Visualizations for Assessing Convergence and Mixing of Markov Chain Monte Carlo Simulations

Jaakko Peltonen [*], Jarkko Venna [1], Samuel Kaski

Helsinki Institute for Information Technology and Adaptive Informatics Research Centre, Department of Information and Computer Science, Helsinki University of Technology, P.O. Box 5400, FI-02015 TKK, Finland

Abstract

Bayesian inference often requires approximating the posterior distribution by Markov chain Monte Carlo sampling. The samples come from the true distribution only after the simulation has converged, which makes detecting convergence a central problem. Commonly, several simulation chains are started from different points, and their overlap is used as a measure of convergence. Convergence measures cannot tell the cause of convergence problems; it is suggested that complementing them with proper visualization will help. A novel connection is pointed out: linear discriminant analysis (LDA) minimizes the overlap of the simulation chains measured by a common multivariate convergence measure. LDA is thus justified for visualizing convergence. However, LDA makes restrictive assumptions about the chains, which can be relaxed by a recent extension called discriminative component analysis (DCA). Lastly, methods are introduced for unidentifiable models and model families with variable number of parameters, where straightforward visualization in the parameter space is not feasible.

Key words: multiple chains, linear discriminant analysis, potential scale reduction factor, posterior predictive distribution, convergence measures, dimensionality reduction

[*] Corresponding author. Tel. +358-9-4514429, Fax. +358-9-4513277
Email addresses: jaakko.peltonen@tkk.fi (Jaakko Peltonen), jarkko.venna@numos.fi (Jarkko Venna), samuel.kaski@tkk.fi (Samuel Kaski).

[1] Currently at Numos Ltd.

1 Introduction

Bayesian inference makes very accurate predictions possible, and provides rigorous methods for model selection and complexity control. In a nutshell, the uncertainty in the data is converted into uncertainty of the model parameters, in the form of a posterior distribution. Inference of parameter values and of predictions is then made based on this distribution.

Although Bayesian inference is potentially very powerful, closed-form solutions are seldom available. Inference has to be based on either sophisticated approximation methods or simulations with Markov chain Monte Carlo (MCMC, [9]) sampling. MCMC sampling is a very versatile yet computationally intensive procedure, which produces samples of parameter values from the posterior distribution. The main practical problem in MCMC simulations is how to assess whether they have converged. The resulting samples come from the true distribution only after convergence.

There are several strategies for monitoring convergence of MCMC sampling [2]. A common approach is to start the simulation from several different initial conditions, and measure when the different simulation chains become sufficiently mixed together. However, a measurement by itself only indicates whether there are problems with the simulations; if there are, the analyst must still discover what is preventing convergence, and resolve the problems by tuning simulation parameters, for instance. Finding the cause of the problems by inspecting the parameter values can be practically impossible for high-dimensional simulations.

In this paper we introduce techniques for visualizing convergence. The visualizations yield low-dimensional overviews that aim to reveal differences between the chains. Such differences are indications of where and how the chains did not mix. The visualizations reduce the burden of analyzing convergence to a manageable level, and complement traditional convergence monitoring.

We start by giving a brief introduction to measuring convergence with two common variance-based measures: the potential scale reduction factor (PSRF) and its multivariate version. We also suggest an information-theoretic alternative, mutual information. Next, we introduce two methods for convergence visualization: *linear discriminant analysis* (LDA) is based on the PSRF measures whereas *discriminative component analysis* (DCA) is based on mutual information; the latter method is theoretically more sound but computationally more complex. We test both methods in a detailed case study. Lastly, we consider more complicated MCMC simulations where LDA and DCA cannot be used directly (due to unidentifiability or changing dimensionality of posterior models); we show two ways to extend the visualization methods and present case studies of both approaches.

This paper extends our previous conference paper [29]. The major additions are the extension to unidentifiable and variable-dimensional model families in Section 5, two new case studies in the same Section, and a theoretical justification in Appendix C.

2 Measuring convergence

We use the following notation: c denotes an MCMC simulation chain. For a posterior sample (model of the data) in such a chain, s and $\mathbf{s}$ denote, respectively, the univariate or multivariate parameters of the sample, and $\breve{s}$ denotes the sample without specifying how it is parameterized. Lastly, $\mathbf{x}$ denotes a posterior predictive sample, that is, a (multivariate) vector in the data space.

2.1 Univariate PSRF

One of the most common quantitative measures for monitoring MCMC convergence is the potential scale reduction factor (PSRF) proposed by Gelman and Rubin [8]. Multiple MCMC sequences are started from different (overdispersed) initial points. At convergence the chains should come from the same distribution, which is assessed by comparing the variance and mean of each chain to those of the combined chain.

The PSRF is defined for one-dimensional data as follows. A number (m) of parallel chains are started, with $2n$ samples each. Only the last n (potentially better converged) samples from each chain are used.

The between-chain variance B/n and pooled within-chain variance W are defined by

$$\frac{B}{n} = \frac{1}{m-1}\sum_{j=1}^{m}\left(\bar{s}_{j\cdot} - \bar{s}_{\cdot\cdot}\right)^2 \quad \text{and} \tag{1}$$

$$W = \frac{1}{m(n-1)}\sum_{j=1}^{m}\sum_{t=1}^{n}\left(s_{jt} - \bar{s}_{j\cdot}\right)^2 \ , \tag{2}$$

where s_{jt} is the parameter value of the t:th sample in the jth chain, $\bar{s}_{j\cdot}$ is the mean of the samples in chain j and $\bar{s}_{\cdot\cdot}$ is the mean of the combined chains.

By taking the sampling variability of the combined mean into account we get a pooled estimate for the posterior variance

$$\hat{V} = \frac{n-1}{n} W + \left(1 + \frac{1}{m}\right) \frac{B}{n}. \tag{3}$$

Finally an estimate $\hat{R}$ of PSRF is obtained by dividing the pooled posterior variance estimate with the pooled within chain variance,

$$\hat{R} = \frac{\hat{V}}{W} = \frac{n-1}{n} + \left(1 + \frac{1}{m}\right) \frac{B}{nW}\ . \tag{4}$$

If the chains have converged, the PSRF is close to one, which makes it a useful indicator of convergence. It does not guarantee convergence, however. The chains might not have traveled the whole state space yet and might still be able to discover possible new areas of high probability if sampling were continued.

2.2 Multivariate PSRF

One weakness of the PSRF measure is that it is only applicable to one variable at a time. Brooks and Gelman [1] have extended it to a multivariate version, MPSRF. It is defined, similarly to the univariate PSRF, based on the estimated posterior covariance matrix $\hat{\mathbf{V}}$, which we get from (3) by replacing the scalar variances B/n and W with the corresponding covariance matrices.

In the multivariate case the comparison of within-chain variance to the pooled variance requires comparing the matrices. Brooks and Gelman summarized the comparison by a maximum root statistic which gives the maximum scale reduction factor of any linear projection of $\mathbf{s}$. The estimate $\hat{R}^p$ of MPSRF is defined by

$$\begin{aligned}
\hat{R}^p &= \max_{\mathbf{a}} \frac{\mathbf{a}^T \hat{\mathbf{V}} \mathbf{a}}{\mathbf{a}^T \mathbf{W} \mathbf{a}} && (5)\\
&= \frac{n-1}{n} + \left(1 + \frac{1}{m}\right) \max_{\mathbf{a}} \frac{\mathbf{a}^T \mathbf{B} \mathbf{a}/n}{\mathbf{a}^T \mathbf{W} \mathbf{a}} && (6)\\
&= \frac{n-1}{n} + \left(1 + \frac{1}{m}\right) \lambda_1, && (7)
\end{aligned}$$

where the λ_1 is the largest eigenvalue of the matrix $\mathbf{W}^{-1}\mathbf{B}/n$.

Implementations of PSRF and MPSRF are available in software packages such as [23].

2.3 Mutual information

Brooks and Gelman [1] noted that any statistic calculated from the separate chains should equal the one calculated from the combined chain when the chains have reached convergence, as the distributions should then be the same. PSRF and MPSRF compare means and covariances. We propose that instead of comparing a statistic, a more general measure would result from comparing the distributions themselves.

Appendix A shows that comparing distributions of posterior samples $\breve{s}$ in two MCMC chains results in measuring the mutual information between the samples and indices c that tell which chain each sample is from. For multiple chains we consider the straightforward generalization, the mutual information $I(c, \breve{s})$ between posterior samples $\breve{s}$ and their chain indices c. $I(c, \breve{s})$ tells how much chains differ: it is nonnegative and is zero if and only if all chains generate the same distribution of models. It would be suitable for a convergence measure.

In Section 3.3 we present a convergence visualization method based on a practically computable measure related to $I(c, \breve{s})$; we further extend the method in Sections 5.1 and 5.3. We also use mutual information in Appendices A–C to prove theoretical connections between our methods.

3 Visualizing convergence

The convergence measures discussed in Section 2 cannot tell *why* simulations did not converge and in some cases they might even be fooled to falsely indicate convergence. It is therefore common practice to complement the measures with *visualizations* of the MCMC chains. The visualizations help assess convergence in detail, and help analyze reasons of convergence problems.

3.1 Current practice

MCMC chains have traditionally been visualized in three ways. Each parameter of the posterior samples can be plotted as a separate time series, or the marginal distributions can be visualized as histograms. The third option is a scatter or contour plot of two parameters at a time, possibly showing the trajectory of the chain on the projection. The obvious problem with these visualizations is that they do not scale up to large models having numerous parameters. The number of displays would be large, and it would be hard to grasp the underlying high-dimensional relationships of the chains based on the component-wise displays.

Some new methods have been suggested. Advanced computer graphics methods can be used to visualize the shape of a three-dimensional distribution [30]. Alternatively, if the outputs of the models can be visualized in an intuitive way, an animation of the MCMC chain can be created whose frames are visualizations of the individual samples [13]. However, these visualizations are applicable only to special models.

Some dimension-free methods can show a large number of variables in a few plots. A parallel coordinate plot (PCP) shows each variable as a column, and each sample as a piecewise linear curve connecting the columns. However, a PCP shows interactions only between adjacent columns (variables); therefore, when visualizing MCMC sampling in a high-dimensional parameter space, some interactions between the parameters may be lost. A generalized association plot (GAP; [6]) visualizes a sequence of matrices, from a correlation matrix to higher-order matrices containing correlations of correlations. A GAP could be used for example to find a clustering of all MCMC samples; a clustering that corresponds to the chains would indicate non-convergence. However, a GAP would try to show all cluster structure, whereas in convergence visualization the goal is to find differences between chains and not structure within chains. Overall, such dimension-free methods can be used in convergence analysis but they do not fully answer the needs of the analyst.

