Adaptive Independent Metropolis-Hastings by Fast Estimation of Mixtures of Normals

Paolo Giordani
Research Department, Sveriges Riksbank
paolo.giordani@riksbank.se

Robert Kohn
School of Economics and School of Banking and Finance
University of New South Wales

August 1, 2006

Abstract

Adaptive Metropolis-Hastings samplers use information obtained from previous draws to tune the proposal distribution. The tuning is carried out automatically, often repeatedly, and continues after the burn-in period. Because the resulting chain is not Markovian, adaptation needs to be done carefully to ensure convergence to the correct ergodic distribution. In this paper we distill recent theoretical advances on non-Markovian chains into simple guidelines to construct adaptive independent Metropolis-Hastings samplers. We then propose one such sampler in which the flexibility of mixtures of normals is exploited to construct the proposal distribution. To take full advantage of the potential of adaptive samplers it is often desirable to update the mixture of normals frequently and starting early in the chain. Algorithms must therefore be built for speed and reliability. The sampler performance is evaluated with simulated examples and with applications to time-varying-parameter, semi-parametric, and stochastic volatility models.

Keywords: Clustering; Gibbs sampling; Markov chain Monte Carlo; Semiparametric regression models; State space models.

1 Introduction

Bayesian methodology using Markov chain Monte Carlo simulation methods has had a large impact on statistical and econometric practice over the last fifteen years because of its ability to estimate complex models and produce finite sample inference. A key component in implementing Markov chain Monte Carlo (MCMC) simulation is the Metropolis-Hastings method (Metropolis et al. 1953; Hastings 1970) which includes the Gibbs sampler as a special case.

Bayesian inference by Metropolis-Hastings sampling requires the specification of one or several proposal distributions. The speed at which the chain converges to its ergodic distribution and its ability to move efficiently across the state space depend crucially on whether the proposal provides a good approximation to the target distribution, and on the local (as in random walk Metropolis) or global (as in independent Metropolis-Hastings with all variables generated jointly) nature of this approximation. Given the key role played by proposal distributions, it is natural to use experience from preliminary runs to tune them. In this broad sense most Metropolis-Hastings samplers are adaptive. In this paper we more narrowly define a Metropolis-Hastings sampler to be adaptive if the tuning of the proposal is carried out automatically and if the tuning continues (at least potentially) for the entire length of the chain.

Since at each iteration the proposal may depend on the entire history of the draws, an adaptive chain is not Markovian, and therefore the standard proof that Metropolis-Hastings (henceforth MH) samplers asymptotically draw from the target distribution no longer applies. This lack of a solid theoretical foundation has for a long time limited the development of adaptive MH schemes, notwithstanding the intuitive appeal of using accumulated knowledge about the target to improve the proposal and hence increase sampling efficiency. Nor was this caution excessive, since it is not difficult to formulate apparently reasonable adaptive schemes which do not converge to the correct ergodic distribution. Theory was therefore needed not only to justify adaptive MCMC schemes, but also to guide in their construction.

The literature on adaptive MCMC methods has followed two main strands. Adaptation by *regeneration* stems from the work of Gilks et al. (1998) In this paper we focus exclusively on *diminishing adaptation* schemes. Important theoretical advances in diminishing adaptation have been made in recent years by Holden (1998) Haario et al. (2001), Andrieu and Robert (2001), Andrieu and Moulines (2006), Andrieu et al. (2005), Atchadé and Rosenthal (2005), and Nott and Kohn (2005). Although more theoretical work can be expected, the existing body of results provides sufficient justification and guidelines to build

adaptive MH samplers for challenging problems. The main theoretical obstacles having been solved, research is now needed to design efficient and reliable adaptive samplers for broad classes of problems.

This more applied literature is still in its infancy, and has mostly focused on random walk Metropolis. Partial exceptions are Gå semyr (2003) who uses normal proposals for an independent Metropolis-Hastings, but limits the tuning of the parameters to the burn-in period, and Hastie (2005),who mixes random walk and independent MH in reversible jump problems. Independent MH schemes are implemented by Nott and Kohn (2005) to sample discrete state spaces in variable selection problems (e.g. to learn if a variable is in or out), and by Giordani and Kohn (2005) to draw interventions in dynamic mixture models (e.g. to learn if an observation is a break, an outlier, or neither).

This paper contributes to the development of algorithms for adaptive independent MH in continuous state spaces. Our aim is to provide samplers that can be expected to perform well for interesting sets of models. Increased sampling efficiency is obviously one important goal, particularly in cases where current best practice (typically some version of random walk Metropolis or Gibbs sampling) is less than satisfactory. But equally important achievements of adaptive schemes may be to expand the set of problems that can be handled efficiently by general purpose samplers and to reduce coding effort. In particular, adaptive schemes can reduce dependence on the use of conjugate priors. Such priors make it easier to implement MCMC schemes, but are less necessary when using adaptive sampling. See, for example, the discussion of the treatment of smoothing parameters in the semiparametric example in section 5.

Our adaptive sampler is built on three main ideas. The first is to ensure that the theoretical conditions for the correct ergodic behavior of the sampler are respected. The second is to estimate mixtures of Gaussians from the history of the draws and use them as proposal distributions for independent MH. The third is to perform this estimation frequently, particularly early on in the chain, starting after a small number of accepted draws (a strategy we call *intensive adaptation*). To apply these ideas successfully, estimation of the mixture parameters needs to be fast, reliable, and robust. We achieve a good balance of these goals by carefully selecting and tailoring to our needs algorithms developed in the clustering literature.

We show that this adaptive sampler performs well in three examples in which commonly used Gibbs schemes can be very inefficient: a time-varying-parameter model, a semiparametric Gaussian model, and a stochastic volatility model. All these examples involve a large number of parameters or latent

variables. However, conditional on a small set of parameters, the others can either be integrated out or have a known distribution, so we update all the parameters jointly and still obtain high acceptance rates.

The structure of the paper is as follows. Section 2 states a set of sufficient conditions for the correct ergodic behavior of adaptive MH samplers, and highlights some practical implications. Section 3 discusses opportunities and risks of intensive adaptation strategies. Section 4 presents our adaptive independent MH sampling schemes. The algorithms are introduced and discussed, with technical details gathered in Appendix A. Section 5 provides applications to time-varying-parameter, semiparametric, and stochastic volatility models. Section 6 concludes.

2 Designing adaptive independent MH samplers: some theory

A *diminishing adaptation* MH sampler performs the accept/reject step like a standard MH, but updates the proposal distribution using the history of the draws. This updating is 'diminishing' in the sense that the proposal distribution settles down asymptotically (in the number of iterations). Theorem 1 in Nott and Kohn (2005), which we restate here, provides sufficient conditions for the ergodicity of an adaptive independent MH sampler.

Let $Z = \{Z_n : n > 0\}$ be a random process on a compact state space Ξ evolving according to a collection of transition probabilities

$$T_n(z, z') = pr(Z_{n+1} = z' | Z_n = z, Z_{n-1} = z_{n-1}, ..., Z_0 = z_0),$$

and let $p(z)$ be be distribution of Z_n and $\pi(z)$ the target distribution we wish to sample from. The proof in Nott and Kohn (2005), is given for a discrete state space, but is readily extended to a compact state space. A more general proof in given in Atchadé and Rosenthal (2005). The assumption of a compact state space is relaxed in Andrieu et al. (2005).

Theorem 1 *Suppose that for every n and $z_0, ..., z_{n-1} \in \Xi$ and for some distribution $\pi(z)$ on Ξ*

$$\sum_{z_n} \pi(z_n) T_n(z_n, z_{n+1}) = \pi(z_{n+1}) \tag{1}$$

$$|T_n(z, z') - T_{n+k}(z, z')| \leq a_n c_k, a_n = O(n^{-r_1}), c_k = O(k^{-r_2}), r_1, r_2 > 0 \tag{2}$$

$$T_n(z, z') \geq \epsilon\pi(z'), \epsilon > 0, \tag{3}$$

where ϵ *does not depend on* $n, z_0, .., z_{n-1}$. *Then, for any initial distribution* $p(z_0)$ *for* Z_0

$$\sup_{z_n} |p(z_n) - \pi(z_n)| \to 0 \text{ for } n \to \infty.$$

By interpreting conditions (1)-(3) we can provide a set of useful guidelines for building adaptive MH schemes.

Invariance condition. Equation (1) is an invariance condition that is similar to that for a standard MH scheme, except that the transition density is indexed by n. This condition is satisfied by any adaptive MH kernel as long as the last available draw z_n is not used to tune the parameters of the proposal distribution.

The proposal distribution must settle down. Equation (2) is a diminished adaptation condition which requires the transition density, and hence the proposal density, to converge to a fixed distribution. This can be achieved by using the full history of the draws (except for the last value) or any suitably increasing sub-sample to design the proposal density. For example, using every *k-th* draw or at some point discarding the first M draws are both feasible options, but continuously using only the last J draws is not.

Bound the tails. Equation (3) is an ergodicity condition. For an adaptive independent MH (henceforth AIMH), equation (3) is equivalent to the condition

$$\frac{g_n(z)}{\pi(z)} \geq \epsilon, \quad \forall z, \quad \epsilon > 0, \tag{4}$$

where $g_n(z)$ is the proposal distribution used at iteration n. To see this, let

$$\alpha_n(z, z') = \left\{ \min 1, \frac{\pi(z')}{g_n(z')} \frac{g_n(z)}{\pi(z)} \right\}$$

and note that $T_n(z, z') = g_n(z')\alpha_n(z, z')$. Thus, if $\alpha_n(z, z') = 1$, then $T_n(z, z')/\pi(z') = g_n(z')/\pi(z')$ and $g_n(z)/\pi(z) \geq g_n(z')/\pi(z')$, and if $\alpha_n(z, z') < 1$, then $T_n(z, z') = g_n(z)\alpha_n(z, z)$. The equivalence of (3) and (4) follows.

