SALT Sampler for Simplex Supported Target Distributions

STAT 689: Sampling for Data Science
Course Project Report

Arhit Chakrabarti[1] and Somjit Roy[1]

[1] Department of Statistics, Texas A&M University, College Station, TX, USA
arhit.chakrabarti@stat.tamu.edu, sroy_123@tamu.edu

Abstract

Sampling complex-structured and constrained posterior distributions can compromise the efficiency and performance of the entire sampling algorithm. This article highlights such a situation which is circumvented by carefully choosing the proposal distribution using the Self-Adjusted Logit Transformation (SALT) technique. The optimized performance of the sampling distribution using the SALT proposal, in case of simplex-supported target distributions, has been illustrated over other two competing techniques based on Dirichlet proposals and Hamiltonian Monte Carlo (HMC) through extensive simulation studies.

1 Introduction

Modelling compositional data (Aitchison, 1982) has immense applications in the field of *species abundance* (Billheimer et al., 2001), *geochemical composition* (Thomas and Aitchison, 2005), and *economics* (Fry et al., 2000). Working with such data involves devising sampling routines over a $k-$dimensional simplex. Compositional data modelling is a major attraction in statistical literature (van den Boogaart and Tolosana-Delgado, 2008), however, computational challenges when sampling and working on a simplex, especially in high-dimensions, poses to be a real challenging problem.

A $k-$dimensional simplex is a collection of points $\boldsymbol{x}_k = (x_1, x_2, \ldots, x_k)$ on the space $\mathbb{R}^k$, constrained such that, $\sum_{i=1}^{k} x_i = c$, where $0 < x_i \leq c$. Geometrically, this simplex can be thought of as a $(k-1)-$dimensional subspace of $\mathbb{R}^k$, by setting $c = 1$. One among many modelling efforts of compositional data on such a simplex extends to defining the *logratio transformation* by Aitchison, 1982,

$$\phi(\boldsymbol{x}) = \log\left(\frac{\boldsymbol{x}_{-k}}{x_k}\right) \tag{1}$$

where $\boldsymbol{x}_{-k}$ denotes the co-ordinates of $\boldsymbol{x}_k$, with the $k-$th co-ordinate removed. This transformed data in (1) is assumed to follow a multivariate normal distribution. Besides transforming the original set of data, dimension-reduction techniques have been developed to work on the simplex (Aitchison and Greenacre, 2002; Filzmoser et al., 2009). However, the existence of a direct and non-trivial sampling scheme of this type of constrained data is still elusive. One of the major contributions in this aspect has been noted by Director et al., 2017, where the *Self-Adjusting Logit Transformation* (SALT) has been introduced for efficiently sampling data points on a simplex following a specified distribution. The SALT, as a Markov Chain Monte Carlo (MCMC) algorithm, proves to be a robust sampling scheme even (i) when we have a high-dimensional simplex and/or (ii) the distribution puts significant mass over the regions where data co-ordinates differ by orders of magnitude. We motivate and illustrate the implementation of sampling from a simplex supported posterior distribution as our target candidate, using the specifically designed proposal distribution for SALT sampler in a widely celebrated nonparametric mixture model as in Section 2. In contrast to our proposed SALT sampler, specific to the underlined mixture model, other competing strategies are also studied.

2 Motivational Problem

Suppose we have observations from two groups. Let x_{ji} denote the observation i from group j and θ_{ji} denote the parameter specifying the mixture component associated with the corresponding

observation. Let $F(\theta_{ji})$ denote the distribution of x_{ji} given θ_{ji} and G_j denote a prior distribution for θ_{ji}. The group-specific mixture model is given by,

$$\begin{aligned} \theta_{ji} \mid G_j &\overset{ind}{\sim} G_j, \\ x_{ji} \mid \theta_{ji} &\overset{ind}{\sim} F(\theta_{ji}). \end{aligned} \tag{2}$$

The celebrated *Dirichlet process* (DP, Ferguson, 1973) has been the backbone of numerous model-based Bayesian nonparametric methods. The DP, $DP(\alpha_0, G_0)$, is a probability measure on probability measures, where $\alpha_0 > 0$ is the concentration parameter and G_0 is a base probability measure. There have been extensive studies on DP mixture models (Antoniak, 1974; Lo, 1984; Escobar and West, 1995; MacEachern and Müller, 1998), where G_j is assigned a DP prior. Such nonparametric mixture models have a wide range of applications including model-based clustering, density estimation, etc. One advantage of such DP mixture model is its ability to perform clustering without having to fix the number of clusters *a priori*. However, assuming a separate DP prior for each groups j do not ensure that clusters are shared across the two populations. To this end, the hierarchical Dirichlet Process (HDP, Teh et al., 2006) prior provides an elegant solution to this problem by assuming $G_j \mid G_0 \sim DP(\alpha_j, G_0)$, $j = 1, 2$, and $G_0 \sim DP(\gamma, H)$.

Consider a simple modification to the specification of the priors on G_j in (2)—motivation to which we provide in Section 4—as,

$$\begin{aligned} G_1 &\sim DP(\alpha_1, G_0), \\ G_2 \mid G_1 &\sim DP(\alpha_2, G_1), \end{aligned} \tag{3}$$

where G_0 is a known base probability measure. The corresponding mixture model can be derived as an infinite limit of the finite mixture model,

$$\begin{aligned} \boldsymbol{\beta}_1 &\sim Dir\left(\alpha_1/L, \ldots, \alpha_1/L\right), \\ \boldsymbol{\beta}_2 \mid \boldsymbol{\beta}_1 &\sim Dir\left(\alpha_2\left(\beta_{11}, \ldots, \beta_{1L}\right)\right), \\ \phi_l \mid G_0 &\sim G_0, \\ z_{ji} \mid \boldsymbol{\beta}_j &\sim \boldsymbol{\beta}_j, \\ x_{ji} \mid z_{ji}, (\phi_l)_{l=1}^{L} &\sim F\left(\phi_{z_{ji}}\right), \end{aligned} \tag{4}$$

as $L \to \infty$. Based on this finite mixture model approximation with a large enough truncation level L, one may consider a blocked Gibbs sampler to facilitate posterior inference. For any given likelihood $F(.)$, assuming conjugate priors on the atoms (ϕ_l), yields closed form Gibbs updates. It is also seen that the posterior update of $\boldsymbol{\beta}_2$ is available in closed form. However, the conditional posterior of $\boldsymbol{\beta}_1$ is given by,

$$\pi(\boldsymbol{\beta}_1 \mid -) \propto \prod_{l=1}^{L} \left\{ \frac{\beta_{1l}^{m_{1l}+\alpha_1/L-1} \beta_{2l}^{\alpha_2\beta_{1l}-1}}{\Gamma(\alpha_2\beta_{1l})} \right\}. \tag{5}$$

