Corradin, Riccardo[1]

joint work with Mario Beraha[2]

[1] *University of Milano-Bicocca*

Department of Economics, Management and Statistics

[2] *Politecnico di Milano*

Department of Mathematics

Approximate estimation of latent random partitions

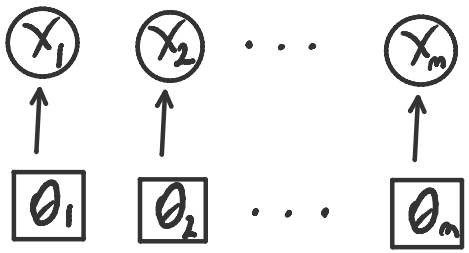
ABC (not) in Svalbard, 13th April 2021

Latent random partitions in BNP

BNP & model based clustering in a nutshell

Let $\boldsymbol{X}_{1:n} = \{X_1, \ldots, X_n\}$ be a set of observations taking values on a regular space $\mathbb{X}$, endowed with its σ-algebra $\mathscr{X}$.

We assume $\boldsymbol{X}_{1:n} = \{X_1, \ldots, X_n\}$ indexed by a set of **exchangeable** latent parameters $\boldsymbol{\theta}_{1:n} = \{\theta_1, \ldots, \theta_n\}$, with the generic θ_i taking values on Θ Polish space, endowed with its σ-algebra $\mathcal{T}$.



The latent parameter θ_i acts on the distribution of X_i through a **kernel** function

$$\mathcal{K}(x, \theta) : \mathbb{X} \times \Theta \to \mathbb{R}^+.$$

The distribution of $\boldsymbol{\theta}_{1:n} = \{\theta_1, \ldots, \theta_n\}$ is governed by a **random probability measure**

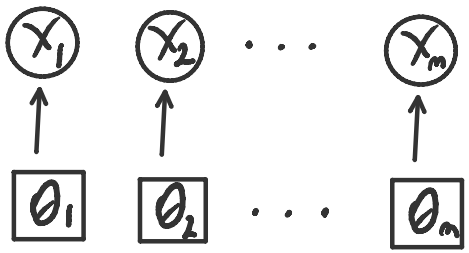
$$\tilde{p} = \sum_{j=1}^{\infty} p_j \delta_{\theta_j^*},$$

with $\{p_j\}_{j \geq 1}$ sequence of random weights, $p_j \geq 0$, $\sum_{j=1}^{\infty} p_j = 1$ a.s., $\{\theta_j^*\}_{j \geq 1}$ i.i.d from P_0 diffuse probability measure on $(\Theta, \mathcal{T})$, and $\tilde{p}$ takes values on $\mathbb{P}_\Theta$ space of a.s. discrete probability measures with support Θ.

BNP & model based clustering in a nutshell

Let $\boldsymbol{X}_{1:n} = \{X_1, \ldots, X_n\}$ be a set of observations taking values on a regular space $\mathbb{X}$, endowed with its σ-algebra $\mathscr{X}$.

We assume $\boldsymbol{X}_{1:n} = \{X_1, \ldots, X_n\}$ indexed by a set of **exchangeable** latent parameters $\boldsymbol{\theta}_{1:n} = \{\theta_1, \ldots, \theta_n\}$, with the generic θ_i taking values on Θ Polish space, endowed with its σ-algebra $\mathcal{T}$.



The latent parameter θ_i acts on the distribution of X_i through a **kernel** function

$$\mathcal{K}(x, \theta) : \mathbb{X} \times \Theta \to \mathbb{R}^+.$$

The distribution of $\theta_{1:n} = \{\theta_1, \ldots, \theta_n\}$ is governed by a **random probability measure**

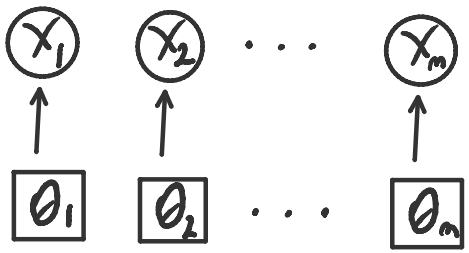
$$\tilde{p} = \sum_{j=1}^{\infty} p_j \delta_{\theta_j^*},$$

with $\{p_j\}_{j \geq 1}$ sequence of random weights, $p_j \geq 0$, $\sum_{j=1}^{\infty} p_j = 1$ a.s., $\{\theta_j^*\}_{j \geq 1}$ i.i.d from P_0 diffuse probability measure on $(\Theta, \mathcal{T})$, and $\tilde{p}$ takes values on $\mathbb{P}_\Theta$ space of a.s. discrete probability measures with support Θ.

BNP & model based clustering in a nutshell

Let $\boldsymbol{X}_{1:n} = \{X_1, \ldots, X_n\}$ be a set of observations taking values on a regular space $\mathbb{X}$, endowed with its σ-algebra $\mathscr{X}$.

We assume $\boldsymbol{X}_{1:n} = \{X_1, \ldots, X_n\}$ indexed by a set of **exchangeable** latent parameters $\boldsymbol{\theta}_{1:n} = \{\theta_1, \ldots, \theta_n\}$, with the generic θ_i taking values on Θ Polish space, endowed with its σ-algebra $\mathcal{T}$.



The latent parameter θ_i acts on the distribution of X_i through a **kernel** function

$$\mathcal{K}(x, \theta) : \mathbb{X} \times \Theta \to \mathbb{R}^+.$$

The distribution of $\theta_{1:n} = \{\theta_1, \ldots, \theta_n\}$ is governed by a **random probability measure**

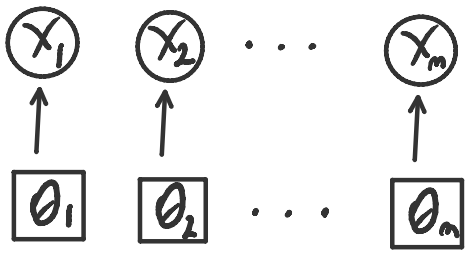
$$\tilde{p} = \sum_{j=1}^{\infty} p_j \delta_{\theta_j^*},$$

with $\{p_j\}_{j \geq 1}$ sequence of random weights, $p_j \geq 0$, $\sum_{j=1}^{\infty} p_j = 1$ a.s., $\{\theta_j^*\}_{j \geq 1}$ i.i.d from P_0 diffuse probability measure on $(\Theta, \mathcal{T})$, and $\tilde{p}$ takes values on $\mathbb{P}_\Theta$ space of a.s. discrete probability measures with support Θ.

BNP & model based clustering in a nutshell

Let $\boldsymbol{X}_{1:n} = \{X_1, \ldots, X_n\}$ be a set of observations taking values on a regular space $\mathbb{X}$, endowed with its σ-algebra $\mathscr{X}$.

We assume $\boldsymbol{X}_{1:n} = \{X_1, \ldots, X_n\}$ indexed by a set of **exchangeable** latent parameters $\boldsymbol{\theta}_{1:n} = \{\theta_1, \ldots, \theta_n\}$, with the generic θ_i taking values on Θ Polish space, endowed with its σ-algebra $\mathcal{T}$.



The latent parameter θ_i acts on the distribution of X_i through a **kernel** function

$$\mathcal{K}(x, \theta) : \mathbb{X} \times \Theta \to \mathbb{R}^+.$$

The distribution of $\boldsymbol{\theta}_{1:n} = \{\theta_1, \ldots, \theta_n\}$ is governed by a **random probability measure**

$$\tilde{p} = \sum_{j=1}^{\infty} p_j \delta_{\theta_j^*},$$

with $\{p_j\}_{j \geq 1}$ sequence of random weights, $p_j \geq 0$, $\sum_{j=1}^{\infty} p_j = 1$ a.s., $\{\theta_j^*\}_{j \geq 1}$ i.i.d from P_0 diffuse probability measure on $(\Theta, \mathcal{T})$, and $\tilde{p}$ takes values on $\mathbb{P}_\Theta$ space of a.s. discrete probability measures with support Θ.

BNP & model based clustering in a nutshell

We then have that $X_1, \ldots, X_n$ are distributed according to a mixture model, with

$$f(x) = \int_\Theta \mathcal{K}(x, \theta)\tilde{p}(\mathrm{d}\theta) = \sum_{j=1}^{\infty} p_j \mathcal{K}(x, \theta_j^*))$$

$\theta_{1:n}$ are **exchangeable** from $\tilde{p} \Rightarrow$ they are **naturally partitioned** in k groups with unique values $\theta_1^*, \ldots, \theta_k^*$ and frequencies $n_1, \ldots, n_k$.

We denote by $\rho_n = \{A_1, \ldots, A_k\}$ the latent partition of the data, with $i \in A_j$ if $\theta_i = \theta_j^*$. The distribution of ρ_n is described by a **symmetric function** (EPPF)

$$\Pi_k^{(n)}(n_1, \ldots, n_k) = \int_{\Theta^k} \mathbb{E}\left[\prod_{j=1}^{k} \tilde{p}^{n_j}(\mathrm{d}\theta_j^*)\right],$$

while $\theta_1^*, \ldots, \theta_k^*$ are i.i.d. from P_0.