The fundamental problem with the usual visualization methods is that they lack the means to focus on visualizing variables or dimensions that are relevant for convergence. This worsens the problems caused by the required large number of plots.

Note that with specific model families, expert knowledge about the model may be available, and thus it may be possible to summarize differences between chains based on known-to-be-useful comparison metrics as in [16] for phylogenetic trees. However, such knowledge is not always available for the models of interest; moreover, in case of non-convergence, detailed analysis using more complicated visualizations could still be needed to find the reasons for the convergence problems.

3.2 Principled visualization: MPSRF and LDA

We next show that the MPSRF criterion in Section 2.2 is very closely related to linear discriminant analysis (LDA; see [27] for a definition).

The goal of (a one-dimensional) LDA is to find the linear transformation $y = \mathbf{a}^T\mathbf{s}$ that maximizes the variance between classes, relative to the variance within classes. Here $\mathbf{s}$ is the multivariate data vector and $\mathbf{a}$ contains the parameters of the transformation. More formally, LDA solves the problem

$$\max_{\mathbf{a}} \frac{\mathbf{a}^T \mathbf{B}_{ss} \mathbf{a}}{\mathbf{a}^T \mathbf{W}_{ss} \mathbf{a}} , \tag{8}$$

where $\mathbf{B}_{ss}$ and $\mathbf{W}_{ss}$ are the between class and within class sum of squares and cross products (SSCP) matrices which differ only by a constant scale from the corresponding covariance matrices. This is a generalized eigenvalue problem, and its solution $\mathbf{a}$ is the eigenvector corresponding to the largest eigenvalue of $\mathbf{W}_{ss}^{-1}\mathbf{B}_{ss}$.

Hence, disregarding the constants, MPSRF (6) equals the cost function of (a one-dimensional) LDA where the MCMC chains are the classes. In other words, optimizing LDA is equivalent to choosing the component that best detects convergence, in the sense of MPSRF. Monitoring convergence by either MPSRF or the LDA cost function is equivalent; if the chains can be discriminated, then they have not converged.

There is no reason to make only one-dimensional visualizations. LDA chooses the second direction or projection axis to be the eigenvector corresponding to the second largest eigenvalue, etc. A K-dimensional LDA then maximizes $\sum_{k=1}^{K} \lambda_k$, the relative between-chain variance representable by the K directions together. For any choice of dimensionality $K \geq 1$, LDA shows as much as possible according to its optimization criteria, so it always shows at least part of the convergence problems.

The more LDA directions are used, the more differences between chains are visualized; it can be shown that K directions are sufficient for discriminating all differences of $K+1$ chains under the LDA assumptions (this follows from the proof of [18] with $d = K$), but that upper limit can be too large in practice. In principle K could be chosen by fitting a model like [14] that tries to discriminate chains (i.e., predict the chain index of a sample from its location); models like [14] have internal methods for choosing K, and more generally for instance Bayesian inference or cross-validation could be applied to choose K for a model. However, for practical visualization a fixed small limit $K \leq 3$ can simply be taken, so that projection to all the LDA directions can be visualized in a single plot which is optimized to show as much of the convergence problems as possible. In the next subsection we will extend LDA to analyze convergence even better with low-dimensional visualizations.

In convergence analysis LDA should typically be used together with a convergence measure such as MPSRF, so that when the measure indicates nonconvergence, the visualization is used to analyze the reason. The number of MCMC chains is typically chosen for the needs of the convergence measure; since the LDA cost function is equivalent to MPSRF, the number of chains is good for LDA when it is good for MPSRF.

In summary, we suggest using LDA for both monitoring and visualizing convergence of MCMC. When LDA is used to visualize MCMC convergence we in effect try to find a linear transformation that visualizes the convergence problems as clearly as possible, in the sense of the (extended) MPSRF measure.

3.3 Extending LDA for principled visualization

Although LDA (or MPSRF) is a good starting point, it has theoretical weaknesses which motivate the need for more principled convergence visualization. Here we discuss the problems, present a recently introduced extension which solves them, and discuss the connection between LDA and the extension.

Problems with LDA. Like the PSRF and MPSRF measures, LDA compares only means and covariances. LDA assumes that each class is normally distributed with the same covariance matrix. If the assumptions are correct, LDA discriminates optimally between two classes. However, this does not hold in general.

Another problem surfaces when generalizing LDA to several classes. The objective considers only pairwise divergences between classes, and no longer results in optimal discrimination. See Appendix B for details.

In the specific case of MCMC convergence visualization, the posterior distributions being simulated are often not Gaussian; even if they were, the MCMC chains do not have the same covariance matrix before convergence. Therefore, LDA is suboptimal for convergence visualization.

Discriminative component analysis. To address the above problems, we suggest to complement LDA-based analysis with a recently developed generalization of LDA. The projection remains linear but the assumptions about the distribution of data are relaxed.

The recently developed generalization finds *discriminative* or *relevant* components by directly maximizing their class-prediction power [18]. Formally, the conditional (log) likelihood

$$L = \sum_{(\mathbf{s},c)} \log p(c|\mathbf{W}^T\mathbf{s}) \tag{9}$$

of classes is maximized within the subspace formed by the components. Here $\mathbf{s}$ is the sample, c is its class, $\mathbf{W}$ is the projection matrix whose columns are the component directions, and p is the estimated probability for the class; in this paper we use a nonparametric estimate (see “Technical note” below for details). The task of finding such components can be called *discriminative component analysis* (DCA). A sketch of the connection between LDA and DCA is presented in Appendix B.

In this paper the c are the indices of the different chains, the $\mathbf{s}$ are the model parameters of the posterior samples, and DCA maximizes the (log) likelihood of correctly guessing which MCMC chain each sample is from. For converged chains one cannot (asymptotically) do better than a random guess; hence, large likelihood indicates the chains have not converged, which can be assessed visually from the DCA projection.

Maximizing the likelihood is asymptotically equivalent to maximizing the mutual information between the (projected) posterior sample parameters and their chain indices; therefore DCA asymptotically stems from the mutual information convergence measure of Section 2.3 in the same way as LDA can be regarded to stem from the MPSRF measure of Section 2.2.

Technical note. With a finite number of samples, we do not know the exact densities $p(c|\mathbf{W}^T\mathbf{s})$, but the projection parameters can be optimized by an estimate. Here we use a nonparametric estimate, without needing to make distributional assumptions. The nonparametric estimate is a Parzen window estimate with a Gaussian window, written as

$$p(c|\mathbf{W}^T\mathbf{s}) = \frac{\sum_{(\mathbf{s}',c')|c'=c} \exp(-||\mathbf{W}^T\mathbf{s} - \mathbf{W}^T\mathbf{s}'||^2/2\sigma^2)}{\sum_{(\mathbf{s}',c')} \exp(-||\mathbf{W}^T\mathbf{s} - \mathbf{W}^T\mathbf{s}'||^2/2\sigma^2)} \quad (10)$$

where the sum in the nominator goes over samples in chain c, the sum in the denominator goes over samples in all chains, and σ is a smoothing parameter. This is a consistent estimator of the conditional density under mild conditions on the choice of σ. For further details on nonparametric DCA and its optimization, see [18]. In this paper we use conjugate (batch) gradient optimization instead of the original stochastic gradient.

It is also possible to use mixture model-based estimation in DCA in place of the nonparametric estimation, as described in [17]. This could be called semiparametric DCA; it can yield faster computation but the estimation can be less accurate for complicated data sets. In this paper we will use the nonparametric estimation for DCA.

Summary. DCA removes two weaknesses of LDA: restrictive assumptions about the distributions and suboptimality for multiple chains. As a result, DCA is able to produce improved visualizations. The disadvantage of nonparametric DCA is the laborious iterative computation. Whether to use LDA, nonparametric DCA, or semiparametric DCA thus depends on how much computation time is available; if there is enough, nonparametric DCA should be used.

It remains an empirical question how much DCA improves the LDA-based visualizations. In [18], nonparametric DCA attained better results than LDA in benchmark comparisons; similarly, in [17] semiparametric DCA outperformed LDA. In Sections 4.1 and 4.2 we apply LDA and nonparametric DCA to assess convergence in a relatively simple task.

Note that both LDA and DCA visualize convergence by finding linear projections that discriminate the MCMC chains. In principle, nonlinear projections could be made based on the same principles that would discriminate the chains even better; however, from nonlinear visualizations it could be hard to analyze the cause of the differences. By contrast, in linear visualizations each coordinate axis corresponds to a weighted sum of original coordinates (model parameters). It is then easy to interpret the visualization; for example, if there are interesting differences between chains along one axis, one can identify which parameters contribute most to those differences simply by checking the weights of the projection matrix.

For both LDA and DCA, the analyst chooses what samples to visualize. For instance, to avoid effects of dependencies between samples within the MCMC chains, one can as usual take only every k:th sample from each chain where k is some appropriate interval; or one can visualize all samples, in which case our methods visualize all differences between chains including possible differences caused by between-sample dependencies.

4 Case study 1: analysis of an MCMC run

To demonstrate visual analysis of an MCMC sampler we have chosen a data set that contains reaction times for schizophrenics and nonschizophrenics. The model and the problem are described in [7] (in Example 16.4, p. 426 of the book) and were also used to illustrate the use of the PSRF measure in the original article [8].

The data are (log) reaction time measurements from 11 nonschizophrenics and 6 schizophrenics. Each person had their reaction time measured 30 times. It is believed that schizophrenics suffer from attentional deficit on some measurements, as well as an overall motor reflex retardation.

For the nonschitzophrenics the reaction time is modeled as a random-effects model with a distinct mean α_j for each person and a common variance σ_y^2. The reaction times for the schizophrenics are modeled with a two-component Gaussian mixture. With probability $(1 - \lambda)$ there is no attention lapse and the response time has mean α_j and variance σ_y^2. With probability λ there is a delay and the response time has mean $\alpha_j + \tau$ and the same variance σ_y^2. To address the question about the amount of motor reflex retardation a hierarchical population model is devised. The means of the reaction times α_j are modeled to be normally distributed with a mean μ for the nonschizophrenics and a mean $\mu + \beta$ for the schizophrenics, and a common variance σ^2.

