i}, X_{2i})_{i=1}^n$
Output: smoothing posterior distribution of ρ_1 and ρ_2

for i in 1:n **do**
Sample c_{1i} from $p(c_{1i} \mid \boldsymbol{c}_1^{-i}, \boldsymbol{c}_2^{-i}, \boldsymbol{X}_1, \boldsymbol{X}_2)$, where

$$p(c_{1i} = m \mid \boldsymbol{c}_1^{-i}, \boldsymbol{c}_2^{-i}, \boldsymbol{X}_1, \boldsymbol{X}_2) \propto \begin{cases} C_m \frac{p(\rho_1^{-i} \cap \{c_{1i} = m\})}{p(\pi_1^{-i})} \frac{\int k_1(X_{1i}, \theta) \prod_{j:c_{1j}=m} k_1(X_{1j}, \theta) \mathrm{d}P_\theta(\theta)}{\int \prod_{j:c_{1j}=m} k_1(X_{1j}, \theta) \mathrm{d}P_\theta(\theta)} & \text{if } m \in \boldsymbol{c}_1^{-i} \\ C_m \frac{p(\rho_1^{-i} \cap \{c_{1i} = m\})}{p(\pi_1^{-i})} \int k_1(X_{1i}, \theta) \mathrm{d}P_\theta(\theta) & \text{otherwise} \end{cases}$$

where C_m is the marginal likelihood of the second layer, i.e., for $m \in \boldsymbol{c}_1^{-i}$,

$$C_m = \sum_s \frac{p(\rho_2^{-i} \cap \{c_{2i} = s\}) \mid \rho_1)}{p(\rho_2^{-i} \mid \rho_1)} \frac{\int k_2(X_{2i}, \theta) \prod_{j:c_{2j}=s} k_2(X_{2j}, \xi) \mathrm{d}P_\xi(\xi)}{\int \prod_{j:c_{2j}=s} k_2(X_{2j}, \xi) \mathrm{d}P_\xi(\xi)}$$

Sample c_{2i} from $p(c_{2i} \mid \boldsymbol{c}_1, \boldsymbol{c}_2^{-i}, \boldsymbol{X}_2)$, where

$$p(c_{2i} = s \mid \boldsymbol{m}, \boldsymbol{s}^{-i}, \boldsymbol{X}_2) \propto \begin{cases} \frac{p(\rho_2^{-1} \cap \{c_{2i} = s\} \mid \rho_1)}{p(\pi_2^{-i} \mid \rho_1)} \frac{\int k_2(X_{2i}, \theta) \prod_{j:c_{2j}=s} k_2(X_{2j}, \xi) \mathrm{d}P_\xi(\xi)}{\int \prod_{j:c_{2j}=s} k_2(X_{2j}, \xi) \mathrm{d}P_\xi(\xi)} & \text{if } s \in \boldsymbol{c}_2^{-i} \\ \frac{p(\rho_2^{-1} \cap \{c_{2i} = s\} \mid \rho_1)}{p(\pi_2^{-i} \mid \rho_1)} \int k_2(X_{2i}, \xi) \mathrm{d}P_\xi(\xi) & \text{otherwise} \end{cases}$$

where: $\prod_{s \in \emptyset} := 1$.

S3.3 Conditional MCMC sampler

Conditional algorithms are a convenient strategy when the full posterior of the random probability is easier to sample compared to the evaluation of the partition's probability mass function. In fact, conditionally on the random probabilities, the full conditional of the allocation variable c_{ti} largely simplifies since it does not depend on observations other than X_{ti}, for t varying.

To derive the conditional sampler for a generic telescopic clustering model, denote with

- $\pi(m,k,t)$ the weight associated to the kth component of $\tilde{p}_m^{(t)}$
- $\theta^\star(m,k,t)$ the atom associated to the kth component of $\tilde{p}_m^{(t)}$

Algorithm 3 General model - Conditional algorithm core structure

Input: Data matrix $(X_{ti}, t = 1,\dots,T)_{i=1}^n$
Output: posterior distribution of ρ_1 and ρ_2

for i in 1:n **do**
Sample $(c_{ti})_t$ from

$$p[(c_{ti})_{t=1}^T = (c_t)_{t=1}^T)] \propto \prod_{t=1}^{T} [\pi(c_{t-1}, c_t, t)\kappa_t(X_{ti}; \theta^\star(c_{t-1}, c_t, t))]$$

Sample $\pi(m,k,t)$ and $\theta^\star(m,k,t)$ (full conditional does not depends on $(\boldsymbol{X}_s)_{s\neq t}$)

Note that the algorithm above requires sampling the joint collection of allocation variables for a single i across all layers. If the number of layers is large, this approach may not always be optimal, as it amounts to sampling a discrete random variable—i.e., the joint path $(c_{ti})_{t=1}^T$—from a full conditional distribution with very large support. In Section S4.2, we show how to modify this core structure to sample one layer at a time instead of the entire path jointly.

S4 Sampling schemes for t-HDP

As already noticed in Section S3, the marginal sampling scheme as devised for a general telescopic sampler as in Algorithm 1 is not a viable alternative for the t-HDP. In particular, adopting the general marginal sampler, require to evaluate Fut_{imt} which is computationally non-feasible, and cannot be solved with the introduction of the typical latent variables employed with the HDP, because will results in a slow mixing, which decreases drastically for layers with a high number of descendants.

In the following, we provide the equivalence of Algorithm 2 as specialized for the t-HDP model and a variant of Algorithm 3. The latter is a fast conditional sampler, obtained by combining block and partially collapsed Gibbs sampling steps, that can be employed for any reasonable number of layers, we tested the performance to up to 100 layers. This partially collapsed conditional block Gibbs sampler serves also to show how Algorithm 3 can be further refined to obtain a scalable algorithm even in the presence of an elevated number of layers.

S4.1 Block Marginal Gibbs sampling for two layers

Referring to the Chinese restaurant metaphor used to describe the predictive law of the hierarchical Dirichlet process as in Teh et al. (2006), denote with c_{ti}, the label of the table at which the ith client is sat at layer t and with c_{ti} the dish eaten by the ith client at layer t.

We recall that all clients that sat at the same table eat the same dish and that the same dish can be served at more than one table.

According to the metaphor, c_{ti} encodes the clustering structure of interest, while c_{ti} are auxiliary latent parameters that are used to simplify the full conditional distribution from which the cluster configuration has to be sampled in a Gibbs sampler.

Denote with

- $\mathcal{C}_1$ the set of tables' labels at layer 1
- $\mathcal{M}_1$ the set of dishes' labels at layer 1
- $\mathcal{C}_2$ the set of tables' labels at layer 2
- $\mathcal{M}_2$ the set of dishes' labels at layer 2
- $\mathcal{C}_{2|m}$ the set of tables' labels at layer 2 restricted to those clients that at layer 1 were eating dish m
- n_{1c} number of customer at layer 1 sat at table c
- $n_{2,c|m}$ number of customer sat at table c at layer 2 and eating dish m at layer 1
- q_{1m} number of tables at layer 1 serving dish m
- q_{2m} number of tables at layer 2 serving dish m
- $d_\ell(c)$ a function returning the label of the dish served at table c of layer ℓ

At layer 1, to sample c_{1i} from $p(c_{1i} \mid \boldsymbol{c}_1^{-i}, \boldsymbol{X}_1, \boldsymbol{X}_2)$, we first sample the table allocation variable c_{1i} from

$$
\begin{aligned}
&p(c_{1i} = c \mid \boldsymbol{c}_1^{-i}, \boldsymbol{c}_1^{-i}, \boldsymbol{X}_1, \boldsymbol{X}_2) \\
&\qquad \propto \begin{cases}
C_{d_1(c)}\, n_{1c}^{-i} \dfrac{\int k_1(X_{1i},\theta) \prod\limits_{j:c_{1j}=c} k_1(x_j^{(1)},\theta)\mathrm{d}P_\theta(\theta)}{\int \prod\limits_{j:c_{1j}=c} k_1(x_j^{(1)},\theta)\mathrm{d}P_\theta(\theta)} & \text{if } c \in \mathcal{C}_1^{-i} \\
\alpha\Bigg(\displaystyle\sum_{m \in \mathcal{M}_1^{-i}} C_m \frac{q_{1m}^{-i}}{q_1^{*-i}+\alpha_0} \frac{\int k_1(X_{1i},\theta) \prod\limits_{j:c_{1j}=m} k_1(x_j^{(1)},\theta)\mathrm{d}P_\theta(\theta)}{\int \prod\limits_{j:c_{1j}=m} k_1(x_j^{(1)},\theta)\mathrm{d}P_\theta(\theta)} + \\
\qquad C_0 \frac{\alpha_0}{q_1^{*-i}+\alpha_0} \int k_1(X_{1i},\theta)\mathrm{d}P_\theta(\theta) \Bigg) & \text{otherwise}
\end{cases}
\end{aligned}
$$

where

1.

$$
\begin{aligned}
C_m = &\sum_{s \in \mathcal{C}_{2|m}^{-i}} \frac{n_{2,s|m}^{-i}}{n_{1m}^{\star-i}+\alpha} \frac{\int k_2(X_{2i},\theta) \prod\limits_{j:c_{2j}=s} k_2(X_{2j},\xi)\mathrm{d}P_\xi(\xi)}{\int \prod\limits_{j:c_{2j}=s} k_2(X_{2j},\xi)\mathrm{d}P_\xi(\xi)} + \\
&+ \frac{\alpha}{(n_{1m}^{\star-i}+\alpha)} \sum_{s \in \mathcal{M}_2^{-i}} \frac{q_{2s}^{-i}}{(q_2^{\star-i}+\alpha_0)} \frac{\int k_2(X_{2i},\theta) \prod\limits_{j:c_{2j}=s} k_2(X_{2j},\xi)\mathrm{d}P_\xi(\xi)}{\int \prod\limits_{j:c_{2j}=s} k_2(X_{2j},\xi)\mathrm{d}P_\xi(\xi)} + \\
&+ \frac{\alpha}{(n_{1m}^{\star-i}+\alpha)} \frac{\alpha_0}{(q_2^{\star-i}+\alpha_0)} \int k_2(X_{2i},\theta)\mathrm{d}P_\xi(\xi)
\end{aligned}
$$

with

$$n_{1m}^{\star -i} = \sum_{s \in \mathcal{C}_{2|m}^{-i}} n_{2,s|m}^{-i} \qquad q_2^{\star -i} = \sum_{s \in \mathcal{M}_2^{-i}} q_{2s}$$

note that $n_{1m}^{\star -i}$ is the number of subjects assigned to dish m at layer 1 (excluding subject i.