Key quantity: the **predictive distribution** of the latent parameters.

$$\mathscr{L}(\theta_{n+1} \mid \boldsymbol{\theta}_{1:n}) = \int_{\mathbb{M}_\Theta} \mathscr{L}(\theta_{n+1}, \tilde{p} \mid \boldsymbol{\theta}_{1:n})\mathrm{d}\tilde{p}$$

$$= \frac{\Pi_{k+1}^{(n+1)}(n_1, \ldots, n_k, 1)}{\Pi_k^{(n)}(n_1, \ldots, n_k)} P_0(\mathrm{d}\theta_{n+1}) + \sum_{j=1}^{k} \frac{\Pi_k^{(n+1)}(\ldots, n_j + 1, \ldots)}{\Pi_k^{(n)}(n_1, \ldots, n_k)} \delta_{\theta_j^*}(\mathrm{d}\theta_{n+1})$$

with a positive **probability of having a new cluster** at the $n+1$ sampling step, and a positive **probability of allocate the $n+1$ realization in an existent cluster**.

BNP & model based clustering in a nutshell

We then have that $X_1, \ldots, X_n$ are distributed according to a mixture model, with

$$f(x) = \int_\Theta \mathcal{K}(x, \theta)\tilde{p}(\mathrm{d}\theta) = \sum_{j=1}^{\infty} p_j \mathcal{K}(x, \theta_j^*))$$

$\boldsymbol{\theta}_{1:n}$ are **exchangeable** from $\tilde{p} \Rightarrow$ they are **naturally partitioned** in k groups with unique values $\theta_1^*, \ldots, \theta_k^*$ and frequencies $n_1, \ldots, n_k$.

We denote by $\rho_n = \{A_1, \ldots, A_k\}$ the latent partition of the data, with $i \in A_j$ if $\theta_i = \theta_j^*$. The distribution of ρ_n is described by a **symmetric function** (EPPF)

$$\Pi_k^{(n)}(n_1, \ldots, n_k) = \int_{\Theta^k} \mathbb{E}\left[\prod_{j=1}^{k} \tilde{p}^{n_j}(\mathrm{d}\theta_j^*)\right],$$

while $\theta_1^*, \ldots, \theta_k^*$ are i.i.d. from P_0.

Key quantity: the **predictive distribution** of the latent parameters.

$$\mathscr{L}(\theta_{n+1} \mid \boldsymbol{\theta}_{1:n}) = \int_{\mathbb{M}_\Theta} \mathscr{L}(\theta_{n+1}, \tilde{p} \mid \boldsymbol{\theta}_{1:n})\mathrm{d}\tilde{p}$$

$$= \frac{\Pi_{k+1}^{(n+1)}(n_1, \ldots, n_k, 1)}{\Pi_k^{(n)}(n_1, \ldots, n_k)} P_0(\mathrm{d}\theta_{n+1}) + \sum_{j=1}^{k} \frac{\Pi_k^{(n+1)}(\ldots, n_j + 1, \ldots)}{\Pi_k^{(n)}(n_1, \ldots, n_k)} \delta_{\theta_j^*}(\mathrm{d}\theta_{n+1})$$

with a positive **probability of having a new cluster** at the $n+1$ sampling step, and a positive **probability of allocate the $n+1$ realization in an existent cluster**.

BNP & model based clustering in a nutshell

We then have that $X_1, \ldots, X_n$ are distributed according to a mixture model, with

$$f(x) = \int_\Theta \mathcal{K}(x, \theta)\tilde{p}(\mathrm{d}\theta) = \sum_{j=1}^{\infty} p_j \mathcal{K}(x, \theta_j^*))$$

$\boldsymbol{\theta}_{1:n}$ are **exchangeable** from $\tilde{p} \Rightarrow$ they are **naturally partitioned** in k groups with unique values $\theta_1^*, \ldots, \theta_k^*$ and frequencies $n_1, \ldots, n_k$.

We denote by $\rho_n = \{A_1, \ldots, A_k\}$ the latent partition of the data, with $i \in A_j$ if $\theta_i = \theta_j^*$. The distribution of ρ_n is described by a **symmetric function** (EPPF)

$$\Pi_k^{(n)}(n_1, \ldots, n_k) = \int_{\Theta^k} \mathbb{E}\left[\prod_{j=1}^{k} \tilde{p}^{n_j}(\mathrm{d}\theta_j^*)\right],$$

while $\theta_1^*, \ldots, \theta_k^*$ are i.i.d. from P_0.

Key quantity: the **predictive distribution** of the latent parameters.

$$\mathscr{L}(\theta_{n+1} \mid \boldsymbol{\theta}_{1:n}) = \int_{\mathbb{M}_\Theta} \mathscr{L}(\theta_{n+1}, \tilde{p} \mid \boldsymbol{\theta}_{1:n})\mathrm{d}\tilde{p}$$

$$= \frac{\Pi_{k+1}^{(n+1)}(n_1, \ldots, n_k, 1)}{\Pi_k^{(n)}(n_1, \ldots, n_k)} P_0(\mathrm{d}\theta_{n+1}) + \sum_{j=1}^{k} \frac{\Pi_k^{(n+1)}(\ldots, n_j + 1, \ldots)}{\Pi_k^{(n)}(n_1, \ldots, n_k)} \delta_{\theta_j^*}(\mathrm{d}\theta_{n+1})$$

with a positive **probability of having a new cluster** at the $n+1$ sampling step, and a positive **probability of allocate the $n+1$ realization in an existent cluster**.

BNP & model based clustering in a nutshell

We then have that $X_1, \ldots, X_n$ are distributed according to a mixture model, with

$$f(x) = \int_\Theta \mathcal{K}(x, \theta)\tilde{p}(\mathrm{d}\theta) = \sum_{j=1}^{\infty} p_j \mathcal{K}(x, \theta_j^*))$$

$\boldsymbol{\theta}_{1:n}$ are **exchangeable** from $\tilde{p} \Rightarrow$ they are **naturally partitioned** in k groups with unique values $\theta_1^*, \ldots, \theta_k^*$ and frequencies $n_1, \ldots, n_k$.

We denote by $\rho_n = \{A_1, \ldots, A_k\}$ the latent partition of the data, with $i \in A_j$ if $\theta_i = \theta_j^*$. The distribution of ρ_n is described by a **symmetric function** (EPPF)

$$\Pi_k^{(n)}(n_1, \ldots, n_k) = \int_{\Theta^k} \mathbb{E}\left[\prod_{j=1}^{k} \tilde{p}^{n_j}(\mathrm{d}\theta_j^*)\right],$$

while $\theta_1^*, \ldots, \theta_k^*$ are i.i.d. from P_0.

Key quantity: the **predictive distribution** of the latent parameters.

$$\begin{aligned}\mathscr{L}(\theta_{n+1} \mid \boldsymbol{\theta}_{1:n}) &= \int_{\mathbb{M}_\Theta} \mathscr{L}(\theta_{n+1}, \tilde{p} \mid \boldsymbol{\theta}_{1:n})\mathrm{d}\tilde{p} \\ &= \frac{\Pi_{k+1}^{(n+1)}(n_1, \ldots, n_k, 1)}{\Pi_k^{(n)}(n_1, \ldots, n_k)} P_0(\mathrm{d}\theta_{n+1}) + \sum_{j=1}^{k} \frac{\Pi_k^{(n+1)}(\ldots, n_j + 1, \ldots)}{\Pi_k^{(n)}(n_1, \ldots, n_k)} \delta_{\theta_j^*}(\mathrm{d}\theta_{n+1})\end{aligned}$$

with a positive **probability of having a new cluster** at the $n+1$ sampling step, and a positive **probability of allocate the $n+1$ realization in an existent cluster**.

The target of our inference

We are interested into **performing inference** on the **latent partition** of the data, with

$$\mathscr{L}(\boldsymbol{X}_{1:n} \mid \boldsymbol{\theta}_{1:n}) = \prod_{i=1}^{n} \mathcal{K}(X_i, \theta_i) = \prod_{j=1}^{k} \prod_{i \in A_j} \mathcal{K}(X_i, \theta_j^*) = \mathscr{L}(X_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*)$$

and a priori $\pi(\boldsymbol{\theta}_{1:n}) = \pi(\rho_n)\pi(\boldsymbol{\theta}_{1:k}^*)$, with $\pi(\rho_n)$ corresponding to the **EPPF** of $\tilde{p}$.

Our inferential interest is on the latent partition ρ_n, we want to produce a sample from

$$\pi(\rho_n \mid \boldsymbol{X}_{1:n}) = \frac{\pi(\rho_n) \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}{\sum_{\rho_n \in \mathbb{P}_n} \pi(\mathrm{d}\rho_n) \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}$$

with $\mathbb{P}_n$ space of possible partition of $\{1, \ldots, n\}$.

The estimation can be mathematically or computationally hardly tractable / intractable. Instead of $\pi(\rho_n \mid \boldsymbol{X}_{1:n})$, we want to sample from

$$\pi_\varepsilon(\rho_n \mid \boldsymbol{X}_{1:n}) = \frac{\pi(\rho_n) \int_{\mathbb{X}^n} \mathbb{1}_{[d(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) < \varepsilon]} \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}{\sum_{\rho_n \in \mathbb{P}_n} \pi(\mathrm{d}\rho_n) \int_{\mathbb{X}^n} \mathbb{1}_{[d(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) < \varepsilon]} \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}$$

for some **distance** $d(\cdot, \cdot) : \mathbb{X}^n \times \mathbb{X}^n \to \mathbb{R}$.