This is the same requirement for the geometric ergodicity of a standard independent MH, except that the proposal density is indexed by n and that the condition must therefore hold for any possible history of the draws. If the target distribution has a finite state space, condition (4) can be enforced by setting a lower bound on $g_n(z)$ (see Nott and Kohn, 2005). Several strategies are available for continuous distributions. The most straightforward is to enforce a compact parameter space (by using a standard prior density with a compact support or by truncating a general density), coupled with one or more of the following designs for the proposal distribution: (i) set a lower bound on the variance implied by $g_n(z)$, such that condition (4) is satisfied at the lower bound, or (ii) let the proposal distribution be a mixture, with one component having a lower bound on the variance, or simply with one component being constant rather than adaptive (see Holden 1998). If the proposal distribution is designed carefully, enforcing a compact state is often unnecessary in practice, and is not implemented in this paper. Alternatively, one could check the conditions given in Andrieu et al. (2005), which make the assumption of a compact space unnecessary.

The next example illustrates that when the target is a mixture of normals, using a normal proposal whose parameters are adaptively estimated from past draws may not satisfy condition (4).

Example 1 Suppose that the target distribution is the two component mixture of normals

$$\pi(z) = 0.8\phi(z; 0, 1) + 0.2\phi(z; 0, 16),$$

where $\phi(z; \mu, \sigma^2)$ denotes the normal density with mean μ and variance σ^2. Suppose further that the initial proposal density is $g_0(z)$ is normal $\phi(z; 0, 16)$, and that after a certain number of draws we would like to estimate a normal proposal density which is based on previous draws. Clearly the initial distribution satisfies condition (4) and would therefore be a valid proposal for a standard IMH. However, estimating the mean and variance from previous iterations will give values of 0 and 4 respectively, on average, as these are the first two moments of $\pi(z)$. To illustrate our point that an adaptive proposal based on a normal distribution will be inadequate, suppose that $g_n(z) = \phi(z; 0, 4)$. Then $g_n(z)/\pi(z) < 10\exp(-3z^2/32)$ implying that condition (4) does not hold, that is, the tails of the proposal are too thin to bound the tails of $\pi(z)$.

The above discussion shows how it is possible to build adaptive independent MH samplers for which the proof of geometric ergodicity is not much more difficult than for a standard independent MH sampler. Of course, in practice both standard and adaptive independent MH schemes can be so inefficient that the theoretically assured geometric ergodicity is of little practical relevance. By

improving the fit of the proposal density to the target adaptively, our aim is to increase the number of problems that can be handled efficiently by IMH.

3 Two-step adaptation and intensive adaptation

A necessary condition for a successful AIMH sampler is that, given a sizable sample drawn from the target $\pi(z)$, the suggested algorithm can build a proposal $g(z)$ which is sufficiently close to the target for IMH to perform adequately. A two step adaptive strategy is feasible whenever the answer is positive. We loosely define *two-step adaptation* as a sampling scheme in which a rather thorough exploration of the target density is carried out in the first part of the chain by a sampler other than IMH (such as random walk Metropolis) before switching to a more efficient IMH sampler with proposal density tuned on the first-stage draws. Hastie (2005) provides an interesting application to reversible jump problems.

Two-step adaptation is relatively simple and safe, and in interesting cases capable of achieving sizable efficiency gains. It has, however, a number of limitations. If the first stage sampler fails to explore a region of the state space, the proposal built for the second stage will not have adequate coverage of that region either. To reduce these risks we may need a very large number of iterations in the first phase. Finally, there is no saving of coding effort, as the first stage sampler still needs to be implemented and tested. This can be a time consuming task, particularly when the first stage sampler uses several conditional distributions, as in Gibbs or Metropolis-within-Gibbs. For some samplers, such as random walk Metropolis, the duplication of coding effort is minimal: we write the likelihood function and then simply switch from a random walk proposal to an independent proposal. However, all the limitations of two-step adaptation are more severe when the first stage sampler is inefficient, which is often the case for random walk schemes. It is in these cases that intensive adaptation is most interesting. The examples provided in Section 6 show that AIMH is a very promising approach to broadening the set of problems that can be handled efficiently with the same convenience of IMH and random walk Metropolis samplers (both of which just require coding the likelihood and prior functions).

We loosely define *intensive adaptation* as an AIMH scheme in which the proposal distribution is updated frequently, particularly in the early part of the chain. Building a sequence of increasingly good proposal densities in intensive

adaptation is a more demanding task than building a proposal density once and with thousands of draws available. The question is whether we can adequately explore the target distributions given an initial proposal $g_0(z)$ but no draws. The answer inevitably depends on the initial proposal $g_0(z)$, on the target $\pi(z)$, and on the details of the sampling scheme. However, it is possible to outline some general dangers and opportunities offered by intensive adaptation.

Among the advantages, if the proposal distribution is sufficiently flexible, frequent tuning of its parameters and continuing adaptation for the entire length of the chain reduces the risk of a long run of rejections and increases the chances of good performance when the initial proposal approximates the target poorly.

Estimating proposal densities based on a small number of draws presents some dangers that the designer of an AIMH scheme should try to minimize. For example, suppose that we predetermine the iteration, say j, at which the proposal is first updated. If the first j draws have all been rejected, then a proposal distribution based entirely on these draws is degenerate and makes the chain reducible. If too few draws have been accepted, the proposal may be very poor. There are various ways of preventing these outcomes. We employ the following. First, we initially update the proposal at a predetermined number of *accepted* draws. Second, after fitting a mixture of normal distributions to past draws, we fatten and stretch its tails. Third, we let the proposal be a mixture where one component is the initial proposal $g_0(z)$, which should of course have long tails. This is similar to the *defensive mixtures* approach studied by Hesterberg (1998) for importance sampling. The same provisions reduce the risk of adapting too quickly to a local mode.

4 A clustering algorithm for fast estimation of mixtures of normals in adaptive IMH

Finite mixtures of normals are an attractive option to construct the proposal density because they can approximate any continuous density arbitrarily well and are quick to sample from and evaluate. See McLachlan and Peel (2000) for an extensive treatment of finite mixture models.

However, estimating mixtures of normals is already a difficult problem when an independent and identically distributed sample from the target is given and estimation needs to be performed only once: the likelihood goes to infinity whenever a component has zero variance (an even more concrete possibility if, as unavoidable in IMH, some observations appear more than once), and its

maximization, whether by the EM algorithm or directly, is plagued by local modes. Although several authors have made substantial advances in dealing with these problems (e.g. Figuereido and Jain 2002; Ueda, Nakano, Ghahramani, and Hinton 2000; Verbeek, Vlassis, and Krose 2003), the EM algorithm does not seem to be sufficiently reliable when the sample is small and contains a non-trivial share of rejected draws. The inevitable short runs of rejections give rise to small clusters with zero covariance matrix at which the EM algorithm frequently gets stuck. Moreover, EM-based algorithm are computationally expensive and slow to converge, which makes them less attractive when the proposal is to be updated frequently.

We have therefore limited our attention to algorithms that estimate mixtures of normals quickly and without explicitly computing the covariance matrix of each component (for robustness). Within this class, the *k-means* is the most popular algorithm. We employ the *k-harmonic means*, an extension of the k-means algorithm that allows for soft membership. Degeneracies can be easily prevented, so the algorithm is remarkably robust even in the presence of long series of rejections. The number of clusters is chosen with the BIC criterion. The increase in speed and reliability is paid for with a decreased fit to the target, as the family of k-means algorithms performs best when an optimal fit requires all components of the mixture to have the same probability and covariance matrix (see Bradly and Fayyad 1998, for a discussion). Hamerly and Elkan (2002) show that the performance of k-harmonic means deteriorates less rapidly than alternatives of similar computational costs with departures from these ideal conditions.

Although the k-harmonic means algorithm is less sensitive to initialization than either k-means or EM (Hamerly and Elkan 2002), in an unsupervised environment it is important to have good starting values. We have found the algorithm of Bradly and Fayyad (1998) to perform very well and at a low computational cost.

One situation (quite common in applications) in which clustering algorithms do not perform well is when a multivariate distribution is normal along most but not all dimensions (the EM algorithm is also in trouble in this case). We have found the following ad hoc solution to work well. Rather than trying to fit a mixture of normals to all parameters of the target distribution, whenever we need to update the mixture we divide the parameter vector θ into two groups, θ_1 and θ_2, where parameters in θ_1 have approximately symmetric marginal distributions while parameters in θ_2 do not. A normal is then fitted to the first group and a mixture of normals to the second. Finally, we compute covariances and build the proposal as a mixture of normals for the entire vector θ (see the

appendix for details). This enhances the ability of the algorithm to deal with the situations mentioned above and reduces computing times whenever some or all parameters are nearly normally distributed.

In principle the proposal could be updated at each iteration, but in practice that would be too costly. We update it at predetermined numbers of accepted draws, more frequently in the earlier stages of the chain. We also recommend updating the proposal following a long sequence of consecutive rejections (excluding the last draw). A large number of consecutive rejections signals a poor fit of the proposal, and it therefore makes sense to update the proposal to give it a chance of better covering that area. This does not violate conditions (1)-(3) because only the decision of when to update the proposal is affected, and not the proposal itself: as the draws accumulate, the proposal gradually settles down.

