Computational and Inferential Difficulties with Mixture Posterior Distributions

Gilles Celeux, Merrilee Hurn, Christian P. Robert

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Computational and inferential difficulties with mixture posterior distributions

Gilles Celeux, Merrilee Hurn, Christian Robert

No 3627

THÈME 4

ISSN 0249-6399

INRIA
RHÔNE-ALPES

Computational and inferential difficulties with mixture posterior distributions

Gilles Celeux, Merrilee Hurn, Christian Robert

Thème 4 — Simulation et optimisation
de systèmes complexes
Projet is2

Rapport de recherche n°3627 — — 34 pages

Abstract: This paper deals with both exploration and interpretation problems related to posterior distributions for mixture models. The specification of mixture posterior distributions means that the presence of $k!$ modes is known immediately. Standard Markov chain Monte Carlo techniques usually have difficulties with well-separated modes such as occur here; the Markov chain Monte Carlo sampler stays within a neighbourhood of a local mode and fails to visit other equally important modes. We show that exploration of these modes can be imposed on the Markov chain Monte Carlo sampler using tempered transitions based on Langevin algorithms. However, as the prior distribution does not distinguish between the different components, the posterior mixture distribution is symmetric and thus standard estimators such as posterior means cannot be used. Since this is also true for most non-symmetric priors, we propose alternatives for Bayesian inference for permutation invariant posteriors, including a clustering device and the call to appropriate loss functions. An important side-issue of this study is the highlighting of the flexibility and adaptability of Langevin Metropolis-Hastings algorithms as quasi-automated Markov chain Monte Carlo algorithms when the posterior distribution is known up to a constant.

Key-words: Classification, label switching, Langevin diffusions, loss functions, Markov chain Monte Carlo, simulated tempering.

(Résumé : tsvp)

This work was partially supported by the TMR network, contract C.E. CT 96–0095. The authors are grateful to the participants to the workshops "McCube" and "MCMC'Ory". G. Celeux is with Inria Rhône-Alpes, M. Hurn is with Crest-Insee and University of Bath, Ch. Robert is with Crest-Insee and Upres-A CNRS 6085, université de Rouen.

Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN (France)
Téléphone : 04 76 61 52 00 - International: +33 4 76 61 52 00
Télécopie : 04 76 61 52 52 - International: +33 4 76 61 52 52

Difficultés numériques et inférentielles dans l'analyse de la loi a posteriori d'un mélange

Résumé : Dans un contexte bayésien, nous nous intéressons aux problèmes d'exploitation et d'interprétation de la loi a posteriori d'un modèle de mélange. Intrinsèquement, la loi a posteriori d'un modèle de mélange présente $k!$ modes. Les méthodes classiques de Monte-Carlo par chaînes de Markov ont en général de grandes difficultés pour restituer ces modes très séparés : l'échantillonneur de Monte-Carlo reste dans le voisinage d'un mode sans parvenir à visiter les autres modes d'égales importances. Nous montrons que l'exploration de ces modes peut être imposée à l'échantillonneur de Monte-Carlo en utilisant des transitions de refroidissement fondées sur les algorithmes de Langevin. Mais dans un cadre non informatif, comme la loi a priori traite les composants du mélange de manière indifférenciée, la loi a posteriori est symétrique et les estimateurs classiques comme les moyennes a posteriori sont inutilisables. Ils le restent d'ailleurs avec des lois a priori informatives qui distinguent les composants entre eux. Nous proposons différentes solutions pour mener à bien l'inférence bayésienne à partir de lois a posteriori invariantes par permutation des indices des composants du mélange. L'une utilise une technique de classification, l'autre se fonde sur la minimisation de fonctions de perte appropriées. Un à-côté important de cette étude est la mise en évidence de la souplesse des algorithmes de Langevin-Metropolis-Hasting qui peuvent être vus comme des algorithmes de Monte-Carlo par chaînes de Markov quasi automatiques lorsque la loi a posteriori n'ait connu qu'à une constante près.

Mots-clé : classification, renversement d'étiquetage, diffusions de Langevin, fonctions de perte, chaînes de Markov de Monte-Carlo, refroidissement simulé.

1 Introduction

Consider the mixture distribution

$$f_\xi(x) = \sum_{j=1}^{k} p_j \, f(x|\zeta_j) \, , \tag{1}$$

where $\xi = (\zeta_1, \ldots, \zeta_k, p_1, \ldots, p_k)$, the weights satisfying $p_j \geq 0$ with $p_1 + \ldots + p_k = 1$, and the $f(\cdot|\zeta_j)$'s being from some parametric family, for example exponential. An identifiability problem stems from the invariance of (1) under permutation of the indices. This problem is well-recognised in the literature (for example, Titterington, Smith and Makov, 1985), and is usually solved by imposing an identifying constraint on the parameters, for instance $p_1 \geq \ldots \geq p_k$ (Mengersen and Robert, 1996, Richardson and Green, 1997, or Stephens, 1997). The identifiability constraint may even be the starting point for a non-informative modelling, as in Mengersen and Robert (1996) or Roeder and Wasserman (1997). The computational difficulties resulting from working with a posterior distribution based on (1) are also well-charted and various Markov chain Monte Carlo strategies have been proposed in the literature (Diebolt and Robert, 1994, or Liu, Liang and Wong, 1998). Inference on the parameters is then derived from the Markov chain Monte Carlo sample, the Bayesian estimates being constructed as ergodic averages using diagnostics of convergence analysed in Guihenneuc *et al.* (1999).

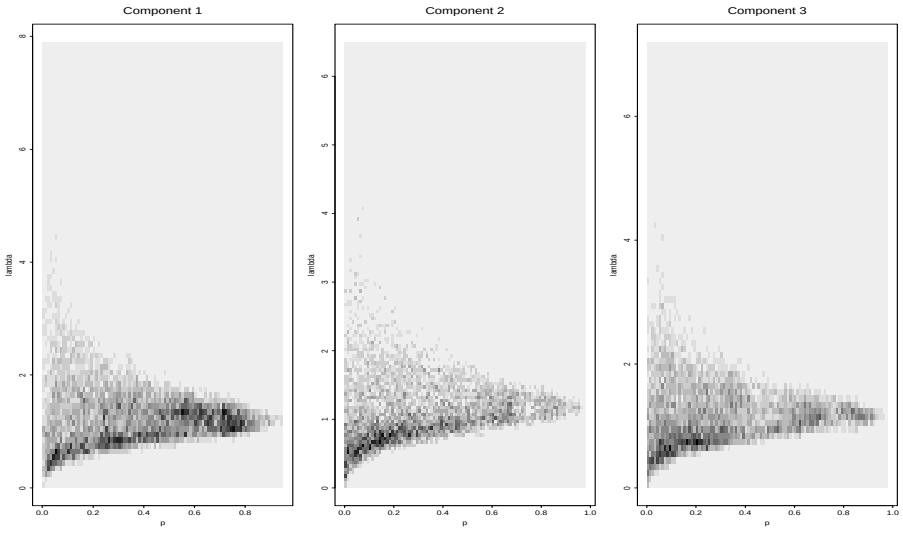
Unfortunately there are difficulties with the model representation, both at the *exploration* and at the *interpretation* stage. Firstly, the influence of the parameter ordering for identifiability, if used, is hard to assess and is certainly less benign than previously thought since it has a bearing both on the design and performance of the Markov chain Monte Carlo sampler and on the resulting inference, as we will demonstrate. Secondly, the exploration of the posterior support by standard Markov chain Monte Carlo samplers may either be too *local*, in the sense that the Markov chain Monte Carlo sample is unable to visit the whole range of the posterior modes, usually staying in the neighbourhood of one of the major modes with rare jumps between modes, or on the contrary too *unstable*, in the sense that the observations are allocated to all the components of the model in the course of the simulation, resulting in similar estimates for the (p_j, ζ_j)'s. Such difficulties are particularly obvious when symmetric priors are used on the parameter pairs (p_j, ζ_j), as in Diebolt and Robert (1994), Chib (1995) and Richardson and Green (1997), since the posterior distribution is then also invariant to permutation of the indices. This results in identical marginals for the parameter pairs, that is marginals which are independent of j. The *interpretation* of the posterior distribution is thus delicate since any inference through posterior means (for example, ergodic averages) is inappropriate since these means arise through the marginal distributions. Monitoring of the Markov chain Monte Carlo sample usually shows that it does not explore the $k!$ equivalent modes of the posterior distribution since the estimates of the marginal posterior distributions are strongly asymmetric in most cases. This is deeply unsatisfactory from an Markov chain Monte Carlo point of view, since it presents a rare setting where we know that some regions of the parameter space have not been visited by

the Markov chain. An equivalent perspective is to consider that these revealed symmetries induce basic control variates which are such that the usual samplers are never stopped against these variates in the wide majority of cases. Although somewhat presumptuous, we consider that almost the entirety of Markov chain Monte Carlo samplers implemented for mixture models has failed to converge! Moreover, we wish to stress that harm can result from the statistical interpretation of Markov chain Monte Carlo samples produced by placing constraints on the parameters. Although, from a statistical point of view, the exploration of the $k!$ modal regions of the posterior distribution is redundant (since the indices j of the components are *not* identifiable), the picture provided by the Markov chain Monte Carlo sample may be, and in many cases is, incomplete because the Markov chain almost never leaves the vicinity of one particular local mode and thus only covers a fraction of the support of the true posterior, even under the identifiability constraint. For this reason we feel that truncations of the parameter space may jeopardize the resulting inference (contrary to previous perspectives adopted in Gruet *et al.*, 1999, or Robert and Mengersen, 1998). Even when the truncation does not modify the prior distribution (which is not exactly the case in Mengersen and Robert, 1996, or Roeder and Wasserman, 1997), the choice of which of the parameters is used for ordering (that is, in the Gaussian case, weight *vs.* mean *vs.* variance) leads to a particular partition of the complete posterior distribution which may modify the corresponding inference. The point at issue here is that the truncation does not necessarily respect the geometry and shape of the unrestricted posterior distribution; while some orderings may nicely isolate a single mode of this distribution, most will involve parts of several modal regions. A consequence of this is that if a posterior mean is then calculated, the estimate will dwell in a valley between the $k!$ modes rather than close to one of them. In addition, removing the ordering constraint from the prior still leaves the exploration problem open; one must decide whether the posterior distribution is sufficiently well described by the Markov chain Monte Carlo sample, that is whether the unexplored part of the support can be recovered by permutations.

Logically it follows that a novel perspective on the simulation of the posterior distribution and the relaxing of the identifiability constraints must also lead to a novel approach at the interpretation stage, that is for Bayesian inference in this setup. Figure 1 illustrates the irrelevance of using posterior means when the identifiability constraint is not enforced, by presenting the case of a three component exponential mixture where components are sufficiently close to allow a standard Gibbs sampler to move freely between the modes. In this case, the three marginal posterior distributions display a high degree of similarity and this leads to nearly identical posterior means, in other words to a representation of the model by essentially a single exponential distribution. Similar phenomena can be observed in Gruet *et al.* (1999) for the constrained case leading to an estimate where one component overwhelms the others which become negligible.

The argument proposed by this paper is that (a) the unconstrained posterior is the most natural (posterior) distribution to consider, especially in non-informative settings, (b) there exist accelerating strategies which fit naturally in the mixture setting, not requiring additional implementation efforts, (c) there exist alternatives to the posterior means which provide

Figure 1: Marginal posterior histograms for the parameters (p_j, λ_j) $(j = 1, 2, 3)$ of a 3 component exponential mixture for the sample produced by a standard Gibbs sampler for an unconstrained symmetric prior (uniform Dirichlet on the vector of p_j's and exponential $\mathcal{E}xp(1)$ on the λ_j's), based on a simulated sample of size 100 and 10,000 iterations. *(Grey levels scale the height of the histogram: the higher the value of the grid, the darker the hue.)*



sensible Bayesian answers to the estimation of the parameters of (1). It is worth noting that while informative priors can make identifying constraints redundant by producing distinct distributions on the component parameters ζ_j, these priors most often still allow for some amount of switching between the components which is not observed for standard algorithms.