The hyperparameters are assigned a noninformative uniform prior. The mixture parameter λ is restricted to the interval $[0.001, 0.999]$, and τ and the variance parameters are restricted to be positive. All necessary conditional distributions were readily available, so Gibbs sampling was used.

Ten chains of 1500 samples each were generated from random starting positions. The MPSRF showed that the sampling had not converged. We calculated the univariate PSRF measures for the 23 variables that we were interested in; the PSRF values showed that several variables had not converged. At this point we still had no idea what had gone wrong with the sampler, or whether the convergence was just slow.

4.1 Visualization with LDA

Gaining insight into the problem. In order to understand the behavior of the chains better, we visualized a part of the simulation. More precisely, we visualized an interval (samples $[200, 600]$) around the point 350, since after the point 350 the MPSRF seemed to have stabilized at a high value.

Note that the purpose of these kinds of convergence visualizations is not to replace convergence measurement methods like MPSRF but to complement them. Visualization can be useful even in cases where it is already known that chains have not converged, as in this case study. Visualization shows differences between chains which are clues to the non-convergence. Visualization then helps quickly find out and correct the cause of convergence problems, which a single convergence measure cannot reveal.

The LDA projection (Fig. 1**a**) shows that there are five distinct clusters. The chains were easy to identify after they were color-coded. Six chains were clustered together and the other four formed a separate cluster each. Three chains were separated from the main cluster on discriminative component 1 and one on component 2. We also checked whether any of the separate chains could still be converging toward the common cluster, by color coding the sampling time. Drift of the distribution should then be visible as a “tail” of gradually changing color; there was no visible hint of that.

At this point, it is not known for certain whether there are differences among the the six chains that were clustered together (chains 1,3-7 in Fig. 1**a**). It could be that the distributions of those six chains are nearly identical, but it could

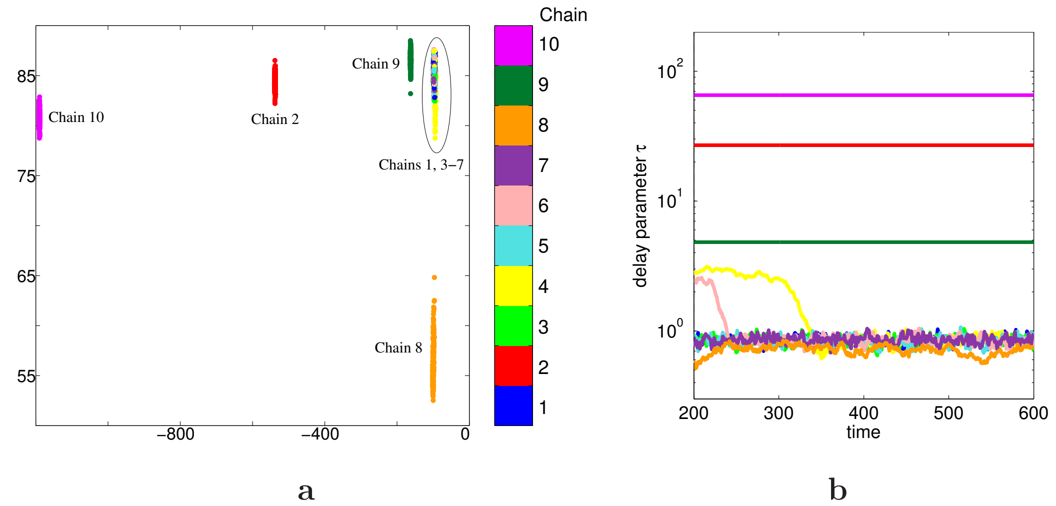

Fig. 1. Convergence visualization for MCMC simulations on the reaction time data. **a**) A two-dimensional LDA projection of all samples from the time interval $[200, 600]$ is shown as a scatterplot. Each chain has been given a unique color. The relative scaling of the axes is meaningful but the absolute values are not informative. **b**) A time series plot of the delay parameter τ for each chain in the same time interval. The three straight lines are degenerate chains 2, 9, and 10. A logarithmic scale is used for τ in order to show the evolution of the other seven chains.

also be that the two-dimensional LDA projection has focused on separating the chains 2, 8, 9, and 10 from each other and from the main cluster, instead of trying to show differences inside the main cluster. At the end of this section we will study this by a new LDA projection that focuses exclusively on the main cluster.

Verifying the findings. A study of the parameter values showed that four chains had quickly ended up in a degenerate part of the parameter space where the mixture model has collapsed to a one-component model. Means and standard deviations for five of the 23 parameters of interest are listed for all chains in Table 1. For three of these degenerate chains (chains 2, 9, and 10) the probability λ of the delayed component was so low that no samples were assigned to it. This was apparent already from the one-dimensional time series plots of these chains: The delay parameter τ had not changed from its starting position. Fig. 1**b** shows a time series plot of τ for all chains; the three chains where τ does not change are clearly visible as straight lines.

The fourth degenerate chain (chain 8 in the LDA visualization, Fig. 1**a**), was harder to diagnose because the time series plots of the chain looked normal, showing clear variation in the parameters. Fig. 2**a** shows three examples of time series plots for the chain. However, only a little additional work was needed to identify the actual problem: Nearly all samples were modeled as delayed measurements, and hence the parameters β and τ became linked. The comparatively high proportion of delayed measurements was easily visible by

Table 1
Mean values and standard deviations of the 10 chains for 5 of the 23 parameters of interest, under the time interval $[200, 600]$.

Chain	σ^2	β	λ	τ	σ_y^2
1	0.16 ± 0.04	0.33 ± 0.09	0.12 ± 0.03	0.85 ± 0.06	0.19 ± 0.01
2	0.19 ± 0.04	0.41 ± 0.11	0.01 ± 0.01	26.94 ± 0.00	0.24 ± 0.01
3	0.16 ± 0.04	0.31 ± 0.09	0.12 ± 0.03	0.85 ± 0.06	0.19 ± 0.01
4	0.38 ± 0.34	0.07 ± 0.43	0.21 ± 0.11	1.42 ± 0.86	0.21 ± 0.02
5	0.16 ± 0.04	0.31 ± 0.09	0.12 ± 0.03	0.84 ± 0.06	0.19 ± 0.01
6	0.19 ± 0.11	0.29 ± 0.14	0.13 ± 0.04	0.97 ± 0.40	0.19 ± 0.01
7	0.16 ± 0.04	0.33 ± 0.09	0.11 ± 0.02	0.86 ± 0.06	0.19 ± 0.01
8	0.26 ± 0.06	-0.16 ± 0.15	0.81 ± 0.04	0.71 ± 0.06	0.20 ± 0.01
9	0.18 ± 0.04	0.41 ± 0.09	0.01 ± 0.01	4.83 ± 0.00	0.24 ± 0.01
10	0.18 ± 0.04	0.41 ± 0.09	0.01 ± 0.01	65.36 ± 0.00	0.24 ± 0.01

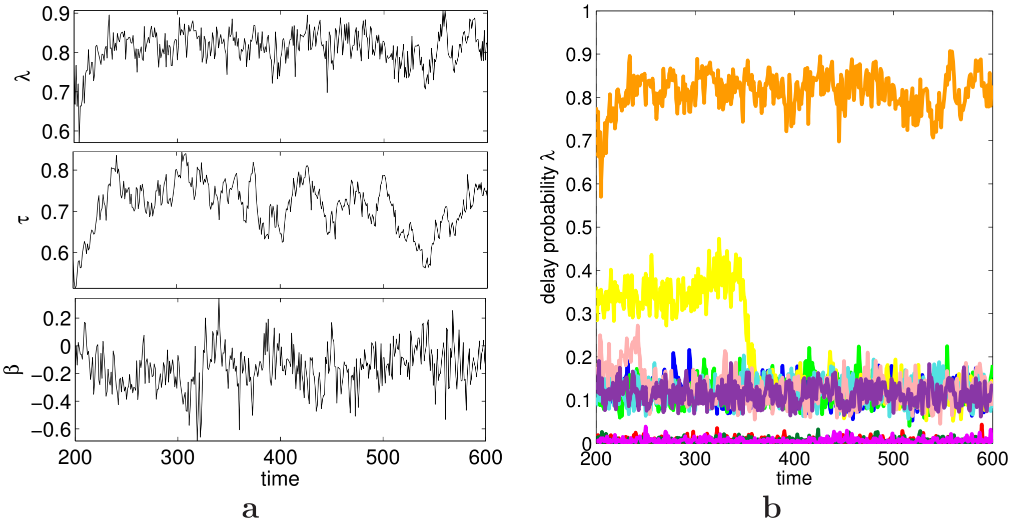

Fig. 2. Time series plots of the fourth degenerate MCMC chain (chain 8 in Fig. 1**a**) whose degeneracy required more investigation. **a**) Three example time series plots of the interval $[200, 600]$, showing the parameters λ, τ, and β. **b**) Time-series plot of λ for all chains (colors as in Fig. 1).

plotting the evolution of the delay probability λ for all chains in a single plot (Fig. 2**b**).

Note that for instance pairwise comparisons of chains using a convergence measure could also reveal which chains are degenerate. However, for large numbers of chains the number of required pairwise comparisons would become very large; moreover, comparison with a convergence measure cannot

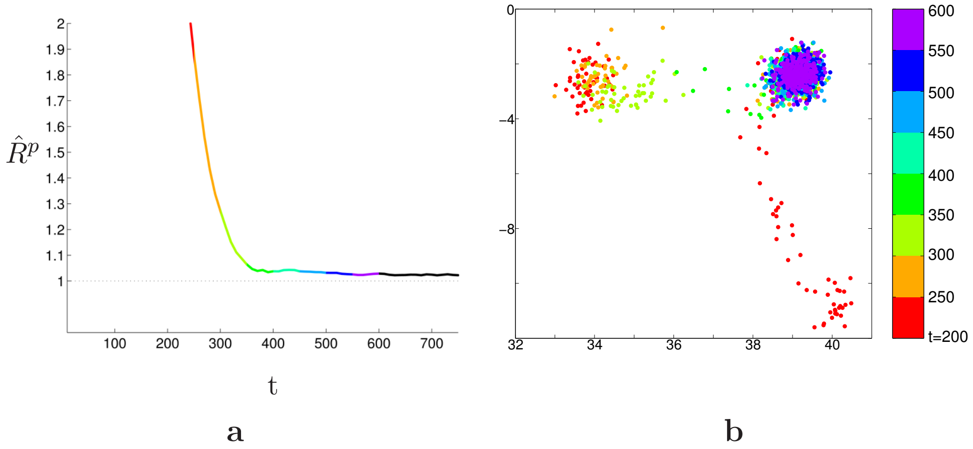

Fig. 3. Convergence visualization for MCMC simulations on the reaction time data, after removing degenerate chains. **a**) MPSRF measure calculated from the nondegenerate chains (1,3-7). **b**) LDA projection computed by applying LDA to the nondegenerate samples from the time interval $[200, 600]$. The colors are time-based, t denotes time. Each color covers 50 time steps.

reveal e.g. ongoing drift of the distributions without further comparisons. By contrast, the LDA visualization used here shows the main differences in a single display.