The target of our inference

We are interested into **performing inference** on the **latent partition** of the data, with

$$\mathscr{L}(\boldsymbol{X}_{1:n} \mid \boldsymbol{\theta}_{1:n}) = \prod_{i=1}^{n} K(X_i, \theta_i) = \prod_{j=1}^{k} \prod_{i \in A_j} K(X_i, \theta_j^*) = \mathscr{L}(X_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*)$$

and a priori $\pi(\boldsymbol{\theta}_{1:n}) = \pi(\rho_n)\pi(\boldsymbol{\theta}_{1:k}^*)$, with $\pi(\rho_n)$ corresponding to the **EPPF** of $\tilde{p}$.

Our inferential interest is on the latent partition ρ_n, we want to produce a sample from

$$\pi(\rho_n \mid \boldsymbol{X}_{1:n}) = \frac{\pi(\rho_n) \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}{\sum_{\rho_n \in \mathbb{P}_n} \pi(\mathrm{d}\rho_n) \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}$$

with $\mathbb{P}_n$ space of possible partition of $\{1, \ldots, n\}$.

The estimation can be mathematically or computationally hardly tractable / intractable. Instead of $\pi(\rho_n \mid \boldsymbol{X}_{1:n})$, we want to sample from

$$\pi_\varepsilon(\rho_n \mid \boldsymbol{X}_{1:n}) = \frac{\pi(\rho_n) \int_{\mathbb{X}^n} \mathbb{1}_{[d(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) < \varepsilon]} \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}{\sum_{\rho_n \in \mathbb{P}_n} \pi(\mathrm{d}\rho_n) \int_{\mathbb{X}^n} \mathbb{1}_{[d(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) < \varepsilon]} \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}$$

for some **distance** $d(\cdot, \cdot) : \mathbb{X}^n \times \mathbb{X}^n \to \mathbb{R}$.

The target of our inference

We are interested into **performing inference** on the **latent partition** of the data, with

$$\mathscr{L}(\boldsymbol{X}_{1:n} \mid \boldsymbol{\theta}_{1:n}) = \prod_{i=1}^{n} K(X_i, \theta_i) = \prod_{j=1}^{k} \prod_{i \in A_j} K(X_i, \theta_j^*) = \mathscr{L}(X_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*)$$

and a priori $\pi(\boldsymbol{\theta}_{1:n}) = \pi(\rho_n)\pi(\boldsymbol{\theta}_{1:k}^*)$, with $\pi(\rho_n)$ corresponding to the **EPPF** of $\tilde{p}$.

Our inferential interest is on the latent partition ρ_n, we want to produce a sample from

$$\pi(\rho_n \mid \boldsymbol{X}_{1:n}) = \frac{\pi(\rho_n) \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}{\sum_{\rho_n \in \mathbb{P}_n} \pi(\mathrm{d}\rho_n) \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}$$

with $\mathbb{P}_n$ space of possible partition of $\{1, \ldots, n\}$.

The estimation can be mathematically or computationally hardly tractable / intractable. Instead of $\pi(\rho_n \mid \boldsymbol{X}_{1:n})$, we want to sample from

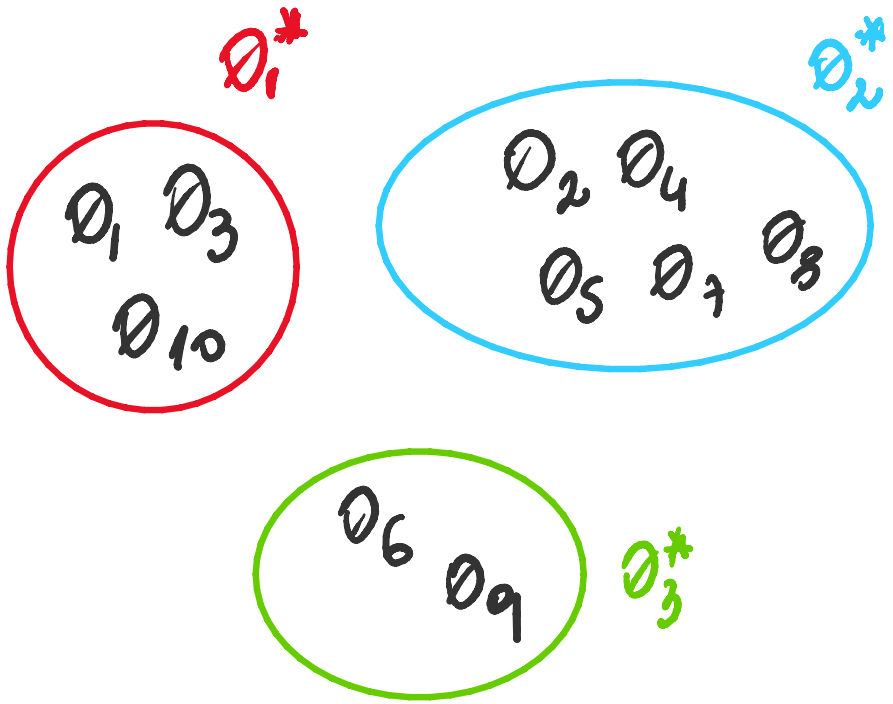
$$\pi_\varepsilon(\rho_n \mid \boldsymbol{X}_{1:n}) = \frac{\pi(\rho_n) \int_{\mathbb{X}^n} \mathbb{1}_{[d(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) < \varepsilon]} \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}{\sum_{\rho_n \in \mathbb{P}_n} \pi(\mathrm{d}\rho_n) \int_{\mathbb{X}^n} \mathbb{1}_{[d(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) < \varepsilon]} \int_{\Theta^k} \mathscr{L}(\boldsymbol{X}_{1:n} \mid \rho_n, \boldsymbol{\theta}_{1:k}^*) \pi(\mathrm{d}\boldsymbol{\theta}_{1:k}^*)}$$

for some **distance** $d(\cdot, \cdot) : \mathbb{X}^n \times \mathbb{X}^n \to \mathbb{R}$.

Sampling a partition

Current state of $\theta_{1:n}$

Suppose that we have a sample of $X_1, \ldots, X_{10}$ realizations, indexed by $\theta_1, \ldots, \theta_{10}$ latent parameters. At the **current state** $\boldsymbol{\theta}_{1:n} = \{\theta_1, \ldots, \theta_{10}\}$, the parameters are divided into $k = 3$ groups with frequencies $n_1 = 3, n_2 = 5, n_3 = 2$, and unique values $\theta_1^*, \theta_2^*, \theta_3^*$.



We want to **update** somehow the partition and the unique values of $\theta_1, \ldots, \theta_{10}$ from the current state $\boldsymbol{\theta}_{1:10}$.

Marginal sampler

In force of the exchangeability of $\boldsymbol{\theta}_{1:n}$, from the predictive distribution we have

$$P(\theta_i \in \mathrm{d}t \mid \boldsymbol{\theta}_{(i)}, X_i) = \begin{cases} \dfrac{\Pi_k^{(n+1)}(n_1,\ldots,n_j+1,\ldots,n_k)}{\Pi_k^{(n)}(n_1,\ldots,n_k)} \mathcal{K}(X_i, \mathrm{d}t), & \text{if } t \in \{\theta_1^*, \ldots, \theta_k^*\} \\ \dfrac{\Pi_{k+1}^{(n+1)}(n_1,\ldots,n_k,1)}{\Pi_k^{(n)}(n_1,\ldots,n_k)} \int_\Theta \mathcal{K}(X_i, t) P_0(\mathrm{d}t) & \text{otherwise} \end{cases}$$

with $\theta_1^*, \ldots, \theta_k^*$ unique values in $\boldsymbol{\theta}_{(i)}$ with frequencies $n_1, \ldots, n_k$.
We can then update **sequentially**