Note that, (5) is a non-standard distribution on a simplex, involving co-ordinates of a simplex in the exponent of another simplex co-ordinates. Although, a natural choice is to consider a Metropolis Hasting step to sample from the conditional posterior as the target distribution, the choice of a suitable proposal density is not straightforward. One naive choice is to consider a Dirichlet distribution as the proposal distribution. Alternatively, one may consider transforming the constrained target distribution (constrained to lie on a simplex) to an un-constrained distribution (distribution supported on a subset of $\mathbb{R}^L$) and use an appropriate Random Walk proposal. However, a Dirichlet proposal usually involves an appropriate tuning parameter, which must be chosen carefully to ensure convergence of the proposed sampler. This can become a challenging problem for different applications. Contrarily, it is not straightforward to transform the constrained target distribution to an un-constrained distribution so that a Random Walk proposal may be employed in the MH step. To mitigate this non-standard sampling issue, we propose to use SALTSampler to efficiently sample from the target simplex distribution (5). We also consider a Dirichlet proposal distribution and appropriately choose the tuning parameter of the proposal distribution. However, we show

that even with such a proposal, the convergence of the sampler may be *sub-optimal*. Furthermore, the performance of the SALT sampler proposal distribution is compared with *Hamiltonian Monte Carlo* (HMC) to sample from (5), implemented using the `Stan` software. This uses the element-wise transformation,

$$y_k = \log\left(\frac{z_k}{1-z_k}\right) - \log\left(\frac{1}{K-k}\right) \tag{6}$$

which represents the famous *stick-breaking* process. In (6) above, K refers to the number of pieces into which the stick is divided and z_k refers to the ratio of the length of the $k-$th piece of stick, x_k, relative to the total length, $1 - \sum_{j=1}^{k-1} x_j$, of the remaining pieces.

3 The General MCMC Proposal for SALT Sampler

Markov Chain Monte Carlo (MCMC) algorithms for sampling are a fundamental tool for Bayesian inference to draw insights from a specified posterior distribution, $p(\boldsymbol{\theta} \mid \boldsymbol{y})$ (Speagle, 2020). Consider the parameter of interest supported on a $k-$dimensional simplex, $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_k)$, with $\boldsymbol{y} = (y_1, y_2, \ldots, y_n) \in \mathbb{R}^n$ being the observed set of data points. An MCMC routine of sampling from a target posterior distribution $p(\boldsymbol{\theta} \mid \boldsymbol{y})$, amounts to proposing random values of the parameter vector $\boldsymbol{\theta}$ from a suitably chosen proposal distribution, where the proposed values are accepted based on a chosen criterion such that, they converge to a sample from $p(\boldsymbol{\theta} \mid \boldsymbol{y})$. Popularly known as the *Metropolis-Hastings* (MH) algorithm (Hastings, 1970), the parameter vector $\boldsymbol{\theta}$ is set to an initial starting value $\boldsymbol{\theta}^0$. For each iteration $t = 1, 2, \ldots$, a new value $\boldsymbol{\theta}'$ is drawn from the chosen proposal distribution $q(\boldsymbol{\theta}' \mid \boldsymbol{\theta})$, and is accepted, $\boldsymbol{\theta}^t = \boldsymbol{\theta}'$ with probability $\min(1, r)$ where,

$$r = \frac{p(\boldsymbol{\theta}' \mid \boldsymbol{y})}{p(\boldsymbol{\theta}^{t-1} \mid \boldsymbol{y})} \cdot \frac{q(\boldsymbol{\theta}^{t-1} \mid \boldsymbol{\theta}')}{q(\boldsymbol{\theta}' \mid \boldsymbol{\theta}^{t-1})} \tag{7}$$

In (7) above, the factor $p(\boldsymbol{\theta}' \mid \boldsymbol{y})/p(\boldsymbol{\theta}^{t-1} \mid \boldsymbol{y})$ is the ratio of the posterior densities and the factor $q(\boldsymbol{\theta}^{t-1} \mid \boldsymbol{\theta}')/q(\boldsymbol{\theta}' \mid \boldsymbol{\theta}^{t-1})$ is the ratio of the transition densities at the $t-$iteration. Sampling on a simplex, under specific cases, using MCMC-type algorithms refers to updating one particular co-ordinate of $\boldsymbol{\theta}$ and correspondingly adjusting the rest, so that, the proposal continues to stay on the simplex.

We illustrate the SALT proposal with an MCMC algorithm, which supports efficient sampling on a simplex. This proposal identifies a candidate point $\boldsymbol{\theta}'$ by moving randomly from a current point $\boldsymbol{\theta}$, where a single element of $\boldsymbol{\theta}$, θ_i is updated and the remaining θ_j's, $j \neq i$, are adjusted. For every MCMC iteration, this process is implemented successively for each co-ordinates of the parameter vector $\boldsymbol{\theta}$, i.e., for all $i = 1, 2, \ldots, k$.

The proposal for θ_i' is given by,

$$\theta_i' = \frac{\exp\left\{\log\left(\frac{\theta_i}{1-\theta_i}\right) + h_i Z\right\}}{1 + \exp\left\{\log\left(\frac{\theta_i}{1-\theta_i}\right) + h_i Z\right\}} \tag{8}$$

where, $Z \sim N(0,1)$ and $h_i \in \mathbb{R}^+$. All other co-ordinates of $\boldsymbol{\theta}'$ are respectively rescaled to maintain the constraint, $\sum_{i=1}^{k} \theta_i = 1$. Consider θ_l', $l \neq i$ and θ_j', $j \neq i, l$, to be defined as,

$$\theta_j' = (1 - \theta_i')\left(\frac{\theta_j}{1-\theta_i} + U_j\right) \tag{9}$$

where, $U_j \overset{ind}{\sim} \text{Unif}(-\epsilon, \epsilon)$ with $\epsilon > 0$ and,

$$\begin{aligned} \theta_l' &= (1 - \theta_i')\left(\frac{\theta_l}{1-\theta_i} - \sum_{j \neq i,l} U_j\right) \\ &= 1 - \sum_{j \neq l} \theta_j' \end{aligned} \tag{10}$$

since, $\theta_l = 1 - \theta_i - \sum_{j \neq i,l} \theta_j$. In (9) above, the factor $\theta_j/(1-\theta_i)$ denotes the proportion of mass that θ_j has relative to all other co-ordinates of $\boldsymbol{\theta}$, except θ_i. The factor $(1-\theta_i')$ in both (9) and (10), rescales each proportion relative to the remaining mass after θ_i' has been determined. The random noise U_j in (9) expands the space on which θ_j, $j \neq i, l$, is defined, thereby reducing the mathematical complexity in computing the transition density $[\boldsymbol{\theta'} \mid \boldsymbol{\theta}]$, which is a well-defined density supported on the $(k-1)-$dimensional subspace of the $k-$dimensional simplex. The transition in (10), provides a candidate $\boldsymbol{\theta'}$, that approximately sums to 1 and has all the co-ordinates bounded as, $-\epsilon \leq \theta_i' \leq 1+\epsilon$, for all $i = 1, 2, \ldots, k$. The case when $\epsilon \neq 0$ suggests that the proposal lies outside the simplex, which is circumvented in Director et al., 2017 by considering ϵ arbitrarily close to 0, restricting the proposals to be supported on the simplex.