Following the recommendations of Section 2, the proposal distribution for MH is built by fattening and stretching the tails of the mixture of normals estimated by k-harmonic means, and by drawing from the initial proposal with a small probability. The proposal density at iteration n is therefore given by the initial proposal $g_0(z)$ before the mixture of normals is estimated for the first time, and afterwards by

$$\pi_1 g_0(z) + \pi_2 \widetilde{g}_n(z) + (1 - \pi_1 - \pi_2) g_n(z),$$

where $g_n(z)$ is the mixture of normals last estimated by the clustering algorithm, and $\widetilde{g}_n(z)$ is a version of $g_n(z)$ with fatter and longer tails: the means and probabilities of each cluster are unchanged, and the variances are multiplied by a user-defined scalar k (we use $k = 4^2$). The probability π_1 can be set to a small number, say 0.05; $\pi_1 > 0$ ensures that condition (4) is satisfied for all n as long as it is satisfied for the initial proposal $g_0(z)$. Setting $\pi_2 > 0$ is not required theoretically but can be very helpful in practice. Fattening the tails can help to satisfy condition (4) as n grows even when it is not satisfied at $g_0(z)$. Maybe more important in practice, setting $\pi_2 > 0$ (at 0.05-0.2, say) often greatly speeds up the exploration of the state space and hence the convergence of the proposal distribution. Although we have used constant π_1 and π_2 in our work, these considerations suggest that a more sophisticated strategy would involve (i) setting π_1 and π_2 rather high in the initial, exploratory phase, and then gradually lowering them (ii) letting π_1 start high (low) and decrease slowly (rapidly) if the quality of the initial proposal $g_0(z)$ is good (bad). A full description of the algorithm is given in the appendix.

5 Discussion

In order to understand the strengths and limitations of our sampler, we have found it useful to consider two desirable qualities of an adaptive IMH scheme. First, given a sufficiently large sample drawn from the target, we wish to construct a proposal density which fits the target as well as possible. This is an approximating ability: we want to draw an accurate 'map' of the areas that we have already explored. Second, we wish the sampler to perform as well as possible when the initial proposal fails to cover part of the support of the target distribution. This is an exploring ability: when we propose in a region where our map is poor, we want to explore that region and quickly update our map.

For example, using a normal proposal when the target is highly non-normal results in little approximating ability. Updating the proposal only once or very rarely results in little exploring ability, since the proposal reacts slowly or not at all to the information that it is fitting poorly at a given point.

Our sampler has several characteristics designed to enhance its exploring ability. Frequent updating, particularly at early iterations, and updating following a sequence of rejections and/or a low MH acceptance probability both quicken the pace at which the proposal adapts to the information that it is not fitting well in a certain area. Long tails are useful not only to satisfy the ergodicity condition (4), but also to improve the chances of venturing into unexplored parts of the state space. Finally, mixtures are ideally suited for this exploration because a new component can be centered on a value causing a sequence of rejections. The long runs of rejections that can plague standard IMH are therefore much less likely using our AIMH sampling scheme because the proposal distribution is updated frequently and will accommodate the cluster of rejections by changing the mixture parameters or adding a new component. If our sampler makes a move in a region where the proposal fits poorly, it should therefore be able to explore it. Of course as the parameter dimension increases, if the initial proposal fails to cover a region we may never explore that region simply because the probability of making a proposal there is too small.

The next example shows that in low dimensions we can often get away with a very poor initial proposal distribution.

Example 2 Suppose that the target is the univariate mixture

$$\pi(z) = 0.5\phi(z; 0, 1) + 0.3\phi(z; -3, 4) + 0.2\phi(z; 6, 0.5),$$

and the initial proposal is $\phi(z; -5, 4)$. This proposal has very high importance weights π/g in a large part of the support of z, but we still quickly converge to

a good approximation of the target. The acceptance rates increase fast initially and then stabilize as the proposal distribution also settles down. See Figure 1

As the dimension increases, a good initial proposal distribution becomes more valuable. This is illustrated in the next example.

Example 3 Consider the fifteen dimensional target distribution which is the mixture of two normals

$$\pi(z) = 0.7\phi(z; 0, I) + 0.3\phi(z; \mu_2, 2I),$$

where $\phi(z; \mu, \Sigma)$ is a multivariate normal density with mean μ and covariance matrix Σ evaluated at z. The vector μ_2 has elements $\mu_{2,i} = 0$ for $i = 1, ..., 14$, and $\mu_{2,15} = -3$. The first fourteen marginals are therefore symmetric but slightly leptokurtic, whereas the fifteenth is also skewed. The proposal distribution is initialized by fattening and stretching the tails of the Laplace approximation, that is, a normal distribution centered at the mode and with covariance set to minus the inverse of the Hessian of the log-likelihood at the mode. The Laplace approximation is nearly equal to $\phi(z; 0, I)$, so we have $g_0(z) \simeq 0.6\phi(z; 0, I) + 0.4\phi(z; \mu_2, 16I)$. The acceptance rates at the initial proposal are not high, but sufficient to start the learning process (see figure 2). The AIMH learns the marginal distribution of the non-normal variable rather well and, since most variables are normal, at very low computational cost since we only estimate the mixture parameters on variables that appear skewed. In constrast, an initial proposal such as $\phi(z; -m, 4I)$, where $m_i = -5$ for $i = 1, ..., 14$ generates such low acceptance rates for this fifteen dimensional distribution that the learning process cannot get successfully started.

6 Applications

State space models and nonparametric models are ideal initial applications for AIMH schemes. Although they can have a large number of parameters, it is often the case that, conditional on a small subset, most parameters can be integrated out or have known analytical form. Therefore it is often possible to draw all parameters in one or two blocks. Exploiting these features, it is also often inexpensive to find the posterior mode, possibly for a simplified version of the model, and therefore obtain a reasonable initialization of the proposal distribution. Finally, the standard approach based on Gibbs and Metropolis-within-Gibbs can be very inefficient, particularly for state space models (see Fruhwirth-Schnatter 2004).

In all the examples below we adopt the notation x for $\{x_1, ..., x_T\}$.

6.1 Time-varying parameter autoregressive models

Consider the following time-varying parameter first order autoregressive (AR(1)) process (the extension to a more general autoregressive process is straightforward):

$$\begin{aligned} y_t &= c_t + \rho_t y_{t-1} + \sigma_\epsilon \epsilon_t \\ c_t &= c_{t-1} + \lambda_0 \sigma_\epsilon u_t \\ \rho_t &= \rho_{t-1} + \lambda_1 v_t, \end{aligned} \tag{5}$$

where ϵ_t, u_t, v_t are all $nid(0,1)$. The model has three parameters $(\sigma_\epsilon^2, \lambda_0^2, \lambda_1^2)$, while c_0 and ρ_0 can be treated either as parameters or (our choice) as states. Given conjugate priors (inverse gamma for the parameters, and normal for c_0 and ρ_0), Gibbs sampling is straightforward (Carter and Kohn 1994). Fruhwirth-Schnatter (2004) reports that Gibbs sampling can be very inefficient in these models. Her assessment is based on inefficiency factors, that is, on the autocorrelation structure of the draws. In the following application we also find that the Gibbs draws are highly autocorrelated and, by comparing posterior statistics from Gibbs sampling and from our AIMH, we add that the autocorrelations do not reveal the full extent of the problem.

6.1.1 Application: US CPI inflation

We apply the model to quarterly U.S. CPI inflation for the period 1960-2005 (184 observations).[1] We use rather dispersed inverse gamma priors for $\sigma_\epsilon^2, \lambda_0^2, \lambda_1^2$ with a common shape parameter of 1. The scale parameters are defined by setting the modes of the priors close to maximum likelihood estimates: σ_{OLS}^2 for σ_ϵ^2 (where σ_{OLS}^2 is the residual variance from an AR(1) model estimated by OLS), at $0.01\sigma_{OLS}^2$ for λ_0^2 and at 0.001^2 for λ_1^2. The modes of λ_0^2 and λ_1^2 are centered at the maximum likelihood estimates to ensure that the bimodality in the posterior distribution of the log of λ_1^2 documented in Figure 4 is not induced by the prior.

For given parameters, the likelihood is easily computed via the Kalman filter. It is therefore simple to find the posterior mode, at which the chain is initialized. Posterior mode values suggest that time variation is nearly absent.

[1] Annualized quarterly CPI inflation, defined as $400(P_t/P_{t-1} - 1)$, where P_t is aggregated from monthly data (averages) on Consumer Price Index For All Urban Consumers: All Items, seasonally adjusted, Series ID CPIAUCSL, Source: U.S. Department of Labor: Bureau of Labor Statistics.

Starting with Gibbs sampling, we draw 40 000 times after a burn-in of 5000. The inefficiency factors for the logs of $\sigma_\epsilon^2, \lambda_0^2$, and λ_1^2, are high (see Table 1). However, the recursive parameter means do seem to settle down (not reported) and the posterior distributions are in line with a normal approximation taken at the mode, suggesting a persistent AR(1) with little sign of parameter variation (see Figure 3). It may therefore seem reasonable to assume that the chain, though mixing slowly, has produced a sample representative of the entire posterior.

However, the AIMH scheme tells a different story. The proposal is initialized at a mixture of two normals

$$g_0(z) = 0.5\phi(z; \widehat{\mu}, \widehat{\Sigma}) + 0.5\phi(z; \widehat{\mu}, 16\widehat{\Sigma}),$$

where $\widehat{\mu}$ is the posterior mode and $-\widehat{\Sigma}$ is the inverse of the Hessian of the log-posterior evaluated at $\widehat{\mu}$. The AIMH soon discovers that the posterior distribution of $\log(\lambda_1^2)$, not to mention λ_1^2, is highly non-normal (see Figure 4), with substantial probability mass around a second mode corresponding to non-trivial amounts of time variation in ρ_t and a lower ρ_1. In spite of this, the inefficiency factors are just 6.9, 2.7, and 6.4 (see Table 1).