The paper is organised as follows: we describe in Section 2 the standard available Markov chain Monte Carlo solutions and show through examples that they fail to achieve the required symmetry. We then introduce in Section 3 a tempering strategy which is shown to overcome the previous difficulty. In Sections 4 and 5, we develop various approaches for deriving Bayes estimates from a component-wise symmetric Markov chain Monte Carlo sample. We concentrate throughout on normal and exponential mixtures, that is

$$\sum_{j=1}^{k} p_j \, \mathcal{N}(\theta_j, \tau_j^2) \qquad \text{and} \qquad \sum_{j=1}^{k} p_j \, \mathcal{E}xp(\lambda_j).$$

However the approach suggested here extends directly to other types of mixtures (Poisson, Pareto, etc.), and also to other categories of latent variable models (Robert, 1998) and non-identifiable problems. Similarly, we only consider the case of symmetric priors, but obviously, modest departures from symmetry would lead to similar behaviour when the

prior does not counter the lack of identifiability sufficiently strongly, as mentioned above. Our choice of symmetric proper priors is justified as in Richardson and Green (1997), namely by an empirical determination of location-scale parameters based on the data-spread. More precisely, we use conjugate priors in both cases, that is $\mathcal{D}(1,\ldots,1)$ distributions on the weights, $\mathcal{E}xp(1)$ distributions on the scale parameters λ_j and τ_j^2 and $\mathcal{N}(0,10\tau_j^2)$ distributions on the means θ_j.

2 Mixing properties of various proposals

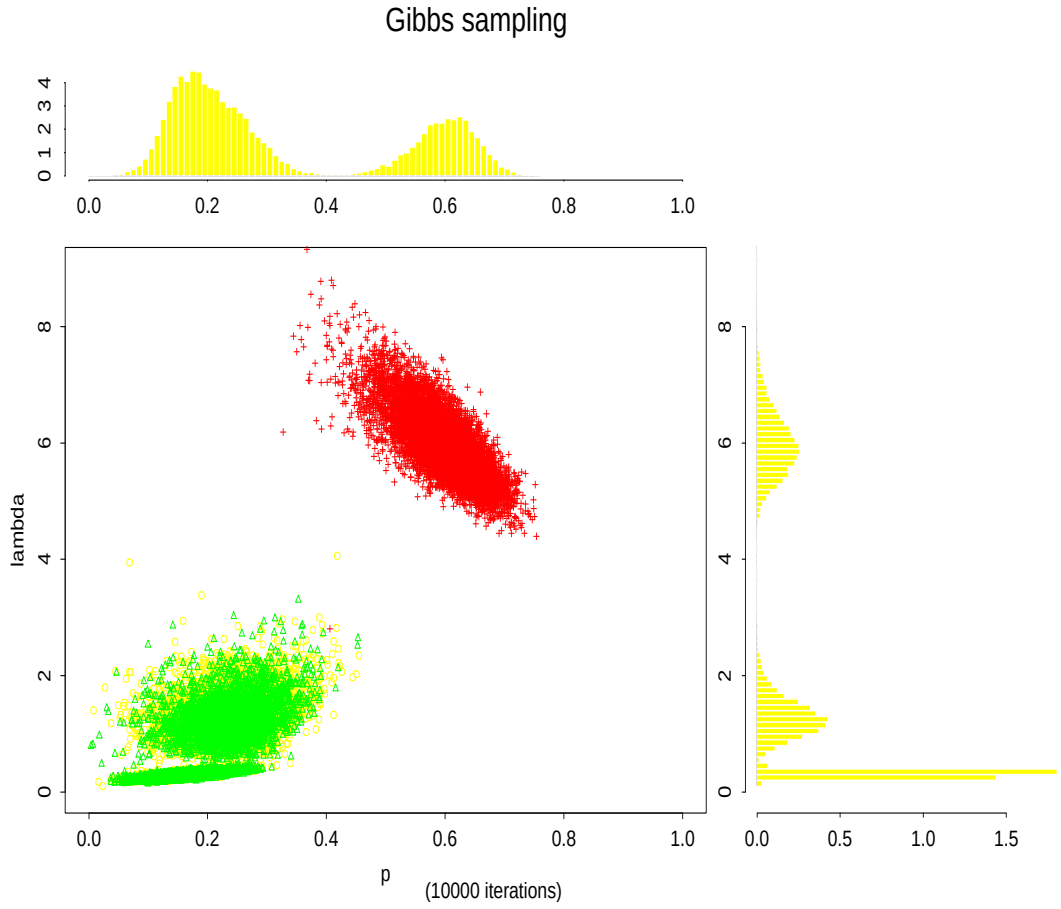
We consider the standard Markov chain Monte Carlo samplers in turn showing that each fails to achieve the symmetry inherent to the whole true stationary distribution. We concentrate in §2.3 on the Metropolis-Hastings algorithm based on the Langevin diffusion which will later be used in a tempering strategy, since, to our knowledge, it has not been yet proposed in the literature in such settings. At this stage we note that it is always possible to achieve label switching between the $k!$ modes by the simple addition of a step in the algorithm which proposes a random permutation of the labels (a Metropolis-Hastings step with acceptance probability one). Our insistence on searching for an algorithm which can achieve symmetry without such a move type is that any concerns over convergence are not necessarily dealt with by such a strategy which simply alleviates the most obvious symptom.

2.1 The Gibbs sampler

The Gibbs sampler is the most commonly used approach in mixture estimation following Diebolt and Robert (1990,1994), and we will not dwell on it longer than necessary; see Chib (1995) or Richardson and Green (1997) for additional details. The basic feature of the Gibbs sampler in this setup is the augmentation of the parameters of (1) by artificial allocation variables z_i, each one being associated with one of the x_i's in order to "de-mix" the observed sample. This allows simulation of the parameters of each component conditionally on the allocations, taking into account only the observations which have been allocated to this component. Note that one of the main defects of the Gibbs sampler in this setting is the ultimate attraction of the local modes, that is the almost impossible simultaneous re-allocation of a group of observations to a different component. In experiments, this behaviour was observed in both the unconstrained and the constrained cases; the result in the constrained case being that the constrained parameter is biased in the event that sufficient observations are wrongly allocated. For instance, in a normal unidimensional mixture, if the constraint $\tau_1 < \tau_2$ bears on the variances and if $x_1,\ldots,x_{10}$ are allocated to component "1", and $x_{11},\ldots,x_{25}$ are allocated to component "2", while the opposite should hold (that is, the common variance of $x_1,\ldots,x_{10}$ is larger than the common variance of $x_{11},\ldots,x_{25}$), the constraint will lead to either an upward bias for τ_2 or a downward bias for τ_1 because an allocation switch will *never* occur.

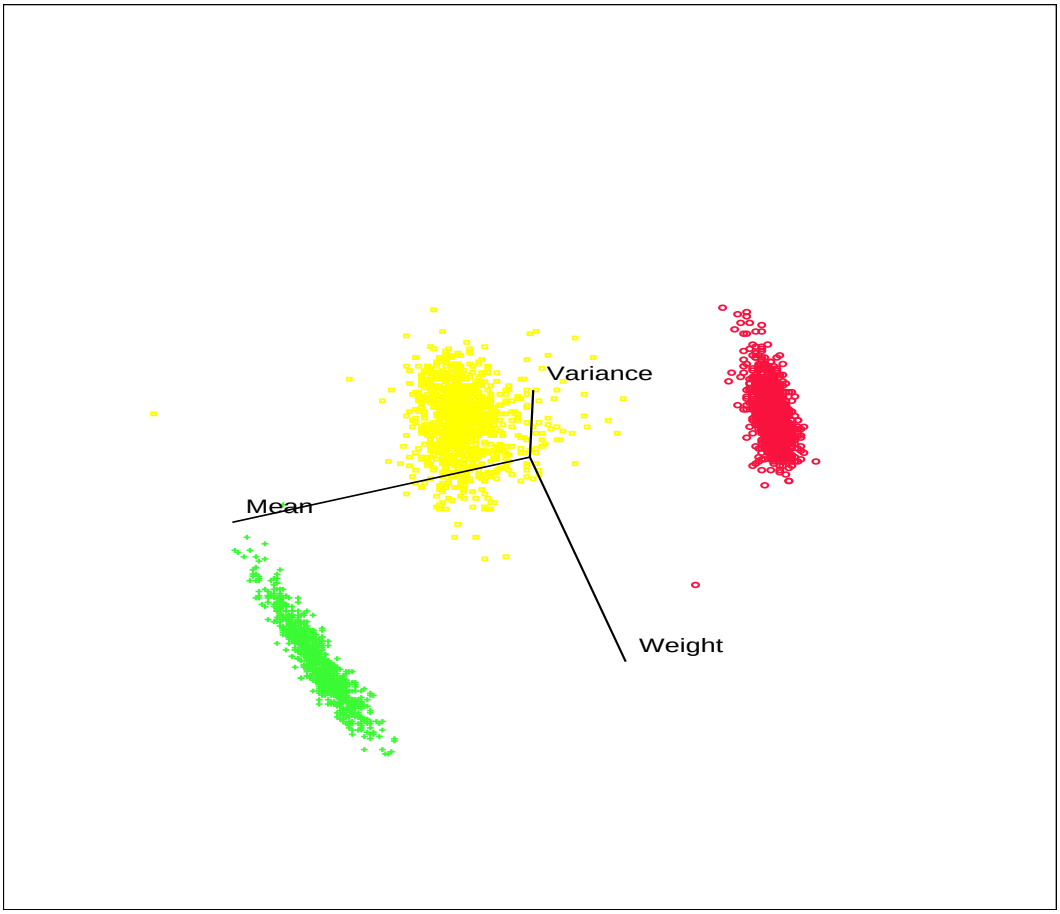
Figures 2 and 3 illustrate the lack of "mixing" of the chains for the exponential and the normal cases, respectively. While the overall Markov chain Monte Carlo sample exhibits

Figure 2: Representation in the (p, λ) plane of the Gibbs sample associated with a sample of $1,000$ simulated points from a 3 component exponential mixture. *(Circles correspond to component* 1*, triangles to component* 2 *and crosses to component* 3*.)* The histograms on top and right of the plot are the estimates of the marginal distributions of p_i *(top)* and λ_i *(right)*.



three zones of importance, the component chains $(p_j^{(t)}, \zeta_j^{(t)})$ never switch in the normal case and exhibit only a moderate degree of switching between two components in the exponential case. (Figure 1 provides an opposite illustration in exponential settings, namely that complete switching may occur.) In many cases, the Gibbs sampler is unable to move the Markov chain to another mode of equal importance because of its inability to step over valleys of low probability. In addition, there is no way one can judge whether the neighbourhood of

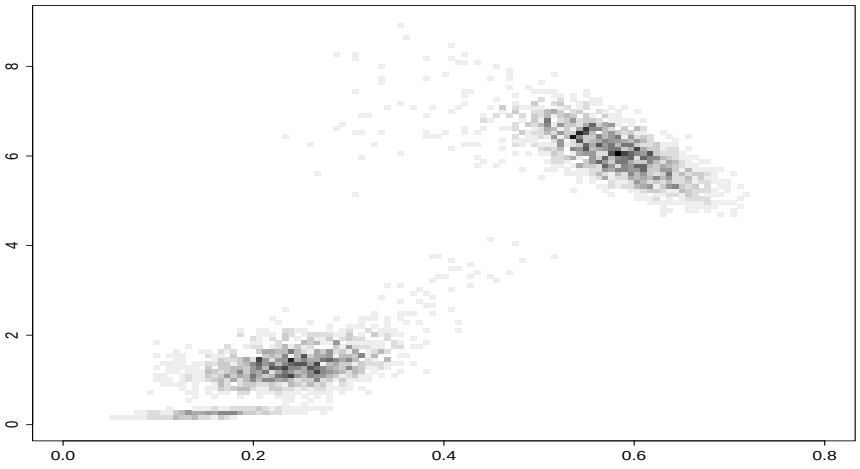
Figure 3: Representation of the Gibbs sample associated with a sample of 500 simulated points from a 3 component normal mixture. *(Same symbols as in Figure 2.)*



a specific mode has been sufficiently explored, even though the path of the Markov chain can be exploited to provide a rough approximation of the marginal posterior distribution of the component parameters (p_i, ζ_i) $(i = 1, \ldots, k)$. This approximation was obtained in the exponential case, Figure 4, by computing the posterior distribution at each iteration of the Gibbs chain and by averaging, for each square of a 50×50 grid in the (p, λ) space, the values of the posterior distributions within the square. Comparing this figure with the plot of the Gibbs sample in Figure 2, while the sample plot covers most of the high regions of the estimated marginal, the ridges between the modal regions are missing. (Note that white

regions in Figure 4 are indicative either of a low posterior probability or of no visit to the corresponding square by the Markov chain.)

Figure 4: Approximation of the (p, λ) marginal posterior distribution for the exponential mixture sample of Figure 2, obtained by averaging the values of the joint posterior at the points of a Gibbs sample occurring in each square of a grid.