Checking the behavior of the sampler near convergence. At this point we could have modified our model or our sampler to remove the problems. A rapid alternative is to discard the degenerate chains, and we did that. We computed the MPSRF again for the remaining chains. It is clear from Fig. 3**a** that they have converged after about 350 samples. For a demonstration we created a new LDA projection of the nondegenerate chains only; that is, we optimized an LDA visualization to show differences between the six nondegenerate chains (1,3-7), by applying LDA to all samples of those six chains in the interval $[200, 600]$. In Fig. 3**b** we can see two “tails” from chains moving toward the common distribution. By color coding the samples based on time we verified that the tails were indeed early samples and that the two chains became combined with the rest after the early samples. Thus we conjecture that the simulation had converged this time.

4.2 Visualization with DCA

We finally compare qualitatively the projection given by nonparametric DCA with the ones given by LDA, to verify that DCA gives the same or better insights on convergence. We again take the samples in the interval $[200, 600]$ from all ten chains, but now we apply DCA instead of LDA to create a two-dimensional convergence visualization.

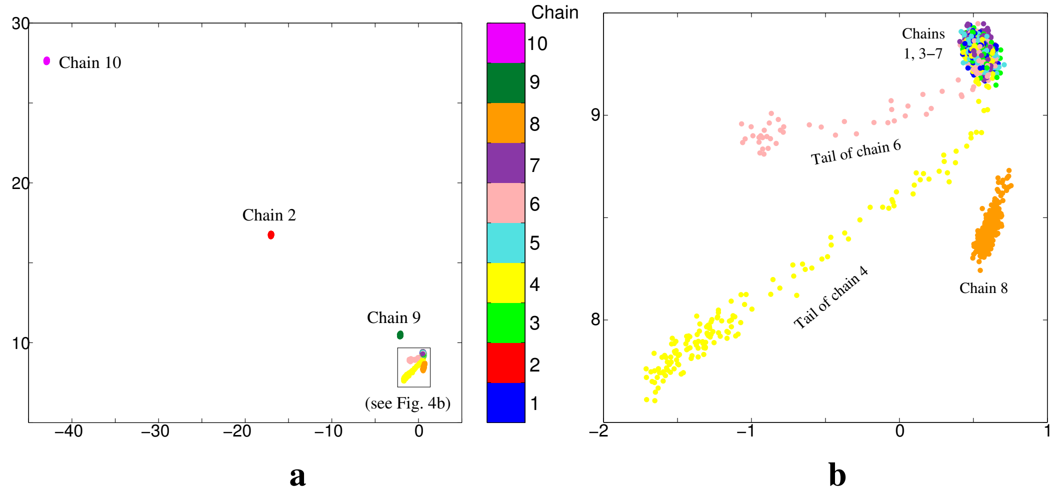

Fig. 4. Discriminative component analysis (DCA) based visualization of MCMC convergence on the reaction time data; a single DCA visualization of all ten chains shows the same information as two LDA visualizations (Figs. 1**a** and 3**b**). **a**) 2D nonparametric DCA projection of all samples from the time interval $[200, 600]$ from all chains. **b**) Enlarged view of the box in lower right corner of **a**.

The resulting DCA projection is shown in Fig. 4. Comparing the DCA projection to the corresponding LDA projection of all chains (Fig. 1**a**), we can see that DCA has discovered the same five clusters as LDA. Four clusters are composed of a single chain each: chains 2, 9, and 10 as seen in Fig. 4**a**, and chain 8 as seen in the enlarged view in Fig. 4**b**. The fifth cluster consists of six chains (chains 1, 3-7). Thus all of the information that we saw in Fig. 1**a** is also visible in the DCA visualization.

However, DCA has also shown additional information that was not visible in Fig. 1**a**: DCA shows the two "tails" of samples, generated by two chains converging toward the multi-chain cluster. These are the same "tails" that we previously found by creating a separate LDA visualization of the nondegenerate chains (Fig. 3**b**).

Comparing the DCA and LDA visualizations of all chains (Fig. 4 and Fig. 1**a**) in more detail, we see that chains 2 and 10 are far from the others in both the LDA and DCA visualizations. However, the LDA visualization kept chain 8 far apart as well, whereas DCA placed it closer to chains 1,3-7 and instead separated the "tails" of early samples of chains 4 and 6 which were not visible in the LDA visualization. Since chain 8 can still be discriminated well in the DCA visualization, the DCA projection is overall more informative than the LDA projection.

In conclusion, the DCA visualization displayed all the discovered convergence properties in a single two-dimensional visualization. No additional studies were required as with LDA. (Just in case, to check whether there could be still more to discover, we computed a DCA visualization of nondegenerate chains corresponding to Fig. 3**b**, and it revealed only the same properties. Thus the original single DCA visualization was enough to show all the information.)

5 Unidentifiable and variable-dimensional model families

LDA and DCA visualize the chains in the parameter space. However, there are useful model families, and MCMC simulations based on them, where LDA or DCA visualization in the parameter space is not possible or does not give good results; we discuss the main reasons, unidentifiability and variable dimensionality, and introduce visualization methods useful for such model families.

Unidentifiability. Label switching (unidentifiability of labels) causes the posterior distribution to have multiple modes, which are identical except for the order of the parameters. Examples are mixture models where the labeling between mixture components is unidentifiable. Such models are useful, so convergence visualization methods are needed for their MCMC simulations.

For unidentifiable models, parameter-space convergence criteria may be too strict. In simulations where label switching occurs, different chains may change modes seldom; when chains inhabit different modes, usual convergence measures indicate non-convergence and visual analysis mostly shows separate clusters. It could be argued that to reach convergence the simulation must go through all the modes, but it would be more useful to disregard the order of components when measuring convergence. One traditional way to handle label switching is to restrict parameter values of different components by a prior; however, developing suitable restrictions can be hard. Samples can also be postprocessed to fix the labeling [25], but this may be computationally demanding and software for it is not readily available.

Changing dimensionality. In some simulations the complexity or dimensionality of the model is varied during the run; for example one might use reversible jump MCMC [11] to cover models with different numbers of parameters. Such simulations can be hard to analyze: one cannot compute the MPSRF measure or the LDA and DCA visualizations, as they assume a fixed parameter space.

Proposed ideas for convergence monitoring include devising a function to monitor that is invariant to the model complexity. Convergence could also be studied within each fixed-dimensional subfamily separately and convergence of the model selection part with yet another method; the number of different models can be high though. The methods in [4] monitor convergence of the model selection without measuring other aspects. In [3], changes in one parameter are studied across both chains and model selections; [5] proposes an improved version with a multivariate extension. Both [3] and [5] require that it is possible to identify one or more parameters having the same meaning in all models.

Varying dimensionality is also a problem for visual analysis of the simulation. If the different-dimensional models are submodels of a more complete model it might be possible to give sensible fixed values for parameters absent from the submodel, and then apply normal visualization methods. This is not possible in general, however. It might be possible to extend the simple LDA-based convergence visualization method in Section 3.2 to the approaches in [3,5], but this would not be a general solution since [3,5] require one or more parameters having the same meaning in all models; also, they do not solve problems caused by unidentifiability. Instead, below we introduce a new solution for visualization in unidentifiable and changing-dimensional cases.

Solution: compare predictive distributions. To bypass the problems caused by unidentifiability and changing dimensionality, we study *what kinds of distributions the parameter values actually represent.* Each posterior sample in an MCMC chain corresponds to some model M and parameter values θ for the model. Such a model generates a *posterior predictive distribution* $p(x|\theta, M)$ over potential observations x in the data space; the form of the distribution depends on the model M. An MCMC chain, having many posterior samples, corresponds to a distribution of such posterior predictive distributions.

The solution to unidentifiability and changing dimensionality is to measure whether the *distribution of the posterior predictive distributions* (defined by the posterior samples) is the same in all chains, as it should be at convergence.

Note that it is possible that the simulation has converged in regard to the posterior predictive distribution but not in regard to all parameters. In this case, the parameters that have not converged have either only a minor effect on the output distribution of the model or their nonconvergence is caused by some form of unidentifiability. On the other hand if the simulation has not converged in regard to the predictive distribution, then there can be no convergence in the parameter space either. (For details see Appendix C).

5.1 Visualizing the model space

To visualize the distribution of the posterior predictive distributions we first change the similarity measure of the posterior samples from measuring differences in parameter values to measuring differences in the predictive distributions. After this all data samples are in the same data space, in contrast to parameter vectors of variable-dimensional models, but since we do not have a handy vectorial representations for the predictive distributions, we cannot apply LDA and DCA directly to them. Instead, the data is now represented by the distance matrix consisting of pairwise distances between all pairs of predictive distributions. Our solution is to apply multidimensional scaling (MDS) to transform the distance matrix to a set of points in a vector space. In the vector space we can again apply LDA or DCA to the chains as previously.

The procedure for visualizing samples has three steps, each of which is discussed below.

Step 1: Compute distances between samples. Select a divergence measure and calculate the distance matrix of pairwise divergences between all posterior samples. This bypasses the problems caused by unidentifiability and changing dimensionality. As a divergence measure we use the Jensen-Shannon divergence [15]:

$$D_{JS}(p_1, p_2) = \sum_{i=1,2} \sum_{x \in X} \pi_i p_i(x) \log \frac{p_i(x)}{\sum_{j=1,2} \pi_j p_j(x)}, \tag{11}$$

where p_1 and p_2 are two posterior predictive distributions and the π_i:s are weights defining the relative importance of the distributions. For MCMC analysis the weights should be equal, $\pi_i = 1/2$.