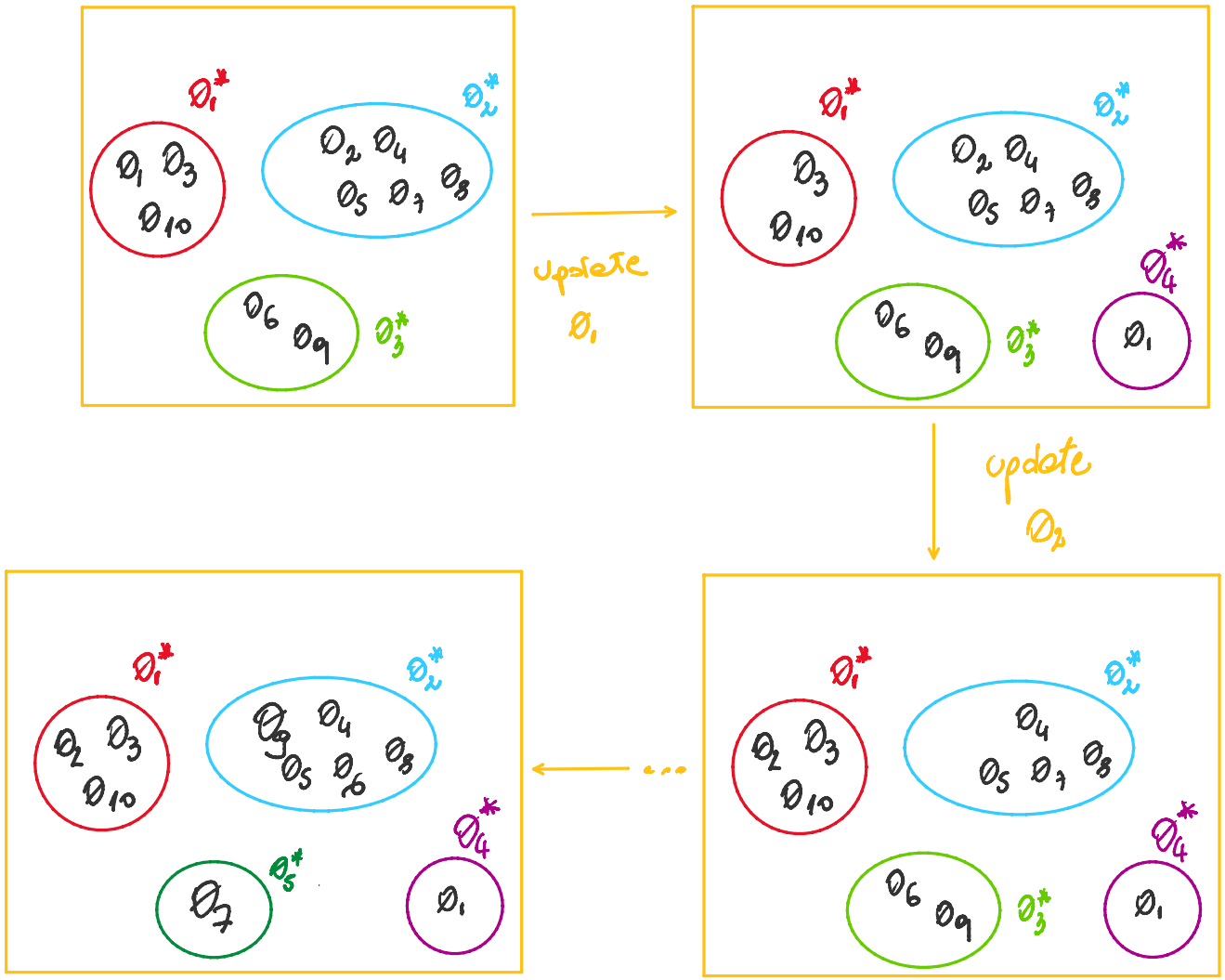
$$\begin{aligned} &\theta_1 \mid \boldsymbol{\theta}_{(1)} \\ &\theta_2 \mid \boldsymbol{\theta}_{(2)} \\ &\vdots \\ &\theta_n \mid \boldsymbol{\theta}_{(n)} \end{aligned}$$

obtaining an updated set of latent parameters $\theta_1, \ldots, \theta_n$ with unique values $\theta_1^*, \ldots, \theta_k^*$, frequencies $n_1, \ldots, n_k$, and corresponding partition ρ_n.

At each step we have to evaluate n times the kernel function for each observed cluster, and compute an integral. **It can be slow** when $n \nearrow$ (and $k \nearrow$).

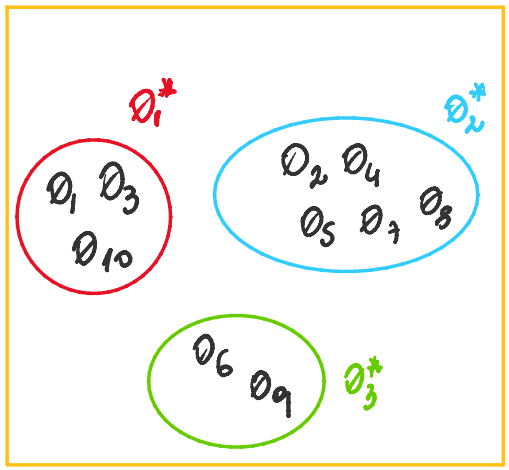
Marginal sampler

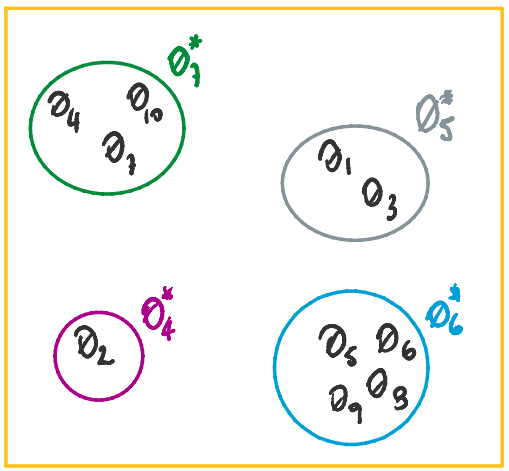
We reallocate all the elements according to their conditional distribution



Independent partitions

A **super efficient** way to generate a partition: exploit independence. For example, we can sample $k \in \{1, \ldots, n\}$ number of distinct elements $\theta_1^*, \ldots, \theta_k^*$, and then sample the allocation of θ_i, $i = 1, \ldots, n$ uniformly on $\{1, \ldots, k\}$.





It's **fast**, but **completely uninformative**, if the current state is a good candidate for the data, we are losing this information by generating a new partition completely independent.

Explore the future!

Starting from a current state of $\theta_1, \dots, \theta_n$, we can go n **step in the future** with the sampling, obtaining $\theta_{n+1}, \dots, \theta_{2n}$.

We can easily express the law of $\boldsymbol{\theta}_{n+1:2n} \mid \boldsymbol{\theta}_{1:n}$ as

$$\mathscr{L}(\boldsymbol{\theta}_{n+1:2n} \mid \boldsymbol{\theta}_{1:n}) = \mathscr{L}(\theta_{n+1} \mid \boldsymbol{\theta}_{1:n})\mathscr{L}(\theta_{n+2} \mid \boldsymbol{\theta}_{1:n+1}) \dots \mathscr{L}(\theta_{2n} \mid \boldsymbol{\theta}_{1:2n-1})$$

and we can sample the generic $\mathscr{L}(\theta_{i+1} \mid \boldsymbol{\theta}_{1:i})$ using the predictive distribution

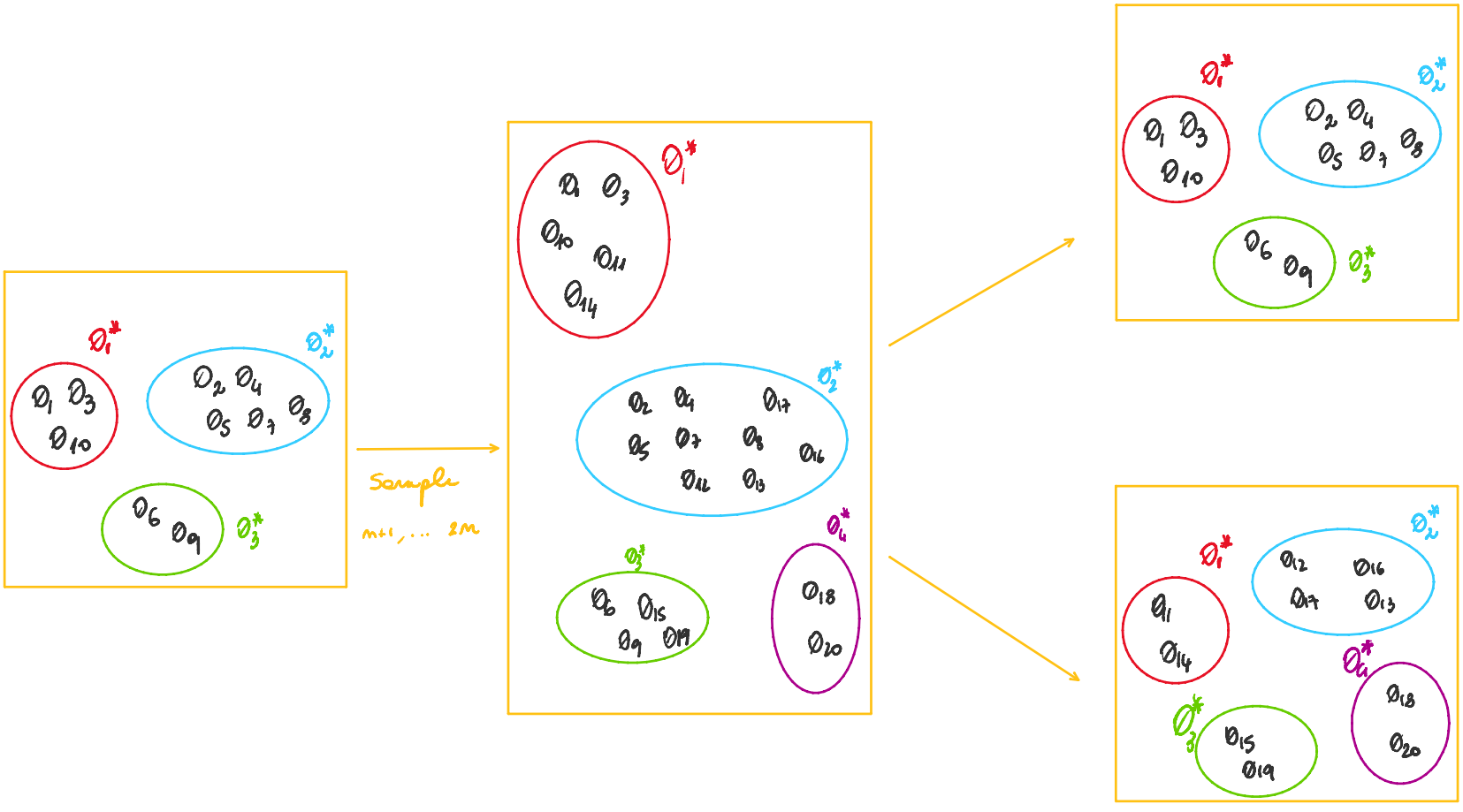
$$P(\theta_{i+1} \in \mathrm{d}t \mid \boldsymbol{\theta}_{1:i}, X_i) = \frac{\Pi_{k+1}^{(n+1)}(n_1, \dots, n_k, 1)}{\Pi_k^{(n)}(n_1, \dots, n_k)} P_0(\mathrm{d}t) + \sum_{j=1}^{k} \frac{\Pi_k^{(n+1)}(n_1, \dots, n_k)}{\Pi_k^{(n)}(n_1, \dots, n_k)} \delta_{\theta_j^*}(\mathrm{d}t)$$

which is a convex combination of the prior guess, expressed in terms of P_0, and the previous realizations $\theta_1, \dots, \theta_i$.