2.2 Random walks

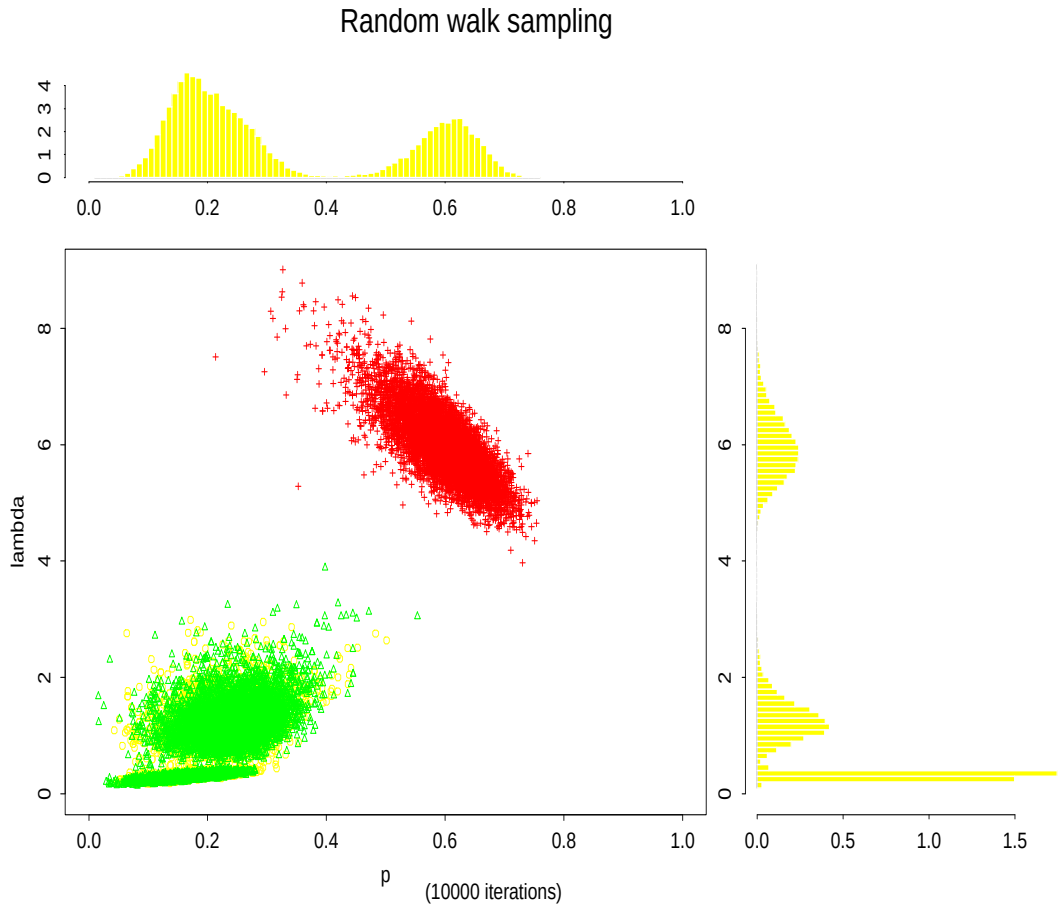
A standard alternative to the Gibbs sampler when the distribution of interest is available up to a constant is the random walk Metropolis–Hastings algorithm. Let $\xi^{(t)}$ denote the whole parameter vector at iteration t, then the next value is proposed as

$$\tilde{\xi} = \xi^{(t)} + \omega\epsilon_t, \qquad \epsilon_t \sim \varphi\,, \tag{2}$$

where φ is usually chosen as a multivariate normal or Cauchy density, and ω is calibrated to achieve a given acceptance rate in the acceptance probability. Following Gelman, Gilks and Roberts (1996), a good range for the acceptance rate of random walk Metropolis–Hastings algorithms is 0.1–0.3.

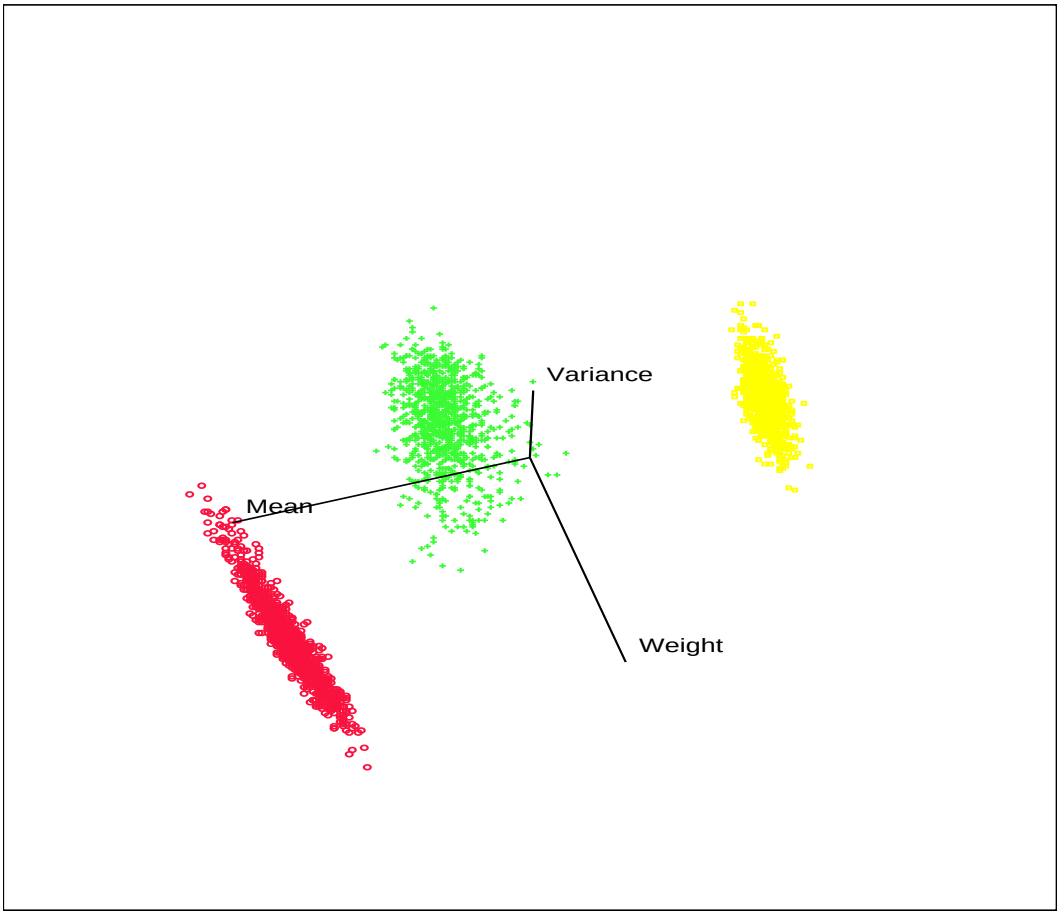
Figure 5 plots the sample obtained using the same simulated data sample as used for Figure 2. The similarity between the figures shows that they both recover the same features of the posterior distribution but fail to achieve symmetry of the marginals. Although we deliberately chose low acceptance rates for the Metropolis–Hastings algorithm, the final acceptance rate being lower than 0.1, in order to attempt to move further into the tails of the stationary distribution, the other modes are still too far away for the proposal to reach.

Figure 5: Representation in the (p, λ) plane of the Markov chain Monte Carlo sample associated with a sample of $1,000$ simulated points from a 3 component exponential mixture and a random walk Metropolis–Hastings algorithm.



A reparameterisation of the exponential parameters as $(\log(p_i/1-p_i), \log(\lambda_i))$ was used so that the random walk (2) would not be hindered by constraints on the support. However, the geometry of the posterior distribution remains too different from a unimodal distribution to allow sufficient acceptance of wide moves. In other words, an isotonic random walk, even with a large value of ω, has too small a chance of encountering one of the remaining $(k! - 1)$ modes. The same comment applies in the normal case, as shown in Figure 6 where no mixing occurs between the three components.

Figure 6: Representation of the Markov chain Monte Carlo sample associated with a sample of 500 simulated points from a 3 component normal mixture and a random walk Metropolis–Hastings algorithm.



2.3 Langevin diffusion

One difficulty with the random walk proposals is that they are "too random" to be able to hit the other modes with sufficient regularity. A more adaptive alternative is the class of Langevin diffusion Metropolis–Hastings algorithms proposed by Roberts and Tweedie (1995) which are based on a random walk drift proposal

$$x_{t+1} = x_t + \frac{\sigma^2}{2}\nabla \log \pi(x_t) + \sigma\varepsilon_t, \quad \varepsilon_t \sim \varphi, \tag{3}$$

where π is the unnormalised density of interest, φ is an arbitrary density, and σ is a simulation scale parameter, which can be calibrated against the acceptance rate of the corresponding Metropolis–Hastings algorithm; see Robert and Casella (1999) for details. This modification introduces a drift term, $\sigma^2 \nabla \log \pi(x_t)/2$, which pulls the Markov chain towards the modal zones of the support of π. As detailed in Appendix 1, the Langevin algorithm can easily be implemented in our setting. We adopt an approximate approach for the computation of the gradient $\nabla \log \pi$, replacing the exact derivative $\partial \log \pi(x)/\partial x_j$ with the finite difference approximation

$$\frac{\log \pi(x + \delta e_i) - \log \pi(x - \delta e_i)}{2\delta},$$

where δ is a small value and e_i the appropriate basis vector. While an exact expression could be derived, we prefer the approximation for several reasons:

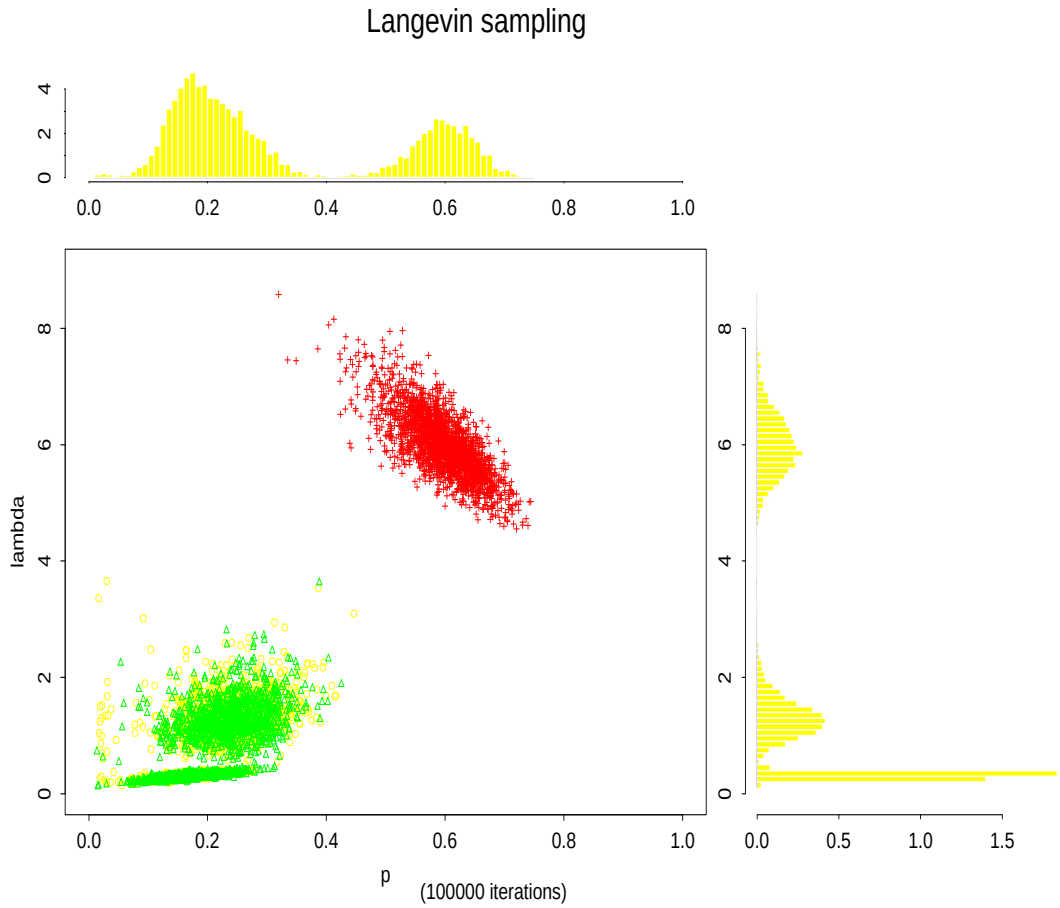
- numerical derivatives can be used in greater generality than analytical derivatives;
- they are usually faster to compute than analytical versions, especially in latent variables settings such as mixture models where the log-derivatives involve many complex terms;
- the approximation effect is taken into account by the proposal distribution in the acceptance probability and is thus corrected: the posterior distribution is still the stationary distribution of the corresponding Markov chain;
- the method opens new horizons with the possibility of semi-automated Metropolis–Hastings algorithms in the sense that the Langevin module/procedure can be implemented independently of the density π, the scale factor σ in (3) being scaled against the acceptance rate in a warmup stage.