We turn to deriving the *acceptance ratio* of the MH algorithm for Bayesian inference, using the SALT sampler proposals considered above. The transition density $q(\boldsymbol{\theta'} \mid \boldsymbol{\theta})$ can be factored as,

$$q(\boldsymbol{\theta'} \mid \boldsymbol{\theta}) = q(\theta_i' \mid \theta_i) \prod_{j \neq i,l} q(\theta_j' \mid \theta_i', \boldsymbol{\theta}) \tag{11}$$

Observe that, from (8) we have,

$$\log\left(\frac{\theta_i'}{1-\theta_i'}\right) \sim N\left(\log\left(\frac{\theta_i}{1-\theta_i}\right), h_i^2\right) \tag{12}$$

Further, standardizing and transforming from the logit scale, i.e., $\text{logit}(x) = \log(x/(1-x))$, to natural scale, $q(\theta_i' \mid \boldsymbol{\theta})$ can be obtained as,

$$\begin{aligned} q(\theta_i' \mid \boldsymbol{\theta}) &= \phi\left(\frac{\log\left(\frac{\theta_i'}{1-\theta_i'}\right) - \log\left(\frac{\theta_i}{1-\theta_i}\right)}{h_i}\right)\left|\frac{d}{d\theta_i'}\left[\frac{\log\left(\frac{\theta_i'}{1-\theta_i'}\right) - \log\left(\frac{\theta_i}{1-\theta_i}\right)}{h_i}\right]\right| \\ &= h_i^{-1}\phi\left(\frac{\log\left(\frac{\theta_i'}{1-\theta_i'}\right) - \log\left(\frac{\theta_i}{1-\theta_i}\right)}{h_i}\right)\frac{1}{\theta_i'(1-\theta_i')} \end{aligned} \tag{13}$$

where, $\phi(.)$ is a $N(0,1)$ density. The transitions $q(\theta_j' \mid \theta_i', \boldsymbol{\theta})$, $j \neq i, l$, we note that, $[\theta_j' \mid \theta_i', \boldsymbol{\theta}]$ is uniformly distributed with,

$$q(\theta_j' \mid \theta_i', \boldsymbol{\theta}) = \frac{1}{2\epsilon(1-\theta_i')} \tag{14}$$

where, θ_j' is given by (9). Therefore, the entire transition density for the transitioning step, $\boldsymbol{\theta'} \leftarrow \boldsymbol{\theta}$ is,

$$\begin{aligned} q(\boldsymbol{\theta'} \mid \boldsymbol{\theta}) &= h_i^{-1}\phi\left(\frac{\log\left(\frac{\theta_i'}{1-\theta_i'}\right) - \log\left(\frac{\theta_i}{1-\theta_i}\right)}{h_i}\right)\frac{1}{\theta_i'(1-\theta_i')}\left[\frac{1}{2\epsilon(1-\theta_i')}\right]^{k-2} \\ &= h_i^{-1}\phi\left(\frac{\log\left(\frac{\theta_i'}{1-\theta_i'}\right) - \log\left(\frac{\theta_i}{1-\theta_i}\right)}{h_i}\right)\frac{1}{\theta_i'}\left[\frac{1}{1-\theta_i'}\right]^{k-1}\left[\frac{1}{2\epsilon}\right]^{k-2} \end{aligned} \tag{15}$$

The Normal distribution is *symmetric*, noting that, the corresponding transition ratio is obtained as,

$$\frac{q(\boldsymbol{\theta} \mid \boldsymbol{\theta'})}{q(\boldsymbol{\theta'} \mid \boldsymbol{\theta})} = \frac{\theta_i'}{\theta_i}\left[\frac{1-\theta_i'}{1-\theta_i}\right]^{k-1} \tag{16}$$

which yields the acceptance ratio as,

$$
\begin{aligned}
r &= \frac{p(\boldsymbol{\theta}' \mid \boldsymbol{y})}{p(\boldsymbol{\theta} \mid \boldsymbol{y})} \cdot \frac{q(\boldsymbol{\theta} \mid \boldsymbol{\theta}')}{q(\boldsymbol{\theta}' \mid \boldsymbol{\theta})} \\
&= \frac{p(\boldsymbol{\theta}' \mid \boldsymbol{y})}{p(\boldsymbol{\theta} \mid \boldsymbol{y})} \cdot \frac{\theta_i'}{\theta_i} \left[\frac{1-\theta_i'}{1-\theta_i} \right]^{k-1}
\end{aligned} \tag{17}
$$

where, r above is independent of ϵ, implying that, ϵ in (9) can be chosen arbitrarily close to 0 such that, proposals of the above form continues to be supported on the simplex. The resulting MH algorithm with the SALT sampler proposals is summarized below in Algorithm 1.

Algorithm 1: Metropolis-Hastings Algorithm based on the SALT Sampler Proposals and Transition Density

Input: Initial value of the $k-$dimensional parameter of interest, $\boldsymbol{\theta}^0$.
Output: Samples from the target/true posterior distribution (or, stationary distribution) $p(\boldsymbol{\theta} \mid \boldsymbol{y})$, $\boldsymbol{\theta}^t$, after $t-$iterations of the MCMC where, $t \geq 0$.

for $t = 1, 2, \ldots,$ *maxiter* **do**
  A value θ_i' is proposed from (8).
  This value proposed above, is *accepted* with probability $\min(1, r)$, where the acceptance ratio r, is given by (17).
  if θ_i' *is accepted* **then**
    Accepted θ_i' is set to the *new proposed* value, i.e., $\theta_i^t = \theta_i'$.
    All θ_j''s, $j \neq i$ are set to their new values, given by (9) and (10), following which, $\theta_j^t = \theta_j'$, for all $j \neq i$.
    Finally, $\boldsymbol{\theta}^t = \boldsymbol{\theta}'$.
  end
  else
    $\boldsymbol{\theta}^t$ is set to $\boldsymbol{\theta}^{t-1}$.
  end
end

4 The Problem Setup and Further Details

Recall, that we consider a simple modification to the specification of the priors on G_j in (2) as,

$$
\begin{aligned}
G_1 &\sim DP(\alpha_1, G_0), \\
G_2 \mid G_1 &\sim DP(\alpha_2, G_1),
\end{aligned} \tag{18}
$$

where G_0 is a known base probability measure. The need for these nested priors arise naturally when there is an inherent dependency between the groups. One such example is time-series data. One might be interested in clustering stocks based on daily prices for each year. Each calendar year is then a group. The groups naturally have time dependence (i.e., one does not expect the clustering of stocks to change dramatically in consecutive years), which may be represented by an autoregressive (AR) model. Particularly, for an AR model with lag 1, the time dependencies can be represented by a simple Directed Acyclic Graph (DAG, see Figure 1). With this specific DAG, it is natural to consider dependency between the time-specific random measures as,

$$
G_1 \mid G_0 \sim DP\left(\alpha_1, G_0\right), \tag{19}
$$

$$
G_t \mid G_{t-1} \sim DP\left(\alpha_t, G_{t-1}\right), \quad t = 2, \ldots, T, \tag{20}
$$

where T denotes the total number of observed time points. In this report, we specifically look at the analysis for $T = 2$ time points, which still, highlights the need to consider efficient sampling strategies on the simplex.