2.

$$C_0 = \sum_{s \in \mathcal{M}_2^{-i}} \frac{q_{2s}^{-i}}{(q_2^{\star -i} + \alpha_0)} \frac{\int k_2(X_{2i}, \theta) \prod_{j:c_{2j}=s} k_2(X_{2j}, \xi) \mathrm{d}P_\xi(\xi)}{\int \prod_{j:c_{2j}=s} k_2(X_{2j}, \xi) \mathrm{d}P_\xi(\xi)} + \\ + \frac{\alpha_0}{(q_2^{\star -i} + \alpha_0)} \int k_2(X_{2i}, \theta) \mathrm{d}P_\xi(\xi)$$

Then the dish allocation variable $c_{1,}$ is sampled from $p(c_{1i} \mid \boldsymbol{m}^{-i}, \boldsymbol{c}, X^{(1)}, \boldsymbol{X}_2)$. Notice that the full conditional is degenerate if at the previous step, the customer has sat at an already occupied table, contrary, if $c_{1i} \notin \mathcal{C}_1^{-i}$,

$$p(c_{1i} = m \mid \boldsymbol{m}^{-i}, \boldsymbol{c}, X^{(1)}, \boldsymbol{X}_2) \\ \propto \begin{cases} C_m \, q_{1m}^{-i} \dfrac{\int k_1(X_{1i}, \theta) \prod_{j:c_{1j}=m} k_1(x_j^{(1)}, \theta) \mathrm{d}P_\theta(\theta)}{\int \prod_{j:c_{1j}=m} k_1(x_j^{(1)}, \theta) \mathrm{d}P_\theta(\theta)} & \text{if } m \in \boldsymbol{c}_1^{-i} \\ C_0 \, \alpha_0 \int k_1(X_{1i}, \theta) \mathrm{d}P_\theta(\theta) & \text{otherwise} \end{cases}$$

The second layer is sampled following a classical marginal MCMC for the HDP (see Teh et al., 2006), which can be obtained from the two full conditionals above setting $C_m = 1$ for all m and $C_0 = 1$.

S4.2 Partially collapsed conditional block Gibbs sampler

Denote with

- $\pi_0(k, \ell)$ the weight associated to the kth component of $\tilde{q}_0^{(\ell)}$
- $\theta_0^\star(k, \ell)$ the atom associated to the kth component of $\tilde{q}_0^{(\ell)}$
- $\pi(m, k, \ell)$ the weight associated to the kth component of $\tilde{p}_m^{(\ell)}$
- $c(\ell, i)$ label of the table at layer ℓ of the ith customer
- $k(\ell, c)$ label of the dish served at layer ℓ at table c
- $m(\ell, i)$ label of the dish eaten at layer ℓ by the ith customer (thus: $m(0, i) = 1$ for all i and $m(\ell, i) = k(\ell, c(\ell, i))$)

The truncated stick breaking version of the t-HDP can be written as follows

$$
\begin{aligned}
\pi_0(\cdot,\ell) = [\pi_0(1,\ell),\dots,\pi_0(H_0,\ell)] &\overset{iid}{\sim} \text{TSB}(\alpha_0, H_0) && \text{for } \ell = 1,\dots,L \\
\pi(m,\cdot,\ell) = [\pi(m,1,\ell),\dots,\pi_0(m,H,\ell)] &\overset{iid}{\sim} \text{TSB}(\alpha, H) && \text{for } m = 1,\dots,H_0 \\
& && \text{and } \ell = 1,\dots,L \\
\theta_0^\star(\cdot,\ell) = [\theta_0^\star(1,\ell),\dots,\theta_0^\star(H_0,\ell)] &\overset{iid}{\sim} \bigtimes_{h=1}^{H_0} P_0 && \text{for } \ell = 1,\dots,L \\
k(\ell,\cdot) = [k(\ell,1),\dots,k(\ell,H_0\times H)] \mid \pi_0(\cdot,\ell) &\overset{ind}{\sim} \bigtimes_{c=1}^{H_0\times H} \left(\sum_h \pi_0(h,\ell)\delta_h\right) && \text{for } \ell = 1,\dots,L \\
c(\ell,\cdot) = [c(\ell,1),\dots,c(\ell,n)] \mid \pi(\cdot,\cdot,\ell), m(\text{par}(\ell),\cdot) & && \\
\overset{ind}{\sim} \bigtimes_{i=1}^{n} \left(\sum_{h=1}^{H} \pi(m(\text{par}(\ell),i),h,\ell)\delta_{[(m(\text{par}(\ell),i)-1)H+h]}\right) && \text{for } \ell = 1,\dots,L \\
m(\ell,i) \mid k(\ell,\cdot), c(\ell,i) &\overset{ind}{\sim} \delta_{k(\ell,c(\ell,i))} && \text{for } i = 1,\dots,n \\
& && \text{and } \ell = 1,\dots,L \\
X_{\ell i} \mid \theta_0^\star, m(\ell,i) &\overset{ind}{\sim} \kappa_\ell(\cdot,\theta_0^\star(m(\ell,i),\ell)) && \text{for } i = 1,\dots,n \\
& && \text{and } \ell = 1,\dots,L
\end{aligned}
$$

Denote also with

- C_ℓ the set of unique values in $c(\ell,\cdot)$ (actually occupied tables)
- $n(\ell,c)$ number of customer at layer ℓ sat at table c
- $q(\ell,h)$ number of tables at layer ℓ serving dish h
- $\bar{n}(\ell,h_1,h_2) = n(\ell,(h_1-1)\times H + h_2)$

Figure S4.1 shows the corresponding graphical model when the number of layers equals three and the dependence across layers is triangular as in Figure 3 in Section 3.3. Algorithm 4 contains the pseudo-code of the conditional algorithm to estimate the t-HDP model for any number of layers and any polytree structure. The algorithm is derived based on the truncated stick-breaking version of the t-HDP, described above, and it is obtained by combining block and partially collapsed Gibbs sampling steps. In particular, $c(\ell,i)$ and $m(\ell,i)$ are sampled as a block from which $\{c(\ell,i), \text{with } \ell \in \text{child}(\ell)\}$ are marginalized out. This drastically improves the mixing of the chain compared to a classical Gibbs sampler and leads to the correct stationary distribution for the chain (cfr., Van Dyk and Park, 2008).