Figures 7 and 8 illustrate the behaviour of the algorithm in the exponential and normal cases respectively. For the exponential mixture, the overlap of the three component chains is much more satisfactory than for the Gibbs chain, although the chain associated with component 1 still fails to visit one of the three zones of importance after $250,000$ iterations. The advantage of using the Langevin algorithm is not so clear in the normal case since the three groups observed on the graphs of Figure 8 still correspond to one component each. While the density φ in (3) can be arbitrary, we found in practice better mixing properties for a Cauchy density than for a normal density, a fact in agreement with the literature (see, e.g., Stramer and Tweedie, 1997).

3 Simulated tempering

Since standard Markov chain Monte Carlo schemes are not able to jump between the equivalent modes of the target distribution, we need an alternative strategy. Rather than trying to refine the Langevin algorithms or transform the problem, we introduce a *tempering* scheme, following Neal (1996). This approach is appealing in that it encourages moves between the different modes in full generality, it is to some extent independent of the underlying model and stationary distribution, and thus could be used for a wide range of models, including latent variable models (Robert, 1998). We will concentrate on one type of tempering strategy, using varying powers of the posterior distribution, but we note that alternatives such

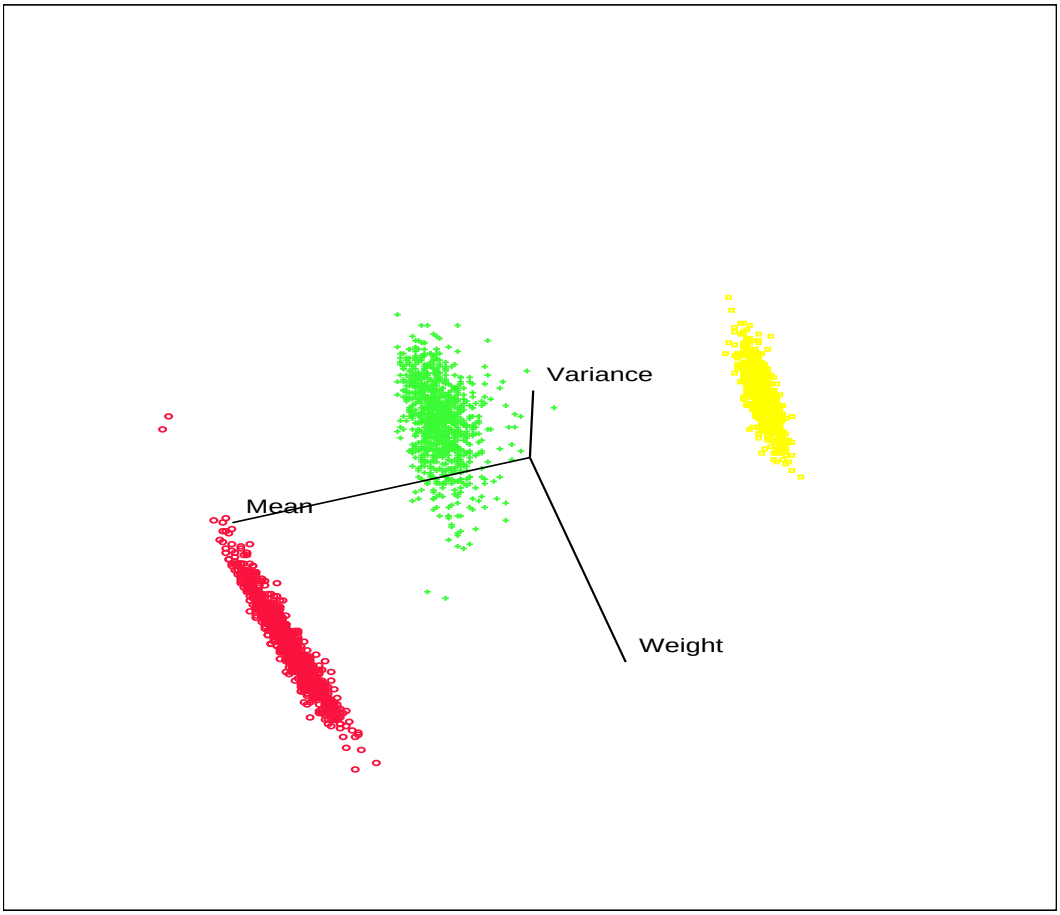
Figure 7: Representation in the (p, λ) plane of the Markov chain Monte Carlo sample associated with the distribution π, for the same exponential sample as in Figure 5 and a Langevin Metropolis–Hastings algorithm.



as convolutions with $\mathcal{N}(0, \varrho^2)$ distributions could also be used, with ϱ playing the role of the temperature.

Simulated tempering originates from the same observation as simulated annealing, namely that the maximisers and minimisers of a function E are identical to those of a monotone transform of the function, for instance powers of E, E^T. While simulated annealing aims to emphasize minima and maxima of the function, tempering tries the converse, flattening peaks and filling valleys, so that a random walk can move more freely on the surface. However, rather than simply simulating from π^T with $T < 1$, which would give a more complete

Figure 8: Representation of the Markov chain Monte Carlo sample associated with the distribution π, for the same normal sample as in Figure 3 and a Langevin Metropolis–Hastings algorithm.



picture of the surface of π but could not be used directly for estimation from π (except perhaps by SIR methods), Geyer and Thomson (1995) and Neal (1996) insert tempering steps within a Markov chain Monte Carlo sampler to focus on the distribution of interest, as explained below.

3.1 Up-and-down power scheme

Neal (1996) shows that a valid way to insert simulations from π^T within an Markov chain Monte Carlo sampler for π is to use an "up-and-down" scheme: Starting at the configuration $x^{(t)}$, the next configuration $x^{(t+1)}$ is proposed using intermediate Markov chain Monte Carlo steps which maintain detailed balance with respect to the flatter π^T. Since T may have to be very small for the associated distribution to be sufficiently flat for good mixing, the huge difference between π and π^T can lead to high rejection rates. Consequently the number of intermediate steps is increased by using a sequence of closely spaced $T_\ell < \ldots < T_1 < 1$, and corresponding sequence of simulations:

1. `Generate` y_1 `from` $x^{(t)}$ `by using an MCMC step from` π^{T_1}

2. `Generate` y_2 `from` y_1 `by using an MCMC step from` π^{T_2}

$\vdots$

ℓ. `Generate` y_ℓ `from` $y_{\ell-1}$ `by using an MCMC step from` π^{T_ℓ}

$\ell+1$. `Generate` $y_{\ell+1}$ `from` y_ℓ `by using an MCMC step from` $\pi^{T_{\ell-1}}$

$\ell+2$. `Generate` $y_{\ell+2}$ `from` $y_{\ell+1}$ `by using an MCMC step from` $\pi^{T_{\ell-2}}$

$\vdots$

$2\ell-1$. `Generate` $y_{2\ell-1}$ `from` $y_{2\ell-2}$ `by using an MCMC step from` π^{T_1} .

Since this scheme constitutes the proposal mechanism for the overall Markov chain Monte Carlo algorithm, a final acceptance step is required. This final step compensates for the fact that the y_i's are not distributed from the "right" distributions. Neal (1996) shows that the acceptance step

`Accept` $x^{(t+1)} = y_{2\ell}$ `with probability`

$$\min\left\{1, \frac{\pi^{T_1}(x^{(t)})}{\pi(x^{(t)})}\cdots\frac{\pi^{T_\ell}(y_{\ell-1})}{\pi^{T_{\ell-1}}(y_{\ell-1})}\frac{\pi^{T_{\ell-1}}(y_\ell)}{\pi^{T_\ell}(y_\ell)}\cdots\frac{\pi(y_{2\ell-1})}{\pi^{T_1}(y_{2\ell-1})}\right\} \tag{4}$$

maintains detailed balance with respect to the target distribution π. Note that this acceptance probability only involves ratios of each of the distributions π^{T_i} and is thus independent of the normalising constants; it is possible to use schemes other than sequences of powers without this result being affected. Equation (4) can be rewritten quite naturally as

$$\min\left\{1, \left(\frac{\pi(y_{2\ell-1})}{\pi(x^{(t)})}\right)^{1-T_1}\left(\frac{\pi(y_{2\ell-2})}{\pi(y_1)}\right)^{T_1-T_2}\cdots\left(\frac{\pi(y_\ell)}{\pi(y_{\ell-1})}\right)^{T_{\ell-1}-T_\ell}\right\}$$

correcting for the wrong distributions of the conditional variables y_i. When the differences $T_i - T_{i+1}$ are small, the overall acceptance rate should thus be high, as should the local

acceptance rates given the small differences between the π^{T_i}'s. However, the wastage of simulations increases with T.

Several modifications can be proposed on this scheme: firstly, as in Neal (1996), additional simulations from π^{T_ℓ} can be generated between stage ℓ and stage ℓ+1 since this facilitates moves between modes while preserving the balance condition (and the acceptance probability (4)). Secondly, as the up-and-down scheme is associated with the global exploration of the posterior support, an arbitrary number of simulations from the target distribution π, using the Langevin proposal of §2.3, can be inserted between two up-and-down proposals to improve the local exploration of the current mode.

3.2 Implementation using Langevin algorithms

Since π is available up to a constant, π^T is also known up to a constant and various Metropolis-Hastings proposals could be used. In particular, the Langevin algorithm of Section 2.3 adapts quite naturally using π^T proposal

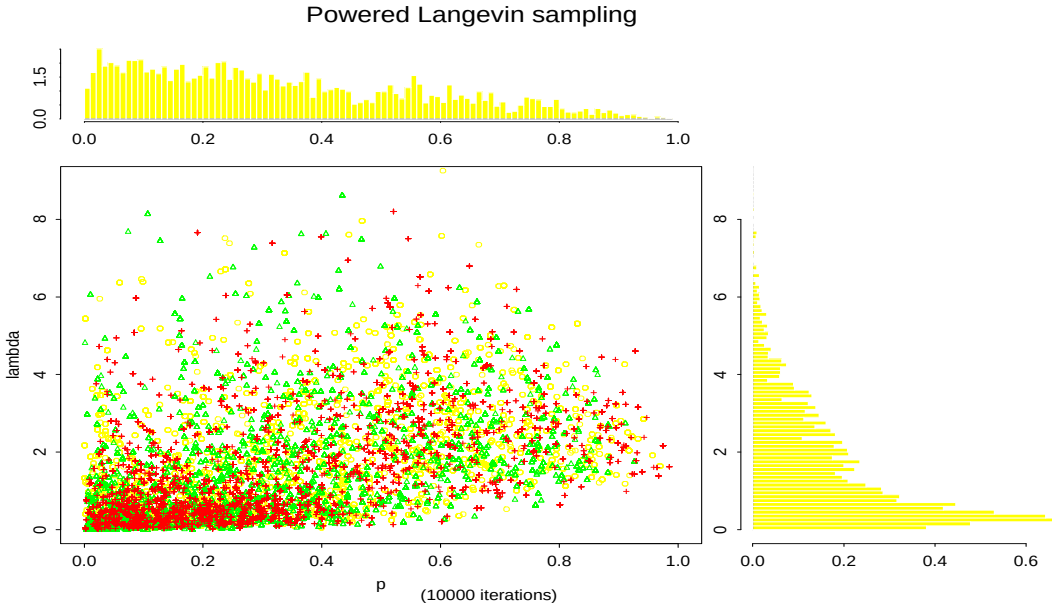
$$x^{(t+1)} = x^{(t)} + \frac{\sigma^2}{2} T \nabla \log \pi(x^{(t)}) + \sigma \varepsilon_t. \tag{5}$$

The proposals in the tempering stages each involve a scale factor σ_i $(1 \leq i \leq \ell)$ which is first calibrated by trial-and-error to obtain an acceptance rate between 0.15 and 0.35, following Roberts and Tweedie (1995). Thus although the importance of the gradient term decreases as T decreases, the move (5) is not necessarily more restricted than with $T = 1$.