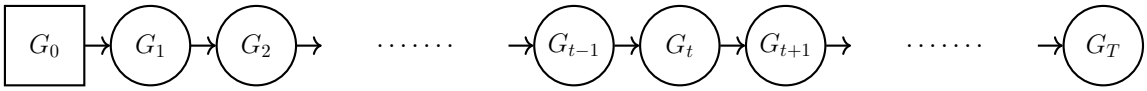


Figure 1: The DAG denoting time-dependency.

The corresponding mixture model can be derived as an infinite limit of the finite mixture model,

$$\begin{aligned}
\boldsymbol{\beta}_1 &\sim Dir\left(\alpha_1/L, \ldots, \alpha_1/L\right), \\
\boldsymbol{\beta}_2 \mid \boldsymbol{\beta}_1 &\sim Dir\left(\alpha_2\left(\beta_{11}, \ldots, \beta_{1L}\right)\right), \\
\phi_l \mid G_0 &\sim G_0, \\
z_{ji} \mid \boldsymbol{\beta}_j &\sim \boldsymbol{\beta}_j, \\
x_{ji} \mid z_{ji}, (\phi_l)_{l=1}^L &\sim F\left(\phi_{z_{ji}}\right),
\end{aligned} \tag{21}$$

as $L \to \infty$. Based on this finite mixture model approximation with a large enough truncation level L, one may consider a blocked Gibbs sampler to facilitate posterior inference. As an illustration, consider a univariate Gaussian likelihood for each of the two groups with unknown mean and known variance, say 1, i.e., $x_{ji} \mid z_{ji}, (\mu_l)_{l=1}^L \sim N\left(\mu_{z_{ji}}, 1\right)$. Furthermore, we assume a hierarchical gamma prior on the concentration parameters, $\alpha_1 \sim Gamma(\alpha_0, 1)$ and $\alpha_2 \mid \alpha_1 \sim Gamma(\alpha_1, 1)$. The base measure G_0 is specified as $N(\mu_0, \lambda^{-1})$. For the given specification, (21) can be expressed as,

$$\begin{aligned}
\alpha_1 &\sim Gamma(\alpha_0, 1), \\
\boldsymbol{\beta}_1 &\sim Dir\left(\alpha_1/L, \ldots, \alpha_1/L\right), \\
\alpha_2 \mid \alpha_1 &\sim Gamma(\alpha_1, 1), \\
\boldsymbol{\beta}_2 \mid \boldsymbol{\beta}_1 &\sim Dir\left(\alpha_2\left(\beta_{11}, \ldots, \beta_{1L}\right)\right), \\
\mu_l &\sim N(\mu_0, \lambda^{-1}), \\
z_{ji} \mid \boldsymbol{\beta}_j &\sim \boldsymbol{\beta}_j, \\
x_{ji} \mid z_{ji}, (\mu_l)_{l=1}^L &\sim N\left(\mu_{z_{ji}}, 1\right),
\end{aligned} \tag{22}$$

With the above distributional structure, Gibbs sampling is straightforward. We use $\pi(.)$ and $\pi(. \mid -)$ to denote the prior distribution and the conditional distribution, respectively, of the parameter specified in the argument.

The conditional posterior distribution for the atoms $((\mu_l)_{l=1}^L)$ is given by,

$$\pi(\mu_l \mid -) \sim N\left(\mu_l \mid \frac{m_{\cdot l}\bar{x}_l + \lambda\mu_0}{m_{\cdot l} + \lambda}, \frac{1}{m_{\cdot l} + \lambda}\right), \quad l = 1, \ldots, L, \tag{23}$$

where $m_{\cdot l} = \sum_{j=1}^{2}\sum_{i=1}^{n_j} \mathbb{1}(z_{ji} = l)$ and $m_{\cdot l}\bar{x}_l = \sum_{j=1}^{2}\sum_{i=1}^{n_j} \mathbb{1}(z_{ji} = l)\, x_{ji}$.

The conditional posterior distributions for the latent cluster labels are given by,

$$P(z_{ji} = l \mid -) \propto \beta_{jl} N(x_{ji} \mid \mu_l, 1), \quad l = 1, \ldots, L, \quad i = 1, \ldots, n_j \quad j = 1, 2. \tag{24}$$

The conditional posterior distribution for the weight $\boldsymbol{\beta}_2$ is given by,

$$\pi(\boldsymbol{\beta}_2 \mid -) \sim Dir(\boldsymbol{m}_2 + \alpha_2\boldsymbol{\beta}_1), \tag{25}$$

where $\boldsymbol{m}_2 = (m_{21}, \ldots, m_{2L})$, with $m_{2l} = \sum_{i=1}^{n_2} \mathbb{1}(z_{2i} = l)$.

The conditional posterior distribution for the weight $\boldsymbol{\beta}_1$ is given by,

$$\pi(\boldsymbol{\beta}_1 \mid -) \propto \prod_{l=1}^{L} \left\{ \frac{\beta_{1l}^{m_{1l} + \alpha_1/L - 1}\beta_{2l}^{\alpha_2\beta_{1l} - 1}}{\Gamma(\alpha_2\beta_{1l})} \right\}, \tag{26}$$

where $m_{1l} = \sum_{i=1}^{n_1} \mathbb{1}(z_{1i} = l)$.

The conditional posterior distribution for the concentration parameter α_1 is given by,

$$\pi(\alpha_1 \mid -) \propto e^{-\alpha_1}\alpha_1^{\alpha_0-1}\alpha_2^{\alpha_1}\frac{\prod_{l=1}^{L}\beta_{1l}^{\frac{\alpha_1}{L}}}{\left\{\Gamma(\frac{\alpha_1}{L})\right\}^L}. \tag{27}$$

The conditional posterior distribution for the concentration parameter α_2 is given by,

$$\pi(\alpha_2 \mid -) \propto e^{-\alpha_2}\alpha_2^{\alpha_1-1}\frac{\Gamma(\alpha_2)}{\prod_{l=1}^{L}\Gamma(\alpha_2\beta_{1l})}\prod_{l=1}^{L}\beta_{2l}^{\alpha_2\beta_{1l}}. \tag{28}$$

Note that the full conditionals of α_1, α_2, and $\boldsymbol{\beta}_1$ are not standard distributions that have direct samplers. We adopt a *Metropolis-within-Gibbs* strategy to sample from their corresponding full conditional distributions. Since α_j's are real-valued, sampling using a Metropolis step is straightforward. However, the main bottleneck in sampling is the weight $\boldsymbol{\beta}_1$, which has a complex structure on the simplex. To mitigate this problem, we use the SALT sampler (Director et al., 2017) for which the implementation is publicly available as an `R` package.

5 Simulations

5.1 Data generation

For simulations, we generated data within each of the two groups from a five-component mixture of univariate Gaussian distributions, the true means of which were taken to be $\phi = (-10, -3, 4, 11, 18)$ and a known precision parameter, $\tau = 1$. Considering true $\alpha_0 = 10$, we first drew the concentration parameter corresponding to population 1, as $\alpha_1 \sim Gamma(\alpha_0, 1)$ and the concentration parameter corresponding to population 2 as $\alpha_2 \mid \alpha_1 \sim Gamma(\alpha_1, 1)$. The group-specific mixture weights were drawn from their corresponding prior distribution in (22), with $L = 5$ (corresponds to the number of true mixture components). Considering sample sizes of 100 and 120 for the two groups, the true cluster indicators of each group were sampled from a multinomial distribution with probabilities equal to the mixture weights. Using these true cluster indices for each group, samples were drawn from the Gaussian distribution with the cluster-specific mean.