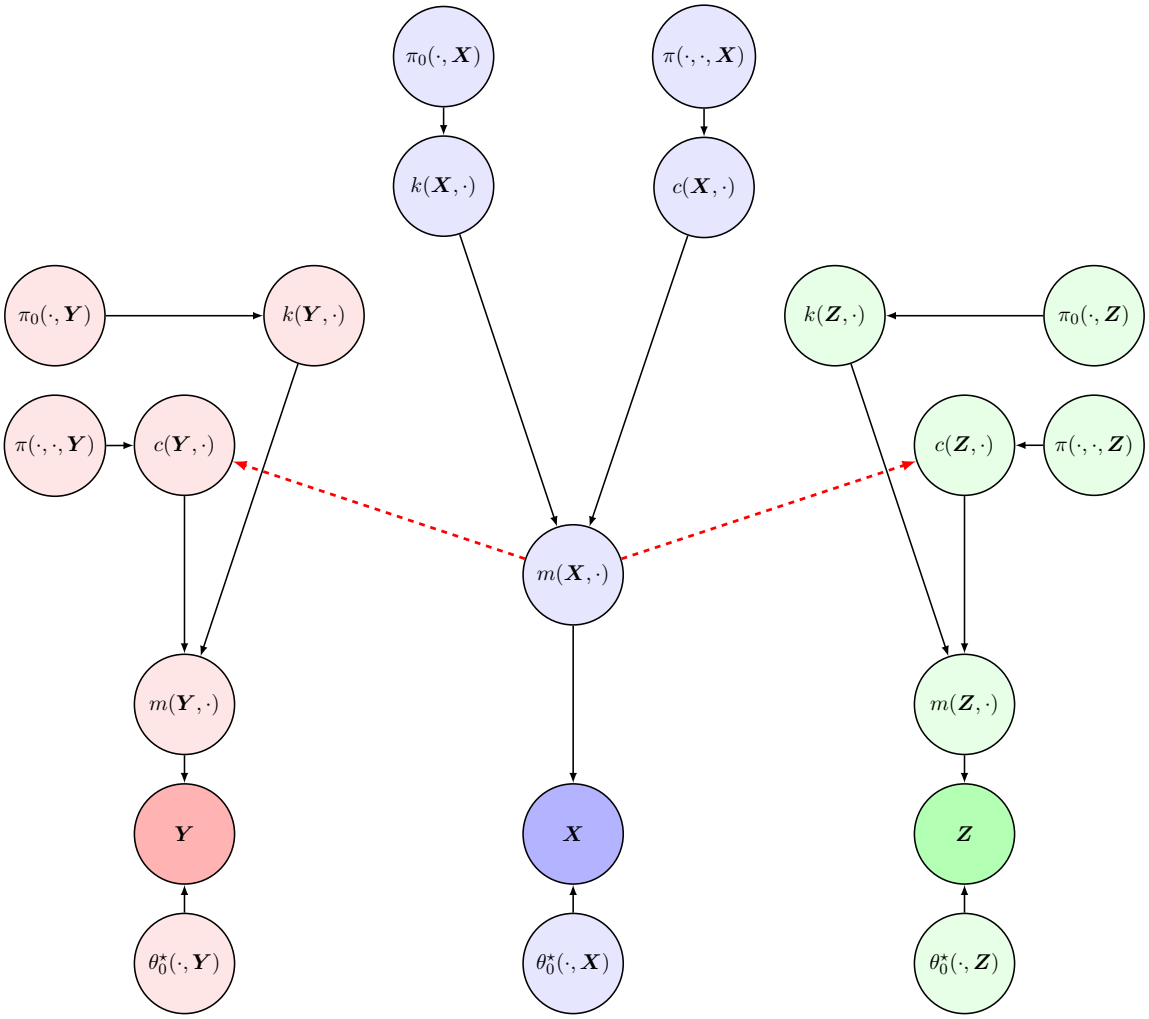


Figure S4.1: Graphical model corresponding to a t-HDP with the truncated stick-breaking representation with a triangular layer dependence.

Algorithm 4 Conditional sampler - t-HDP

Input: Data matrix $(X_{1i}, X_{2i})_{i=1}^n$
Output: smoothing posterior distribution of ρ_1 and ρ_2

for ℓ in 1:L **do**
  for h_1 in 1:H_0 **do**
    Sample $b_0(h_1, \ell)$ from Beta $\left(1 + q(\ell, h_1), \alpha_0 + \sum_{s=h_1+1}^{H_0} q(\ell, s)\right)^\diamond$
    $\pi_0(h_1, \ell) \leftarrow b_0(h_1, \ell) \prod_{s=1}^{h_1-1} b_0(s, \ell)^\diamond$
    Sample $\theta_0^\star(h_1, \ell)$ from $p(\theta) \propto \prod_{i:m(\ell,i)=h_1} \kappa_\ell(X_{\ell i}, \theta) P_0(\mathrm{d}\theta)^\diamond$
    for h_2 in 1:H **do**
      Sample $b(h_1, h_2, \ell)$ from Beta $\left(1 + \bar{n}(\ell, h_1, h_2), \alpha_0 + \sum_{s=h_2+1}^{H} \bar{n}(\ell, h_1, s)\right)^\diamond$
      $\pi(h_1, h_2, \ell) \leftarrow b(h_1, h_2, \ell) \prod_{s=1}^{h_2-1} b(h_1, s, \ell)^\diamond$

for ℓ in 1:L **do**
  for i in 1:n **do**
    $m \leftarrow m(\ell - 1, i)$
    $f \leftarrow m(\ell + 1, i)$
    Sample $c(\ell, i)$ from $p(c)$ with $c \in \{(m-1)H + 1, \ldots, mH\}$, where

$$p(c) \propto \pi(m, c, \ell) \times \kappa_\ell(X_{\ell i}\,; \theta_0^\star(k(\ell, c), \ell)) \times \prod_{\ell^\star \in \mathrm{child}(\ell)} \left(\sum_{d \in \mathcal{M}_{\ell c f}} \pi(k(\ell, c), d, \ell^\star) \right)$$
$$\text{where } \mathcal{M}_{\ell c f} = \{d : k(\ell^\star, [k(\ell, c) - 1]H + d) = f\}$$

  for c in $1 : (H \times H_0)$ **do**
    Sample $k(\ell, c)$ from $p(k)$, with, for $k \in \{1, \ldots H_0\}$,

$$p(k) \propto \pi_0(k, \ell) \prod_{i:c(\ell,i)=c} \kappa_\ell(X_{\ell i}\,; \theta_0^\star(k, \ell))^\diamond$$

  for i in 1:n **do**
    $m(\ell, i) \leftarrow k(\ell, c(\ell, i))$
  for h in 1:H_0 **do**
    $q(\ell, h) \leftarrow \sum_{c=1}^{H \times H_0} \mathbb{1}(k(\ell, c) = h)\, \mathbb{1}(c \in \mathcal{C}_\ell)$
  for c in 1:$H \times H_0$ **do**
    $n(\ell, c) \leftarrow \sum_{i=1}^{n} \mathbb{1}(c(\ell, i) = c)$

$\diamond$ we use the conventions: $\sum_{s=H_0+1}^{H_0} q(\ell, s) := 0$, $\prod_{s=1}^{0} b_0(s, \ell) := 1$, $\prod_{i \in \emptyset} x_i = 1$

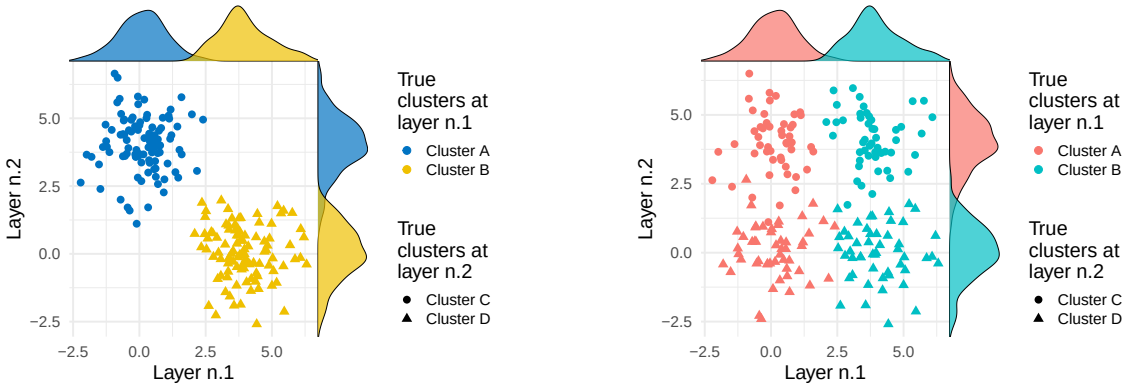
S5 Simulation studies

Scenario	Num. of items	Num. of layers	Num. of var. per layer	Num. of clusters	ARI	Mispecified
n.A	200	2	1	2	1.000	No
n.B	200	2	1	2	0.010	No
n.1	200	10	1	2	0.809	No
n.2	200	100	1	2	0.921	No
n.C	200	2	2	3	0.914	Yes

Table S5.1: Simulation scenarios summaries: number of layers, layers' dimension (i.e., number of variables per each layer), number of clusters at each layer, adjusted Rand index between partitions at consecutive layers, whether the t-HDP estimated over the simulated data has a mispecified kernel or not.

Scenario n.A and n.B

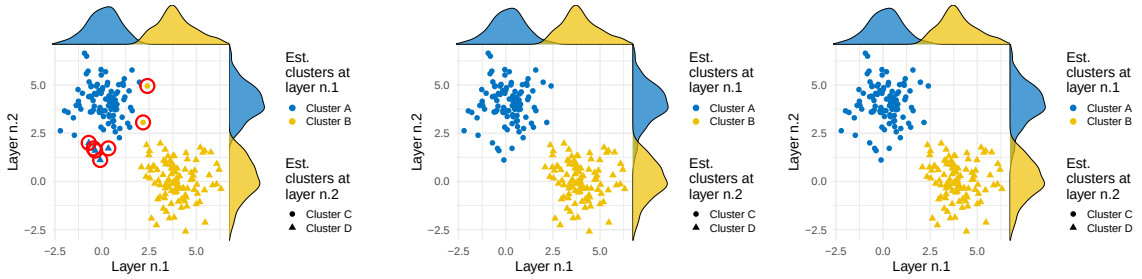
Simulating scenario description: In Scenario A and B, data for $n = 200$ observational units and $L = 2$ layers are generated. Scenario A is obtained keeping the clustering structure constant across the two layers. In particular, the first cluster is composed by 100 observations such that $(X_{1i}, X_{2i}) \overset{iid}{\sim} \mathcal{N}(0,1) \times \mathcal{N}(4,1)$, while the remaining 100 observations form a second cluster and are sampled according to $(X_{1i}, X_{2i}) \overset{iid}{\sim} \mathcal{N}(4,1) \times \mathcal{N}(0,1)$. Figure S5.1a shows the simulated data and highlights how the two clusters are well-separated both at layer 1 (on the x-axis) and at layer 2 (on the y-axis). Differently, Scenario B is obtained by imposing two highly different clustering structures at the two layers while keeping the number of clusters and the clusters' frequencies fixed across layers. This is achieved by re-assigning half of the observations in each cluster to the other cluster while moving from one layer to the next. Denoting with c_{1i} and c_{2i} the allocation variables at layer 1 and 2 respec-



(a) Scenario 1: RI = 1, ARI = 1

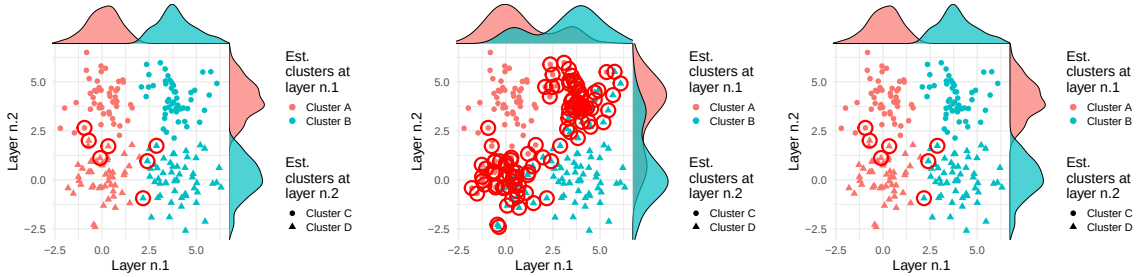
(b) Scenario 2: RI = 0.55, ARI = 0.01

Figure S5.1: Simulation study: simulated data and *true* cluster allocation for Scenarios A and B. Each point corresponds to an item, colours denote clusters at layer 1 and shapes are clusters at layer 2. Under scenario A, the clustering structure is the same at both layers, and the adjusted Rand index between the two partitions is equal to 1. Under scenario B, the clustering structure drastically changes between the two layers and the adjusted Rand index (ARI) between the two partitions is approximately 0.