Given the formal freedom in the choice of the "flatter" distributions at level i in the up-and-down scheme, we work with a tempering schedule where at level i rather than raise the whole posterior to the power T_i, we power up just the likelihood while the prior contribution is left unchanged. This choice was motivated by the uncertainty about the integrability of the pseudo-distribution π^{T_i} for small values of T_i, and led to better results in practice.

The number of levels and the rate of decrease of the power must be chosen with care to achieve a sufficient rate of label switching at the lowest power. We observed sensitivity of the method to the choice of (a) the lowest power T_ℓ, (b) the number of levels ℓ, and (c) the rate of increase of the power, $T_i - T_{i-1}$. First, the lowest power must be chosen small enough to ensure that moves between modes are frequent at this power; this can be assessed by a preliminary check. The flatter the target distribution π^T, the easier the moves between the modes, as can be observed in practice in Figure 9 which shows the simulated sample from the exponential mixture at power $T = 10/n$ and with an average acceptance rate of 0.2. Figure 10 illustrates the same mixing phenomenon for the normal model with a power of 0.01; the marginal histograms of the parameters (p, θ, τ) are approximately symmetric. While a low enough power is bound to achieve a good mixing rate, the generation from the Langevin proposal associated with π^{T_ℓ} in the up-and-down scheme does not necessarily enjoy the same properties when π^{T_ℓ} and $\pi^{T_{\ell-1}}$ are too different. Monitoring y_ℓ along the tempering iterations shows in particular that the mixing properties are not the same as for a Langevin chain with stationary distribution π^{T_ℓ}. The reason for this apparent discrepancy

Figure 9: Representation in the (p, λ) plane of the Markov chain Monte Carlo sample associated with the distribution proportional to $\ell^{10/n}\pi$, for the same sample as in Figure 5 ($n = 1000$) and a Langevin Metropolis–Hastings algorithm.



is that the conditioning value $y_{\ell-1}$ is not distributed from π^{T_ℓ} but from $\pi^{T_{\ell-1}}$, which is more concentrated around the modes of π, thus preventing moves further away from these modes. We found in practice that geometric decrease rates such as $T_i = 1 - (i/\ell)^{0.1}(1 - T_\ell)$ perform better than linear decrease rates, while increasing the number of levels led to faster mixing at level ℓ. Figure 11 illustrates the switching behavior of the chain of the θ_j's, in the normal case, with long stays around each mode, in contrast in the much faster mixing of the chain associated with $\ell^{.01}\pi$, shown in Figure 10.

Figures 12 and 13 give the results of an implementation with $\ell = 50$ and $\ell = 45$ levels in the exponential and normal cases, respectively. Both are quite satisfactory in terms of mixing, since the marginal samples for the three components are comparable. The stepwise acceptance rates are all calibrated between 0.15 and 0.35, and remain in the same range along the up-and-down iterations. Significantly the marginals in these figures are quite similar in Figure 12 and Figure 7 which validates *(a posteriori)* the output of the original Markov chain Monte Carlo samplers.

The obvious drawback of the tempering scheme is the considerable computational burden it entails; using 50 levels multiplies the total number of simulations by 100, or in other words, we only use 1% of the simulations. Moreover, not all of these computationally expensive proposals will be accepted; the low acceptance rate motivates the use of additional steps of

Figure 10: Histograms of the componentwise Markov chain Monte Carlo samples associated with the distribution proportional to $\ell^{.01}\pi$, for the same sample as in Figure 6 ($n = 500$) and a Langevin Metropolis–Hastings algorithm, in the (p, θ) *(left)*, (τ, θ) *(middle)* and (p, τ) planes *(right)* $(50,000$ iterations).

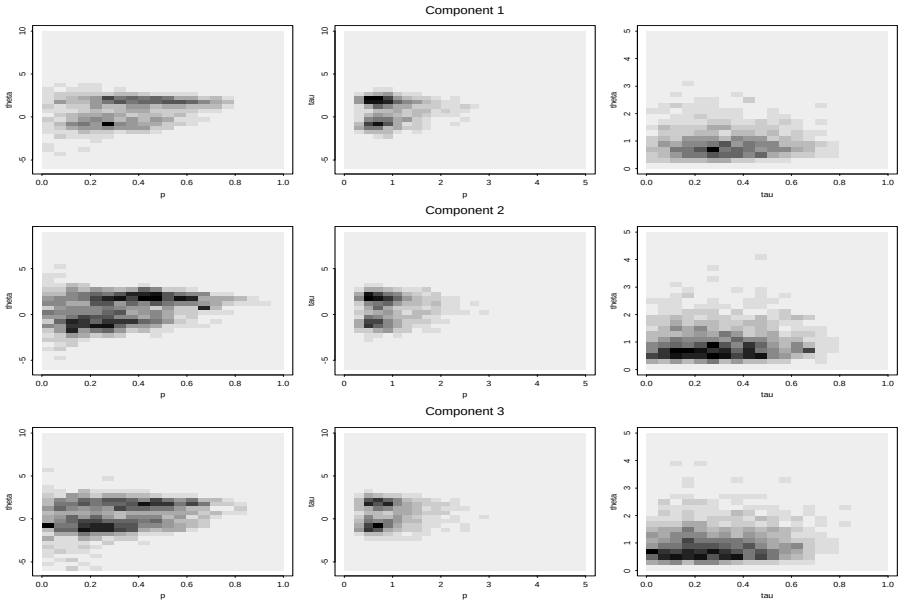
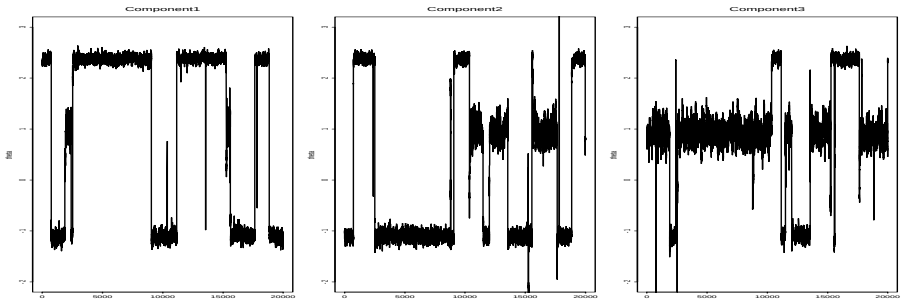
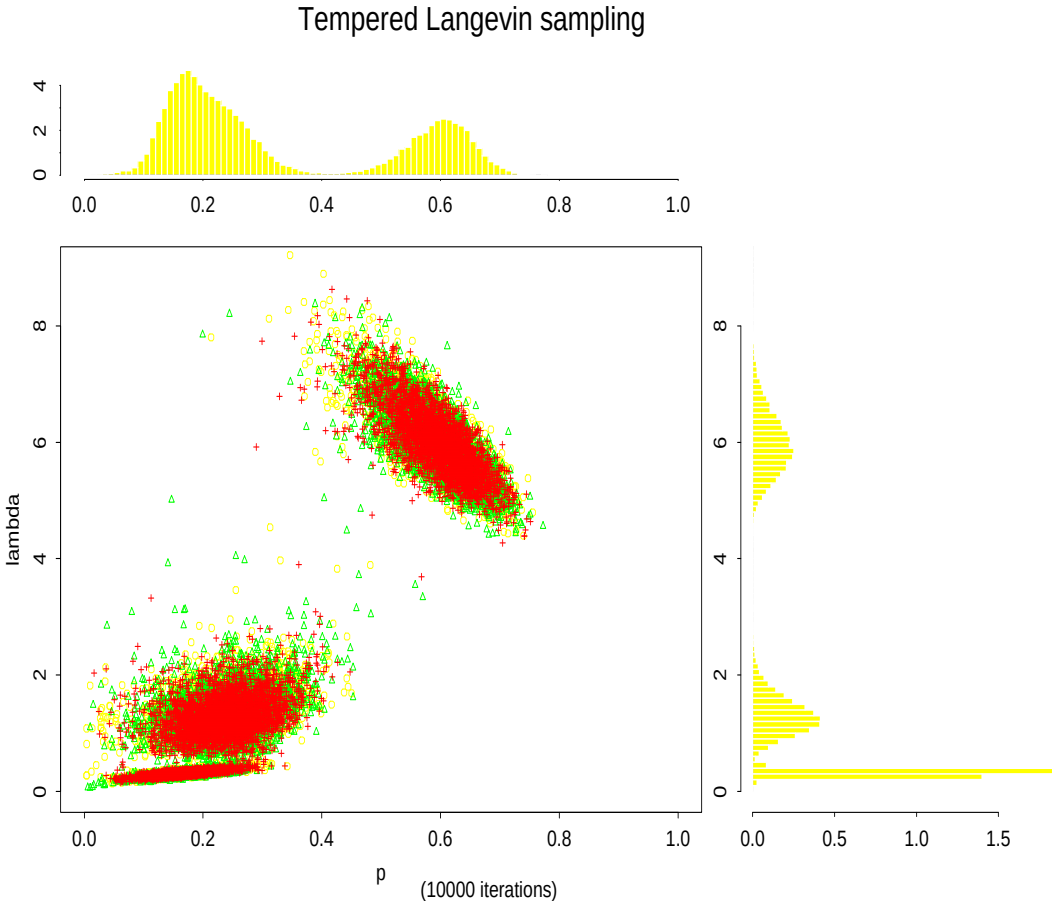


Figure 11: Rawplots of the chains θ_j $(j = 1, 2, 3)$ for the tempered Langevin Metropolis–Hastings algorithm $(\ell = 45, T_\ell = .005)$ and the same sample as in Figure 6 $(n = 500)$.



the Langevin proposal of §2.3 between tempering steps. It is worth noting that it should be possible to make use of the rejected simulations by recycling them through a SIR scheme

Figure 12: Representation in the (p, λ) plane of the Markov chain Monte Carlo sample associated with the distribution π, for the same exponential sample as in Figure 2 and a tempered Langevin Metropolis–Hastings algorithm ($\ell = 50$ levels).

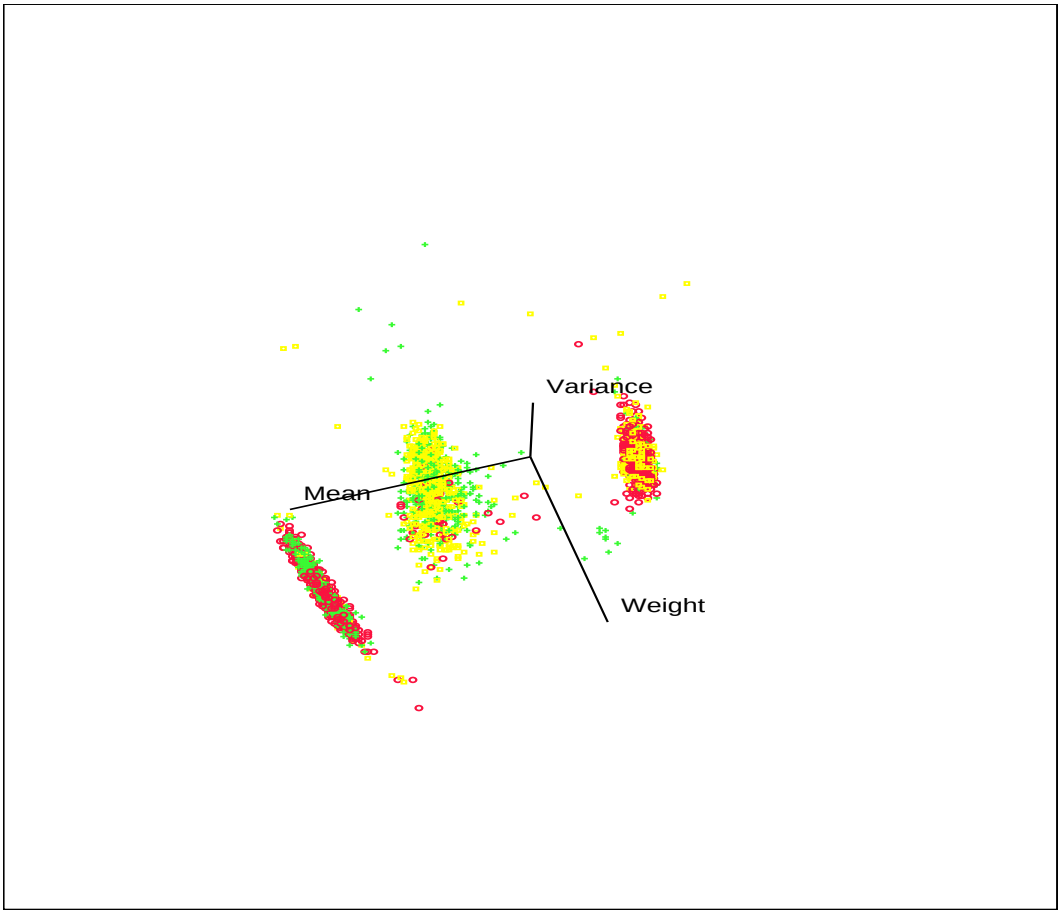


(Rubin, 1987), that is by resampling from the simulated values by weighting them with weights proportional to the true posterior, which is computed at each simulation.