In this sense, the set $\theta_{n+1}, \dots, \theta_{2n}$ combine the **prior guess** P_0 and the **latent empirical information** of previous state $\theta_1, \dots, \theta_n$.

Explore the future!

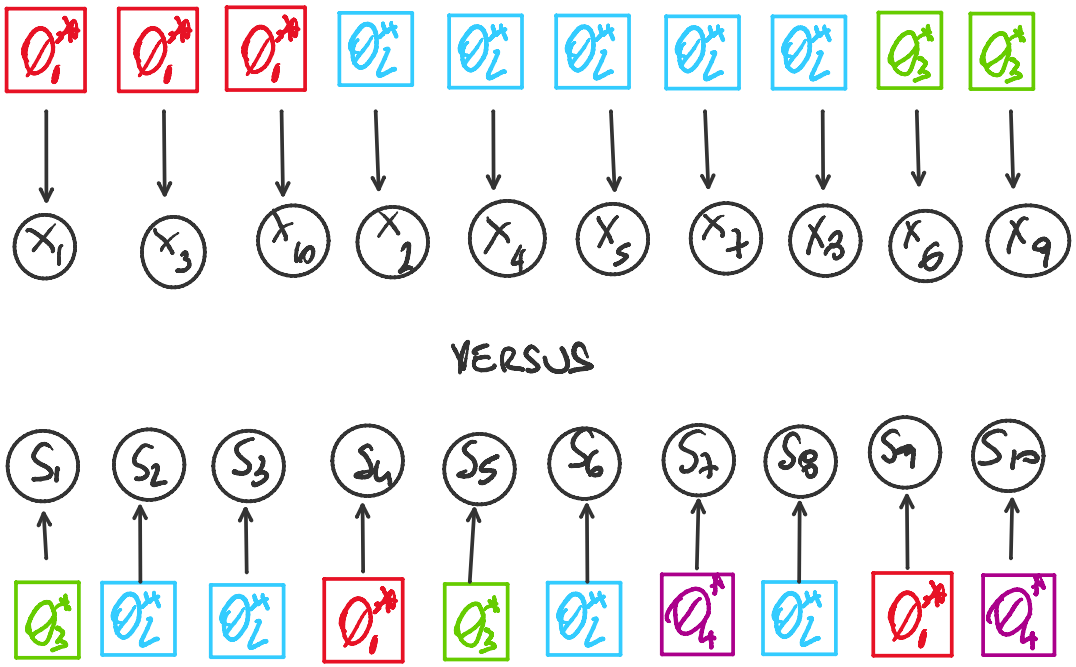
Graphically we have:



Going approximate with Wasserstein

New partition $\rightarrow$ new data

Once we get a **raw** new state of the latent parameters $\boldsymbol{\theta}' = \{\theta_{n+1}, \ldots, \theta_{2n}\}$, with corresponding partition ρ_n', we sample a **raw** set of **synthetic data** $S_1, \ldots, S_n$ according to a data generating process, with $S_i \sim \mathcal{K}(S_i, \theta_i'), i = 1, \ldots, n$



We can use the exchangeability assumption: both $\boldsymbol{X}_{1:n}$ and $\boldsymbol{S}_{1:n}$ are exchangeable, i.e. their distribution is invariant with respect to any permutation of the data

$$\mathscr{L}(S_1, \ldots, S_n) \stackrel{d}{=} \mathscr{L}(S_{\sigma(1)}, \ldots, S_{\sigma(n)})$$

with $\sigma : \{1, \ldots, n\} \rightarrow \{1, \ldots, n\}$ a permutation of the indices.

The Wasserstein distance

In force of the exchangeability, a reasonable distance to compare $\boldsymbol{X}_{1:n}$ with $\boldsymbol{S}_{1:n}$ is the **Wasserstein distance**, in the spirit of Bernton et al. (2019b,a).

We denotes by $\mathcal{W}_q(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n})$ the Wasserstein distance between to finite set of elements with the same cardinality is equal to

$$\mathcal{W}_q(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) = \left\{ \min_{P \in M_{n \times n}} \sum_{i=1}^{n} \sum_{j=1}^{n} ||X_i - S_j||_q P_{i,j} \right\}^{1/q}$$

where P is **transport matrix** with only one non-zero element for each row and for each column, and $||\cdot||_q$ denotes the L_q norm.

The optimal P solving the minimization problem of the $\mathcal{W}_q$-distance is also providing an **optimal order** of the synthetic data $\boldsymbol{S}_{1:n}$, and consequentially of the raw sequence of latent parameters $\theta'_{1:n}$, say $\theta''_{1:n}$.

The sequence of latent parameters $\theta''_{1:n}$, associated with a latent partition ρ''_n, is **playing the role of candidate** to update the current state $\theta_{1:n}$.

The Wasserstein distance

In force of the exchangeability, a reasonable distance to compare $\boldsymbol{X}_{1:n}$ with $\boldsymbol{S}_{1:n}$ is the **Wasserstein distance**, in the spirit of Bernton et al. (2019b,a).

We denotes by $\mathcal{W}_q(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n})$ the Wasserstein distance between to finite set of elements with the same cardinality is equal to

$$\mathcal{W}_q(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) = \left\{ \min_{P \in M_{n \times n}} \sum_{i=1}^{n} \sum_{j=1}^{n} ||X_i - S_j||_q P_{i,j} \right\}^{1/q}$$

where P is **transport matrix** with only one non-zero element for each row and for each column, and $||\cdot||_q$ denotes the L_q norm.

The optimal P solving the minimization problem of the $\mathcal{W}_q$-distance is also providing an **optimal order** of the synthetic data $\boldsymbol{S}_{1:n}$, and consequentially of the raw sequence of latent parameters $\boldsymbol{\theta}'_{1:n}$, say $\boldsymbol{\theta}''_{1:n}$.

The sequence of latent parameters $\boldsymbol{\theta}''_{1:n}$, associated with a latent partition ρ''_n, is **playing the role of candidate** to update the current state $\boldsymbol{\theta}_{1:n}$.

An ABC–MCMC sampling scheme

We can then use the proposed value $\boldsymbol{\theta}''_{1:n}$ in an ABC–MCMC scheme, **thanks to the echangeability** we have that $\mathscr{L}(\boldsymbol{\theta}''_{1:n} \mid \boldsymbol{\theta}_{1:n}) = \mathscr{L}(\boldsymbol{\theta}'_{1:n} \mid \boldsymbol{\theta}_{1:n})$.

We use $q(\boldsymbol{\theta}_{1:n} \to \boldsymbol{\theta}''_{1:n}) = \mathscr{L}(\boldsymbol{\theta}''_{1:n} \mid \boldsymbol{\theta}_{1:n})$ as **proposal distribution**.

Once that $\mathcal{W}_q(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) \leq \varepsilon$, we perform a MH step from the prior distribution (see Marjoram et al., 2003).

We notice that, until the $\mathcal{W}_q$-distance is smaller than a threshold ε, we **always accept the new state**

$$\alpha(\boldsymbol{\theta}''_{1:n}, \boldsymbol{\theta}_{1:n}) = 1 \wedge \frac{\mathscr{L}(\boldsymbol{\theta}''_{1:n}) q(\boldsymbol{\theta}''_{1:n} \to \boldsymbol{\theta}_{1:n})}{\mathscr{L}(\boldsymbol{\theta}_{1:n}) q(\boldsymbol{\theta}_{1:n} \to \boldsymbol{\theta}''_{1:n})} = 1 \wedge \frac{\mathscr{L}(\boldsymbol{\theta}''_{1:n}) \mathscr{L}(\boldsymbol{\theta}_{1:n}, \boldsymbol{\theta}''_{1:n}) \mathscr{L}(\boldsymbol{\theta}_{1:n})}{\mathscr{L}(\boldsymbol{\theta}_{1:n}) \mathscr{L}(\boldsymbol{\theta}''_{1:n}, \boldsymbol{\theta}_{1:n}) \mathscr{L}(\boldsymbol{\theta}''_{1:n})} = 1,$$

in the spirit of Clarté et al. (2020).

An ABC–MCMC sampling scheme

We can then use the proposed value $\boldsymbol{\theta}''_{1:n}$ in an ABC–MCMC scheme, **thanks to the echangeability** we have that $\mathscr{L}(\boldsymbol{\theta}''_{1:n} \mid \boldsymbol{\theta}_{1:n}) = \mathscr{L}(\boldsymbol{\theta}'_{1:n} \mid \boldsymbol{\theta}_{1:n})$.