5.2 The Use of Dirichlet proposal in Metropolis Sampling

First, we ran a Metropolis-within-Gibbs sampler by iteratively sampling from the conditional posterior distributions in equations (23) - (28). Since the number of clusters (mixture components) are assumed to be unknown apriori, we considered a large enough truncation level of our proposed model, i.e., $L = 10$ so that our model can adaptively estimate the number of clusters in a data-driven manner. The proposal distribution for sampling the concentration parameters, α_1 and α_2 were taken as their prior distributions. Particularly for the mixture weight $\boldsymbol{\beta}_1$, we considered several choices of proposal distributions. Particularly, at iteration t, we use the proposal $q(\boldsymbol{\beta}_1 \mid -)$, where we condition on other parameters.

- $Dir(\rho/L, \ldots, \rho/L)$, where ρ is a tuning parameter.
- $Dir(\rho\,\alpha_1^{(t-1)}/L, \ldots, \rho\,\alpha_1^{(t-1)}/L)$, where ρ is a tuning parameter.
- $Dir(\rho\,((\boldsymbol{m}_1^{(t)} + \alpha_1^{(t-1)})/L))$, where $\boldsymbol{m}_1^{(t)} = (m_{11}^{(t)}, \ldots, m_{1L}^{(t)})$, with $m_{1l}^{(t)} = \sum_{i=1}^{n_1}\mathbb{1}(z_{1i}^{(t)} = l)$ and ρ is a tuning parameter.

We considered several choices of the tuning parameters in small pilot runs of our sampler to monitor the acceptance rate for sampling $\boldsymbol{\beta}_1$. We ran 5,000 iterations of the sampler in parallel and after discarding the initial 2,000 samples, we calculated the acceptance rates. For the different choices of the tuning parameters, the first two proposals had no acceptances. However, for the third proposal distribution, the acceptance rates are shown in Table 1. This highlights the difficulty of choosing the proposal distribution carefully. We see that the choice $\rho = 0.99$ provides a reasonable acceptance rate and we choose the tuning parameter $\rho = 0.99$ in the final MCMC sampler. Furthermore, the

Choices of ρ	0.1	0.2	0.5	0.9	0.99	1	1.1	1.5	2
Acceptance Rate	0	0.001	0.08	0.293	0.311	0.355	0.358	0	0

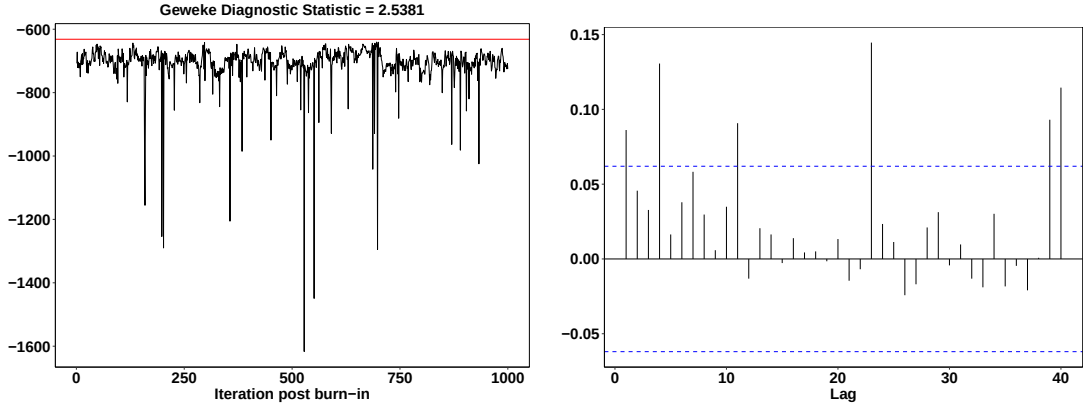
Table 1: Acceptance rate of $\boldsymbol{\beta}_1$ for several choices of the tuning parameters ρ

tuning of the parameter ρ can become troublesome and time consuming in higher dimensions or even for different datasets, which call the need for efficient methods to sample from the simplex.

We ran our proposed MCMC chain for 10,000 iterations. We discarded the first 5,000 iterations as burn-in and thinned the remaining chain by retaining every fifth sample, yielding a total of 1,000 posterior samples. The MCMC run achieved an acceptance rate of 33.1% for $\boldsymbol{\beta}_1$. Convergence diagnostics were assessed via traceplots of the log-likelihood, as shown in Figure 2a. Additionally, we calculated Geweke's diagnostic statistic [Geweke, 1991], reported at the top of the figure, which indicated some evidence of possible non-convergence. Furthermore, when we considered 50,000 iterations of our sampler, discarded the first 25,000 iterations as burn-in, and thinning by a factor of 25 the resulting chain of still showed some signs of non-convergence as indicated by the high value of Geweke's diagnostic statistic (Figure 3a). Furthermore, the traceplots of the co-ordinates of the mixture weight $\boldsymbol{\beta}_1$ are shown in Figure 4, which possibly shows high correlations. The posterior estimates of the mixture weights $\boldsymbol{\beta}_1$ along with the true values are shown in Table 2, which show that the estimates are quite far off from the true values (upto permutation of the order of the co-ordinates, which is due to label switching in mixture models).

Co-ordinates of $\boldsymbol{\beta}_1$	β_{11}	β_{12}	β_{13}	β_{14}	β_{15}	β_{16}	β_{17}	β_{18}	β_{19}	β_{110}
True	0.1537	0.1117	0.3494	0.2337	0.1516	0	0	0	0	0
Estimated	0.1024	0.0912	0.0991	0.1024	0.1060	0.1130	0.0932	0.084	0.1073	0.1015

Table 2: True and Posterior Estimated mixture weights (rounded upto 4 decimal places).



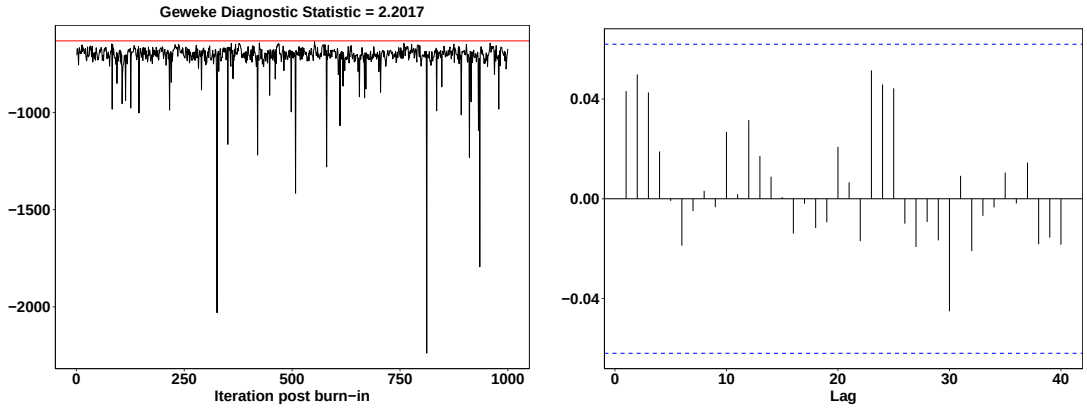
(a) Traceplot of log-likelihood.