4 Inference using MCMC samples

When label switching of the components is a prerequisite of any Markov chain Monte Carlo convergence, there is automatically a complication created for inference from the resulting samples. The mixing of labels means that it is not possible to form ergodic averages over

Figure 13: Representation of the Markov chain Monte Carlo sample associated with the distribution π, for the same normal sample as in Figure 6 ($n = 500$) and a tempered Langevin Metropolis–Hastings algorithm ($\ell = 45$ levels, minimum power 0.005).



labels since, if the sampler is behaving properly, all resulting parameters estimates will be close (and ideally, they should be equal). In this section, we consider some of the existing alternatives which attempt to deal with this problem before developing in Section 5 a new approach based on loss functions.

Table 1: Estimates of the parameters of a three component normal mixture, obtained for the same simulated sample as in Figure 3 using a Gibbs sampler and re-ordering according to one of three constraints, $p: p_1 < p_2 < p_3$, $\theta: \theta_1 < \theta_2 < \theta_3$, or $\tau: \tau_1 < \tau_2 < \tau_3$.

order	p_1	p_2	p_3	θ_1	θ_2	θ_3	τ_1	τ_2	τ_3
p	0.231	0.311	0.458	0.321	−0.55	2.28	0.41	0.471	0.303
θ	0.297	0.246	0.457	−1.1	0.83	2.33	0.357	0.543	0.284
τ	0.375	0.331	0.294	1.59	0.083	0.379	0.266	0.34	0.579
true	0.22	0.43	0.35	1.1	2.4	−0.95	0.3	0.2	0.5

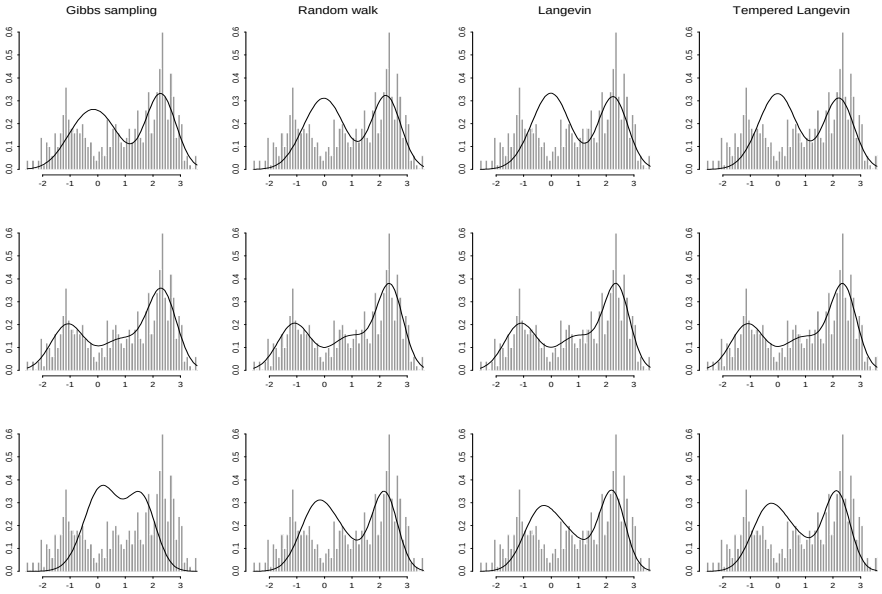
4.1 Ordering constraints

By imposing some form of order constraint on the parameters, for example in terms of the means of the components, it is possible to create a k group structure over which averages can be formed. As noted in Stephens (1997, Proposition 3.1), the ordering constraint can be imposed *ex post*, that is after the simulations have been completed. This is interesting since it offers a direct and easy comparison of the biasing effects of different possible ordering constraints based on the same Markov chain Monte Carlo sample. Applying the different order constraints to the samples obtained via the different samplers of Section 2, the results are disturbing. The estimates for the normal model are shown in Table 1, they can be seen to be markedly different for the three orders. Since it is well known that different sets of parameters can lead to the same estimation of the mixture density, as observed in Robert and Mengersen (1998) for instance, a more realistic comparison is based on the estimated densities; see Figure 14. Quite clearly the choice of the identifying constraint has a strong influence on the resulting estimate and, *ergo*, creates a bias. In this case, the best fit is provided by the constraint on the θ_i's, which is natural given that the true θ_i's are quite different (see Table 1). However, this natural ordering is not obvious a priori and requires a post-data processing which is difficult to define formally. Another interesting feature of this comparison is that the different sampling schemes give very similar estimates for the three constraints, barring the Gibbs estimate under the τ constraint. This indicates that the three schemes have "reached" stationarity for the number of iterations considered in this experiment. We note that similar differences between the different estimates do not appear in the simulated exponential mixture where the orderings by true parameter values, either by p or by λ, are the same.

4.2 Classifying

As already discussed, ordering constraints fail to isolate one of the $k!$ modes of the posterior distribution because these constraints are usually not in agreement with the geometry of the posterior surface. Ideally, since the Markov chain Monte Carlo sample should exhibit a good amount of symmetry between the $k!$ modes. If this sample could be reordered by

Figure 14: Estimations of the density of the sample (represented by the histogram), for four different Markov chain Monte Carlo samplers and the three possible orderings, $p : p_1 < p_2 < p_3$, $\theta : \theta_1 < \theta_2 < \theta_3$, and $\tau : \tau_1 < \tau_2 < \tau_3$ (50,000 simulations).



suitable permutations so that all points in the sample come from the same modal region of the posterior distribution, the means of each component could be used as estimators. One way to come close to this goal is to use clustering-like tools (see the discussion of Richardson and Green 1997). Following Stephens (1997), a simple on-line procedure has been proposed in Celeux (1998): one of the modal regions is selected using the first iterations of the Markov chain Monte Carlo sampler, the following points in the sampler being permuted according to a $k!$-means-type algorithm, selecting at each iteration which of the $k!$ permutations is closest to the current cluster means. Note that this approach avoids the storage of the complete Markov chain Monte Carlo sample.

More precisely, the clustering procedure works as follows: Let $\xi^1, \xi^2, \ldots$ be the sequence of d-dimensional Markov chain Monte Carlo vector samples (for example, for a 3 component exponential mixture, $d = 3 \times 2$). The procedure is initiated using the first m vectors $\xi^1, \ldots, \xi^m$, where m is typically 100 or so (the choice is not highly sensitive, it must be large enough to ensure that the initial estimates are a reasonable crude approximation of the posterior means, but not so large that a label switch has already occurred, which would

worsen the approximation). Reference centres for $(i = 1, \ldots, d)$ are defined

$$\bar{\xi}_i = 1/m \sum_{j=1}^{m} \xi_i^j \,,$$

together with componentwise variances

$$s_i = \frac{1}{m} \sum_{j=1}^{m} (\xi_i^j - \bar{\xi}_i)^2 \,.$$

We set $s_i^{[0]} = s_i$, $i = 1, \ldots, d$. If we denote $\bar{\xi}_1^{[0]} = \bar{\xi}$, the $(k! - 1)$ other centres $\bar{\xi}_2^{[0]}, \ldots, \bar{\xi}_{k!}^{[0]}$ can be deduced from $\bar{\xi}_1^{[0]}$ by permuting the labeling of the mixture components. After this initialisation stage, the rth iteration of the clustering procedure runs as follows:

1. Allocate ξ^{m+r} to the cluster $j^\star$ which minimises the normalized squared distance $(j = 1, \ldots, k!)$

$$||\xi^{m+r} - \bar{\xi}_j^{[r-1]}||^2 = \sum_{i=1}^{d} \frac{(\xi_i^{m+r} - \bar{\xi}_{ij}^{[r-1]})^2}{s_i^{[r-1]}}, \tag{6}$$

   where $\bar{\xi}_{ij}^{[r-1]}$ is the ith coordinate of $\bar{\xi}_j^{[r-1]}$.
   If $j^\star \neq 1$, permute the coordinates of ξ^{m+r} to get $j^\star = 1$.

2. Update the $k!$ centres and the d normalizing coefficients

   a. Compute
$$\bar{\xi}_1^{[r]} = \frac{m + r - 1}{m + r} \bar{\xi}_1^{[r-1]} + \frac{1}{m + r} \xi^{m+r} \,.$$

   b. Derive the $k! - 1$ other centres by permutation.

   c. Update the variances as $(i = 1, \ldots, d)$

$$\begin{aligned} s_i^{[r]} &= \frac{m + r - 1}{m + r} s_i^{[r-1]} + \frac{m + r - 1}{m + r} (\bar{\xi}_{i1}^{[r-1]} - \bar{\xi}_{i1}^{[r]})^2 \\ &\quad + \frac{1}{m + r} (\xi_i^{m+r} - \bar{\xi}_{i1}^{[r]})^2 . \end{aligned}$$

The mode of reference thus corresponds to $j = 1$ at each iteration. Note that the normalization of the distances (6) makes the procedure independent of location-scale transformations of the mixture parameters, even though the resulting estimates depend on the parameterisation of ξ. Tables 2 and 3 display the resulting estimates for the exponential and normal mixtures examples respectively, based on the Markov Chain Monte Carlo samples obtained in Sections 2 and 3. In these tables, the number of times the cluster labels change

Table 2: Exponential mixture estimates (arranged by scale) derived by the clustering procedure

	True values	Gibbs sampling	Random walk	Langevin	Tempered Langevin
Swaps		(96)	(36)	(32)	(41)
$\hat{p}_1$	0.15	0.164	0.162	0.162	0.166
$\hat{\lambda}_1$	0.25	0.317	0.314	0.320	0.316
$\hat{p}_2$	0.22	0.239	0.236	0.239	0.240
$\hat{\lambda}_2$	1.1	1.299	1.278	1.288	1.326
$\hat{p}_3$	0.63	0.597	0.602	0.599	0.594
$\hat{\lambda}_3$	5.4	5.993	5.948	5.976	6.010

Table 3: Normal mixture estimates (arranged by means) derived by the clustering procedure

	True values	Gibbs sampling	Random walk	Langevin	Tempered Langevin
Swaps		(206)	(26)	(20)	(174)
$\hat{p}_1$	0.35	0.299	0.303	0.300	0.300
$\hat{\theta}_1$	-0.95	-1.031	-1.057	-1.080	-1.057
$\hat{\tau}_1^2$	0.5	0.371	0.354	0.341	0.359
$\hat{p}_2$	0.22	0.247	0.269	0.277	0.278
$\hat{\theta}_2$	1.1	0.778	0.906	0.924	0.886
$\hat{\tau}_2^2$	0.3	0.530	0.507	0.526	0.573
$\hat{p}_3$	0.43	0.454	0.428	0.423	0.422
$\hat{\theta}_3$	2.4	2.319	2.368	2.386	2.376
$\hat{\tau}_3^2$	0.2	0.284	0.230	0.220	0.229

between two consecutive runs of the MCMC sampler is displayed at the top of each column between parentheses. This number gives an indication of the sampler's ability to move from one mode to another, although it may suffer from artifacts, as shown in the case of the Gibbs sampler, when the method reallocates the simulated vectors, even though Figure 3 shows that label switching never occurs for the corresponding sample. The variation between the estimates is wider on parameter λ_3 in Table 2, but this is to be expected, given the poor identifiability of exponential mixtures in the tails. As shown by Figure 15, this does not jeopardize the estimated density.

4.3 Annealing

Once the support of the posterior density has been adequately explored, the $k!$ modes are equivalent from an inferential point of view. One could thus take advantage of the exploration stage to implement an annealing inferential stage, starting from the highest posterior density value in the Markov chain Monte Carlo sample and running simulations from π^T with increasing powers T in order to locate one of the equivalent $k!$ MAP estimates. The implementation cost is negligible, given that the simulation from π^T has already been programmed for the tempering step. Strictly speaking we use a modification of simulated annealing since we do not increase the power T at each iteration; the Langevin algorithm requires a calibration stage to determine acceptable values of the scale σ for a fixed value of T, and so the power T is increased gradually, with rounds of calibration and exploration for each selected power. The approach is costly in terms of additional simulations since the Markov chain Monte Carlo sample is only used for the starting value. The results on our two examples are interesting in that the estimates we obtain are almost identical to those following a series of Gibbs or of tempered Langevin steps. This feature is only mildly surprising since both algorithms are already known to have good features in terms of exploration of local modes. We also observed variations depending on the increase rate of the power, with better performances when T was increased slowly.