6.2 Additive semiparametric Gaussian models

In this example we consider a semiparametric regression model with Gaussian errors, with some of the covariates entering linearly and the others entering more flexibly. The model is additive, implying that its flexibility does not extend to interaction effects. Thus, we consider the model

$$y_i = \sum_{j=1}^{m} \gamma_j z_{ji} + \sum_{h=1}^{H} f_h(x_{h,i}) + \sigma_\epsilon \epsilon_i \tag{6}$$

where ϵ_i is $nid(0,1)$. The z is a vector of regressors that enter linearly and the x_h, $h = 1, ..., H$ are covariates that enter more flexibly by using the quadratic polynomial spline functions

$$\begin{aligned} f_h(x_{h,i}) &= \beta_{0,h} x_{h,i} + \sum_{j=1}^{J} \beta_{h,j} (x_{h,i} - \tilde{x}_{h,j})_+^2 \\ &= \beta_{0,h} x_i + g_h(x_{h,i}), \end{aligned} \tag{7}$$

where $x_+ = x$ if $x > 0$ and 0 otherwise and $\{\widetilde{x}_{h,1}, ..., \widetilde{x}_{h,k}\}$ are points (or 'knots') on the abscissae of x_h such that $\min(x_h) = \widetilde{x}_{h,1} < ... < \widetilde{x}_{h,J} < \max(x_h)$. In

this paper we choose 30 knots so that each interval contains the same number of observed values of x_h. For a discussion of quadratic spline bases and other ralated bases see chapter 3 of Ruppert, Wand and Carroll (2003). We assume that a global intercept term is included in z in (6) and for simplicity we include the parameters $\beta_{h,0}$, $h = 1, \ldots, H$ in the vector γ and x_h, $h = 1, \ldots, H$ as part of the vector z. This transforms the nonparametric model into an highly parametrized linear model

$$y = \widetilde{Z}\widetilde{\gamma} + \epsilon. \tag{8}$$

The prior for the linear parameters γ can be normal with a diagonal covariance matirx

$$\gamma \sim N(0, v_\gamma^2 I),$$

where v_γ can be set to a large number. It is also convenient to assume a normal prior for the nonparametric part, with all parameters independent and

$$\beta_{h,j} \sim N(0, \tau_h^2), \ , j = 1, ..., J, \ h = 1, ..., H.$$

However, with this prior there is a high risk of over-fitting if we simply set τ_h^2 to a large number. The variance τ_h^2 is often chosen by cross-validation, but in a fully Bayesian setting we can treat τ_h^2 as a parameter. To illustrate the advantage of AIMH in working with different priors, we experiment with two options for the prior τ_h^2. The first prior is log-normal and rather dispersed:

$$\ln(\tau_h^2) \sim N(0, 5^2),$$

the second is inverse gamma with shape parameter 1 and scale parameter implied by setting the mode at 0.1^2. The prior for σ_ϵ^2 is inverse gamma with shape parameter one and scale parameter implied by setting the prior mode at the OLS residual variance estimated on (8). The prior for $\widetilde{\gamma} = (\gamma, \beta_1, ..., \beta_H)$ is therefore jointly normal conditional on $\tau^2 = \{\tau_1^2, .., \tau_H^2\}$

$$\widetilde{\gamma}|\tau \sim N(\mathbf{0}, V_{\widetilde{\gamma}}(\tau)),$$

where

$$V_{\widetilde{\gamma}}(\tau) = \begin{bmatrix} v_\gamma^2 I & 0 & 0 & 0 \\ 0 & \tau_1^2 I & 0 & 0 \\ 0 & 0 & .. & 0 \\ 0 & 0 & 0 & \tau_H^2 I \end{bmatrix}.$$

One way to estimate the posterior density of the semiparametric model is to use Gibbs or Metropolis-within-Gibbs sampling as proposed by Wong and Kohn

(1996) and Hastie and Tibshirani (2000) (who called it Bayesian backfitting). Such an approach proceeds as follows. The parameters $\widetilde{\gamma} = \{\gamma, \beta_1, ..., \beta_H\}$ are conjugate given $\theta = \{\sigma_\epsilon^2, \tau_1^2, ..., \tau_H^2\}$, and σ_ϵ^2 is conjugate given $\widetilde{\gamma}$. Each variance τ_h^2 can be updated with a Gibbs step for the inverse gamma prior, or with a Metropolis-Hastings step for the log-normal prior. In this second case, we use a Laplace approximation of $p(\ln(\tau_h^2)|\beta_h)$, which is very fast to compute using analytical derivatives. However, the correlation between τ_h^2 and $\{\beta_{h,1}, .., \beta_{h,J}\}$ could be quite high using either prior for τ_h^2. In addition, using a log normal prior for τ_h^2 leads to high rejection rates in the Metropolis-Hastings step when generating the τ_h^2. Both problems are elegantly solved using AIMH.

We now show how to update all parameters in one block with an efficient AIMH sampler. We first note that, conditional on θ, $\widetilde{\gamma}$ can be integrated out, making it possible to compute $p(\theta|y) \propto p(y|\theta)p(\theta)$, where $y|\theta \sim N(\mathbf{0}, \sigma_\epsilon^2 I + \widetilde{Z} V_{\widetilde{\gamma}}(\tau) \widetilde{Z}')$. An efficient method for computing this likelihood is given in appendix B.

6.2.1 Application: Boston housing data

We use the Gaussian semiparametric model to study the Boston housing data analyzed semiparametrically by Smith and Kohn (1996). [2] There are 506 observations. The dependent variable is the log of MV, the median value of owner-occupied homes. We use all 13 available covariates (see Smith and Kohn or the web-site for a full description) in the linear part and the following six in the nonparametric part (Smith and Kohn use only the first five):

$X_5 = NOX$, nitrogen oxide concentration,
$X_6 = RM$, average number of rooms,
$X_8 = DIS$, logarithm of the distance from five employment centers,
$X_{10} = TAX$, property tax rate,
$X_{13} = STAT$, proportion of the population that is lower status,
$X_1 = CRIM$, per capita crime rate by town.

The proposal distribution for the seven parameters $\theta = \{\ln(\sigma_\epsilon^2), \ln(\tau_5^2), ..., \ln(\tau_1^2)\}$ is initialized by fattening the tails of the Laplace approximation. To find the Laplace approximation, we simply apply Newton-Raphson optimization (with numerical derivatives) to $\ln p(y|\theta) + \ln p(\theta)$, which involves no extra

[2] The dataset is available at www.cs.utoronto.ca/~delve/data/boston.

coding effort since both distributions are needed to compute the MH acceptance ratio. Figure 5 provides results for the case of a log-normal prior on τ_h^2, $h = 1, ..., H$. The acceptance rates (all seven parameters updated jointly) quickly improves and stabilizes at around 60% (see Figure 5). Most parameters are approximately lognormally distributed, except those connected to the variables TAX and $CRIM$, which benefit from the added flexibility of mixtures. The correlation matrix of the smoothing parameters $\{\ln(\tau_5^2), ..., \ln(\tau_1^2)\}$ is nearly diagonal. This suggests that the AIMH could handle large numbers of smoothing parameters efficiently by updating them in blocks (with a different proposal density estimated adaptively on each block), since the blocks would be nearly independent of each other.

Table 1 shows that AIMH is about twice as efficient as Gibbs sampling when both samplers use the inverse gamma prior on τ_h^2, and nine times as efficient when both samplers use the log-normal prior. Reported results are for the average inefficiency factor (over both h and i) of $f_h(x_{h,i})$. Looking at the autocorrelation of the log-parameters gives similar inefficiency ratios.

Boston	mean $f_h(x_{h,i})$		**Inflation**	$\log(\sigma_\epsilon^2)$	$\log(\lambda_0^2)$	$\log(\lambda_1^2)$
AIMH, IG	3.5		AIMH	6.9	2.7	6.4
Gibbs, IG	6.3		Gibbs	9.4	113.3	156.4
AIMH, LN	2.1					
M-Gibbs, LN	18.4					

Table 1: Inefficiency factors for semiparametric (Boston) and state space (inflation) models. Inefficiency factors in columns, parameters in rows. AIMH for adaptive independent Metropolis-Hastings. M-Gibbs for Metropolis-within-Gibbs. IG and LN for inverse gamma and log-normal prior.

6.3 Stochastic volatility models

The simplest stochastic volatility model can be written for mean corrected data as

$$\begin{aligned} y_t &= e^{0.5h_t}\epsilon_t \\ h_t &= \mu + \rho(h_{t-1} - \mu) + \sigma v_t, \end{aligned} \tag{9}$$

where ϵ_t is $nid(0, 1)$ and the model parameters are $\theta = \{\mu, \rho, \sigma\}$. We square and take logs of the observation equation, and we approximate the distribution

of $\ln(\epsilon_t^2)$, which is the log of a chi-squared 1, by a mixture of normals as in Kim et al. (1998). The model then has a conditionally Gaussian state space form

$$\begin{aligned} \widetilde{y}_t &= g(K_t) + h_t + G(K_t)u_t \\ h_t &= \mu + \rho(h_{t-1} - \mu) + \sigma v_t, \end{aligned} \tag{10}$$

where ϵ_t is $nid(0,1)$, $\widetilde{y}_t = \ln(y_t^2)$, and $g(K_t)$ and $G(K_t)$ are known given K_t.