(b) ACF plot of log-likelihood.

Figure 2: The traceplot and ACF of log-likelihood considering 10,000 iterations of our sampler post burn-in of 5,000 samples and thinning by a factor of 5. The red line corresponds to the true log-likelihood value.

5.3 The Use of SALT sampler proposal in Metropolis Sampling

Next, we considered the proposed SALT sampler in sampling the mixture weight $\boldsymbol{\beta}_1$. The Gibbs sampling steps for all other model parameters were as before. As before, we considered 10,000 iterations of our sampler, which took $<$ 1 minute on a MacBook Pro with M1 chip and 16GB RAM. After discarding the first 5,000 iterations as burn-in, we thinned the remaining chain by retaining every fifth sample, yielding a total of 1,000 posterior samples. The traceplots of the log-likelihood is shown in Figure 5a along with the Geweke's diagnostic statistic (reported at the



(a) Traceplot of log-likelihood.

(b) ACF plot of log-likelihood.

Figure 3: The traceplot and ACF of log-likelihood considering 50,000 iterations of our sampler post burn-in of 25,000 samples and thinning by a factor of 25. The red line corresponds to the true log-likelihood value.

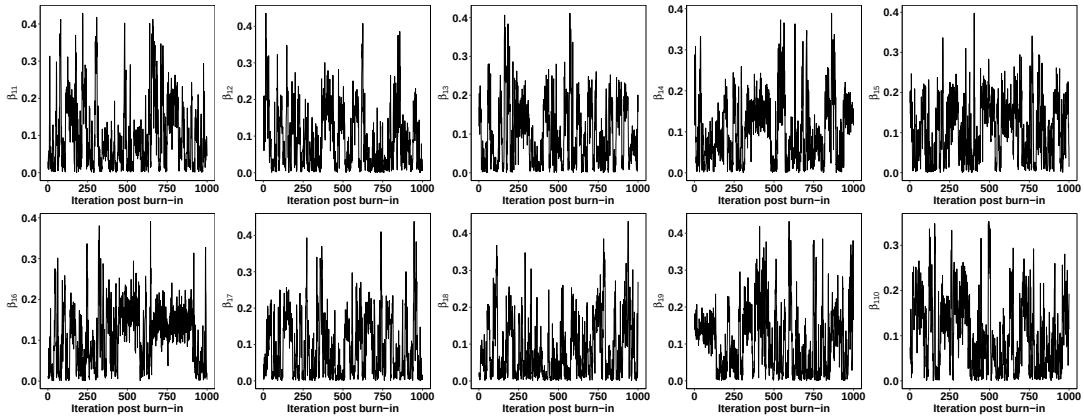


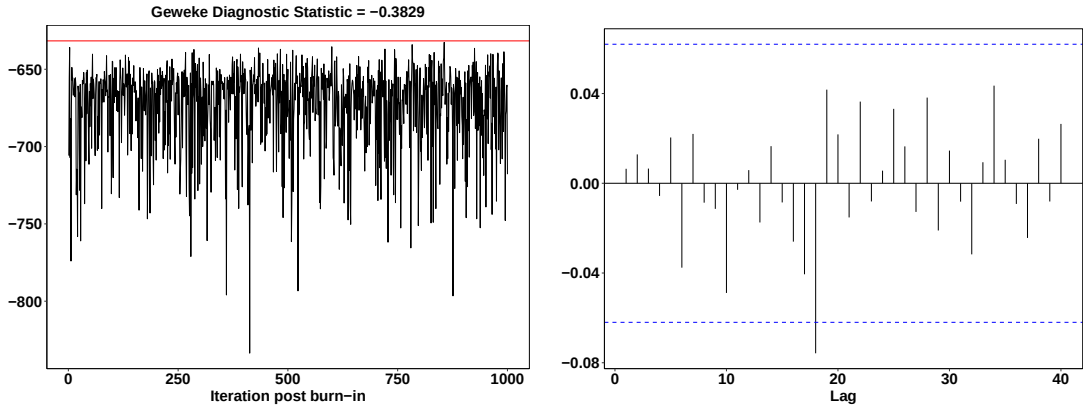
Figure 4: Traceplot of the co-ordinates of the mixture weight $\boldsymbol{\beta}_1$ post burn-in and thinning.

top of Figure), which indicated no evidence of non-convergence. Furthermore, the posterior log-likelihood values are very close to the true value as indicated by the red line. The corresponding ACF plot (Figure 5b) shows no significant autocorrelations. The traceplots of the co-ordinates of the mixture weight $\boldsymbol{\beta}_1$ are shown in Figure 6. The posterior estimates of the mixture weights $\boldsymbol{\beta}_1$ along with the true values are shown in Table 3, which show that the estimates are close to the true values (upto permutation of the order of the co-ordinates, which is due to label switching in mixture models).

We estimated the clusters by minimizing the variation of information [Meilă, 2003, Wade and Ghahramani, 2018] and they were compared with the true cluster labels for evaluation using adjusted Rand Index [Hubert and Arabie, 1985]. Figure 7 shows that the clusters were estimated perfectly for both the populations. Furthermore, the heatmaps of posterior co-clustering probabilities of the observations in Figure 8, highlight the lower uncertainty using SALT sampler in comparison to Dirichlet Metropolis proposal.

Co-ordinates of $\boldsymbol{\beta}_1$	β_{11}	β_{12}	β_{13}	β_{14}	β_{15}	β_{16}	β_{17}	β_{18}	β_{19}	β_{110}
True	0.1537	0.1117	0.3494	0.2337	0.1516	0	0	0	0	0
Estimated	0.3138	0.1942	0.1702	0.1259	0.1960	0	0	0	0	0

Table 3: SALTSampler: True and Posterior Estimated mixture weights (rounded upto 4 decimal places).



(a) Traceplot of log-likelihood.

(b) ACF plot of log-likelihood.

Figure 5: SALTSampler: The traceplot and ACF of log-likelihood considering 10,000 iterations of our sampler post burn-in of 5,000 samples and thinning by a factor of 5. The red line corresponds to the true log-likelihood value.