For discrete distributions it is always possible to calculate D_{JS}, but for continuous distributions the sum would be replaced by an integral which typically cannot be computed analytically; thus it must be estimated. Here we suggest a simple estimate: for each posterior sample, generate a set of data-space points from the posterior predictive distribution, and use a binned density estimator to compute a discrete estimate of the density. The discrete estimates can then be used to compute D_{JS}. In a case study (Section 5.2), a simple estimator with 10 bins and 100 data-space points per posterior sample sufficed, since the data in the case study is low-dimensional (one-dimensional).

For higher-dimensional data, the number of bins and data-space points required for accurate estimation would quickly grow, which would increase computational cost. Such growth of the number of bins and points would be necessary, as inaccurate estimation with an insufficient number of bins or points could cause noise (bias): insufficient bins could overlook local differences in the models, and insufficient points could introduce sampling noise. In Section 5.3 we will describe an alternative, fast approach for high-dimensional data.

Step 2: Transform samples to a vector space. Having computed the distances between all sample pairs, select a multidimensional scaling (MDS) method to transform the distance matrix to a configuration of points in a vector space. The configuration of the points will be an appropriate representation of the differences between the predictive distributions.

Here we use linear MDS [28,10] to find the configuration of vectorial points that represent the predictive distributions. It can be shown [10] that the configuration found by linear MDS equals the configuration found by Principal Component Analysis [12] of the same dimensionality. Thus the MDS solution of high enough dimensionality contains all the variance information inherent in the data.

Step 3: Create the visualization. Now that we have a vector-space representation for each posterior sample, we can use LDA or DCA to create a visualization that discriminates between the different simulation chains. The procedure is the same as described in Section 3; the only difference is that LDA or DCA is applied to the vector-space points found in Step 2, rather than some existing parameter vectors.

In Section 5.2 (below) we test the three-step approach described above in a case study.

5.2 Case study 2: Simulation of a one-dimensional Gaussian mixture model

The data chosen for this illustration is the galaxy data described by Roeder ([21]; first presented by Postman et al. [19], with one additional data point) and later analyzed with mixture models by several authors, including Richardson and Green [20] and Stephens [25,24].

The data are the velocities of 82 distant galaxies diverging from us. There seems to be multimodality in the data. We fitted a mixture of seven Gaussian components to the data by Gibbs sampling, with priors picked from [20]. We simulated 10 chains of 2000 samples each, and computed both the PSRF and MPSRF convergence measures along the run.

Judging from the PSRF and MPSRF measures the simulation did not converge. Some parameters like the hyperprior of component variances converged early, but others kept fluctuating during the simulation. Based on the LDA projection of the samples at the end of the simulation (Fig. 5) it is clear that while some mixing of chains is going on there are still several areas where one or two chains are predominant. Thus the convergence measures indicate that the chains have not converged in the parameter space.

In order to measure convergence of the predictive distributions we transformed the samples to points in the model space by estimating the pairwise divergences and finding a configuration of the points with linear MDS. The dimensionality of the MDS solution was selected to hold 99% of the variance.

More specifically, for each posterior sample, we generated 100 data-space points, and used a 10-bin density estimator to estimate a discretized ver-

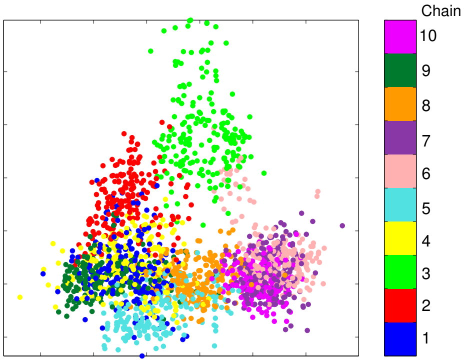

Fig. 5. Posterior simulation of a mixture model on galaxy velocity data; convergence visualization in the parameter space, by LDA projection of the last 200 samples. Each chain is marked with a unique color.

sion of the predictive distribution. These estimates were used to compute the pairwise Jensen-Shannon divergences between each pair of posterior samples. MDS was applied to the resulting matrix of divergences.

Based on the MPSRF measure calculated in the MDS space the simulation seems to have roughly converged after approximately 100 samples (Fig. 6**a**). A common rule of thumb is that a simulation has converged when the convergence measure stays under 1.2 [7]. From the LDA visualization of the first 100 samples in the MDS space (Figs. 6**c** and **d**) it can be seen that there are areas where some of the chains dominate. There are clearly more red and dark green points on the left side of the image while light green and dark blue are more dominant on the right. This can be compared to the visualization from the last 200 samples (Fig. 6**b**) where it is impossible to separate any of the chains from the others. In summary, the new visualizations corrected for the apparent divergence caused by unidentifiability in the parameter space.

Additionally, we did the same analysis using the reversible jump MCMC method for mixture models, developed by [20]. The results were very similar except that it was not possible to visualize samples in the parameter space (as we did in Fig. 5) due to the changing model dimensionality.

It should be noted that even though we used MDS to monitor convergence of the simulation in the model space (by the MPSRF measure) the method is in general better suited for visual inspection. The computational complexity of forming the distance matrix and finding the MDS solution restricts the applicability to only comparatively small sets of samples. While this is a setback for convergence monitoring it still allows visual inspections which is crucial when the simulations do not converge.

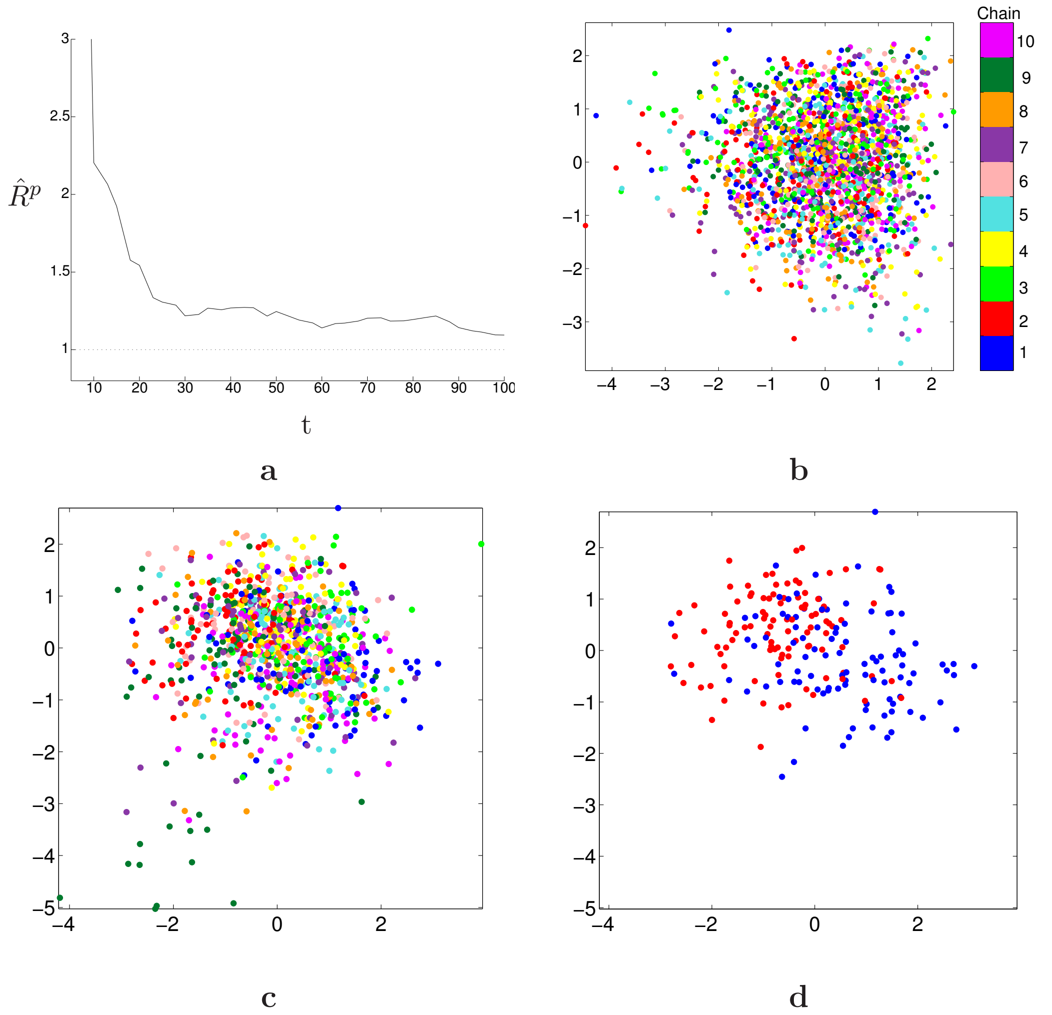

Fig. 6. Posterior simulation of a mixture model on galaxy velocity data; convergence visualization of the model space through multidimensional scaling (MDS). **a**) Change in the MPSRF measure during the first 100 samples of the simulation, calculated in the MDS space. An often used rule of thumb is that a simulation has converged when the convergence measure stays under 1.2 [7]. **b**) MDS-LDA projection of predictive distributions of the last 200 samples of each chain. **c**) MDS-LDA projection of predictive distributions of the first 100 samples of each chain. **d**) As **c**, but only two chains are shown to illustrate some of the differences. Samples are color coded based on the chain they belong to.

5.3 Fast and simple solution for high-dimensional data

In Sections 5.1 and 5.2, chains were visualized based on Jensen-Shannon divergences between models. The divergences were estimated by generating posterior predictive samples (data-space points) from the models. However, if the predictive distribution is high-dimensional, approximating the divergence takes much computational time.

A solution is to study the pooled posterior predictive distribution of each chain. Instead of forming the matrix of pairwise divergences, pool all generated posterior predictive samples from each chain and study the distribution of these samples in the data space. If convergence has been reached, all chains have the same distribution of pooled posterior predictive samples. Deviation of some chains from the rest can be visualized by applying LDA or DCA to the pooled posterior predictive samples.

The above method needs less samples than the previous approach because it need not estimate a distribution for each model; it only needs to generate enough samples to represent the pooled distribution. On the other hand, the pooled distribution does not contain as much information as the posterior predictive distributions of all individual models. It turns out that this kind of indirect visualization has a simple justification: it optimizes a lower bound of a justified but computationally expensive cost function. The cost function directly compares models instead of samples generated from them; see Appendix C for details.

5.4 Case study 3: Simulation of a five-dimensional mixture model

We illustrate use of the pooled data samples in a setting with two characteristics: 1) the data are relatively high-dimensional, which makes binned density estimation infeasible, and 2) we use a model family which does not contain the true model, which causes non-trivial convergence problems. The data set contained 500 data points generated from a mixture of five-dimensional multivariate Gaussians. The mixture had one wide component generating background noise and three small tight components, forming modes in the data.