The indicators K can be sampled in one block given θ and h as in Carter and Kohn (1994). The distribution of θ given h is conjugate, but Kim et al. (1998) show that θ and h are highly correlated and recommend drawing θ given y and K but integrating h out. This is accomplished with a Metropolis-Hastings step, where $p(y|K,\theta)$ is computed via the Kalman filter. Since the posterior mode is not readily available, Kim et al. (1998) use IMH, where the proposal distribution is calibrated once from draws obtained with a less efficient sampling scheme. This is less efficient than our scheme and requires coding two different samplers. An alternative we now explore is to use AIMH from the beginning of the chain.

6.3.1 Application: USD-GBP daily returns

We analyze daily U.S. dollar - British pound returns (defined as the first difference of the log of the nominal exchange rate) for the period January 1990 to March 2004. The parameter μ can be integrated out (see Kim et al. (1998)). To initialize the proposal distribution, we approximate the distribution of $\ln(\epsilon_t^2)$ as a normal with mean -1.27 and variance 2.22^2. This gives a standard Gaussian state space models, for which the Laplace approximation is easily available. We also use the mode to center the priors for ρ and $\ln(\sigma^2)$, which are normal and disperse. The prior for ρ is truncated at 1. With fattened tails, the initial proposal gives an acceptance rate of around 10%, and it takes only a few hundred iterations for the acceptance rates to increase to around 50% (see Figure 6). This number is satisfactory given that the proposal approximates $p(\theta|y)$ while the acceptance probability is computed on $p(\theta|y,K)$. (see Figure 6)

7 Conclusion

The key message of this paper is that there is now sufficient theoretical background on adaptive Metropolis-Hastings to move to the important task of devising efficient and reliable adaptive samplers for important classes of problems. The most interesting applications arise when current best practice is inefficient

or cumbersome and, in our opinion, when adaptation starts early. Our article provides a fast and reliable algorithm which performs well in three interesting models. Better algorithms can surely be devised. Another promising area for research is the development of adaptive samplers for conditional distributions when it is not feasible or desirable to update all the parameters in one block.

References

Andrieu, C., D., M., and Priouret, P. (2005), "Stability of stochastic approximation under verifiable conditions," *SICON*, 44, 283–312.

Andrieu, C. and Moulines, D. (2006), "On the ergodicity properties of some adaptive MCMC algorithms," *Annals of Applied Probability*, forthcoming.

Andrieu, C. and Robert, C. P. (2001), "Controlled MCMC for optimal sampling," Technical report, University of Bristol.

Atchadé, Y. and Rosenthal, J. (2005), "On adaptive Markov chain Monte Carlo algorithms," *Bernoulli*, 11, 815–828.

Bradly, P. and Fayyad, U. (1998), "Refining initial points for k-means clustering," *Proceedings of the 15th International Conference on Machine Learning*, 91–99.

Carter, C. and Kohn, R. (1994), "On Gibbs sampling for state-space models," *Biometrika*, 83, 589–601.

Figuereido, M. and Jain, A. (2002), "Unsupervised learning of finite mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24, 381–396.

Fruhwirth-Schnatter, S. (2004), "Efficient Bayesian parameter estimation," in *State space and unobserved component models*, eds. Harvey, A., Koopman, S., and Shephard, N., Cambridge: Cambridge University Press.

Gå semyr, J. (2003), "On an adaptive version of the Metropolis-Hastings algorithm with independent proposal distribution," *Scandinavian Journal of Statistics*, 30, 159–173.

Gilks, W., Roberts, G., and Sahu, S. (1998), "Adaptive Markov chain Monte Carlo through regeneration," *Journal of the American Statistical Association*, 93, 1045–1054.

Giordani, P. and Kohn, R. (2005), "Efficient Bayesian inference for multiple change-point and mixture innovation models," Submitted for publication.

Haario, H., Saksman, E., and Tamminen, J. (2001), "An adaptive Metropolis algorithm," *Bernoulli*, 7, 223–242.

Hamerly, G. and Elkan, C. (2002), "Alternatives to the k-means algorithm that find better clusterings," in *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, eds. Kalpakis, K., Goharian, N., and Grossmann, D., New York: Academic Press, pp. 600–607.

Hastie, D. (2005), "Towards automatic reversible jump Markov chain Monte Carlo," Unpublished PhD dissertation, Department of Mathematics, University of Bristol.

Hastings, W. (1970), "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, 57, 97–109.

Hesterberg, T. C. (1998), "Weighted average importance sampling and defensive mixture distributions," *Technometrics*, 37, 185–194.

Holden, L. (1998), "Adaptive chains," Manuscript, Norwegian Computing Center, Oslo.

Kim, S., Shepherd, N., and Chib, S. (1998), "Stochastic volatility: likelihood inference and comparison with ARCH models," *Review of Economic Studies*, 65, 361–394.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., H., T. A., and Teller, E. (1953), "Equation of state calculations by fast computing machines," *Journal of Chemical Physics*, 21, 1087–1092.

Nott, D. and Kohn, R. (2005), "Adaptive sampling for Bayesian variable selection," *Biometrika*, 92, 747–763.

Smith, M. and Kohn, R. (1996), "Nonparametric regression using Bayesian variable selection," *Journal of Econometrics*, 75, 317–343.

Ueda, N., Nakano, R., Ghahramani, Z., and Hinton, G. (2000), "SMEM algorithm for mixture models," *Neural Computation*, 12, 2109–2128.

Verbeek, J., Vlassis, N., and Krose, B. (2003), "Efficient Greedy Learning of Gaussian Mixture Models," *Neural Computation*, 15, 469–485.

A Appendix: Adaptive sampling scheme

The proposal distribution at iteration n is given by $\pi_1 g_0 + \pi_2 \widetilde{g}_n + (1-\pi_1-\pi_2) g_n$, where g_0 is the initial proposal, for example a Laplace approximation (possibly with fattened tails), g_n is the mixture of normals last estimated by the clustering algorithm, and $\widetilde{g}_n$ is a version of g_n with fatter tails: the means and probabilities of of each cluster are unchanged, the variances are multiplied by a user-defined scalar k (we use $k = 16$). In this paper we set $\pi_1 = 0.05$, and $\pi_2 = 0.15$. These values have been found to work well but are not optimal in any well-specified sense, but the speed of convergence and the efficiency of our sampler could surely be further improved with a more careful (and possible adaptive) choice of these parameters. Any other value of k in the range 9-25 and of π_1 and π_2 in the range $0.05 - 0.3$ worked well for the examples given in this paper, and π_1 could be set to 0 without affecting the results.

The distribution g_n is a mixture of normals. It is updated at predetermined numbers of *accepted* draws. We start updating after $20, 30, 50, 100, 200, 300, 500, 1000, 2000, 3000, 5000$ and multiples of 5000 accepted draws, though updating would probably optimally start later unless the acceptance rate from g_0 is very low. We also update after a series of more than M consecutive rejections (excepting the last draw), where M is set to the number of parameters multiplied by ten, after checking that the MH acceptance probability was lower than 1% (i.e. that the rejections were due to poor fitting rather than bad luck).

The estimation of the mixture of normals can get slow when the number of iterations is large. To avoid this problem, after 1000 accepted draws we only add every *j-th* draw to the sample used to estimate the mixture parameters, where j is chosen so that the mixture is not estimated often on more than 10000 observations.

When most parameters are nearly normally distributed, fitting a mixture of normals to all the parameters is problematic in the sense that the chances of finding a local mode with all parameters normally distributed is quite high (though depending on the starting value of course). This is true of clustering algorithms and also of EM-based algorithms. To improve the performance of the sampler in these situations, we divide the parameter vector θ into two groups, θ_1 and θ_2, where parameters in θ_1 are classified as approximately normal and parameters in θ_2 are skewed.[3] A normal is then fit to the first group and

[3]Our rule of thumb is to place a parameter in the 'normal' group if its marginal distribution has $|s| < 0.2$, where s is the skeweness. Our fattening the tails of the mixture should handle the kurtosis, though this would optimally be done with mixtures of more flexible distributions

a mixture of p normals to the second. For θ_1, we can compute the mean μ_{θ_1} and covariance matrix Σ_{θ_1} inexpensively from the draws. For θ_2, we fit a mixture of normals as detailed below, estimating probabilities $\pi_1, ..., \pi_p$, means $\mu_1, ..., \mu_p$, and covariance matrices $\Sigma_1, ..., \Sigma_p$. We then need to build a mixture for $\theta = \{\theta_1, \theta_2\}$. The mean is straightforward: for the normal parameters, all components have the same mean. The diagonal blocks of the covariance matrices Ω_i corresponding to $var(\theta_1)$ and $var(\theta_2)$ for component i are also straightforward. The off-diagonal blocks of Ω_i, corresponding to $cov(\theta_1, \theta_2)$ is computed as

$$\Omega_i^{12} = \sum_{t=1}^{n} \pi_{i,t}^* [(\theta_{1,t} - \mu_{\theta_1})(\theta_{2,t} - \mu_i)] / \sum_{t=1}^{n} \pi_{i,t}^*,$$

where $\pi_{i,t}^* = prob(K_t = i|\theta_{2,t})$ is the probability of $\theta_{2,t}$ coming from the i-th component.

A.1 k-harmonic means clustering

We estimate the mixture of normal parameters using the k-harmonic means clustering algorithm (see Hamerly and Elkan 2002, for a discussion) . The algorithm runs as follows. Let p be the number of clusters.

1. Initialize the algorithm with $c_1, ..., c_p$, the component centers. The starting values are chosen with the procedure of Bradly and Fayyad (1998) . We depart slightly from Bradley and Fayyad in using the harmonic k-means algorithm (rather than k-means) in the initialization procedure.