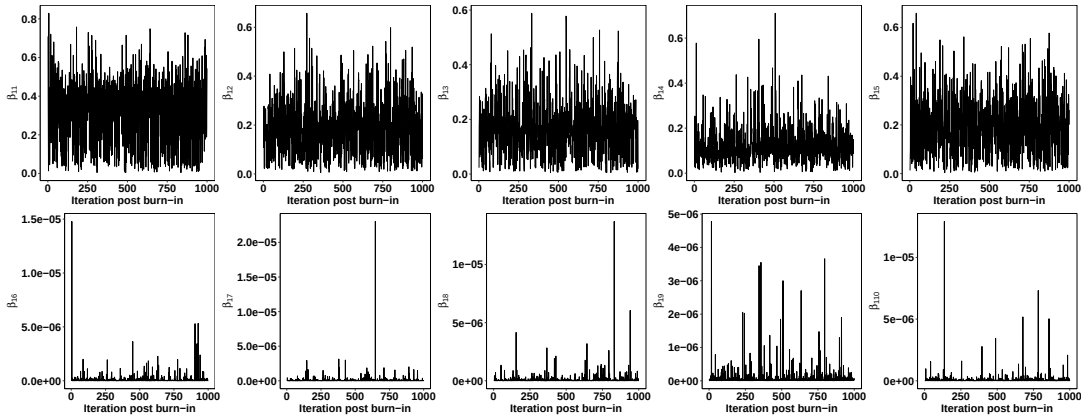


Figure 6: SALTSampler: Traceplot of the co-ordinates of the mixture weight $\boldsymbol{\beta}_1$ post burn-in and thinning.

5.4 Posterior Analysis Using the Probabilistic Programming Language Stan

Stan [Team, 2023] is a powerful, open-source probabilistic programming language designed for statistical modeling and high-performance computation. Developed for Bayesian inference, Stan allows users to define complex models and leverage state-of-the-art MCMC methods, such as the No-U-Turn Sampler (NUTS, Homan and Gelman, 2014), a variant of Hamiltonian Monte Carlo (HMC, Neal, 2011). Stan's efficient algorithms make it possible to estimate parameters of high-dimensional models that might be computationally prohibitive with traditional techniques. Furthermore, as previosuly highlighted, in the Stan software, all constrained variables are transformed to unconstrained variables to simplify HMC. Stan uses the element-wise transformation,

$$y_k = \log\left(\frac{z_k}{1-z_k}\right) - \log\left(\frac{1}{K-k}\right) \tag{29}$$

which represents the famous *stick-breaking* process. In (6) above, K refers to the number of pieces into which the stick is divided and z_k refers to the ratio of the length of the $k-$th piece of stick, x_k, relative to the total length, $1 - \sum_{j=1}^{k-1} x_j$, of the remaining pieces.

We ran $10,000$ iterations of the built-in NUTS in Stan, which took nearly 24 minutes on a MacBook Pro with M1 chip and 16GB RAM, which is significantly longer than running our

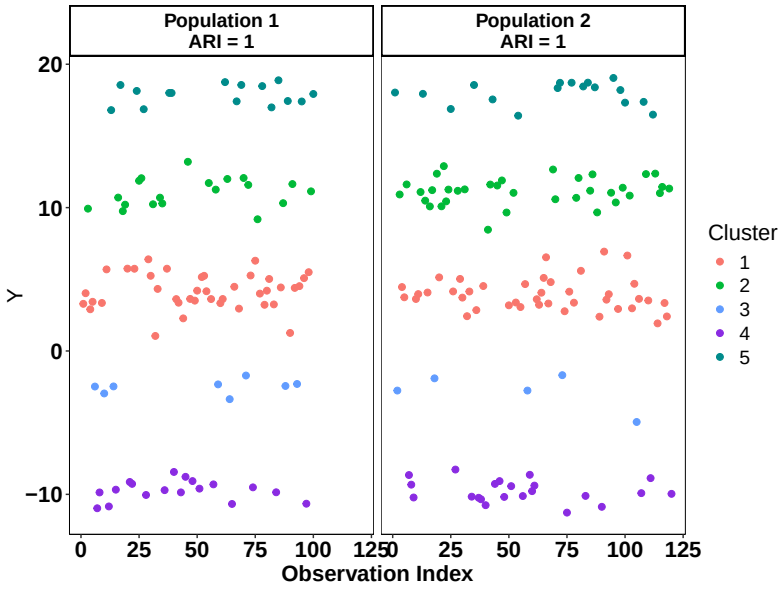
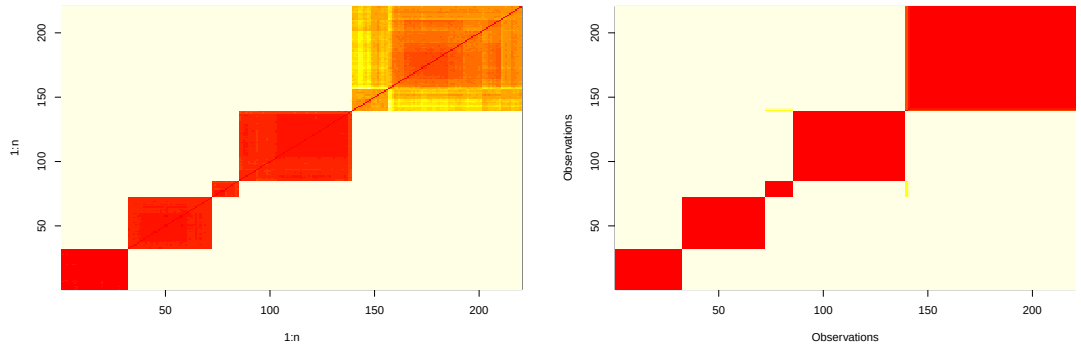


Figure 7: SALTSampler: Clustering performance for the two populations. The colors indicate the estimated clusters. Adjusted Rand index is reported at the top of each panel.



(a) Dirichlet Proposal in Metropolis Hastings.

(b) SALTSampler.

Figure 8: Posterior co-clustering probabilities of observations assigned to clusters by (a) Dirichlet Proposal in Metropolis Sampling and (b) SALTSampler.

proposed SALTSampler. We discarded the first 5,000 iterations as burn-in and retained every 5th iteration of posterior samples. Table 4 shows the true and posterior estimate of $\boldsymbol{\beta}_1$ obtained from Stan. We see that the estimated mixture weights are quite different from the true values even upto a permutation of the order of the co-ordinates. Furthermore, the estimated clusters (Figure 9) indicate that the clustering accuracy is lower than that obtained by our proposed SALTSampler (indicated by lower ARI values, reported at the top of the plot).

Co-ordinates of $\boldsymbol{\beta}_1$	β_{11}	β_{12}	β_{13}	β_{14}	β_{15}	β_{16}	β_{17}	β_{18}	β_{19}	β_{110}
True	0.1537	0.1117	0.3494	0.2337	0.1516	0	0	0	0	0
Estimated	0.0045	0.0059	0.0144	0.1674	0.0810	0.1715	0.1427	0.076	0.1874	0.1492

Table 4: STAN: True and Posterior Estimated mixture weights (rounded upto 4 decimal places).