The model family to be fitted was a restricted mixture of uniform distributions. Each model in the family was a mixture of four uniform distributions in the shape of hypercubes. The first component was fixed and was set to cover the whole domain of data. The other three components were unit hypercubes with equal mixture weights. Only the locations of the three small hypercubes were allowed to change; that is, their locations were the parameters in the family. A wide Gaussian prior centered on the data mean was used on these location parameters.

We simulated six chains with 5000 samples in each using the Metropolis-Hastings algorithm, jumping along one parameter at a time. The jumping kernel was Gaussian; the average acceptance ratio of a chain was around 0.4. As was expected from a mixture model because of the label switching problem, no convergence was indicated in the parameter space. Fig. 7 shows an LDA projection of the last 200 samples; all chains are clearly separate.

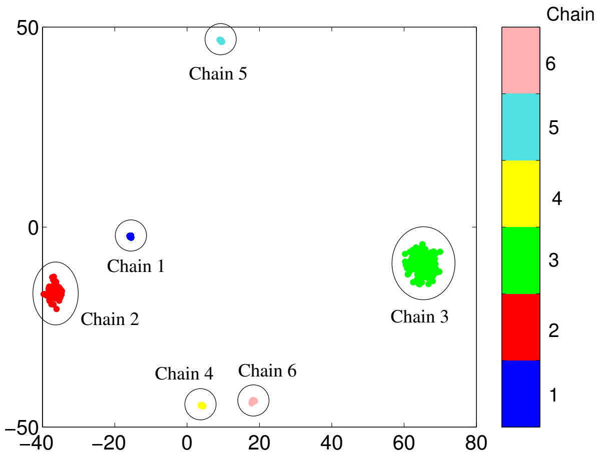

Fig. 7. MCMC simulation of a multivariate mixture; parameter-space convergence visualization by LDA projection of the last 200 samples. Each chain is illustrated with a different color.

Because of the relatively high dimensionality (five) of the data space we do not try to estimate the pairwise divergences as was done in Section 5.2 but study the pooled posterior predictive samples directly. We generated one posterior predictive data sample from each parameter sample of the posterior and calculated the MPSRF measure directly from the generated samples.

The result can be seen in Fig. 8**a**. The MPSRF measure first falls rapidly but then stays stable near a level of about 1.1. Based on the rule of thumb we could say that the simulation has converged. This is not the case, however. A visual inspection by projecting the chains with LDA helps see what is going on (Fig. 8**b**). There are clearly areas where one chain dominates. For example chains two (red) and three (green) form clusters. The means of the chains just happen to lie close to each other because of the multimodal nature of the distribution, and this fools MPSRF to give a low value even when the chains have not converged. This case study suggests that it is always a good idea to check the results visually. If there had been no problems aside from unidentifiability, the posterior predictive samples from each chain would have looked mixed in the visualization. In this case study the visualization reveals that there really is a problem in convergence which is not caused by the unidentifiability in the model.

If we enlarge the center part of the LDA projection and add the projection of the original data points to the visualization, as is done in Figs. 8**c** and **d**, it is easier to get an idea of what is happening. Chains four (yellow) and five (light blue) have found the three modes in the data. Chains one (dark blue) and six (pink) have found two of the modes. Each has two components describing one of these modes. Chains two (red) and three (green) have only found one mode and both have two components that have got stuck in the background noise generated by the wide Gaussian. It seems the simulation is very sensitive to

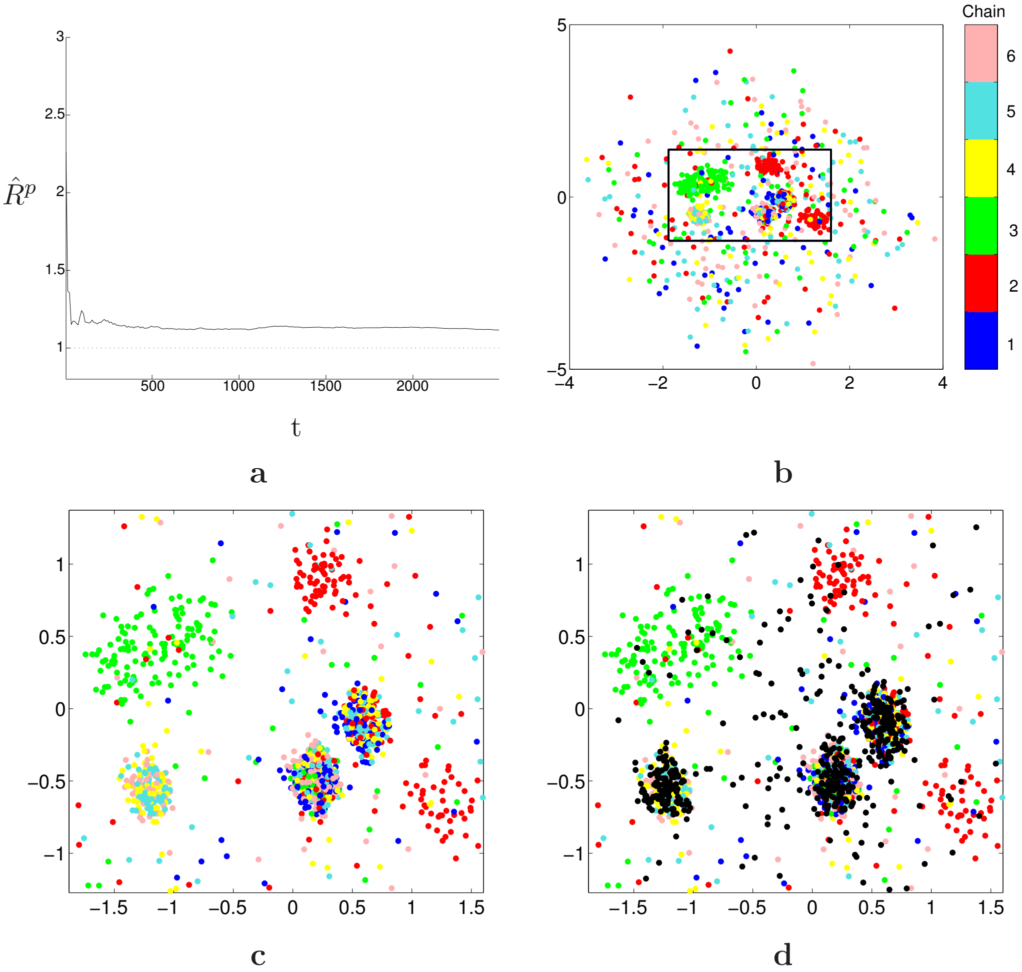

Fig. 8. MCMC simulation of a multivariate mixture; convergence visualization by studying the pooled posterior predictive distribution of each chain. **a**) Change in the MPSRF measure of the pooled posterior predictive distributions. **b**) LDA projection of the last 2500 pooled posterior predictive samples. Only every 10th sample is shown. **c**) Enlarged view of the center area of the LDA projection. **d**) As **c** but with the training data (black dots) projected in the same image. Each chain is illustrated with a different color.

slight variation in the density of data.

This behavior was verified by running the simulation several times with different jumping kernel widths. The same behavior was evident in all runs.

As a side note, we also checked how a Gaussian mixture model simulated with a Gibbs sampler performed on the data. The posterior predictive distributions of the chains converged in about 100 steps. A visual inspection showed that the chains were completely mixed and the model fit the data quite well. Hence, in this case, the problems in convergence clearly stem from using the wrong model family. This case study is realistic in the sense that the model family rarely is perfect.

6 Discussion

We have shown how to create visualizations for MCMC convergence analysis. Problems can be identified quickly using only a few visualizations. For most simulations we recommend linear discriminant analysis (LDA) for visualization. Justification for LDA comes from its connection to a common convergence measure: Its goal is to separate the different simulation chains, and if it is successful the simulation has not converged. This was demonstrated in a case study. We also discussed and demonstrated an alternative method, *discriminative component analysis* (DCA), which is based on an improved objective but is computationally more complex.

Convergence analysis by studying mixing of multiple chains is problematic for some common models, where label switching causes unidentifiability, and for simulations covering several models with different numbers of parameters. These problems can be bypassed by studying the posterior predictive distributions that the posterior samples define. For discrete or low-dimensional continuous predictive distributions this can be done by estimating the distributions by binning, and comparing them with suitable divergence measures. The chains can then be visualized by first transforming all samples into a vector space with MDS, and then applying LDA as usual. For high-dimensional continuous distributions divergence measures can be hard to estimate, but we can still study convergence by comparing pooled sets of samples from posterior predictive distributions of each chain in the data space. Two case studies of these methods were given. Fig. 9 summarizes the proposed visualization methods.

In this paper we focused on visualizations that try to separate different simulation chains from each other. It would also be possible to discriminate based on some other parameter of interest. We could for example try to see whether models with different complexities can be separated based on their generative distributions. These ideas could be combined in an interactive visualization tool.

Some of the ideas we have presented could also be used to define measures for convergence. A K-dimensional LDA maximizes the relative between-chain variance representable by the K directions together. Using all K eigenvalues for measuring convergence takes directly into account deviation in several directions instead of only the dominant one, as is done by the commonly used PSRF. We could also measure convergence in the model space. This is computationally very costly, however. Finally, the objective function of DCA could serve as a measure of convergence, when compared with a naive estimate that simply predicts the overall chain proportions. If the values are different, MCMC has not converged.

- If the simulation has a fixed parameter set and the parameters are identifiable, use LDA, as described in Section 3.2. Replace LDA by nonparametric DCA (or by semiparametric DCA), as described in Section 3.3, if computational resources are not scarce or if LDA is not informative enough.
- If the parameter space is not fixed or the model is unidentifiable, and the output space is low-dimensional, 1) generate output samples from each model, 2) compute binned estimates of their distribution, 3) compute Jensen-Shannon divergences between all distribution pairs, 4) convert the samples into a vector space with MDS, and 5) apply LDA or DCA. This method is described in Section 5.1.
- If the parameter space is not fixed or the model is unidentifiable, and the output space is high-dimensional, 1) generate output samples from each model in each chain and pool them. 2) Visualize the samples in the data space with LDA or DCA. Details of the method are in Section 5.3.

Fig. 9. A summary of the proposed ways to visualize MCMC simulations.