We use $q(\boldsymbol{\theta}_{1:n} \to \boldsymbol{\theta}''_{1:n}) = \mathscr{L}(\boldsymbol{\theta}''_{1:n} \mid \boldsymbol{\theta}_{1:n})$ as **proposal distribution**.

Once that $\mathcal{W}_q(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) \leq \varepsilon$, we perform a MH step from the prior distribution (see Marjoram et al., 2003).

We notice that, until the $\mathcal{W}_q$-distance is smaller than a threshold ε, we **always accept the new state**

$$\alpha(\boldsymbol{\theta}''_{1:n}, \boldsymbol{\theta}_{1:n}) = 1 \wedge \frac{\mathscr{L}(\boldsymbol{\theta}''_{1:n}) q(\boldsymbol{\theta}''_{1:n} \to \boldsymbol{\theta}_{1:n})}{\mathscr{L}(\boldsymbol{\theta}_{1:n}) q(\boldsymbol{\theta}_{1:n} \to \boldsymbol{\theta}''_{1:n})} = 1 \wedge \frac{\mathscr{L}(\boldsymbol{\theta}''_{1:n}) \mathscr{L}(\boldsymbol{\theta}_{1:n}, \boldsymbol{\theta}''_{1:n}) \mathscr{L}(\boldsymbol{\theta}_{1:n})}{\mathscr{L}(\boldsymbol{\theta}_{1:n}) \mathscr{L}(\boldsymbol{\theta}''_{1:n}, \boldsymbol{\theta}_{1:n}) \mathscr{L}(\boldsymbol{\theta}''_{1:n})} = 1,$$

in the spirit of Clarté et al. (2020).

An ABC–MCMC sampling scheme

We can then use the proposed value $\boldsymbol{\theta}''_{1:n}$ in an ABC–MCMC scheme, **thanks to the echangeability** we have that $\mathscr{L}(\boldsymbol{\theta}''_{1:n} \mid \boldsymbol{\theta}_{1:n}) = \mathscr{L}(\boldsymbol{\theta}'_{1:n} \mid \boldsymbol{\theta}_{1:n})$.

We use $q(\boldsymbol{\theta}_{1:n} \to \boldsymbol{\theta}''_{1:n}) = \mathscr{L}(\boldsymbol{\theta}''_{1:n} \mid \boldsymbol{\theta}_{1:n})$ as **proposal distribution**.

Once that $\mathcal{W}_q(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) \leq \varepsilon$, we perform a MH step from the prior distribution (see Marjoram et al., 2003).

We notice that, until the $\mathcal{W}_q$-distance is smaller than a threshold ε, we **always accept the new state**

$$\alpha(\boldsymbol{\theta}''_{1:n}, \boldsymbol{\theta}_{1:n}) = 1 \wedge \frac{\mathscr{L}(\boldsymbol{\theta}''_{1:n}) q(\boldsymbol{\theta}''_{1:n} \to \boldsymbol{\theta}_{1:n})}{\mathscr{L}(\boldsymbol{\theta}_{1:n}) q(\boldsymbol{\theta}_{1:n} \to \boldsymbol{\theta}''_{1:n})} = 1 \wedge \frac{\mathscr{L}(\boldsymbol{\theta}''_{1:n}) \mathscr{L}(\boldsymbol{\theta}_{1:n}, \boldsymbol{\theta}''_{1:n}) \mathscr{L}(\boldsymbol{\theta}_{1:n})}{\mathscr{L}(\boldsymbol{\theta}_{1:n}) \mathscr{L}(\boldsymbol{\theta}''_{1:n}, \boldsymbol{\theta}_{1:n}) \mathscr{L}(\boldsymbol{\theta}''_{1:n})} = 1,$$

in the spirit of Clarté et al. (2020).

A first sampling scheme

Algorithm 1: ABC-MCMC for random partitions

input a set of data $\boldsymbol{X}_{1:n}$, a threshold ε, and possibly hyperparameters for $\mathcal{K}(\cdot, \theta)$;
set admissible initial values for $\boldsymbol{\theta}^{(0)}$;
for r *in* $1 : R$ **do**
    repeat
        propose a move from $\boldsymbol{\theta}^{(r-1)}$ to $\boldsymbol{\theta}'$ according to a transition kernel $q(\boldsymbol{\theta}^{(r-1)} \to \boldsymbol{\theta}')$, with related partition ρ_n';
        sample $\boldsymbol{S}_{1:n} \mid \boldsymbol{\theta}'$ vector of synthetic data, where $S_i \sim \mathcal{K}(\cdot, \theta_i')$;
    until $\mathcal{W}_p(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) \leq \varepsilon$;
    accept ρ_n'', the permuted version of ρ_n', as realization from $\pi_\varepsilon(\rho_n \mid \boldsymbol{X}_{1:n})$;
end

The chain has invariant distribution $\pi_\varepsilon(\rho_n \mid \boldsymbol{X}_{1:n})$.

We early realized that the threshold ε **has a strong impact** on the results. If ε is too large, the latent partitions are completely uninformative, while if ε is too small, it is difficult to accept a new partition, especially in the early part of the chain.

Going adaptive

Instead of a fixed threshold ε, we consider a **sequence** $\{\varepsilon_\ell\}_{\ell \geq 1}$ such that as $\ell \to \infty$, the sequence $\{\varepsilon_\ell\}_{\ell \geq 1}$ converges to a **limit value** ε^*. $\{\varepsilon_\ell\}_{\ell \geq 1}$ should be **monotonically decreasing**.

Algorithm 2: adaptive ABC-MCMC for random partitions

input a set of data $\boldsymbol{X}_{1:n}$, a threshold ε, and possibly hyperparameters for $\mathcal{K}(\cdot, \theta)$;
set admissible initial values for $\boldsymbol{\theta}^{(0)}$, set $\ell = 1$;
for r *in* $1 : R$ **do**
    repeat
        propose a move from $\boldsymbol{\theta}^{(r-1)}$ to $\boldsymbol{\theta}'$ according to a transition kernel $q(\boldsymbol{\theta}^{(r-1)} \to \boldsymbol{\theta}')$, with related partition ρ_n';
        sample $\boldsymbol{S}_{1:n} \mid \boldsymbol{\theta}'$ vector of synthetic data, where $S_i \sim \mathcal{K}(\cdot, \theta_i')$;
        update $\varepsilon_\ell = g(w_{1:\ell-1}, \varepsilon^*)$;
        set $\ell = \ell + 1$;
    until $\mathcal{W}_p(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}) \leq \varepsilon_\ell$;
    accept ρ_n'', the permuted version of ρ_n', as realization from $\pi_\varepsilon(\rho_n \mid \boldsymbol{X}_{1:n})$;
end

An adaptive sampling scheme

$\{\varepsilon_\ell\}_{\ell \geq 1}$ can be chosen in many ways. One possible way is to consider a **convex combination** of two terms

$$\varepsilon_\ell = a \times g(w_{1:\ell-1}) + (1 - a) \times \varepsilon^*$$

where

- $g(w_{1:\ell-1})$ can be for example the quantile of $w_{1:\ell-1}$, or the rolling quantile of $w_{\ell-\gamma:\ell-1}$, for $\ell > \gamma$;
- a is tuning the speed of convergence of ε_ℓ to ε^*, for example $a = \frac{1}{\ell}$ or $a = \frac{1}{\ell^2}$.

Under the previous conditions, we can prove the following result.

Thrm: Invariant distribution of the adaptive scheme

Let $\{\varepsilon_\ell\}_{\ell \geq 1}$ be $\mathbb{R}^+$-valued sequence of elements, such that $\lim_{\ell \to +\infty} |\varepsilon_\ell - \varepsilon^*| = 0$. Let $\{\rho_{n,1}, \rho_{n,2}, \ldots\}$ be a sample from an ABC-MCMC scheme according to algorithm 2, with proposal $q(\theta_n \to \theta'_n)$. Let $p(w)$ be the density function of $W_p\left(\boldsymbol{X}_{1:n}, \boldsymbol{S}_{1:n}^{(\ell)}\right)$, where $\boldsymbol{S}_{1:n}^{(\ell)}$ denotes the ℓ-th synthetic sample, and assume $0 < p(w) < M$ for all ℓ. Then, for $\ell \to +\infty$, we have that $\pi_{\varepsilon^*}(\rho_n \mid \boldsymbol{X}_{1:n})$ is a.s. the invariant distribution of the chain.

An adaptive sampling scheme

$\{\varepsilon_\ell\}_{\ell \geq 1}$ can be chosen in many ways. One possible way is to consider a **convex combination** of two terms

$$\varepsilon_\ell = a \times g(w_{1:\ell-1}) + (1-a) \times \varepsilon^*$$

where

- $g(w_{1:\ell-1})$ can be for example the quantile of $w_{1:\ell-1}$, or the rolling quantile of $w_{\ell-\gamma:\ell-1}$, for $\ell > \gamma$;
- a is tuning the speed of convergence of ε_ℓ to ε^*, for example $a = \frac{1}{\ell}$ or $a = \frac{1}{\ell^2}$.

Under the previous conditions, we can prove the following result.