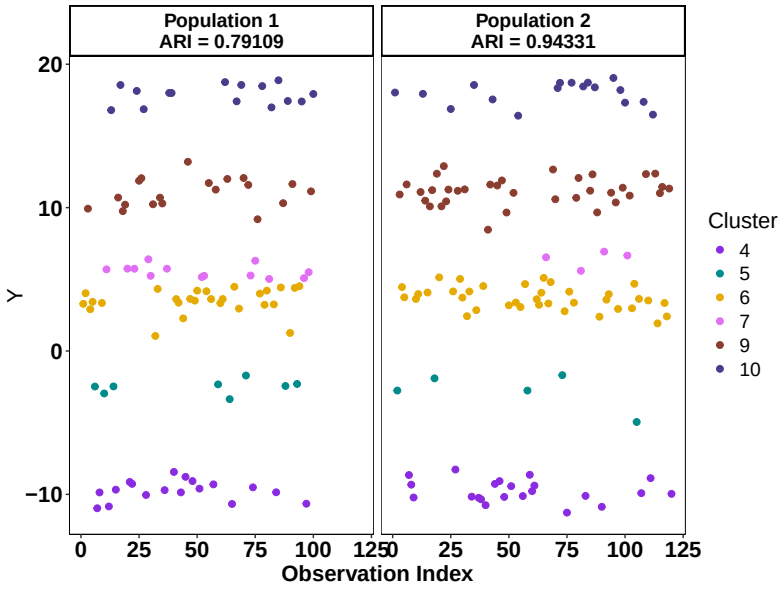


Figure 9: Stan: Clustering performance for the two populations. The colors indicate the estimated clusters. Adjusted Rand index is reported at the top of each panel.

6 Conclusion

The SALT proposal used in the Metropolis-within-Gibbs sampling algorithm implemented to infer from the simplex-supported full-conditional posterior distribution of $\boldsymbol{\beta}_1$, as in (26), has been illustrated to be efficient over the competing sampling algorithms using the Dirichlet proposal and the NUTS-based HMC routine in `Stan`. The results are highlighted through our simulation routines and numerics in Section 5. It has been observed that, while using the Dirichlet proposal, tuning the hyper-parameter ρ is quite challenging. Also, the posterior estimates of $\boldsymbol{\beta}_1$ has been reported keeping in mind the label switching phenomenon that is quite prevalent in mixture models. Furthermore, a study of clustering accuracy in each of the sampling algorithms highlights the superior performance of the SALT-based MCMC technique over the other two. The corresponding `R` code for performing the simulations and numerics can be accessed at https://github.com/Roy-SR-007/SALT-Simplex.

Therefore, conducting MCMC over complex spaces such as, a simplex-supported target distribution in our case can pose challenges. These bottlenecks are surmounted by carefully devising the proposal distribution which is being used in the MH algorithm using the SALT technique. We witness a complex structured full-conditional of $\boldsymbol{\beta}_1$, where the posterior distribution takes the "simplex raised to the power of a simplex" form that has been quite efficiently sampled by using the SALT proposal. Beyond our specific case, the SALT-based MCMC technique can be generalized to running multiple parallel chains as well as accommodating interaction among them.

References

J. Aitchison. The statistical analysis of compositional data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 44(2):139–177, 1982. ISSN 00359246. URL http://www.jstor.org/stable/2345821.

Dean Billheimer, Peter Guttorp, and William F Fagan. Statistical interpretation of species composition. *Journal of the American Statistical Association*, 96(456):1205–1214, 2001. doi: 10.1198/016214501753381850. URL https://doi.org/10.1198/016214501753381850.

Christopher W Thomas and John Aitchison. Compositional data analysis of geological variability and process: A case study. *Mathematical Geology*, 37(7):753–772, October 2005.

Jane M. Fry, Tim R. L. Fry, and Keith R. McLaren. Compositional data analysis and zeros in micro data. *Applied Economics*, 32(8):953–959, 2000. doi: 10.1080/000368400322002. URL https://doi.org/10.1080/000368400322002.

K. Gerald van den Boogaart and R. Tolosana-Delgado. "compositions": A unified r package to analyze compositional data. *Computers and Geosciences*, 34(4):320–338, 2008. ISSN 0098-3004. doi: https://doi.org/10.1016/j.cageo.2006.11.017. URL https://www.sciencedirect.com/science/article/pii/S009830040700101X.

John Aitchison and Michael Greenacre. Biplots of Compositional Data. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 51(4):375–392, 10 2002. ISSN 0035-9254. doi: 10.1111/1467-9876.00275. URL https://doi.org/10.1111/1467-9876.00275.

Peter Filzmoser, Karel Hron, and Clemens Reimann. Principal component analysis for compositional data with outliers. *Environmetrics*, 20(6):621–632, 2009. doi: https://doi.org/10.1002/env.966. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/env.966.

Hannah M. Director, James Gattiker, Earl Lawrence, and Scott Vander Wiel. Efficient sampling on the simplex with a self-adjusting logit transform proposal. *Journal of Statistical Computation and Simulation*, 87(18):3521–3536, 2017. doi: 10.1080/00949655.2017.1376063. URL https://doi.org/10.1080/00949655.2017.1376063.

Thomas S. Ferguson. A bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209–230, 1973. doi: 10.1214/aos/1176342360. URL https://doi.org/10.1214/aos/1176342360.

Charles E. Antoniak. Mixtures of dirichlet processes with applications to bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174, 1974. ISSN 00905364. URL http://www.jstor.org/stable/2958336.

Albert Y. Lo. On a Class of Bayesian Nonparametric Estimates: I. Density Estimates. *The Annals of Statistics*, 12(1):351 – 357, 1984. doi: 10.1214/aos/1176346412. URL https://doi.org/10.1214/aos/1176346412.

Michael D. Escobar and Mike West. Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, 90(430):577–588, 1995. doi: 10.1080/01621459.1995.10476550. URL https://www.tandfonline.com/doi/abs/10.1080/01621459.1995.10476550.

Steven N. MacEachern and Peter Müller. Estimating mixture of dirichlet process models. *Journal of Computational and Graphical Statistics*, 7(2):223–238, 1998. ISSN 10618600. URL http://www.jstor.org/stable/1390815.

Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006. doi: 10.1198/016214506000000302. URL https://doi.org/10.1198/016214506000000302.

Joshua S. Speagle. A conceptual introduction to markov chain monte carlo methods, 2020. URL https://arxiv.org/abs/1909.12313.

W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 04 1970. ISSN 0006-3444. doi: 10.1093/biomet/57.1.97. URL https://doi.org/10.1093/biomet/57.1.97.

John Geweke. Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. Staff Report 148, Federal Reserve Bank of Minneapolis, 1991. URL https://ideas.repec.org/p/fip/fedmsr/148.html.

Marina Meilă. Comparing clusterings by the variation of information. In Bernhard Schölkopf and Manfred K. Warmuth, editors, *Learning Theory and Kernel Machines*, pages 173–187, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg. ISBN 978-3-540-45167-9.

Sara Wade and Zoubin Ghahramani. Bayesian Cluster Analysis: Point Estimation and Credible Balls (with Discussion). *Bayesian Analysis*, 13(2):559 – 626, 2018. doi: 10.1214/17-BA1073. URL https://doi.org/10.1214/17-BA1073.

Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985. doi: 10.1007/BF01908075. URL https://doi.org/10.1007/BF01908075.

Stan Development Team. *Stan: A Probabilistic Programming Language*, 2023. URL https://mc-stan.org. Version 2.33.

Matthew D. Homan and Andrew Gelman. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623, January 2014. ISSN 1532-4435.

Radford M. Neal. MCMC using hamiltonian dynamics. In Steve Brooks, Andrew Gelman, Galin L. Jones, and Xiao-Li Meng, editors, *Handbook of Markov Chain Monte Carlo*, pages 113–162. Chapman & Hall/CRC, Boca Raton, FL, 2011.