(a) Independent clustering (b) Constant clustering (c) Telescopic HDP

Figure S5.2: Simulation study: results for Scenario A. Red circles denote observations assigned to the wrong cluster at least for one layer.



(a) Independent clustering (b) Constant clustering (c) Telescopic HDP

Figure S5.3: Simulation study: results for Scenario B. Red circles denote observations assigned to the wrong cluster at least for one layer.

tively, the data generating process is: $(X_{1i}, X_{2i}) \mid c_{1i} = m, c_{2i} = s \overset{iid}{\sim} \mathcal{N}(\theta_m, 1) \times \mathcal{N}(\xi_s, 1)$, where the locations at layer 1 are $\boldsymbol{\theta} = (\theta_1, \theta_2) = (0, 4)$ and the locations at layer 2 are $\boldsymbol{\xi} = (\xi_1, \xi_2) = (4, 0)$. In this scenario, the true clustering structure coincides with the expected value of a random assignment procedure, where, moving from layer 1 to layer 2, each observation is reassigned to the other cluster with probability 1/2. Figure S5.1b shows the simulated data. Note that the layer-marginal distributions are the same in both simulation studies, what truly differentiates the two scenarios is how single items are reallocated moving from layer 1 to layer 2.

Model: For both simulated datasets A and B, as baseline comparisons, we estimate the clustering configuration independently at each layer as well as a constant clustering model, which assumes the same configuration at both layers. We compare such approaches with the results from the t-HDP model, based on CPE. The first two models are mixtures of the hierarchical Dirichlet process as described in Camerlenghi et al. (2018). All three models have univariate Normal kernel with mean μ and variance equal to 1. Prior distribution for the mean is Normal centered in 0 and variance equal to 0.1. Concentration parameters are fixed to 0.1. What differentiates the three models is the type of dependence between layer-specific partitions, from independence to complete dependence to CPE.

Algorithm for t-HDP: 100 000 iterations of the block partially collapsed conditional sampler in Algorithm 4 are performed and the first half is disregarded as burn-in. The chain is initialized to the k-means solutions computed independently for each layer.

Results: Figure S6.2 and S5.3 show the point estimates of the clustering allocations obtained minimizing the variation of information loss (Meilă, 2007).

The constant clustering approach performs extremely well under scenario A, since the prior distribution is degenerate on the truth of a unique clustering configuration (cf. Figure S5.2b). The same model performs badly in the second simulation scenario since the true clustering configuration does not belong to the support of the prior (cf. Figure S5.3b).

On the other hand, the independent model performs worse than the constant model in simulation scenario A, since it does not allow for any borrowing of information and modeling of within-subject dependence/ subjects' identity (cf. Figure S6.2a). In simulation scenario B, the independent model has an advantage with respect to both the constant clustering and CPE, because under the truth there is no within-subject dependence and borrowing of information between clustering configurations is undesirable. Nonetheless, in this second scenario, the independent approach lead to seven allocation errors (four at layer 1 and three at layer 2), which can be explained by the fact that they correspond to observations that are more likely to be generated under the other mixture component (cf. Figure S5.3a).

The results of the telescopic clustering model coincide with the best performance in both scenarios. In fact, the model achieves the same results as the constant model when the clustering configuration is indeed constant (Scenario A) and the same results as the independent model when the clustering configurations are the expectation of a random assignment (Scenario B). The telescopic clustering appears able to detect the dependence structure between layers and accurately recover the clustering configuration of the observations.

Scenario n.1

Simulating scenario description: In Scenario 1, we generate data on $n = 200$ items and $T = 10$ layers. At each layer, marginally we assume two clusters simulated from two univariate Normal distributions with unitary variance and centred in 0 and 4 respectively. From one layer to the next, 10 items (5% of the total) are selected at random and moved to the other cluster, so that the adjusted Rand index from one layer t to the next $t + 1$ equals 0.809. Simulated data are shown in Figure S5.4. See also Section 7.1.

Model: t-HDP model with univariate Normal kernel with mean μ and variance σ^2. Prior distribution for the mean is Normal centered in 0 and variance equal to $\sigma^2/0.1$. Concentration parameters are fixed to 0.1. The prior for the precision $1/\sigma^2$ is a Gamma distribution with shape and rate parameters equal to 0.1. We compare the results of t-HDP with three competitors We compare four methods: (i) k-means fitted independently at each layer, where the number of clusters is determined by the gap statistics (Tibshirani et al., 2001); (ii) the estimate obtained with a logit stick-breaking process (LSBP) (Ren et al., 2011); and (iii) the estimate from an Enriched Dirichlet process (E-DP) (Wade et al., 2011).

Algorithm for t-HDP: 100 000 iterations of the block partially collapsed conditional sampler in Algorithm 4 are performed and the first half is disregarded as burn-in. The chain is initialized to the k-means solutions computed independently for each layer.

Results: The t-HDP model identifies the true clustering configuration at all layers with at most three out of 200 wrongly allocated subjects and a rand index between the truth and the estimate always higher than 0.97, the average rand index equals 0.99 and the average number of wrongly allocated subject per layer is 1.1 out 200. It outperforms the competitors both consistently at each layer and overall.

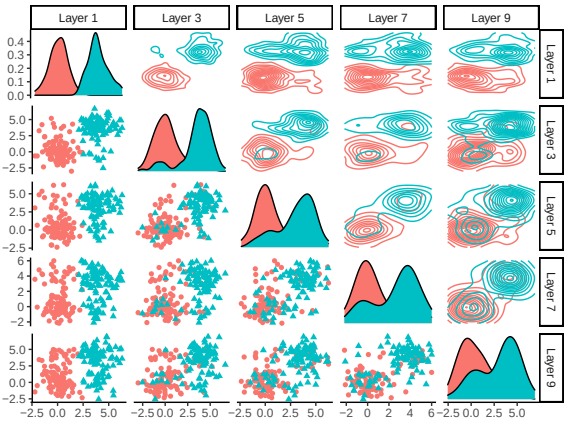


Figure S5.4: Simulation study: simulated data for Scenario 1. Plots refer to the observations in layers 1, 3, 5, 7, and 9. Colours and shapes denote the true clustering at layer 1. The diagonal plots show the marginal distribution at each layer, colour coded according to the clustering allocation at layer 1. Upper off-diagonal plots display the joint distribution of two pairs of layers, colour coded according to the clustering allocation at layer 1. Lower off-diagonal plots show the scatter plot of the data at the corresponding layers, colour coded according to the clustering allocation at layer 1. The adjusted Rand index between two consecutive configurations is 0.809.

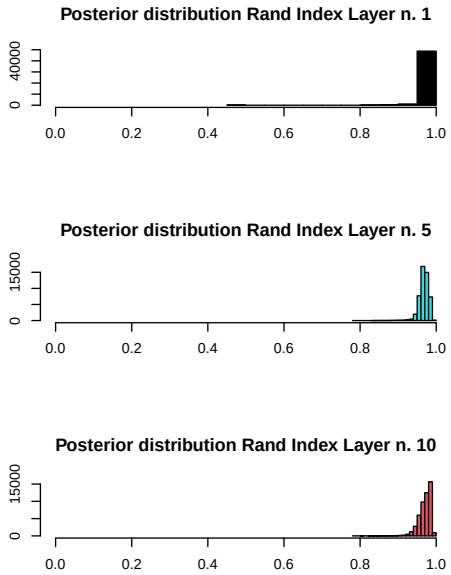


Figure S5.5: Simulation study: results for Scenario 1. Posterior distributions of Rand indexes between the posterior configurations and the truth for t-HDP model for layers 1, 5, 10.

Independent k-means and the LSBP perform well, even if they do not include within-subject dependence. This is to be expected in this scenario since the true clusters are well-separated (cf., with results of Scenario 4 below, where the k-means solution is often unable to identify the true number of clusters, even though the cluster have still Gaussian shapes). Nevertheless, both the k-means solution and the LSBP estimates are always dominated by the t-HDP estimates.

Finally, the enriched Dirichlet process is the worst-performing model, as a direct consequence of the degeneracy issue of the model discussed in Section 2. Recall that under the enriched Dirichlet process, once two items are assigned to two different clusters at layer t, they cannot be assigned to the same cluster at any subsequent layer s, for $s > t$.