5 Alternative Bayes estimators

Another possibility for estimating the parameters ξ is to consider designing a loss function $L(\xi, \hat{\xi})$ for which the lack of labeling is immaterial, and then to find the Bayes estimator $\hat{\xi}^*$ corresponding to this loss function:

$$\hat{\xi}^* = \arg\min_{\hat{\xi}} \mathbb{E}_{\xi|x} L(\xi, \hat{\xi}). \tag{7}$$

Depending on the loss function, the complexity of the problem may mean that the calculation of many estimators is not analytically feasible. Consequently, the choice of loss function has usually been restricted to those for which the form of the estimator is known, for example the squared loss function $L(\xi, \hat{\xi}) = ||\xi - \hat{\xi}||^2$ which leads to the posterior marginal means as estimates but relies on the parameter labeling. It has recently been demonstrated however that apparently intractable estimators in Bayesian image analysis can be approximated using Markov chain Monte Carlo (Frigessi and Rue, 1997; Rue, 1995). We consider a similar approach here, suggesting two loss functions, the first when inference for the parameters is the issue, and the second when the predictive mixture distribution is of more interest.

5.1 A loss function for the parameters

We return to Figures 2 and 3 to motivate our approach; here the mixture models are represented by points in the (p, λ) or (p, θ, τ) spaces. Removing any labeling of the points, as would be natural from a point process perspective, suggests that the components could

be viewed as points in a unlabeled point process (in these cases fixed dimension point processes). So in formulating an appropriate loss function what we are looking for is a way of measuring distance between two point configurations (of possibly different dimensions in a more general setting).

We suggest the following, loosely based on the Baddeley Δ metric (Baddeley, 1992) used as a loss function by Frigessi and Rue (1997) in a discrete pixel setting. The Δ metric measures the distance between two binary images and is based on the difference for every pixel between the shortest distance from that pixel to the closest foreground pixel in both of the two images. Working as we are in a continuous space, we will need to modify the definition while trying to maintain the same spirit. We begin with a collection of points $t_1, \ldots, t_n$ in the same space as the components of $\xi = (\xi_j)$, that is in the exponential case the t_i's are in (p, λ) space and in the normal case they are in (p, θ, τ^2) space. For each point t_i we define $d(t_i, \xi)$ to be the distance between t_i and the closest of the ξ_j's ($j = 1, \ldots, k$), where the distance d is Euclidean but applies to the transformed variables $\ln(p/(1-p))$, $\ln \lambda$, θ, $\ln \sqrt{\tau^2}$ (rather than to p, λ, θ, τ). We then define the loss function

$$L(\xi, \hat{\xi}) = \sum_{i=1}^{n} (d(t_i, \xi) - d(t_i, \hat{\xi}))^2, \tag{8}$$

which says that for each of the fixed points t_i, there is a contribution to the loss function if the distance from t_i to the nearest ξ_j is not the same as the distance from t_i to the nearest $\hat{\xi}_j$. Clearly the choice of the t_i's plays an important role here; ideally we want $L(\xi, \hat{\xi}) = 0$ only if $\xi = \hat{\xi}$, and for the loss function to respond appropriately to changes in the two point configurations. For the first point, the t_i's should be sufficiently numerous and located in such a way that it is possible to determine a particular ξ given the t_i's and the corresponding $d(t_i, \xi)$; in the exponential case, this means that each component of a particular ξ needs to be the closest one for at least three t_i's, while for the normal mixture model, which is one dimension higher, at least four t_i's are required for each component of that ξ. For the second point, the t_i's are best positioned in high posterior density regions of the ξ_j's space. To this end, we have chosen to divide our simulation effort so that the first half of the simulations are allocated to selecting a suitable set of t_i locations by randomly selecting one of the components in each realisation as t_i. The second half of the simulations are used to estimate the corresponding $\mathbb{E}_{\xi|x}[d(t_i, \xi)]$ using the two-step procedure proposed by Rue (1995). First note that we can write

$$\mathbb{E}_{\xi|x}\left[\sum_{i=1}^{n} (d(t_i, \xi) - d(t_i, \hat{\xi}))^2\right] = \sum_{i=1}^{n} \left(\mathbb{E}_{\xi|x}\left[d(t_i, \xi)^2\right] - 2d(t_i, \hat{\xi})\mathbb{E}_{\xi|x}\left[d(t_i, \xi)\right] + d(t_i, \hat{\xi})^2\right), \tag{9}$$

since the expectation is with respect to the posterior $\pi(\xi|x)$ and $\hat{\xi}$ is not involved. It is possible to estimate $\gamma_i = \mathbb{E}_{\xi|x}[d(t_i, \xi)]$ by simulating from the posterior (that is, using the

MCMC sample) and using the usual ergodic averaging (there is no issue of labeling here since we are always talking about the closest point for a fixed t_i). This leaves the minimisation task which, ignoring terms not involving $\hat{\xi}$, is

$$\hat{\xi}^* = \arg\min_{\hat{\xi}} h(\hat{\xi}) = \arg\min_{\hat{\xi}} \sum_{i=1}^{n} (-2\hat{\gamma}_i d(t_i, \hat{\xi}) + d(t_i, \hat{\xi})^2). \tag{10}$$

This task can be tackled using simulated annealing to search for the modes of the distribution proportional to $\exp(-h(\hat{\xi}))$. At this stage, any proposed $\hat{\xi}$ without the requisite number of t_i points in its neighbourhood can be flagged and, if necessary, further t_i can be added and associated γ_i can be estimated from the existing simulations. Tables 4 and 5 gives the resulting estimates for the exponential and normal mixtures examples respectively.

5.2 A loss function for the predictive distribution

When the object of inference is the predictive distribution, it may be more sensible to devise a more global loss function which measures distributional discrepancies in some way. (This was the approach adopted in Mengersen and Robert (1996) when testing the number of components in a normal mixture.) One such possibility is the integrated squared difference

$$L(\xi, \hat{\xi}) = \int_{\mathcal{R}} (f_\xi(y) - f_{\hat{\xi}}(y))^2 dy\,, \tag{11}$$

where f_ξ denotes the density of the mixture (1). This form of loss function again lends itself to the two-step "estimation-then-maximisation" algorithm in that we can decompose the posterior expected loss as follows:

$$\mathbb{E}_{\xi|x}\left[\int_{\mathcal{R}} (f_\xi(y) - f_{\hat{\xi}}(y))^2 dy\right] =$$
$$\int_{\mathcal{R}} \left(\mathbb{E}_{\xi|x}\left[f_\xi(y)^2\right] - 2 f_{\hat{\xi}}(y)\mathbb{E}_{\xi|x}\left[f_\xi(y)\right] + f_{\hat{\xi}}(y)^2\right) dy,$$

assuming that the order of integration may be interchanged. For all the $k!$ possible labelings of the mixture distribution, the final part of this integral may be evaluated analytically for the normal and exponential cases

$$\int_{\mathcal{R}} f_{\hat{\xi}}(y)^2 dy = \begin{cases} \displaystyle\sum_{i,j=1}^{k} \frac{\hat{p}_i \hat{p}_j}{\sqrt{2\pi(\hat{\tau}_i^2 + \hat{\tau}_j^2)}} \exp\left\{-\frac{(\hat{\theta}_i - \hat{\theta}_j)^2}{2(\hat{\tau}_i^2 + \hat{\tau}_j^2)}\right\} & \text{for normals,} \\ \displaystyle\sum_{i,j=1}^{k} \frac{\hat{p}_i \hat{p}_j}{\hat{\lambda}_i + \hat{\lambda}_j} & \text{for exponentials.} \end{cases}$$

For the remaining relevant term,

$$\int_{\mathcal{R}} -2 f_{\hat{\xi}}(y)\mathbb{E}_{\xi|x}\left[f_\xi(y)\right] dy\,,$$

Table 4: The exponential mixture estimates (arranged by means) under the two loss functions (8) and (11) for the different simulation schemes.

		True values	Gibbs sampling	Random walk	Langevin	Tempered Langevin
Local	$\hat{p}_1$	0.15	0.169	0.170	0.154	0.163
loss	$\hat{\lambda}_1$	0.25	0.260	0.257	0.358	0.239
function	$\hat{p}_2$	0.22	0.224	0.239	0.251	0.246
	$\hat{\lambda}_2$	1.1	1.294	1.187	1.199	1.206
	$\hat{p}_3$	0.63	0.607	0.592	0.596	0.591
	$\hat{\lambda}_3$	5.4	6.814	5.128	5.233	5.237
Global	$\hat{p}_1$	0.15	0.188	0.177	0.156	0.145
loss	$\hat{\lambda}_1$	0.25	0.352	0.335	0.289	0.322
function	$\hat{p}_2$	0.22	0.146	0.225	0.305	0.242
	$\hat{\lambda}_2$	1.1	1.220	1.324	1.688	1.069
	$\hat{p}_3$	0.63	0.666	0.598	0.538	0.613
	$\hat{\lambda}_3$	5.4	5.457	5.878	6.584	5.942

we propose first estimating $\psi = \mathbb{E}_{\xi|x}[f_\xi(y)]$ on a grid of y_i values using simulations from the posterior distribution. We then use numerical integration on the same grid of y_i values to estimate $\int_{\mathcal{R}} -2 f_{\hat{\xi}}(y)\hat{\psi} dy$. Again we are left with a complicated minimisation problem for $\hat{\xi}$ which can be tackled using simulated annealing. Tables 4 and 5 give the resulting estimates for the exponential and normal mixtures examples respectively. (We refer to this loss function as *global*, and to that based on the parameters as *local*.)

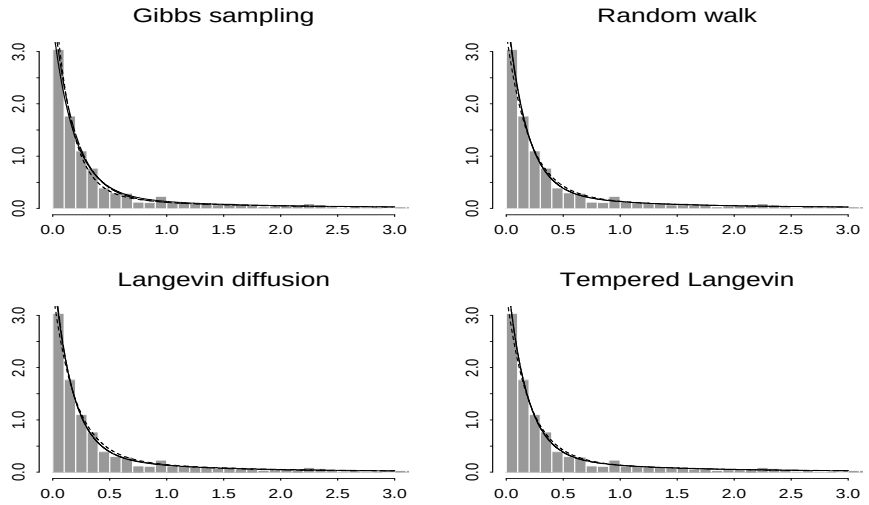
6 Conclusions

Figure 15 provides a comparison of the estimated densities corresponding to Tables 2 and 4, while Figure 16 similarly analyses Tables 3 and 5. Several comments can be made about these graphs. First, the fit provided by the different estimates is reasonable, especially in the exponential case (as is usually the case). Second, the differences between the samplers are negligible for a given estimation method, even though the corresponding parameters (p_j, ζ_j) seem to vary considerably (this is particularly true for the clustering procedure). Third, we can detect a strong similarity between the clustering and the global loss function approaches, even though intuition would rather suggest a similarity with the local loss function. This

Table 5: The normal mixture estimates (arranged by means) under the two loss functions (8) and (11) for the different simulation schemes.