2. For each data point θ_t, compute a membership function $m(c_i|\theta_t)$ and a weight function $w(\theta_t)$, for $t = 1, ..., p$. These are defined as

$$w(\theta_t) = \frac{\sum_{i=1}^{p} ||\theta_t - c_i||^{-p-2}}{(\sum_{i=1}^{p} ||\theta_t - c_i||^{-p})^2}$$

$$m(c_i|\theta_t) = \frac{||\theta_t - c_i||^{-p-2}}{\sum_{i=1}^{p} ||\theta_t - c_i||^{-p-2}},$$

   where $||\theta_t - c_i||$ is the Euclidean or Mahalanobis distance. Following Bradly and Fayyad (1998), we put a lower boundary ϵ on $||\theta_t - c_i||$ (to

than the normal.

avoid degeneracies when trying $||c_i - c_i||$). The membership function softens the sharp membership of the k-means algorithm, so one observation can belong to more than one cluster in differing degrees. The weight function gives more weight to observations that are currently covered poorly (i.e. that are far from the nearest center).

3. Update each center c_i

$$c_i = \frac{\sum_{t=1}^{n} m(c_i|\theta_t) w(\theta_t) \theta_t}{\sum_{t=1}^{n} m(c_i|\theta_t) w(\theta_t)}.$$

4. Repeat until convergence. This gives the cluster centers, which we take as estimates of the component means. The other mixture parameters can then be estimated for $i = 1, ..., k$ as

$$\begin{aligned} V_i &= \frac{\sum_{t=1}^{n} m(c_i|\theta_t) w(\theta_t)(\theta_t - c_i)(\theta_t - c_i)'}{\sum_{t=1}^{n} m(c_i|\theta_t) w(\theta_t)} \\ \pi_i &\propto \sum_{t=1}^{n} m(c_i|\theta_t) w(\theta_t). \end{aligned}$$

5. The number of clusters is chosen with the BIC criterion given a maximum number (5 in our examples).

We notice that the covariance matrices V_i are only estimated once, after convergence. k-means type algorithms also differ from the EM algorithm in that they do not evaluate the likelihood $p(\theta|c_1, ..., \pi_1, V_1, ...)$. This sub-optimal use of information in fact turns out to be a great advantage for our purposes. Fewer iterations than for EM are needed for convergence, and each iteration is faster. Even more importantly, the algorithm does not get stuck in the small degenerate clusters caused by rejections in the sense that, unlike for the EM algorithm with freely estimated covariances, these small clusters are not absorbing. If k-harmonic means does find a degenerate cluster, this causes no trouble for convergence, and after convergence we can use a predefined matrix in place of any non-positive-definite covariance matrix (for example, if V_i is not positive definite we set it to $0.5^2 Var(\theta)$). If desired, the mixture parameters can be refined with a few steps of the EM algorithm. In this case, we recommend not updating the the covariance matrices for the reasons just discussed.

B Appendix: Computing the marginal likelihood for the semiparametric Gaussian model

In the semiparametric example of section 6.2 we wish to efficiently compute the likelihood

$$y|\theta \sim N(\mathbf{0}, \sigma_\epsilon^2 I + \widetilde{Z} V_{\widetilde{\gamma}}(\tau) \widetilde{Z}'). \tag{11}$$

The employ the *QR decomposition* to evaluate $p(y|\theta)$ quickly. The QR decomposition of a $n \times k$ matrix X returns a $n \times n$ matrix Q and a $k \times k$ matrix R such that $Q'X = (R', 0')'$ and $Q'Q = I$. The decomposition is numerically stable and is quickly performed even for n in the several thousands, and it needs computing only once. From (8)

$$Q'y = Q'\widetilde{Z}\widetilde{\gamma} + Q'\epsilon,$$

where $\widetilde{\epsilon} = Q'\epsilon$ has covariance $\sigma_\epsilon^2 I$ since $Q'Q = I$. We can therefore write

$$\begin{bmatrix} \widetilde{y}_1 \\ \widetilde{y}_2 \end{bmatrix} = \begin{bmatrix} R\widetilde{\gamma} \\ 0 \end{bmatrix} + \begin{bmatrix} \widetilde{\epsilon}_1 \\ \widetilde{\epsilon}_2 \end{bmatrix}$$

and evaluate $p(y|\theta)$ as

$$p(y|\theta) = \exp(-0.5\widetilde{y}_2'\widetilde{y}_2\sigma_\epsilon^{-2})\sigma_\epsilon^{-0.5(n-k)} \exp(-0.5\widetilde{y}_1' V_1^{-1}(\tau)\widetilde{y}_1)|V_1(\tau)|^{-0.5}, \tag{12}$$

and k is the dimension of $\widetilde{\gamma}$, and where

$$V_1(\tau) = \sigma_\epsilon^2 I + V_{\widetilde{\gamma}}(\tau).$$

The evaluation of (12) requires computing the determinant of a $k \times k$ matrix and therefore has computational complexity $O(nk)$.

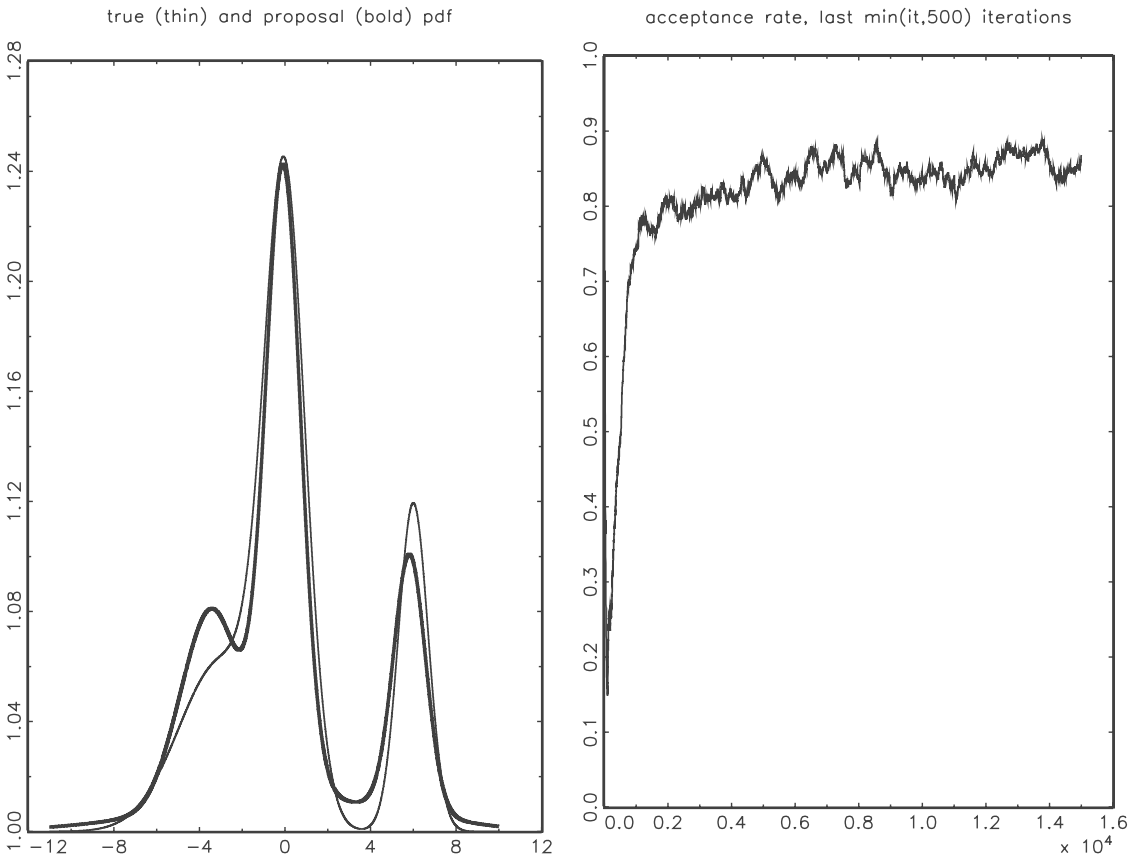

Figure 1: Left panel: Proposal distribution after 15 000 iterations, initialized with a normal $\phi(z; -5, 4)$. The target density is a univariate mixture $0.5\phi(z; 0, 1) + 0.3\phi(z; -3, 4) + 0.2\phi(z; 6, 0.5)$. Right panel: Recursive updates of the acceptance rate in the last 500 iterations.