Figure S5.5 shows the posterior distribution of the Rand index between the true clustering configuration and the configurations visited by the posterior algorithm of the t-HDP model for layers 1, 5, and 10 after burn-in. The posterior is concentrated around 1, which corresponds to the truth, exhibiting small uncertainty around the estimated clustering configuration. See also Section 7.1.

Scenario n.2

Simulating scenario description: Data for 100 layers are simulated. At each layer there are two clusters and data are univariate. In particular, at layer 1 half of the dataset forms the first cluster, i.e., $c_{1i} = 1$ for $i = 1, \dots, 50$, and the other half the second cluster, i.e. $c_{1i} = 2$ for $i = 51, \dots, 100$. At layer 1, values are sampled from

$$X_{1i} \mid c_{1i} \overset{ind}{\sim} \mathcal{N}(0,1)\mathbb{1}(c_{1i} = 1) + \mathcal{N}(3,1)\mathbb{1}(c_{1i} = 2)$$

Then, from layer ℓ to layer $\ell + 1$, 2% of the observations are selected at random and moved to the cluster they were not assigned to.

Model: t-HDP model with univariate Normal kernel with mean μ and variance σ^2. The prior distribution for the mean is Normal centred in 0 and variance equal to $\sigma^2/0.1$. The prior for the precision $1/\sigma^2$ is a Gamma distribution with shape and rate parameters equal to 0.1. The concentration parameters of the t-HDP have prior Gamma with rate and shape parameters equal to 3. Compare with independent k-mean at each layer.

Algorithm: 70 000 iterations of the block partially collapsed conditional sampler in Algorithm 4 are performed and the first 20 000 are disregarded as burn-in. The chain is initialized to the k-means solutions computed independently for each layer.

Results: see Section 7.1.

Scenario C

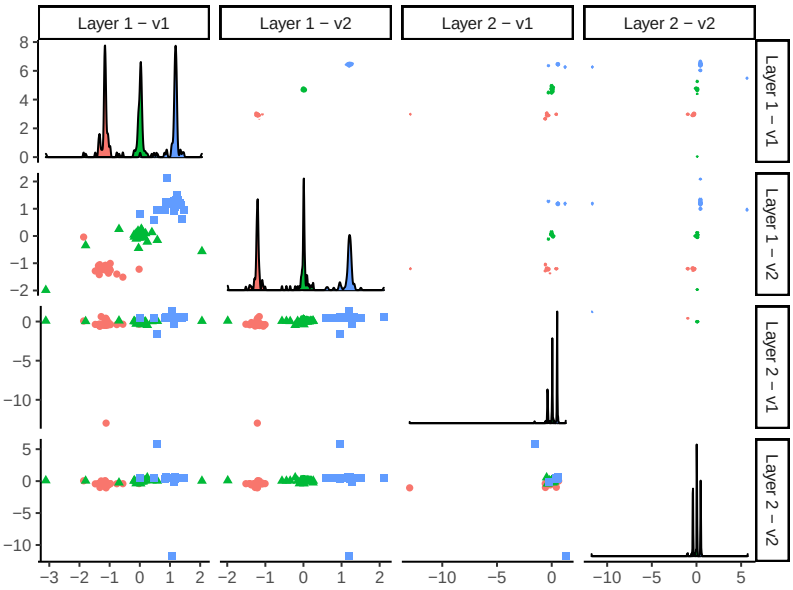


Figure S5.6: Simulation study: simulated data for Scenario C. Colours and shapes denote the true clustering at layer 1. The diagonal plots show the marginal distribution of each variable at each layer, colour coded according to the clustering allocation at layer 1. Upper and lower off-diagonal plots display the joint distribution of two pairs of variables, colour coded according to the clustering allocation at layer 1.

Simulating scenario description: Data for two layers are simulated. At each layer, there are three clusters and data are bi-variate. In particular, at layer 1 approximately

one-third of the dataset forms the first cluster, i.e., $c_{1i} = 1$ for $i = 1, \dots, 66$, approximately one-third forms the second cluster, i.e. $c_{1i} = 2$ for $i = 67, \dots, 132$ and the remaining observations form a third cluster. At layer 1, bivariate values are sampled from bivariate student t distributions

$$X_{1i} \mid c_{1i} \overset{ind}{\sim} \mathcal{T}_2(\boldsymbol{\mu_1}, 1, \Sigma_1)\mathbb{1}(c_{1i} = 1) + \mathcal{T}_2(\boldsymbol{\mu_2}, 1, \Sigma_2)\mathbb{1}(c_{1i} = 2) + \mathcal{T}_2(\boldsymbol{\mu_3}, 1, \Sigma_3)\mathbb{1}(c_{1i} = 3)$$

where $\mathcal{T}_2(\boldsymbol{\mu}, \nu, \Sigma)$ denotes a bivariate t-Student distribution with ν degrees of freedom, centered in $\boldsymbol{\mu}$ and with scale matrix given by Σ.

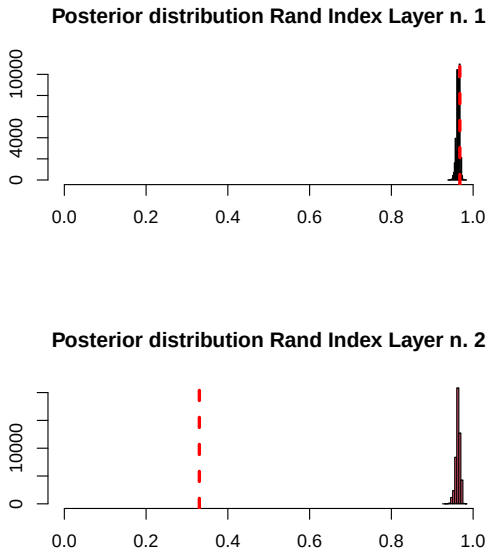


Figure S5.7: Scenario C: posterior distributions of Rand indexes between the posterior and the truth for t-HDP model. Red dashed vertical lines denote the Rand indexes corresponding to the k-means' solution.

Then, from layer 1 to layer 2, 5% of the observations in the first two clusters are selected at random and moved to the cluster they were not assigned to, while the third cluster is kept constant. Bivariate values for the second layer are sampled from

$$\begin{aligned} X_{2i} \mid c_{2i} \overset{ind}{\sim} & \mathcal{T}_2(\boldsymbol{\mu_1}, 1, \Sigma_1)\mathbb{1}(c_{2i} = 1) \\ & + \mathcal{T}_2(\boldsymbol{\mu_2}, 1, \Sigma_2)\mathbb{1}(c_{2i} = 2) \\ & + \mathcal{T}_2(\boldsymbol{\mu_3}, 1, \Sigma_3)\mathbb{1}(c_{2i} = 3) \end{aligned}$$

The true clusters' means are $\boldsymbol{\mu_1} = (0, 0)^T$, $\boldsymbol{\mu_2} = (4, 4)^T$, and $\boldsymbol{\mu_3} = (8, 8)^T$.

Model: t-HDP model with univariate Normal kernel with mean $\boldsymbol{\mu}$ and diagonal variance and covariance matrix Σ^2. The prior distribution for the mean and variance and covariance matrix is a Normal-Inverse-Chi-Squared-distribution, in particular, for $j = 1, 2$, μ_j are a priori independent and Normal distributed with mean 0 and variance $\sigma_j^2/0.1$, while σ_j^2 are independently distributed accordingly to an inverse Chi-Squared with 1 degrees of freedom. The concentration parameters of the t-HDP have prior Gamma with rate and shape parameters equal to 3.

Algorithm: 100 000 iterations of the block partially collapsed conditional sampler in Algorithm 4 are performed and the first half is disregarded as burn-in. The chain is initialized to the k-means solutions computed independently for each layer for 10 clusters.

Results: The Rand index between the true configuration and the point estimates derived minimizing the variation of information loss function (Meilă, 2007; Wade and Ghahramani, 2018) are 0.97 and 0.96 for layer 1 and layer 2 respectively. The same values obtained with two independent k-means algorithms where the number of clusters is chosen based on the gap statistics (Tibshirani et al., 2001), are respectively 0.97 and 0.33. Figure S5.7 shows the distribution of the Rand index between the true clustering configuration and the configurations visited by the posterior algorithm of the t-HDP model after burn-in.