Thrm: Invariant distribution of the adaptive scheme

Let $\{\varepsilon_\ell\}_{\ell \geq 1}$ be $\mathbb{R}^+$-valued sequence of elements, such that $\lim_{\ell \to +\infty} |\varepsilon_\ell - \varepsilon^*| = 0$. Let $\{\rho_{n,1}, \rho_{n,2}, \ldots\}$ be a sample from an ABC-MCMC scheme according to algorithm 2, with proposal $q(\boldsymbol{\theta}_n \to \boldsymbol{\theta}'_n)$. Let $p(w)$ be the density function of $W_p\left(\boldsymbol{X}_{1:n}, \boldsymbol{S}^{(\ell)}_{1:n}\right)$, where $\boldsymbol{S}^{(\ell)}_{1:n}$ denotes the ℓ-th synthetic sample, and assume $0 < p(w) < M$ for all ℓ. Then, for $\ell \to +\infty$, we have that $\pi_{\varepsilon^*}(\rho_n \mid \boldsymbol{X}_{1:n})$ is a.s. the invariant distribution of the chain.

Simulation results

Simulation setup

We first investigate the performances of the **ABC-MCMC** algorithm with simulated data. We generate a set of n realizations from $f_0(x)$, a balance mixture of two Gaussian distributions, with

$$f_0(x) = 0.5\phi(x; -3, 1) + 0.5\phi(x; 3, 1)$$

for different sample sizes $n \in \{40, 80, 160, 320\}$.

We consider as **data generating process**, to simulate the synthetic observations, a **Gaussian distribution** with both location and scale parameters group specific.

Our prior guess on the latent partition is expressed by setting the mixing measure of the starting mixture model equal to a **Pitman-Yor process**, i.e. $\tilde{p} \sim PY(P_0, \alpha, \lambda)$, with $P_0 \stackrel{d}{=} NIG(0, 0.5, 2, 2)$, and the **corresponding EPPF** of $\tilde{p}$ is equal to

$$\Pi_k^{(n)}(n_1, \ldots, n_k) = \frac{\prod_{j=1}^{k-1}(\alpha + j\lambda)}{(\alpha + 1)_{n-1}} \prod_{j=1}^{k}(1 - \lambda)_{n_j - 1},$$

where $(a)_b$ denotes the Pochammer symbol.

Simulations: time

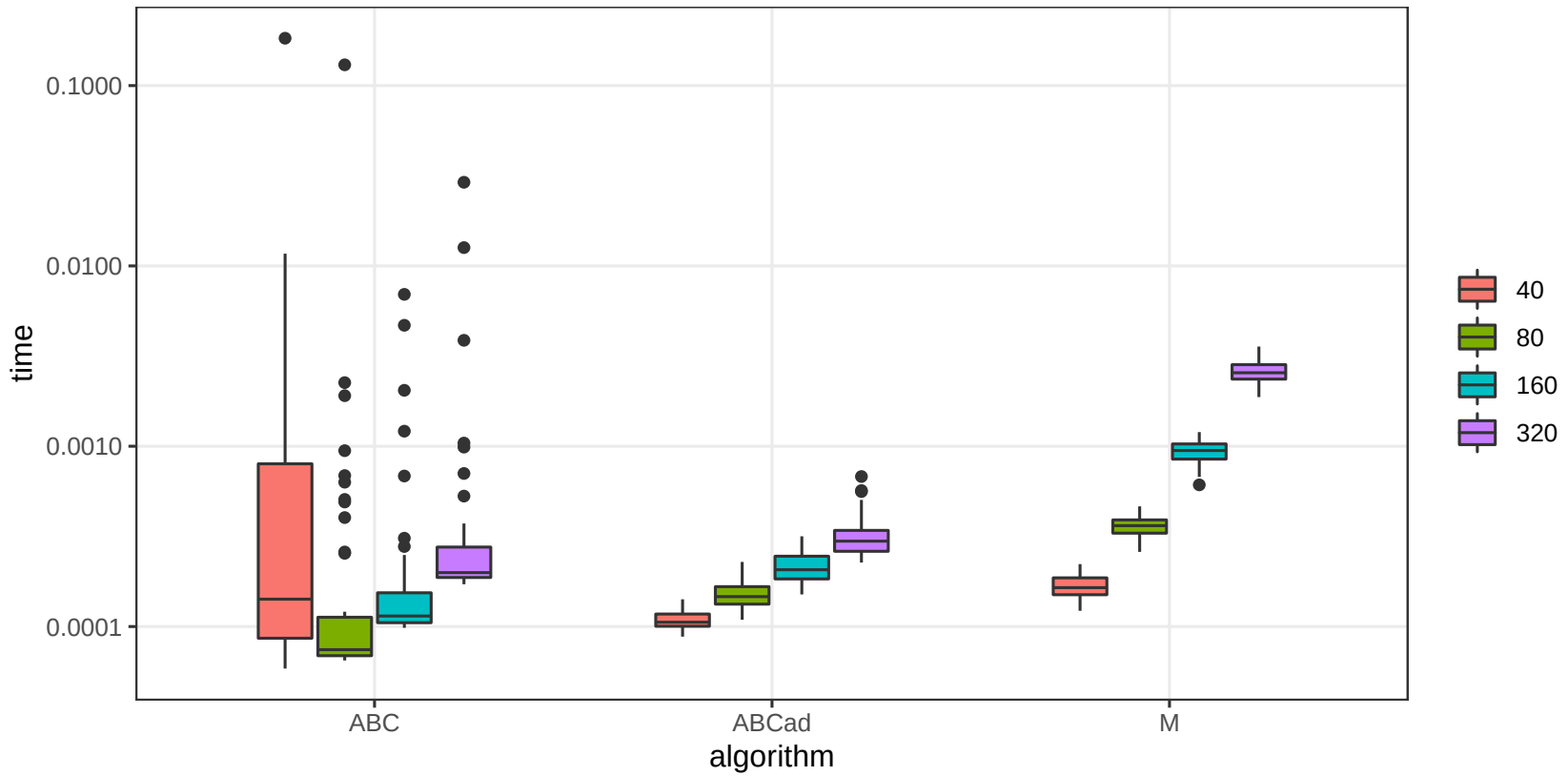


Figure 1: Avg. time for a single realization from the posterior distribution, 100 replications, on a $\log_{10}$-scale. Different algorithms: ABC stands for the basic ABC-MCMC scheme. ABCad stands for the adaptive ABC-MCMC scheme. MAR is the marginal algorithm.

Simulations: RAND

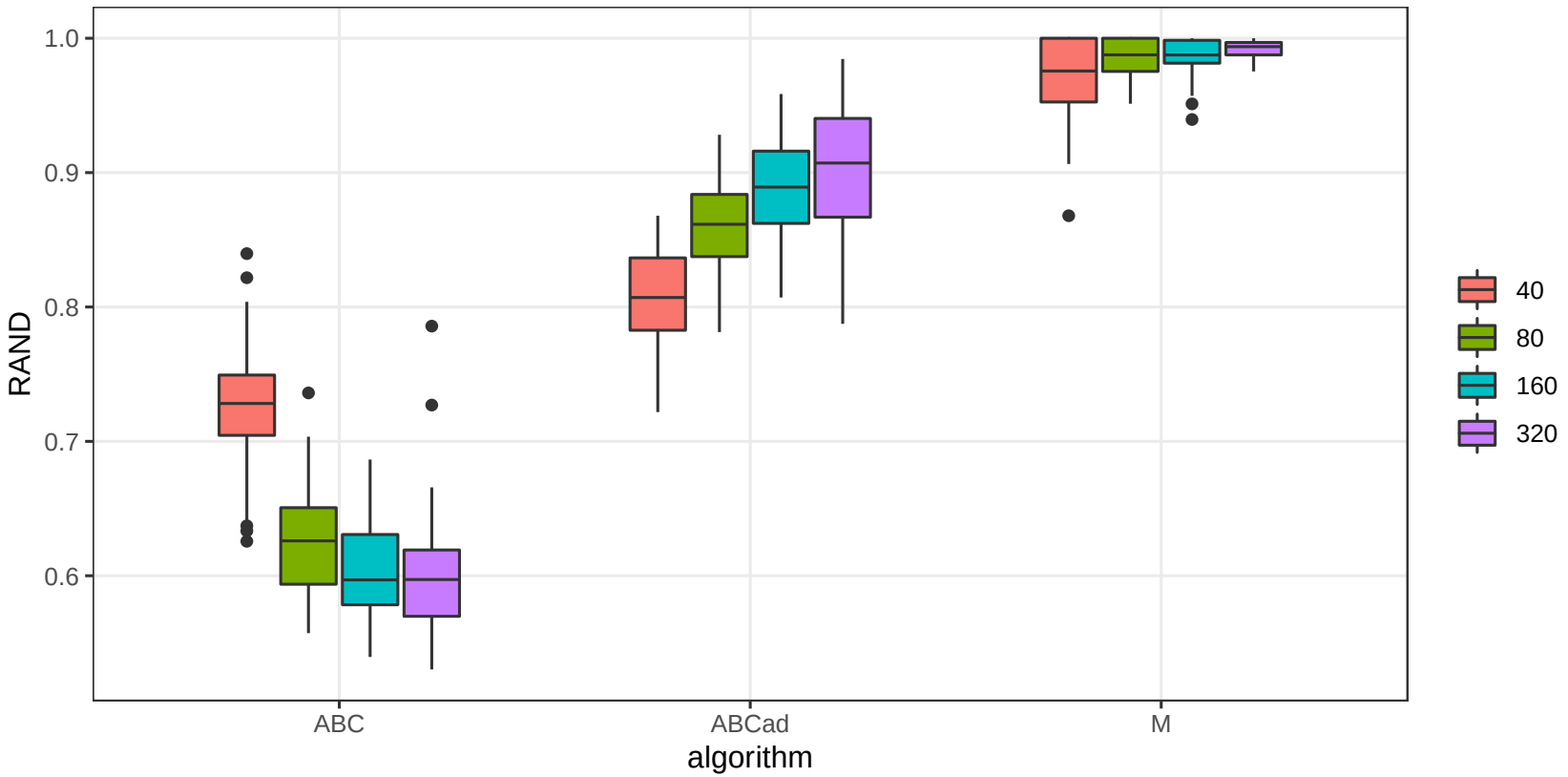


Figure 2: RAND index of true vs point estimate of the latent partition (with Binder loss function), 100 replications, on a $\log_{10}$-scale. Different algorithms: ABC stands for the basic ABC-MCMC scheme. ABCad stands for the adaptive ABC-MCMC scheme. MAR is the marginal algorithm.