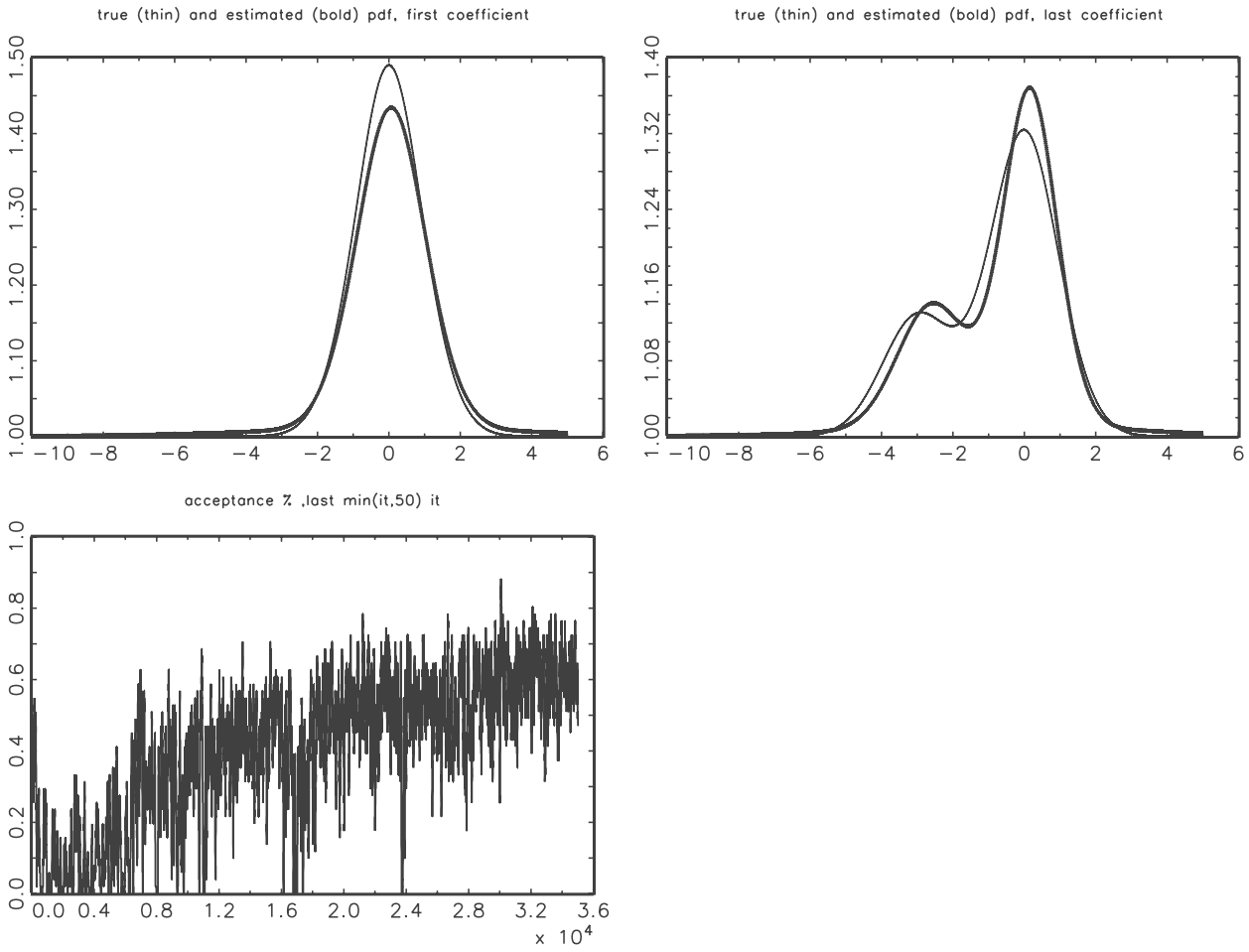

Figure 2: Proposal distribution after 35 000 iterations. The target distribution is a 15-dimensional mixture. The graph plots the true marginal distributions for the first and last variable together with the corresponding marginal distributions implied by the mixture of normals estimated on the full history of the draws, and with recursive updates of the acceptance rate in the last 500 iterations.

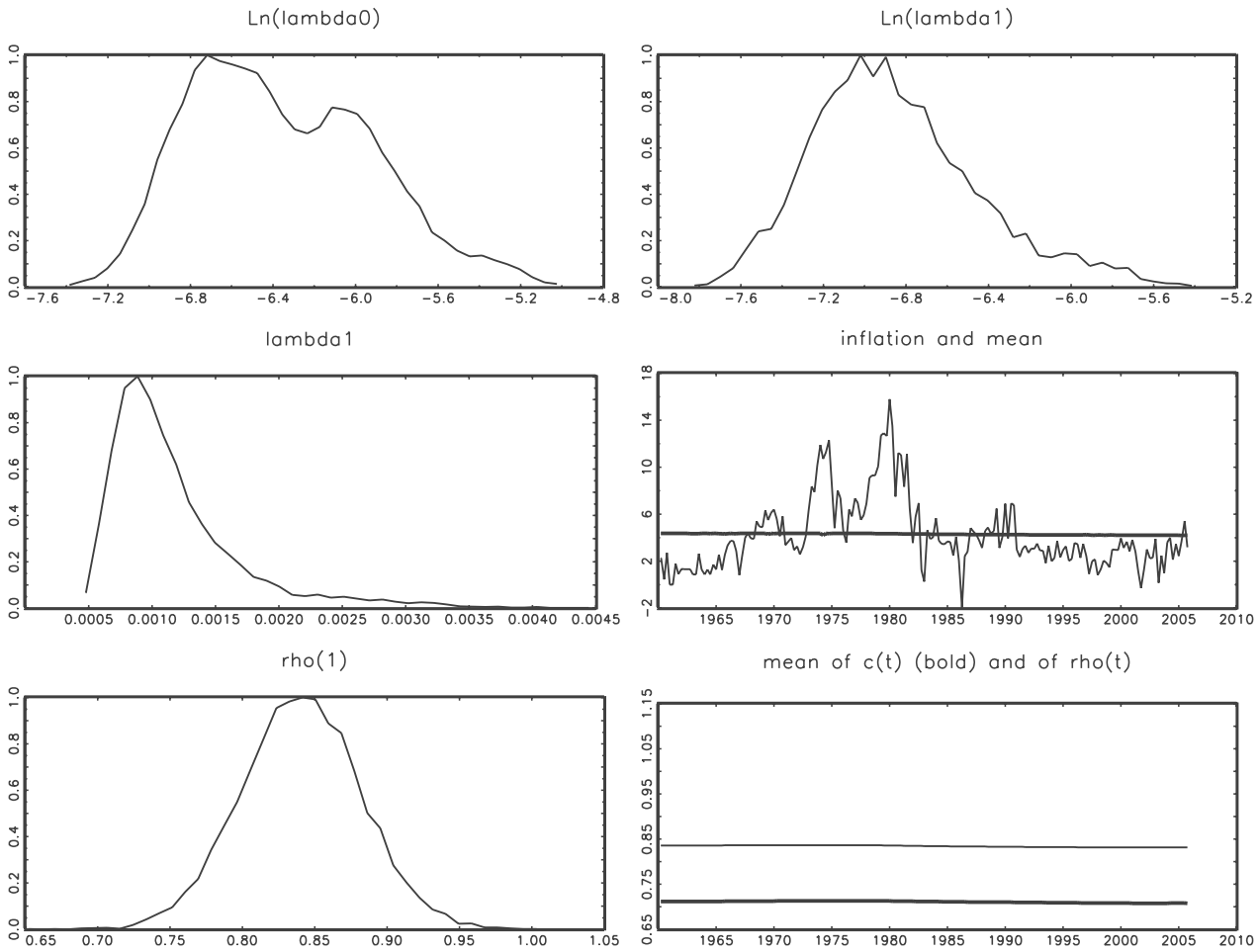

Figure 3: Inference for a time varying parameter AR(1) model for US inflation by Gibbs sampling. (a) marginal distribution of $\ln(\lambda_0)$ (b) marginal distribution of $\ln(\lambda_1)$ (c) marginal distribution λ_1 (d) inflation plot and mean, estimated as $E[(c_t/1-\rho_t)|y]$ (e) marginal distribution of $\rho_0|y$ (f) $E(c_t|y)$ (bold line) and $E(\rho_t|y)$.

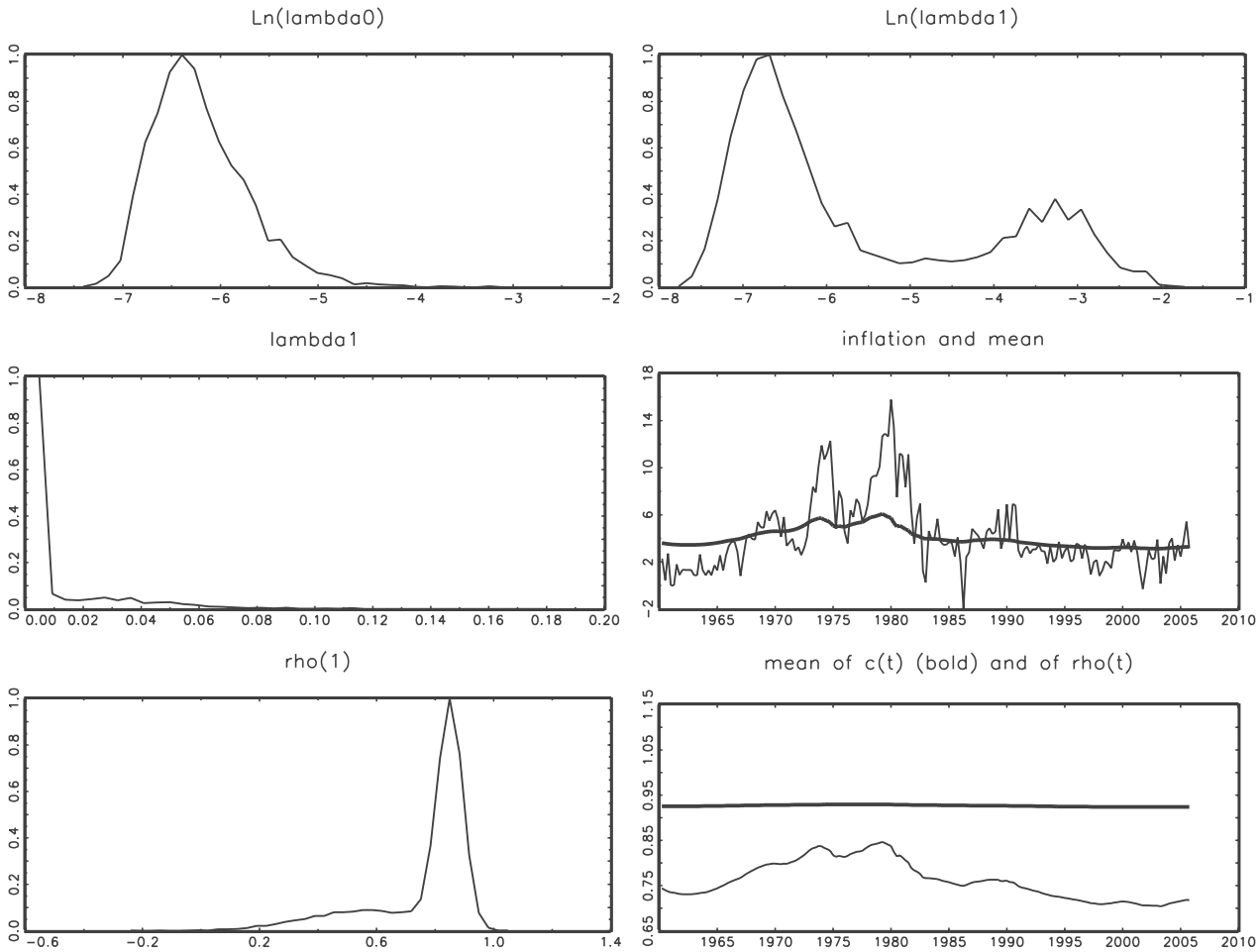

Figure 4: Inference for the model of Figure 3 by adaptive IMH. The interpretation of the panels is the same as in figure 3.