S6 Sensitivity to first-layer prior in t-HDP

In this section, we compare the results obtained using the t-HDP, as described in Section 4, with a variant of the model in which the first layer marginally follows a Dirichlet process mixture model, while the specification of subsequent layers remains unchanged. That is, the variant of the model is given by

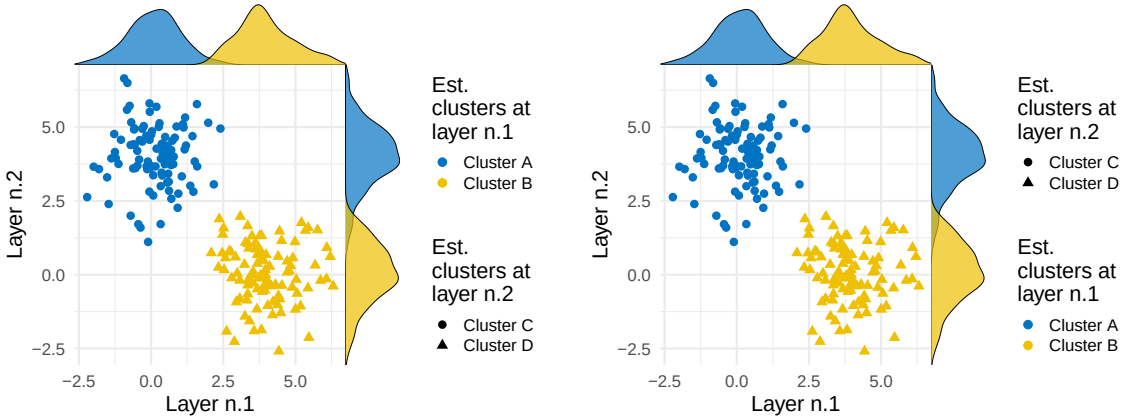
$$X_{1i} \mid \tilde{p}_1 \overset{iid}{\sim} \int_{\Theta_1} k_1(X_{1i}, \theta)\tilde{p}_1(\mathrm{d}\theta), \qquad \tilde{p}_1 \mid \gamma \sim DP(\gamma, P_\theta),$$

whereas the second-layer conditional law is

$$X_{2i} \mid \boldsymbol{c}_1, (\tilde{p}_{21}, \tilde{p}_{22}, \ldots) \overset{ind}{\sim} \int f(X_{2i}, \theta)\tilde{p}_{2c_i}(\mathrm{d}\theta),$$
$$\tilde{p}_{2m} \mid \alpha, \tilde{q}_0 \overset{iid}{\sim} DP(\alpha, \tilde{q}_0), \qquad \tilde{q}_0 \mid \alpha_0 \sim DP(\alpha_0, P_\xi),$$

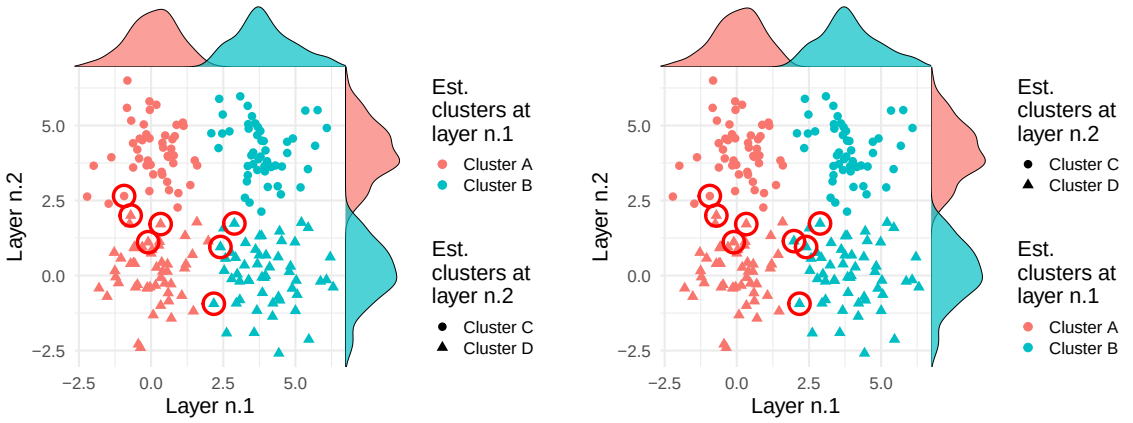
where $DP(\alpha, P)$ denotes a Dirichlet process with concentration parameter α and base distribution P.
The comparison is derived for the simulation studies A, B, and 1 as described in the previous section and summarized in Table S5.1. The kernel, base measures, and priors on the concentration parameters remain unchanged between the two specifications and are as described in the previous section for each simulation study. The number of iterations and the burn-in period are also as specified in the previous section.

Scenario n.A and n.B



(a) t-HDP as in Section 4 - clustering (b) t-HDP with DP variant

Figure S6.1: Simulation study: results for Scenario A. Red circles denote observations assigned to the wrong cluster at least for one layer. The absence of red circles denotes perfect recovery.



(a) t-HDP as in Section 4 - clustering

(b) t-HDP with DP variant

Figure S6.2: Simulation study: results for Scenario b. Red circles denote observations assigned to the wrong cluster at least for one layer.

Scenario n.1

	Rand Index					# Mistakes				
Layer	k-means	t-HDP	LSBP	E-DP	DP-t-HDP	k-means	t-HDP	LSBP	E-DP	DP-t-HDP
n.1	0.98	0.98	0.98	0.50	0.98	2	2	2	100	2
n.2	0.98	1.00	0.98	0.90	1.00	2	0	2	10	0
n.3	0.92	0.98	0.92	1.00	0.98	8	2	8	0	2
n.4	0.98	1.00	0.98	0.92	1.00	2	0	2	17	0
n.5	0.92	0.97	0.91	0.89	0.97	8	3	9	21	3
n.6	0.97	0.98	0.97	0.86	1.00	3	2	3	31	0
n.7	0.94	0.99	0.92	0.83	0.99	6	1	8	40	1
n.8	0.95	1.00	0.95	0.79	1.00	5	0	5	44	0
n.9	0.93	1.00	0.93	0.79	1.00	7	0	7	47	0
n.10	0.91	0.99	0.89	0.75	0.99	9	1	11	54	1
average	0.95	0.99	0.83	0.82	0.99	5.2	1.1	5.7	36.4	0.9

Table S5.1: Scenario 1, Rand indexes between the estimated and true clustering configurations and numbers of items allocated to the wrong cluster. The variant under study is denoted with DP-t-HDP

S7 Application to metabolic concentrations in obese children: additional details and results

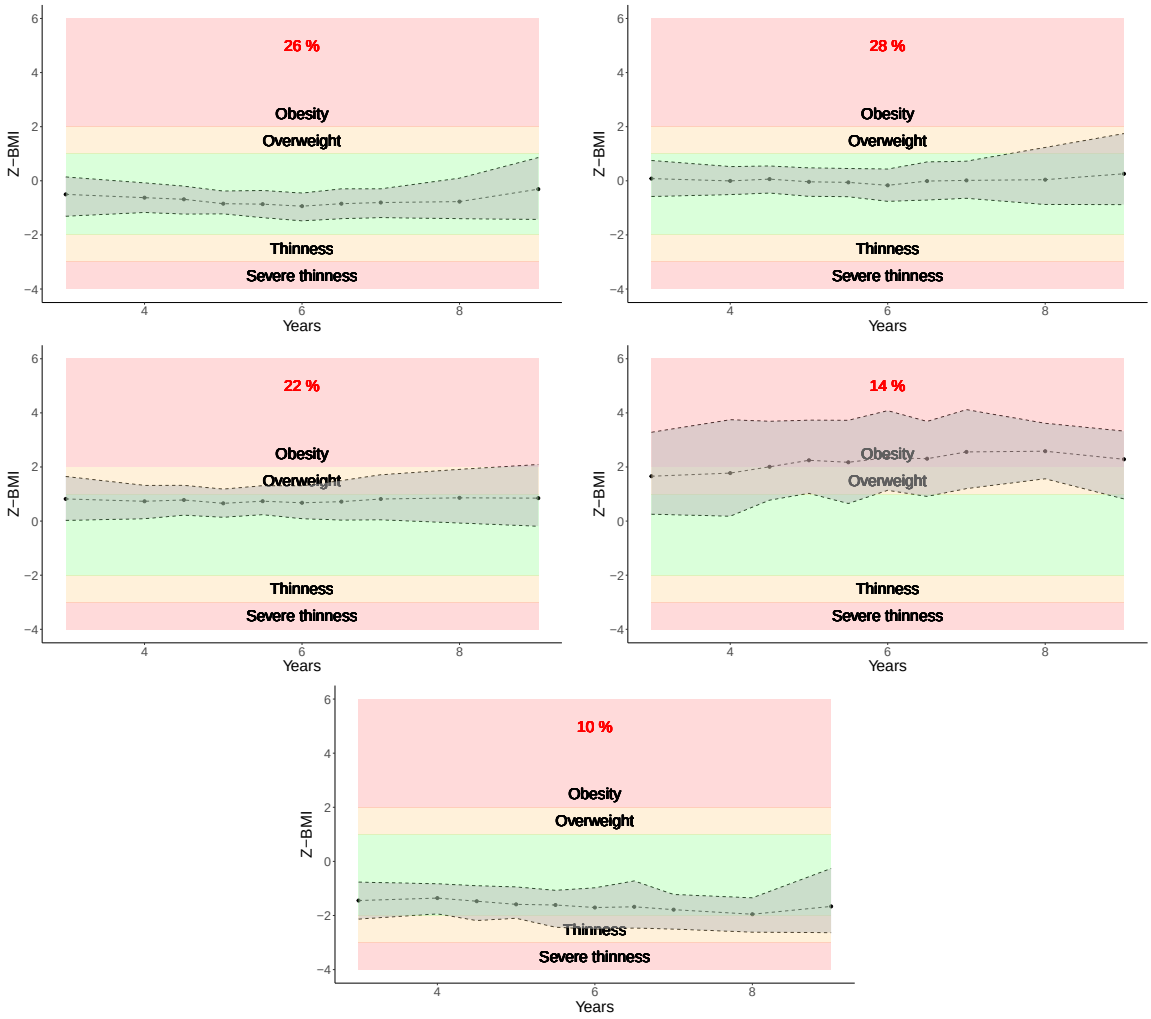


Figure S7.1: Average growth trajectories in the five estimated clusters at the z-BMI layer, shaded area includes 95% of the observations assigned to the cluster, bands in the background correspond to WHO classification of growth trajectories into Obesity, Overweight, Normal, Thinness, and Severe thinness. Percentages correspond to the proportions of children assigned to each of the five clusters.