Implementations of LDA are available for example in the MASS package for R, available from `http://cran.r-project.org/web/packages/VR/index.html`. An implementation of DCA is available from `http://www.cis.hut.fi/projects/mi/software/dca/`.

Acknowledgments

The authors were supported by the Academy of Finland, decision numbers 108515 and 123983. This work was also supported in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence. This publication only reflects the authors' views.

References

[1] S. Brooks, A. Gelman, General methods for monitoring convergence of iterative simulations, Journal of Computational and Graphical Statistics 7 (4) (1998) 434–456.

[2] S. Brooks, A. Gelman, Some issues in monitoring convergence of iterative simulations, in: Proceedings of the Section on Statistical Computing, ASA, 1998.

[3] S. P. Brooks, P. Giudici, MCMC convergence assesment via two-way ANOVA, Journal of Computational and Graphical Statistics 9 (2000) 266–285.

[4] S. P. Brooks, P. Giudici, A. Philippe, Nonparametric convergence assessment for MCMC model selection, Journal of Computational and Graphical Statistics 12 (2003) 1–22.

[5] J. M. Castelloe, D. L. Zimmerman, Convergence assesment for reversible jump MCMC samplers, Tech. rep., Department of Statistics and Actuarial Science, University of Iowa (Feb 2002).

[6] C.-H. Chen, Generalized association plots: Information visualization via iteratively generated correlation matrices, Statistica Sinica 12 (2002) 7–29.

[7] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Rubin, Bayesian Data Analysis, Texts in Statistical Science, Chapman & Hall/CRC, Boca Raton, Florida, 1995.

[8] A. Gelman, D. B. Rubin, Inference from iterative simulation using multiple sequences, Statistical Science 7 (4) (1992) 457–472.

[9] W. R. Gilks, S. Richardson, D. J. Spiegelhalter, Markov Chain Monte Carlo in Practice, Interdisciplinary Statistics, Chapman & Hall/CRC, Boca Raton, Florida, 1995.

[10] J. C. Gower, Some distance properties of latent root and vector methods used in multivariate analysis, Biometrika 53 (3/4) (1966) 325–338.

[11] P. J. Green, Reversible jump Markov chain Monte Carlo computation and Bayesian model determination, Biometrika 82 (1995) 711–732.

[12] H. Hotelling, Analysis of a complex of statistical variables into principal components, Journal of Educational Psychology 24 (1933) 417–441,498–520.

[13] N. A. Lazar, J. B. Kadane, Movies for the visualization of MCMC output, Journal of Computational and Graphical Statistics 11 (4) (2002) 836–874.

[14] K.-C. Li, Sliced inverse regression for dimension reduction, Journal of the American Statistical Association 86 (414) (1991) 316–327.

[15] J. Lin, Divergence measures based on the Shannon entropy, IEEE Transactions on Information Theory 37 (1) (1991) 145–151.

[16] J. A. A. Nylander, J. C. Wilgenbusch, D. L. Warren, D. L. Swofford, AWTY (are we there yet?): a system for graphical exploration of MCMC convergence in Bayesian phylogenetics, Bioinformatics 24 (2008) 581–583.

[17] J. Peltonen, J. Goldberger, S. Kaski, Fast semi-supervised discriminative component analysis, in: K. Diamantaras, T. Adali, I. Pitas, J. Larsen, T. Papadimitriou, S. Douglas (eds.), Machine Learning for Signal Processing XVII, IEEE, 2007, pp. 312–317.

[18] J. Peltonen, S. Kaski, Discriminative components of data, IEEE Transactions on Neural Networks 16 (2005) 68–83.

[19] M. Postman, J. P. Huchra, M. J. Geller, Probes of large-scale structure in the corona borealis region, The Astronomical Journal 92 (1986) 1238–1247.

[20] S. Richardson, P. J. Green, On Bayesian analysis of mixtures with an unknown number of components, Journal of the Royal Statistical Society, Series B 59 (4) (1997) 731–792.

[21] K. Roeder, Density estimation with confidence sets exemplified by superclusters and voids in the galaxies, Journal of the American Statistical Association 85 (1990) 617–624.

[22] N. Slonim, N. Tishby, Agglomerative information bottleneck, in: S. A. Solla, T. K. Leen, K.-R. Müller (eds.), Advances in Neural Information Processing Systems 12, MIT Press, Cambridge, MA, 2000, pp. 617–623.

[23] B. J. Smith, boa: an R package for MCMC output convergence assessment and posterior inference, Journal of Statistical Software 21 (11) (2007).

[24] M. Stephens, Bayesian analysis of mixtures with an unknown number of components — an alternative to reversible jump methods, Annals of Statistics 28 (2000) 40–74.

[25] M. Stephens, Dealing with label-switching in mixture models, Journal of the Royal Statistical Society, Series B 62 (2000) 795–809.

[26] S. Theodoridis, K. Koutroumbas, Pattern Recognition, Academic Press, 1999.

[27] N. H. Timm, Applied Multivariate Analysis, Springer Texts in Statistics, Springer-Verlag, New York, 2002.

[28] W. S. Torgerson, Multidimensional scaling: I. theory and method, Psychometrika 17 (4) (1952) 401–419.

[29] J. Venna, S. Kaski, J. Peltonen, Visualizations for assessing convergence and mixing of MCMC, in: N. Lavrac, D. Gamberger, H. Blockeel, L. Todorovski (eds.), Proceedings of the 14th European Conference on Machine Learning (ECML 2003), Springer, Berlin, 2003, pp. 432–443.

[30] E. J. Wegman, Q. Luo, On methods of computer graphics for visualizing densities, Journal of Computational and Graphical Statistics 11 (1) (2002) 137–162.

Appendix A Jensen-Shannon divergence and mutual information

We show that, in a special case, the Jensen-Shannon divergence between two distributions is equal to a mutual information measure; this is essentially a particular case of the result shown in [22].

Let y be a scalar variable whose distribution is a mixture of several classes, including c_1 and c_2. In particular, in this paper y can be a scalar parameter of a posterior sample $\check{s}$ (or a scalar-valued function of several parameters of $\check{s}$), and the classes c_1 and c_2 can be two MCMC chains generating such samples. Denote proportions of the class prior probabilities by $p_{c_1} = p(c_1)/(p(c_1)+p(c_2))$ and $p_{c_2} = p(c_2)/(p(c_1) + p(c_2))$, and set $q(y) = p_{c_1}p(y|c_1) + p_{c_2}p(y|c_2) = p(y|c_1 \vee c_2)$, where $c_1 \vee c_2$ refers to the distribution containing only the classes c_1 and c_2.

The Jensen-Shannon divergence between $p(y|c_1)$ and $p(y|c_2)$, with weights p_{c_1} and p_{c_2}, is

$$\begin{aligned}
D_{JS}(p(y|c_1), p(y|c_2); p_{c_1}, p_{c_2}) &= p_{c_1} D_{KL}\left(p(y|c_1), q(y)\right) + p_{c_2} D_{KL}\left(p(y|c_2), q(y)\right) \\
&= p_{c_1} \int p(y|c_1) \log \frac{p(y|c_1)}{q(y)} dy + p_{c_2} \int p(y|c_2) \log \frac{p(y|c_2)}{q(y)} dy \\
&= \int \sum_{c=c_1,c_2} p(y|c) p_c \log \frac{p(y|c)}{q(y)} dy \\
&= \int \sum_{c=c_1,c_2} p(y, c|c_1 \vee c_2) \log \frac{p(y, c|c_1 \vee c_2)}{p(y|c_1 \vee c_2) p_c} = I(y, c|c_1 \vee c_2) \qquad \text{(A.1)}
\end{aligned}$$

which is the mutual information between y and the index c that tells which category y is from, given the assumption that y is from c_1 or c_2.

Appendix B Connection between LDA and DCA

Reformulating LDA. For simplicity, consider only the first LDA component $\mathbf{a}$. Denote $\sigma_{\mathbf{a}}^2 = \mathbf{a}^T \mathbf{W}_{ss} \mathbf{a}/N$, where N is the total number of samples. The LDA objective equals the variance of class centers along the projection direction, relative to the within-class variance:

$$\frac{\mathbf{a}^T \mathbf{B}_{ss} \mathbf{a}}{\mathbf{a}^T \mathbf{W}_{ss} \mathbf{a}} = \frac{1}{N\sigma_{\mathbf{a}}^2} \mathbf{a}^T \mathbf{B}_{ss} \mathbf{a} = \sum_c \frac{n_c}{N} \frac{(\mathbf{a}^T(\bar{\mathbf{s}}_{c.} - \bar{\mathbf{s}}_{..}))^2}{\sigma_{\mathbf{a}}^2} \ . \qquad \text{(B.1)}$$

Since, for a scalar variable s, $E_{p(s_1)p(s_2)}[(s_1 - s_2)^2] = 2E_{p(s)}[s^2] - 2(E_{p(s)}[s])^2 = 2E_{p(s)}[(s - E_{p(s)}[s])^2]$, the objective further equals (up to a constant multiplier) the weighted sum of squared distances between class pairs:

$$\frac{2}{N\sigma_{\mathbf{a}}^2} \mathbf{a}^T \mathbf{B}_{ss} \mathbf{a} = \sum_{c_1,c_2} \frac{n_{c_1} n_{c_2}}{N^2} \frac{(\mathbf{a}^T(\bar{\mathbf{s}}_{c_1.} - \bar{\mathbf{s}}_{c_2.}))^2}{\sigma_{\mathbf{a}}^2} \ . \qquad \text{(B.2)}$$

Since $\mathbf{a}^T\mathbf{a} = 1$, each Gaussian class has a variance of $\sigma_{\mathbf{a}}^2$ along the projection dimension. Then, for each pair of classes c_1 and c_2, the rightmost term equals the squared *Mahalanobis distance* of the projected class centers along the projection. This in turn equals the following *symmetrized Kullback-Leibler* *divergence* between the distributions along the projection [26]:

$$\frac{1}{\sigma_{\mathbf{a}}^2}(\mathbf{a}^T(\bar{\mathbf{s}}_{c_1\cdot} - \bar{\mathbf{s}}_{c_2\cdot}))^2 = D_{KL}(p(\mathbf{a}^T\mathbf{s}|c_1), p(\mathbf{a}^T\mathbf{s}|c_2)) + D_{KL}(p(\mathbf{a}^T\mathbf{s}|c_2), p(\mathbf{a}^T\mathbf{s}|c_1)) \tag{B.3}$$

LDA thus maximizes a sum of symmetrized Kullback-Leibler divergences between the classes along the projection, weighted by the fractions $n_{c_1}n_{c_2}/N^2$.