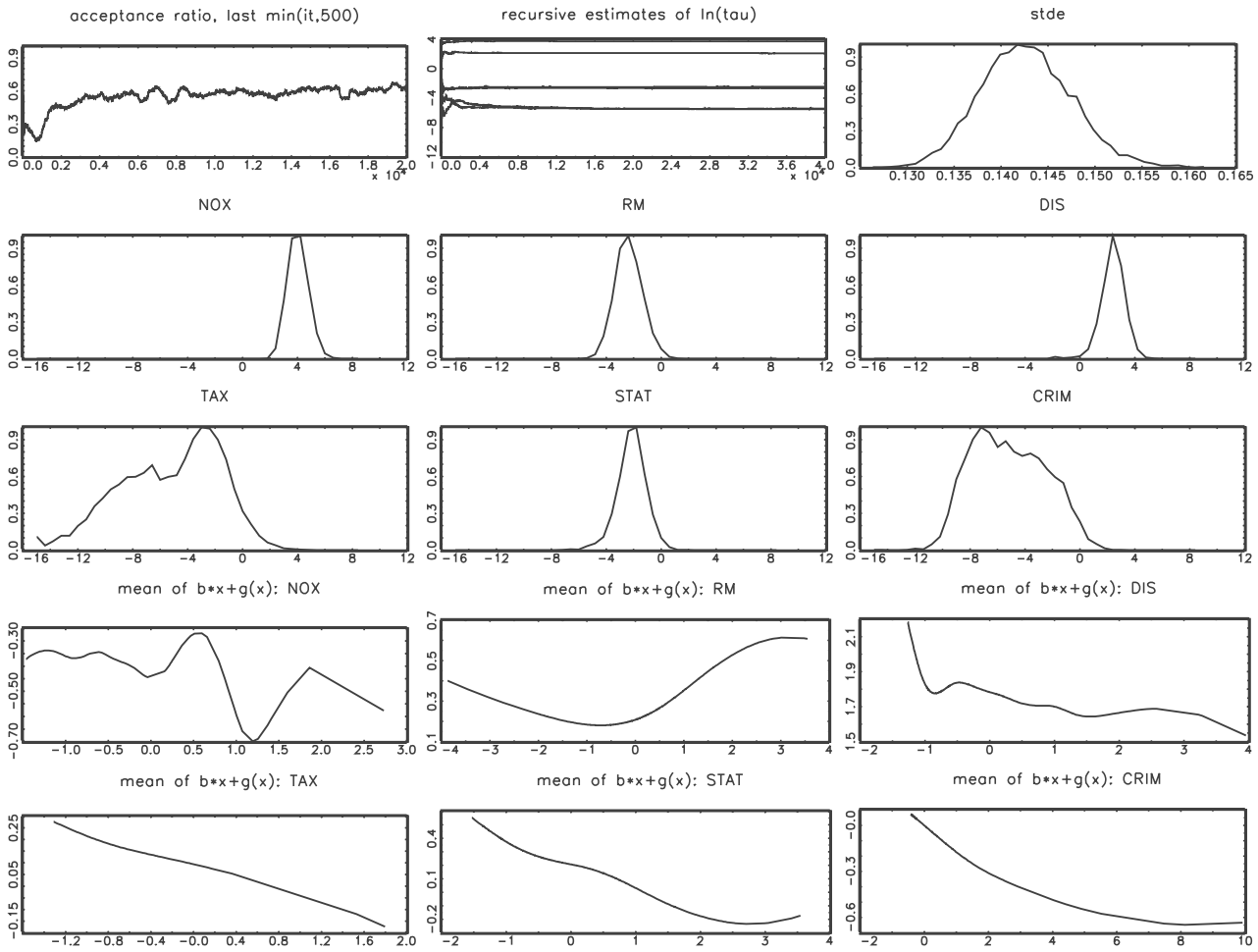

Figure 5: Inference for semiparametric model of housing prices by adaptive IMH. First row: recursive acceptance rate for the last min(it,500) iterations, recursive means of $\ln(\tau_i)$, marginal of σ_ϵ. Second and third rows: marginals of $\ln(\tau_i)$. Fourth and fifth rows: means of $\beta_i x + g_i(x)$.

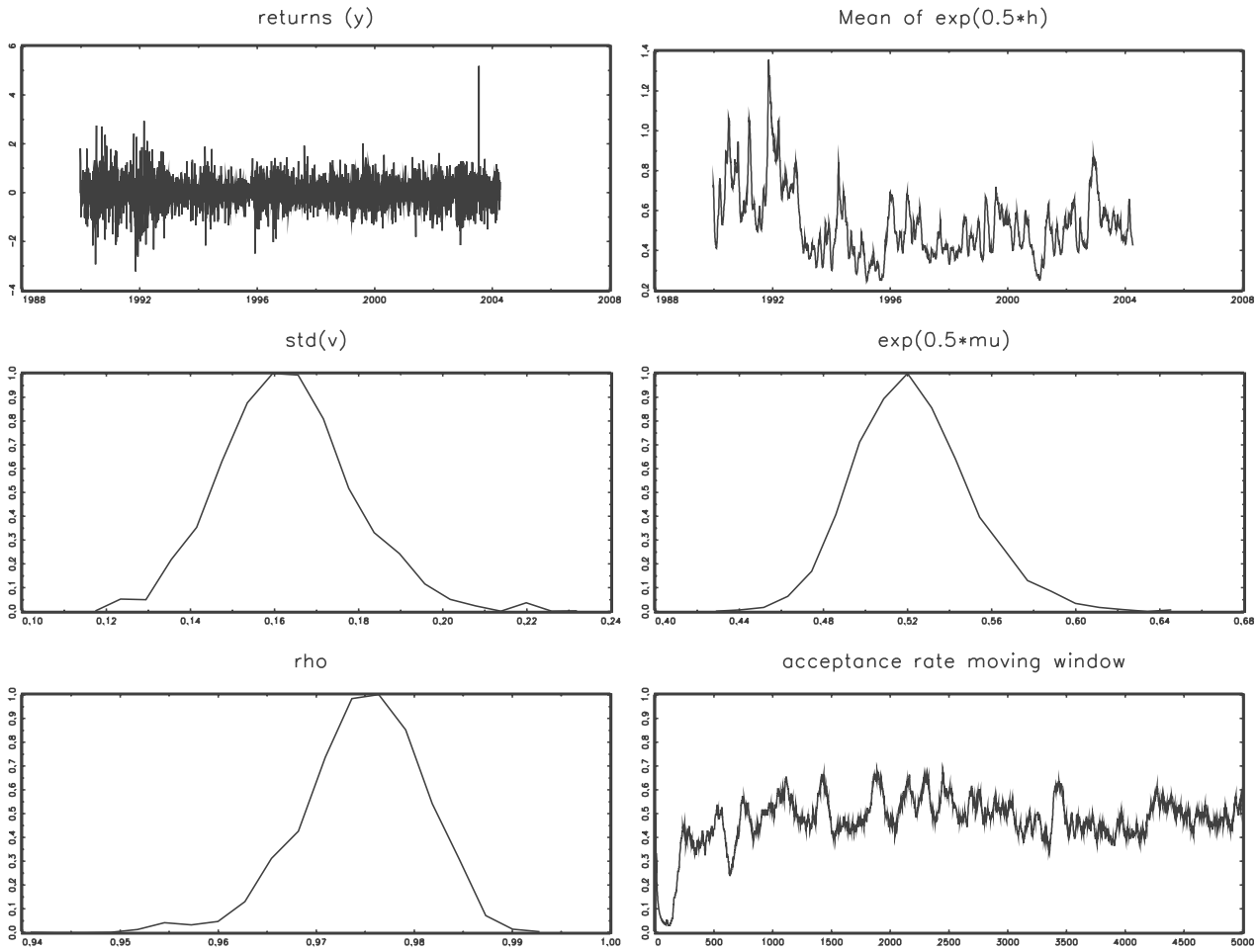

Figure 6: Inference for the daily US-GBP exchange rate by AIMH. (a) exchange rate returns (b) mean of $0.5\ln(h_t)$ (c) marginal of σ_v (d) marginal of $0.5\exp(\mu)$ (e) marginal of ρ (f) moving window of the acceptance rate for the last min(it,500) iterations.