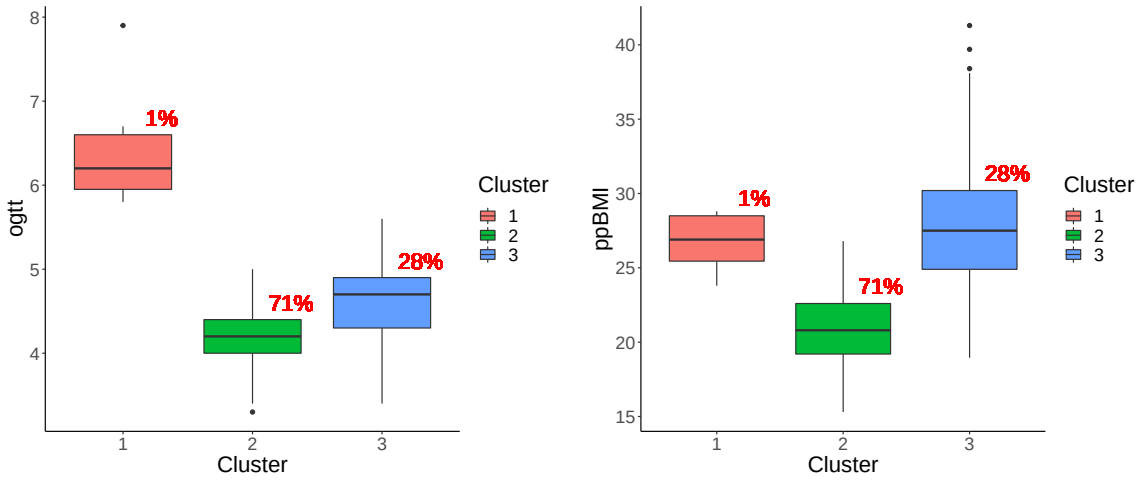


Figure S7.2: Boxplots by cluster assignment of the variables OGTT and PPBMI corresponding to the mother layer. Percentages correspond to the proportions of mothers assigned to each of the three clusters.

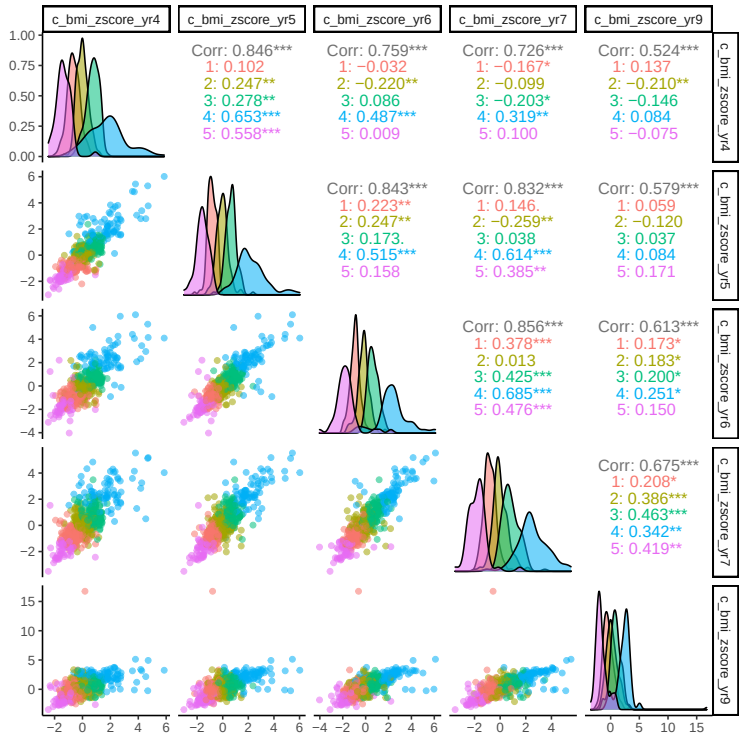


Figure S7.3: Scatter plots, density estimates and correlation values of z-BMI scores by cluster assignment at years 4, 5, 6, 7, and 9. Colours denote the cluster assignment of the children at the growth trajectory layer. The diagonal plots show the marginal distribution of the z-BMIs at each time point, colour coded according to the clustering allocation. Upper off-diagonal plots display the correlation between any two pairs of time points, overall and by cluster. Lower off-diagonal plots show the scatter plot of the data, colour coded according to the clustering allocation.

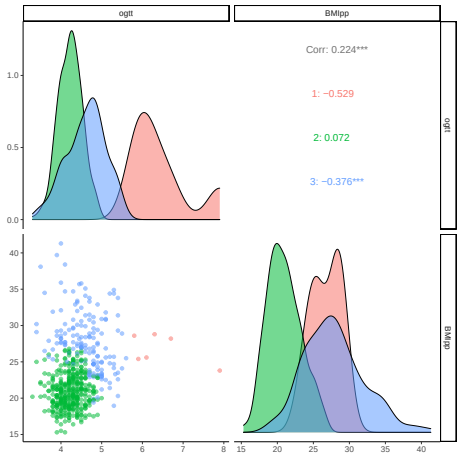


Figure S7.4: Scatter plots, density estimates and correlation values of the OGTT and PPBMI variables. Colours denote the cluster assignment at the mother layer. The diagonal plots show the marginal distribution of OGTT and PPBMI. Upper off-diagonal plots display the correlation between the two variables overall and by cluster. Lower off-diagonal plots show the scatter plot of the data.

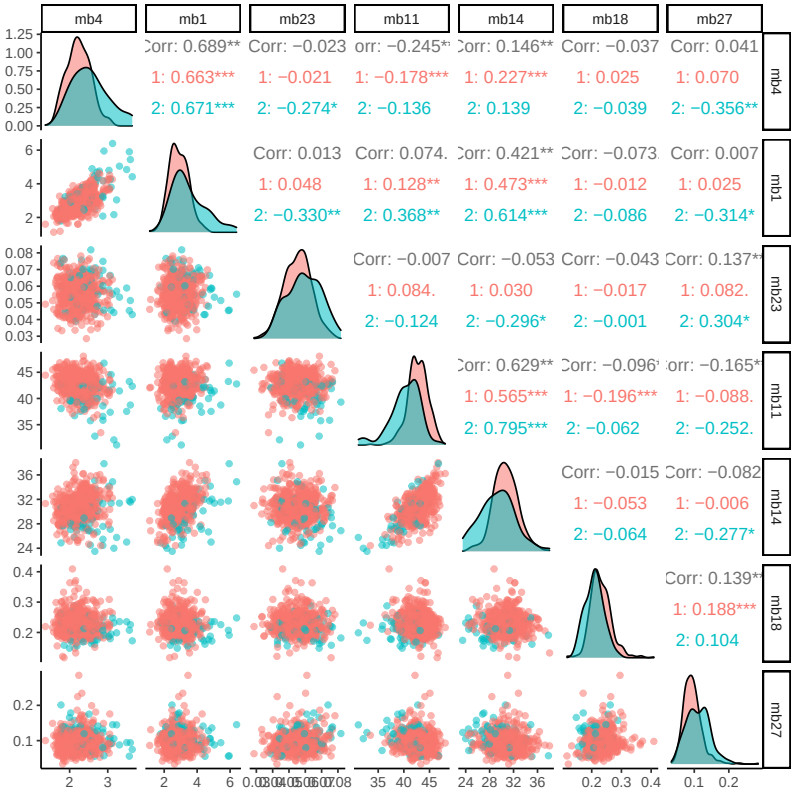


Figure S7.5: Scatter plots, density estimates and correlation values of seven randomly selected metabolites. Colors denote the cluster assignment at the metabolites layer. The diagonal plots show the marginal distributions. Upper off-diagonal plots display the correlation overall and by cluster. Lower off-diagonal plots show the scatter plot of the data.

Mother	Metabolites	Growth trajectory					Total
		Underweight	*Normal low*	*Normal*	*Normal high*	*Obesity*	
Low	*Conf. 1*	45	107	112	68	26	64.74%
	Conf. 2	1	11	5	3	12	5.79%
High	*Conf. 1*	8	20	39	42	21	23.51%
	Conf. 2	0	2	1	7	17	4.88%
Outliers	*Conf. 1*	0	2	0	2	1	0.90%
	Conf. 2	0	0	0	0	1	0.18%
Total		9.76%	25.68%	28.39%	22.06%	14.10%	

Table S7.1: Three-way cross-table of the estimated clustering configurations. Values within the table are absolute frequencies; the last row indicates the percentages of children in different growth trajectory clusters; the last column contains percentages of children assigned to different combinations of mother and metabolites clusters.