Application

Clustering a set of networks

We consider a set of $n = 52$ **companies serving the US airports** in the study. We know which airports are connected by each company.

We represent each realization in terms of **graph** $G_i = \{V_i, E_i\}$, where V_i is the set of nodes for the observation i-th, and E_i denotes the set of tuples $(j, k) \in V_i \times V_i$.

In our specific framework, the **airports** considered are **shared by the different companies**, i.e. $V_i = V$ for all $i = 1, \ldots, n$, and we assume the graph **undirected**.

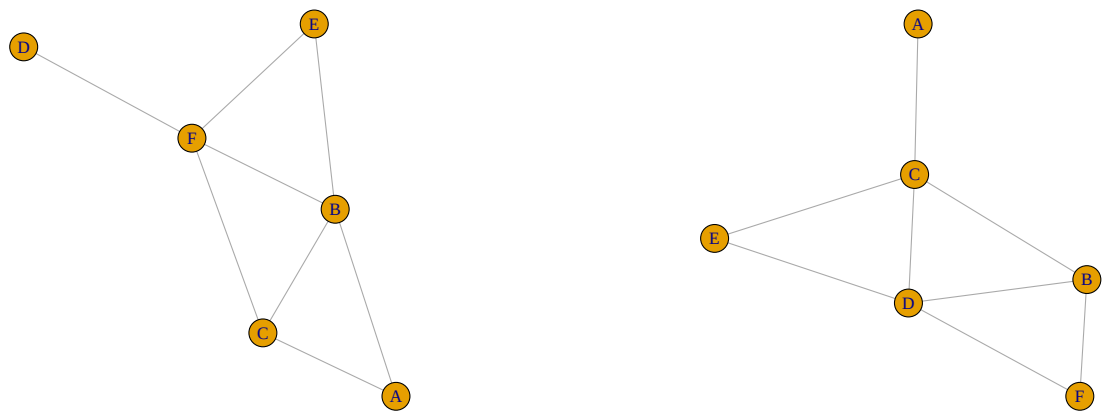


Figure 3: An example of two graphs differing only on the labeling of the nodes, but with the same topology. The graph in the right picture is recovered starting from the graph in the left picture by renaming the nodes as $D \to A$, $C \to B$, $F \to C$, $B \to D$, $A \to F$, $E \to E$.

Clustering a set of networks

Remark. *The main difference between this application and the previous sections is that the observed data $\boldsymbol{G}_{1:n}$ are not a subset of $\mathbb{R}^d$ anymore. Nonetheless, the formulation of the Wasserstein distance can be generalized to this scenario. Here we incorporated the spectral distance between graphs (Gu et al., 2015).*

We consider a **Pitman-Yor mixture models**, with data generating process equal to an **Exponential Random Graph Model (ERGM)**, a distribution of the Gibbs form on the network space (Robins et al., 2007b,a).

The assumption underlying ERGMs is that the topology of an observed graph Y_i can be explained by a **set of statistics** $s(Y_i)$, with

$$P(Y_i = y_i \mid \boldsymbol{\theta}_i) = \frac{1}{\mathrm{C}_{\theta_i}} \mathrm{e}^{\boldsymbol{\theta}_i s(y_i)}$$

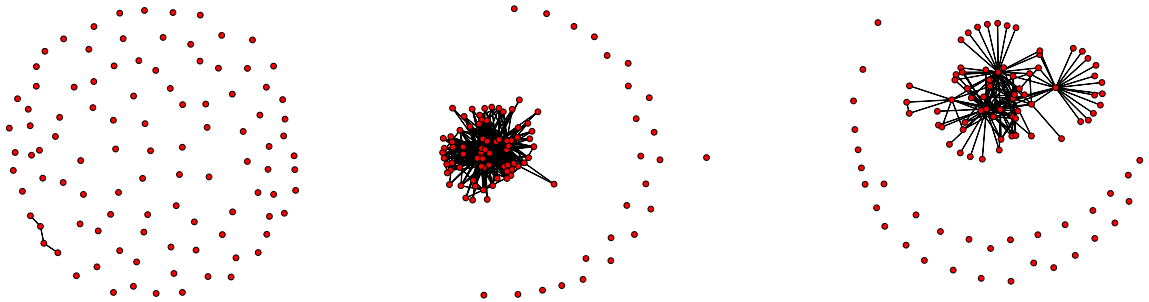
where Y_i is the binary matrix representation of the i-th graph, and C_{θ_i} is a normalization constant, **not available in a closed form**.

We consider as statistics $s(Y_i)$ a vector of counts where the first element is the total number of edges in the i-th graph, and the rest of the counts are distinct degrees

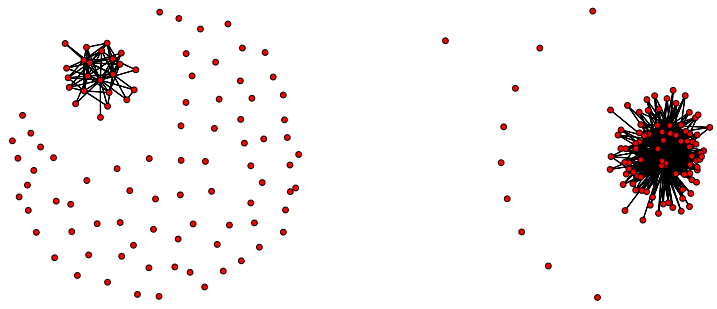
$$s(Y_i, \tau) = \sum_{j=1}^{m} \mathbb{1}_{[(\sum_{k=1}^{m} Y_{j,k}) = \tau]}$$

with $\tau \in \{0, 1, 10, 50, 70\}$.

Clustering a set of networks



(a) Cluster 1, 44 obs. **(b)** Cluster 2, 4 obs. **(c)** Cluster 3, 1 obs.



(d) Cluster 4, 1 obs. **(e)** Cluster 5, 2 obs.

Figure 4: Medoids of the 5 clusters of the partition obtained minimizing the Binder loss function.

Final highlights

Final highlights

- The sampling scheme can easily accommodate **intractable kernel functions** and several prior processes, such as Gibbs-type priors (Gnedin and Pitman, 2005; De Blasi et al., 2015) and NRMI (Regazzini et al., 2003);

- The method suffers from the (double) **curse of dimensionality**, where both the data and the synthetic data become sparse as far as the dimension increases. Indeed, we can work on latent spaces;

- The sampling scheme **works also for similar models**, e.g. product partition models

$$p(\rho_n = \{A_1, \ldots, A_k\}) = \mathrm{K} \prod_{j=1}^{k} c(A_j)$$

  with $c(A_j)$ cohesion function, and K normalization constant;

- We are currently developing an **efficient C++ library** to perform approximate inference on latent partitions;

- On going research:
  - **extensions to the partially exchangeable case**;
  - **different adaptive strategies**;
  - **non-MCMC strategies**;

Thanks for your attention!

References

Bernton, E., Jacob, P. E., Gerber, M., and Robert, C. P. (2019a). Approximate bayesian computation with the wasserstein distance. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 81(2):235–269.

Bernton, E., Jacob, P. E., Gerber, M., and Robert, C. P. (2019b). On parameter estimation with the Wasserstein distance. *Information and Inference: A Journal of the IMA*, 8(4):657–676.

Clarté, G., Robert, C. P., Ryder, R. J., and Stoehr, J. (2020). Componentwise approximate Bayesian computation via Gibbs-like steps. *Biometrika*.

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(2):212–229.

Gnedin, A. and Pitman, J. (2005). Exchangeable Gibbs partitions and Stirling triangles. *Zap. Nauchn. Sem. S.-Peterburg. Otdel. Mat. Inst. Steklov. (POMI)*, 325(Teor. Predst. Din. Sist. Komb. i Algoritm. Metody. 12):83–102, 244–245.

Gu, J., Hua, B., and Liu, S. (2015). Spectral distances on graphs. *Discrete Applied Mathematics*, 190:56–74.

Marjoram, P., Molitor, J., Plagnol, V., and Tavaré, S. (2003). Markov chain monte carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 100(26):15324–15328.

Regazzini, E., Lijoi, A., and Prünster, I. (2003). Distributional results for means of normalized random measures with independent increments. *The Annals of Statistics*, 31(2):560 – 585.

Robins, G., Pattison, P., Kalish, Y., and Lusher, D. (2007a). An introduction to exponential random graph (p*) models for social networks. *Social networks*, 29(2):173–191.

Robins, G., Snijders, T., Wang, P., Handcock, M., and Pattison, P. (2007b). Recent developments in exponential random graph (p*) models for social networks. *Social networks*, 29(2):192–215.