		True values	Gibbs sampling	Random walk	Langevin	Tempered Langevin
Local	$\hat{p}_1$	0.35	0.319	0.323	0.293	0.322
loss	$\hat{\theta}_1$	-0.95	-1.259	-1.238	-1.184	-1.264
function	$\hat{\tau}_1^2$	0.5	0.361	0.355	0.395	0.369
	$\hat{p}_2$	0.22	0.206	0.233	0.298	0.241
	$\hat{\theta}_2$	1.1	0.982	1.032	0.794	1.095
	$\hat{\tau}_2^2$	0.3	0.807	0.823	0.340	0.941
	$\hat{p}_3$	0.43	0.475	0.444	0.408	0.436
	$\hat{\theta}_3$	2.4	2.228	2.253	2.235	2.252
	$\hat{\tau}_3^2$	0.2	0.209	0.165	0.216	0.157
Global	$\hat{p}_1$	0.35	0.302	0.303	0.300	0.300
loss	$\hat{\theta}_1$	-0.95	-1.085	-1.100	-1.103	-1.103
function	$\hat{\tau}_1^2$	0.5	0.366	0.347	0.342	0.345
	$\hat{p}_2$	0.22	0.233	0.264	0.271	0.276
	$\hat{\theta}_2$	1.1	0.835	0.920	0.927	0.942
	$\hat{\tau}_2^2$	0.3	0.442	0.473	0.483	0.535
	$\hat{p}_3$	0.43	0.465	0.433	0.430	0.424
	$\hat{\theta}_3$	2.4	2.336	2.382	2.391	2.385
	$\hat{\tau}_3^2$	0.2	0.283	0.228	0.218	0.223

Figure 15: Estimations by the clustering and the loss function procedures of §4.2 and 5 of the density of the same exponential sample as in Figure 2 (represented by the histogram), for the different samplers. *(Full lines stand for the clustering procedure, short dashes to the local loss function and long dashes to the global loss function. The later cannot be distinguished from the clustering estimate on the picture.)*

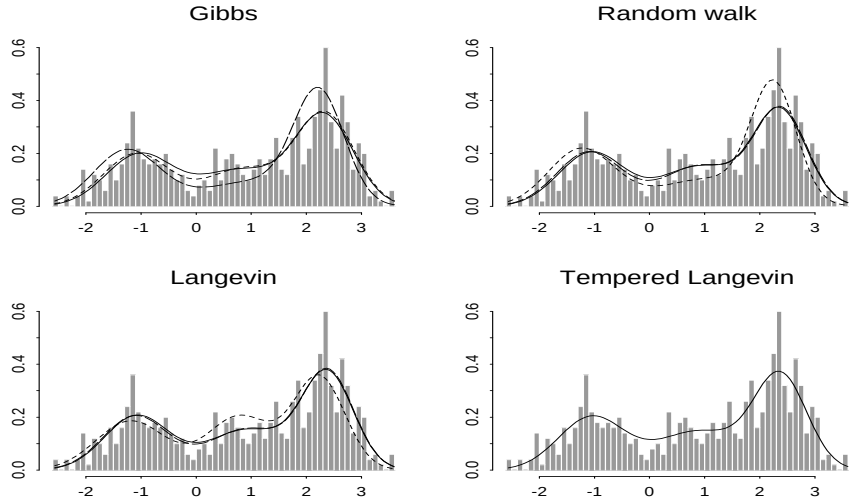


correspondence is interesting because it indicates that the clustering technique is a good approximation to a method with sounder foundations and, besides, that it is not overly dependent on the choice of the distance (6).

Despite the fact that the various estimates are generally in accord, the overall impression of this paper may be that it complicates rather than clarifies certain aspects of mixture modelling. We argue that the picture is more complicated than previously thought but we are equally convinced that this level of complexity is necessary if we want to present a thorough analysis of the mixture problem. Indeed, we have clearly demonstrated that identifiability constraints can have a potentially detrimental effect on the estimates with the corresponding implication that the full posterior distribution must be simulated. We have also shown that the standard Markov chain Monte Carlo samplers are fundamentally unable to overcome the attraction effect of local modes, and established the appeal of tempered Langevin diffusion algorithms as a semi-automated Metropolis-Hastings technique. On the inferential side of the problem, we have suggested two approaches to the statistical analysis of both symmetric and non-symmetric Markov chain Monte Carlo samples one based on a clustering approximation and the other on new choices of loss function.

While the overall message is indeed one of increased complexity, we do not see this as a deterrent because mixtures of distributions (and other latent variable models) are a

Figure 16: Estimations by the clustering and the loss function procedures of the density of the same normal sample as in Figure 3 (represented by the histogram), for the different samplers. *(Full lines stand for the clustering procedure, short dashes for the local loss function and long dashes for the global loss function.)*



complex area and one which therefore calls for advanced procedures. Note moreover that the automated aspect which has been stressed in several parts of the paper allows for a partial recovery from this complexity, in the sense that both the Langevin and the tempering parts of the procedure can be exported to other setups at little additional programming cost.

References

Baddeley, A. (1992) Errors in binary images and a L^p version of the Hausdorff Metric. *Nieuw Archief voor Wiskunde* **10** 157–183.

Celeux, G. (1998) Bayesian inference for mixtures: The label switching problem. In *COMPSTAT 98*, R. Payne and P. Green (Eds.), Physica-Verlag, 227–232.

Chib, S. (1995) Marginal likelihood from the Gibbs output. *J. Amer. Statist. Assoc.* **90** 1313–1321.

Diebolt, J. & Robert, C.P. (1990) Estimation des paramètres d'un mélange par échantillonnage bayésien. *Notes aux Comptes–Rendus de l'Académie des Sciences I* **311**, 653–658.

Diebolt, J. & Robert, C.P. (1994) Estimation of finite mixture distributions by Bayesian sampling. *J. Royal Statist. Soc.* (Ser. B) **56**, 363–375.

Frigessi, A. & Rue, H. (1997) Bayesian Image Classification using Baddeley's Delta Loss. *J. Comput. Graphical Statis.* **6**, 55–73.

Gelman, A., Gilks, W.R. & Roberts, G.O. (1996) Efficient Metropolis jumping rules. In *Bayesian Statistics 5*, J.O. Berger, J.M. Bernardo, A.P. Dawid, D.V. Lindley and A.F.M. Smith (Eds.), 599–608. Oxford University Press, Oxford.

Geyer, C.J. & Thompson, E.A. (1995) Annealing Monte Carlo maximum likelihood with application to pedigree analysis. *J. Amer. Statist. Assoc.* **90**, 909–920.

Gruet, M.A., Philippe, A., & Robert, C.P. (1999) Markov chain Monte Carlo control spreadsheets for exponential mixture estimation. *J. Comput. Graphical Statis.* (to appear).

Guihenneuc, C., Knight, S., Mengersen, K.L., Richardson, S., & Robert, C.P. (1999) MCMC diagnostics in action. Tech. report, CREST, Paris.

Liu, J.S., Liang, F., & Wong, W.H. (1998) The use of multiple-try method and local optimization in Metropolis sampling. Tech. report, Dept. of Statistics, Stanford University.

Mengersen, K. & Robert, C.P. (1996) Testing for mixtures: a Bayesian entropic approach. In *Bayesian Statistics 5*, J.O. Berger, J.M. Bernardo, A.P. Dawid, D.V. Lindley and A.F.M. Smith (Eds.), 255-276. Oxford University Press, London.

Neal, R. (1996) Sampling from multimodal distributions using tempered transitions. *Statistics and Computing* **4**, 353–366

Richardson, S. and Green, P.J. (1997) On Bayesian analysis of mixtures with an unknown number of components (with discussion). *J. Royal Statist. Soc.* (Ser. B) **59**, 731–792.

Robert, C.P. (1996) Inference in mixture models. In *Markov Chain Monte Carlo in Practice*, W.R. Gilks, S. Richardson and D.J. Spiegelhalter (Eds.), 441–464. Chapman and Hall, London.

Robert, C.P. (1997) Discussion of Richardson and Green' paper. *J. Royal Statist. Soc.* (Ser. B) **59**, 758–764.

Robert, C.P. (1998) Specifics of latent variable models. The label switching problem. In *COMPSTAT 98*, R. Payne and P.G. Green (Eds.), Physica-Verlag, 101–112.

Robert, C.P. & Casella, G. (1999) *Monte Carlo Statistical Methods.* Springer-Verlag (to appear).

Robert, C.P. & Mengersen, K.L. (1998) Reparametrization issues in mixture estimation and their bearings on the Gibbs sampler. *Comput. Statist. Data Anal.* **29**(3), 325–343.

Roberts, G.O. & Tweedie, R.L. (1995) Exponential convergence for Langevin diffusions and their discrete approximations. Res. report, Stat. Lab., U. Cambridge, Cambridge.

Roeder, K. & Wassserman, L. (1997) Practical Bayesian density estimation using mixtures of normals. *J. Amer. Statist. Assoc.* **92**, 894–902.

Rue, H. (1995) New Loss Functions in Bayesian Imaging. *J. Amer. Statist. Assoc.* **90**, 900–908.

Stephens, M. (1997) Bayesian methods for mixtures of normal distributions. Ph.D. Thesis, Oxford Uni.

Stramer, O. & Tweedie, R.L. (1997) Geometric and Subgeometric Convergence of Diffusions with Given Stationary Distributions, and Their Discretizations. Tech. report, University of Iowa.

Appendix 1: Langevin algorithms for normal and exponential mixtures

1. Normal mixture The Langevin algorithm uses the random walk drift

$$x_{t+1} = x_t + \frac{\sigma^2}{2}\nabla \log f(x_t) + \sigma\varepsilon_t, \quad \varepsilon_t \sim \varphi,$$

the new value being accepted with probability

$$\rho = \min\left\{1, \frac{f(x_{t+1})}{f(x_t)} \frac{\varphi\left[\left(x_t - x_{t+1} - \frac{\sigma^2}{2}\nabla \log f(x_{t+1})\right)\Big/\sigma\right]}{\varphi\left[\left(x_{t+1} - x_t - \frac{\sigma^2}{2}\nabla \log f(x_t)\right)\Big/\sigma\right]}\right\}.$$

The choice of the parameterization of the model is thus important because it appears both in $\nabla \log f$ and in the acceptance probability ρ. The constraints imposed on the parameters τ_j and p_j must in addition be taken into account in the choice of φ and we thus opt for an unconstrained parameterization, namely

$$\varrho_j = \log(p_j/(1-p_1)), \ j > 1, \qquad \omega_j = \log(\tau_j^2), \ j \geq 1,$$

which allows for a standard distribution φ. For spread reasons, we will prefer the Cauchy distribution rather than the normal distribution. The change of parameterization involves a Jacobian term, which is

$$\frac{d(p_1, \cdots, p_k, \tau_1, \cdots, \tau_k)}{d(\varrho_1, \cdots, \varrho_k, \omega_1, \cdots, \omega_k)} = \tau_1^2 \cdots \tau_k^2 p_1 \cdots p_k,$$

since

$$\begin{aligned}
\frac{d(p_2, \cdots, p_k)}{d(\varrho_2, \cdots, \varrho_k)} &= \begin{vmatrix} (1-p_1)p_2 & & \cdots & & -p_2p_k \\ & \ddots & & & \\ -p_2p_i & & (1-p_i)p_i & & -p_ip_k \\ & & & \ddots & \\ & & & & (1-p_k)p_k \end{vmatrix} \\
&= p_2 \cdots p_k \begin{vmatrix} 1-p_2 & -p_3 & \cdots & -p_k \\ -1 & 1 & & 0 \\ & & \ddots & \\ -1 & 0 & & 1 \end{vmatrix} = p_1 \cdots p_k \,.
\end{aligned}$$

Given the choice of priors on $(p_1, \cdots, p_k, \theta_1, \cdots, \theta_k, \tau_1^2, \cdots, \tau_k^2)$, this leads to the following expression of $\pi(\varrho_1, \cdots, \varrho_k, \theta_1, \cdots, \theta_k, \omega_1, \cdots, \omega_k)$, up to normalization,

$$\prod_{i=1}^{n}\left(\sum_{j=1}^{k} p_j \tau_j^{-1} \exp\{-(x_i-\theta_j)^2/2\tau_j^2\}\right) \times \prod_{j=1}^{k} \tau_j^{-3} \exp\left(-\frac{\theta_j^2}{20\tau_j^2} - \frac{1}{\tau_j^2}\right).$$

2. Exponential mixtures The arguments are similar. The scale parameters λ_j are reparameterized as $\omega_j = \log(\lambda_j)$ and the weights p_j as above. The posterior distribution on $(\zeta_1, \cdots, \zeta_k, \omega_1, \cdots, \omega_k)$

is thus proportional to

$$\prod_{i=1}^{n} \left(\sum_{j=1}^{k} p_j \lambda_j e^{-x_i \lambda_j} \right) \prod_{j=1}^{k} \lambda_j p_j e^{-\lambda_j} \ .$$

Unit´e de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unit´e de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unit´e de recherche INRIA Rhône-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unit´e de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unit´e de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
http://www.inria.fr
ISSN 0249-6399