	Average		IQR		Kruskal-Wallis
Metabolite	cluster 1	cluster 2	cluster 1	cluster 2	p-value
Clinical LDL Cholesterol	2.8918	3.4800	0.8383	1.3361	0.0000
HDL Cholesterol	1.6027	1.5545	0.3234	0.4035	0.0366
Triglycerides	0.7663	1.2641	0.3232	0.7728	0.0000
Phosphoglycerides	2.2372	2.5376	0.4324	0.7174	0.0000
Cholines Phosphoglycerides	2.5614	2.8604	0.4464	0.6645	0.0000
Sphingomyelins	0.5001	0.5468	0.0936	0.1424	0.0014
APO A1	1.4819	1.4974	0.2747	0.3638	0.7219
APO B	0.8092	0.9997	0.2181	0.3844	0.0000
Omega 3	0.4182	0.4702	0.1377	0.1862	0.0077
Omega 6	4.2964	4.7016	0.5689	0.8934	0.0000
Poly-Unsaturated FA (PUFA)	42.7171	40.2715	2.5448	3.4999	0.0000
Mono-Unsaturated FA (MUFA)	23.2455	24.8581	1.8430	2.8237	0.0000
Saturated FA (SFA)	34.0375	34.8704	1.0974	1.5096	0.0000
Linoleic acid	30.7019	29.3047	2.7123	4.0315	0.0001
Docosahexaenoic acid (DHA)	2.1145	1.9018	0.5387	0.4947	0.0005
Alanine	0.3079	0.3502	0.0901	0.1114	0.0000
Glutamine	0.5775	0.5375	0.1343	0.1423	0.0026
Glycine	0.2310	0.2091	0.0436	0.0412	0.0000
Histidine	0.0877	0.0878	0.0128	0.0136	0.8621
Isoleucine	0.0512	0.0650	0.0126	0.0126	0.0000
Leucine	0.1020	0.1230	0.0203	0.0254	0.0000
Valine	0.2311	0.2722	0.0422	0.0395	0.0000
Phenylalanine	0.0553	0.0594	0.0114	0.0140	0.0032
Tyrosine	0.0686	0.0802	0.0139	0.0224	0.0000
Glucose	4.8412	4.9235	0.5471	0.4521	0.0219
Lactate	2.0602	2.5197	0.7823	0.8316	0.0000
Pyruvate	0.0952	0.1108	0.0314	0.0461	0.0001
Citrate	0.1059	0.1034	0.0176	0.0198	0.1641
beta-Hydroxybutyric acid	0.1237	0.2036	0.1257	0.1227	0.4402
Acetate	0.0357	0.0294	0.0156	0.0101	0.0000
Acetoacetate	0.0434	0.0709	0.0406	0.0468	0.5898
Acetone	0.0183	0.0258	0.009	0.0091	0.2576
Creatinine	45.3778	47.9555	9.1402	12.7015	0.0318
Albumin	42.5924	43.4704	3.7106	4.6384	0.3151
Glycoprotein acetyls	0.8306	0.9525	0.1344	0.2099	0.0000

Table S7.2: Summary of metabolite clusters: average concentration of each metabolite by cluster, interquartile range (IQR) of the metabolites concentration distribution by cluster and p-value of the Kruskal-Wallis test for difference in distribution between the two clusters.

S8 Computational cost and mixing performance of posterior algorithms

Simulation study n.1: n=200, P=2, L=2			
ESS / N layer n.1	ESS / N layer n.2	time for 1 iteration	time for 1000 effective draws
0.03830	0.02410	0.0132 sec	9.12 min

Simulation study n.2: n=200, P=2, L=2			
ESS / N layer n.1	ESS / N layer n.2	time for 1 iteration	time for 1000 effective draws
0.05762	0.04580	0.0134 sec	4.88 min

Simulation study n.3: n=200, P=10, L=10				
ESS / N layer n.1	ESS / N layer n.5	ESS / N layer n.10	time for 1 iteration	time for 1000 effective draws
0.09904	0.05823	0.03056	0.072 sec	39.26 min

Simulation study n.4: n=200, P=100, L=100				
ESS / N layer n.1	ESS / N layer n.50	ESS / N layer n.100	time for 1 iteration	time for 1000 effective draws
0.0254	0.02035	0.0872	1.028 sec	841 min

Table S8.1: Effective sample size per iteration (ESS/N) after burn-in for the Rand index between chain and truth, time in seconds per iteration, and time in minutes for 1000 effective draws. The latter is computed as the maximum of the value $(time) \times 1000/(ESS/N)$ across layers. n denotes the sample size, P is the total number of considered variables, and L is the total number of layers to which the variables are assigned. Algorithms are coded in `R` and run on Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz CPU.

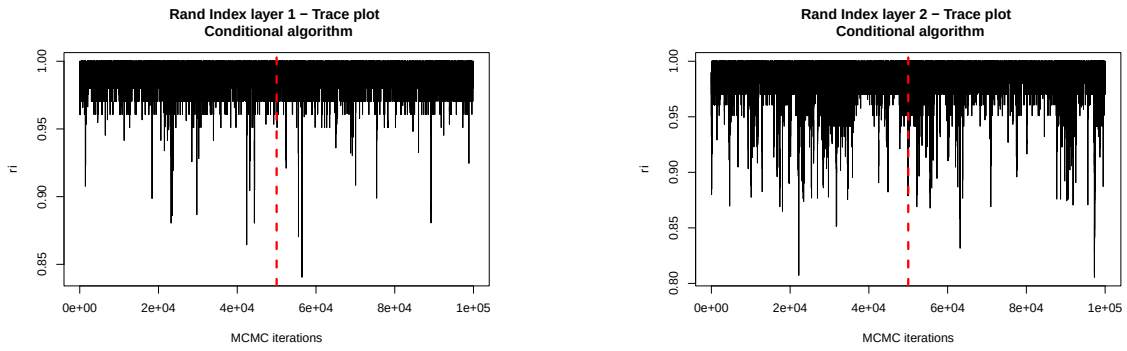


Figure S8.1: Simulation study: Scenario n.1. Trace plots of the Rand index between the chain configuration and the true configuration. Vertical dashed lines locate the burn-in period.

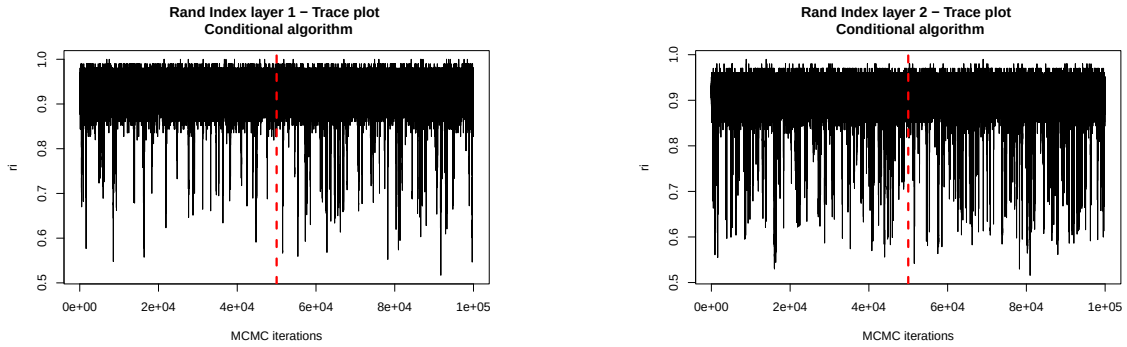


Figure S8.2: Simulation study: Scenario n.2. Trace plots of the Rand index between the chain configuration and the true configuration. Vertical dashed lines locate the burn-in period.

References

Camerlenghi, F., A. Lijoi, and I. Prünster (2018). Bayesian nonparametric inference beyond the Gibbs-type framework. *Scandinavian Journal of Statistics 45*(4), 1062–1091.

De Finetti, B. (1937). La prévision: ses lois logiques, ses sources subjectives. In *Annales de l'institut Henri Poincaré*, Volume 7, pp. 1–68.

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems. *The Annals of Statistics 1*(2), 209–230.

Ferguson, T. S. (1983). Bayesian density estimation by mixtures of normal distributions. In *Recent advances in statistics*, pp. 287–302. Elsevier.

Ghosal, S. and A. Van der Vaart (2017). *Fundamentals of nonparametric Bayesian inference*, Volume 44. Cambridge University Press.

Lo, A. Y. (1984). On a class of Bayesian nonparametric estimates: I. Density estimates. *The Annals of Statistics 12*(1), 351–357.

MacEachern, S. N. (2000). Dependent Dirichlet processes. Technical report, The Ohio State Univ.

Meilă, M. (2007). Comparing clusterings–an information based distance. *Journal of Multivariate Analysis 98*(5), 873–895.

Page, G. L., F. A. Quintana, and D. B. Dahl (2022). Dependent modeling of temporal sequences of random partitions. *Journal of Computational and Graphical Statistics 31*(2), 614–627.

Pitman, J. (1996). Some developments of the Blackwell-MacQueen urn scheme. *Lecture Notes-Monograph Series*, 245–267.

Pitman, J. and M. Yor (1997). The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *The Annals of Probability 25*(2), 855–900.

Ren, L., L. Du, L. Carin, and D. B. Dunson (2011). Logistic stick-breaking process. *Journal of Machine Learning Research 12*, 203–239.

Rodríguez, A., D. B. Dunson, and A. E. Gelfand (2008). The nested Dirichlet process (with discussion). *Journal of the American Statistical Association 103*(483), 1131–1154.

Teh, Y., M. Jordan, M. Beal, and D. Blei (2006). Hierarchical Dirichlet processes. *Journal of the American Statistical Association 101*(476), 1566–1581.

Tibshirani, R., G. Walther, and T. Hastie (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63*(2), 411–423.

Van Dyk, D. A. and T. Park (2008). Partially collapsed gibbs samplers: Theory and methods. *Journal of the American Statistical Association 103*(482), 790–796.

Wade, S. and Z. Ghahramani (2018). Bayesian cluster analysis: point estimation and credible balls (with discussion). *Bayesian Analysis 13*(2), 559–626.

Wade, S., S. Mongelluzzo, and S. Petrone (2011). An enriched conjugate prior for Bayesian nonparametric inference. *Bayesian Analysis 6*(3), 359–385.