Improving the cost function. Optimizing the objective (B.2) does not result in optimal discrimination. We will improve it in two steps. First, for each class pair (c_1, c_2), replace the symmetrization in (B.3) with the *Jensen-Shannon divergence*. This helps reinterpret the objective in a form that can be easily generalized. For brevity, denote $y = \mathbf{a}^T\mathbf{s}$ and denote proportions of the class prior probabilities by $p_{c_1} = p(c_1)/(p(c_1)+p(c_2))$ and $p_{c_2} = p(c_2)/(p(c_1)+p(c_2))$. As shown in Appendix A, the Jensen-Shannon divergence here is

$$D_{JS}(p(y|c_1), p(y|c_2); p_{c_1}, p_{c_2}) = I(y, c|c_1 \vee c_2) \tag{B.4}$$

which is the mutual information between a projected sample and its class, when only the classes c_1 and c_2 are possible. LDA then finds (roughly, due to the different symmetrization) the direction that maximizes the sum of pairwise mutual informations between classes, weighted by the class proportions. This suggests the natural extension to consider more than just pairwise class interactions, and maximize the complete mutual information $I(c, y)$ between classes and projected data. It can be shown that as the amount of data grows, the likelihood objective of DCA asymptotically equals $I(c, y)$, up to a constant; see [18]. DCA is then a finite-data implementation of an extension of LDA.

Appendix C Justification for the indirect visualization method

The method in Section 5.3 visualizes convergence indirectly, from pooled predictive samples instead of the underlying chains of models. It is meant for situations where directly visualizing the chains is not feasible.

Here we justify the method: we show its cost function is a lower bound to a cost function which would directly use the models instead of samples generated from them. Thus, if the visualization finds differences between chains, the direct measure would indicate non-convergence.

The proof is in two parts. First, we show that a difference measure with no assumptions about the model family leads to the DCA visualization method for pooled predictive samples. Next, we show that a similar difference measure, specific to Gaussian models, leads to the LDA visualization method for pooled predictive samples. The latter method is recommended in practice since it is faster.

General result. The mutual information $I(c, \breve{s})$ between models and the indices of the chains that generated them would be a suitable convergence measure, as discussed in Section 2.3. However, here the model space is difficult to work with (it may have variable-dimensional parameterizations or unidentifiability). We therefore consider a third variable: $\mathbf{x}$, denoting a posterior predictive sample generated by some model in some chain.

We can derive a lower bound: $I(c, \breve{s}) \geq I(c, \mathbf{x})$, as follows: we have $p(c, \breve{s}, \mathbf{x}) = p(c)p(\breve{s}|c)p(\mathbf{x}|\breve{s})$ since $\mathbf{x}$ is conditionally independent of c given $\breve{s}$. Therefore $p(c, \breve{s}, \mathbf{x}) = p(c|\breve{s})p(\breve{s}, \mathbf{x})$, and it then follows that

$$I(c, \breve{s}) - I(c, \mathbf{x}) = E_{p(c,\breve{s},\mathbf{x})}\left[\log\left(\frac{p(c, \breve{s})p(c)p(\mathbf{x})}{p(c)p(\breve{s})p(c, \mathbf{x})}\right)\right]$$
$$= E_{p(\breve{s},\mathbf{x})}\left[E_{p(c|\breve{s})}\left[\log\left(\frac{p(c|\breve{s})}{p(c|\mathbf{x})}\right)\right]\right] = E_{p(\breve{s},\mathbf{x})}\left[D_{KL}(p(c|\breve{s}), p(c|\mathbf{x}))\right] \geq 0 \; . \quad \text{(C.1)}$$

If $\mathbf{x}$ is a vector variable, we further have $I(c, \mathbf{x}) \geq \max_{\mathbf{W}} I(c, \mathbf{W}\mathbf{x})$ where $\mathbf{W}$ are projections to a smaller-dimensional space (the proof is similar to the above, with $\mathbf{x}$ and $\mathbf{W}\mathbf{x}$ in place of $\breve{s}$ and $\mathbf{x}$, respectively). The DCA visualization method maximizes a cost function which asymptotically approaches $I(c, \mathbf{W}\mathbf{x})$; therefore DCA asymptotically maximizes a lower bound to $I(c, \breve{s})$.

Gaussian-specific result that leads to LDA. We do not know a direct connection from the mutual information discussed above to the LDA method for pooled predictive samples. However, a weaker justification for LDA can be derived from an alternative cost function suitable for measuring convergence. Here we propose such a cost function, show that it is suitable under a Gaussianity assumption, and derive a connection to LDA.

We propose that convergence can be measured by how mutually different the models within each chain are on average, compared to how different all models are on average. If models within chains are less different, the chains have not converged. The difference between two models $\breve{s}_1$ and $\breve{s}_2$ can be measured by the difference between the data distributions they generate, that is, $p_{\breve{s}_1}$ and $p_{\breve{s}_2}$. We will consider one-dimensional marginal distributions along a direction $\mathbf{w}$, namely $p^{\mathbf{w}}_{\breve{s}_1}$ and $p^{\mathbf{w}}_{\breve{s}_2}$, instead of the full distributions. (This will lead to a connection to a single LDA component; behavior with several components is discussed at the end of the proof.)

We will measure the difference between the distributions by symmetrized Kullback-Leibler divergence $D_{SKL}(p, q) = D_{KL}(p, q) + D_{KL}(q, p)$. The comparison of average differences (within chains vs. between any models irrespective of chains) can then be done by the following measure:

$$\frac{E_{p(c_1)p(c_2)p(\check{s}_1|c_1)p(\check{s}_2|c_2)}[D_{SKL}(p^{\mathbf{w}}_{\check{s}_1}, p^{\mathbf{w}}_{\check{s}_2})]}{E_{p(c)p(\check{s}_1|c)p(\check{s}_2|c))}[D_{SKL}(p^{\mathbf{w}}_{\check{s}_1}, p^{\mathbf{w}}_{\check{s}_2})]} - 1 \ . \tag{C.2}$$

We will next show that (C.2) is suitable for measuring convergence. Assume that each model $\check{s}$ in each chain c is Gaussian with the same covariance matrix $\mathbf{\Sigma}$, but different means $\theta_{\check{s}}$. It is easy to see that the measure is zero when the chains are identical. To show it is always at least zero, denote $\sigma^2_{\mathbf{w}} = \mathbf{w}^T\mathbf{\Sigma}\mathbf{w}$. As noted in Appendix B, for Gaussians with equal variances, D_{SKL} becomes a Mahalanobis distance. The measure (C.2) then becomes

$$\begin{aligned}
\frac{E_{p(c_1)p(c_2)p(\check{s}_1|c_1)p(\check{s}_2|c_2)}[(\mathbf{w}^T(\theta_{\check{s}_1} - \theta_{\check{s}_2}))^2/\sigma^2_{\mathbf{w}}]}{E_{p(c)p(\check{s}_1|c)p(\check{s}_2|c)}[(\mathbf{w}^T(\theta_{\check{s}_1} - \theta_{\check{s}_2}))^2/\sigma^2_{\mathbf{w}}]} - 1 &= \frac{Var_{p(c,\check{s})}[\mathbf{w}^T\theta_{\check{s}}]}{E_{p(c)}[Var_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]} - 1 \\
&= \frac{Var_{p(c)}[E_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]] + E_{p(c)}[Var_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]}{E_{p(c)}[Var_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]} - 1 \\
&= \frac{Var_{p(c)}[E_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]}{E_{p(c)}[Var_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]} \geq 0 \ .
\end{aligned} \tag{C.3}$$

where the first equality follows from the general relation between variance and average squared distance for scalar variables (see Appendix B) and the second equality follows by adding and subtracting the chain-wise mean inside the difference used in the definition of variance. Hence (C.2) is a justified measure of convergence.

Finally, we show that, under the same Gaussianity assumption, LDA optimizes a lower bound to the convergence criterion (C.2) and therefore is a justified visualization method. The first LDA component for chains of generated data maximizes

$$\begin{aligned}
\frac{Var_{p(c)}[E_{p(\mathbf{x}|c)}[\mathbf{w}^T\mathbf{x}]]}{E_{p(c)}[Var_{p(\mathbf{x}|c)}[\mathbf{w}^T\mathbf{x}]]} &= \frac{Var_{p(c)}[E_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]}{E_{p(c)}[E_{p(\mathbf{x},\check{s}|c)}[(\mathbf{w}^T\mathbf{x} - E_{p(\mathbf{x},\check{s}|c)}[\mathbf{w}^T\mathbf{x}])^2]]} \\
&= \frac{Var_{p(c)}[E_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]}{E_{p(c)}[E_{p(\mathbf{x},\check{s}|c)}[(\mathbf{w}^T\mathbf{x} - \mathbf{w}^T\theta_{\check{s}} + \mathbf{w}^T\theta_{\check{s}} - E_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}])^2]]} \\
&= \frac{Var_{p(c)}[E_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]}{E_{p(c)}[E_{p(\mathbf{x},\check{s}|c)}[(\mathbf{w}^T\mathbf{x} - \mathbf{w}^T\theta_{\check{s}})^2 + (\mathbf{w}^T\theta_{\check{s}} - E_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}])^2]]} \\
&= \frac{Var_{p(c)}[E_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]}{\sigma^2_{\mathbf{w}} + E_{p(c)}[Var_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]} \leq \frac{Var_{p(c)}[E_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]}{E_{p(c)}[Var_{p(\check{s}|c)}[\mathbf{w}^T\theta_{\check{s}}]]} \ .
\end{aligned} \tag{C.4}$$

Therefore, the first LDA component maximizes a quantity that is a lower bound to (C.3). Both quantities are tightly lower bounded by zero.

For several LDA components we do not know as strict optimality results, but we do know the following. Assume the data has been preprocessed so that the within-chain covariance matrix is an identity matrix. Then LDA component directions are orthogonal and they can be found one by one, by removing the previous components from the data. Each component by itself is optimal for data where the previous ones have been removed, but the components together may not be optimal for discriminating the chains of pooled posterior predictive samples. Similarly, each LDA component maximizes a lower bound to (C.2) when that measure is computed for data where the previous LDA components have been removed, but the components together do not maximize a lower bound to a generalization of (C.2) to multidimensional marginal distributions.