Machine Learning, 50, 73–94, 2003

Mixtures of Factor Analysers. Bayesian Estimation and Inference by Stochastic Simulation

ERNEST FOKOUÉ ernest@stat.ohio-state.edu
Department of Statistics, The Ohio State University, 404 Cockins Hall, 1958 Neil Avenue, Columbus OH 43210-1247, USA

D.M. TITTERINGTON mike@stats.gla.ac.uk
Department of Statistics, University of Glasgow, 15 University Gardens, Glasgow, G12 8QW, UK

Editors: Nando de Freitas, Christophe Andrieu, Arnaud Doucet

Abstract. Factor Analysis (FA) is a well established probabilistic approach to unsupervised learning for complex systems involving correlated variables in high-dimensional spaces. FA aims principally to reduce the dimensionality of the data by projecting high-dimensional vectors on to lower-dimensional spaces. However, because of its inherent linearity, the generic FA model is essentially unable to capture data complexity when the input space is nonhomogeneous. A finite Mixture of Factor Analysers (MFA) is a globally nonlinear and therefore more flexible extension of the basic FA model that overcomes the above limitation by combining the local factor analysers of each cluster of the heterogeneous input space. The structure of the MFA model offers the potential to model the density of high-dimensional observations adequately while also allowing both clustering and local dimensionality reduction. Many aspects of the MFA model have recently come under close scrutiny, from both the likelihood-based and the Bayesian perspectives. In this paper, we adopt a Bayesian approach, and more specifically a treatment that bases estimation and inference on the stochastic simulation of the posterior distributions of interest. We first treat the case where the number of mixture components and the number of common factors are known and fixed, and we derive an efficient Markov Chain Monte Carlo (MCMC) algorithm based on Data Augmentation to perform inference and estimation. We also consider the more general setting where there is uncertainty about the dimensionalities of the latent spaces (number of mixture components and number of common factors *unknown*), and we estimate the complexity of the model by using the sample paths of an ergodic Markov chain obtained through the simulation of a continuous-time stochastic birth-and-death point process. The main strengths of our algorithms are that they are both efficient (our algorithms are all based on familiar and standard distributions that are easy to sample from, and many characteristics of interest are by-products of the same process) and easy to interpret. Moreover, they are straightforward to implement and offer the possibility of assessing the goodness of the results obtained. Experimental results on both artificial and real data reveal that our approach performs well, and can therefore be envisaged as an alternative to the other approaches used for this model.

Keywords: mixtures, factor analysis, birth-and-death process, data augmentation, point process, prior, posterior, Gibbs sampling, Markov chain, MCMC, stochastic simulation, equilibrium (stationary) distribution

1. Introduction

The main goal of factor analysis (FA) is to describe the covariance relationships among many variables in terms of fewer underlying latent (unobservable) constructs represented

by random quantities known as *factors*. Factor analysis is therefore a *data reduction or dimensionality reduction* technique, since the number of factors is always assumed to be far less than the number of originally observed variables. Finite mixtures of distributions are used to model the distribution of a random variable when the input space is assumed to be heterogeneous (nonhomogeneous). Mixture models therefore allow the partition of the input space into clusters and can therefore be used for classification. In some settings where a mixture of densities is an adequate representation of the density of the data, mixture models can be used for density estimation as an alternative to traditional nonparametric kernel density estimators. The mixture of factor analysers (MFA) model is a globally nonlinear extension of the basic factor analysis (FA) model. Unlike the fundamentally linear FA model, the MFA model is essentially more flexible, with its inherent ability to partition a heterogeneous input space into clusters while simultaneously achieving local dimensionality reduction in each of the derived subspaces. Under the assumption of orthogonal factor analysis, the MFA is a reduced-dimensional mixture of multivariate Gaussians that can be used as an approximate method of density estimation in high-dimensional space, especially in cases where samples are of small sizes. In fact, while a plain mixture of multivariate Gaussians with full covariance matrices would be prone to overfitting when the number of mixture components is increased, the MFA model allows one to control or avoid overfitting by varying the dimensionalities of the latent subspaces (i.e. the number of common factors), thereby reducing the number of free model parameters significantly without imposing such strong constraints as forcing the covariance matrices of the local Gaussians to be isotropic.

Both FA and finite mixture models have been extensively studied by the Machine Learning community for a variety of applications, ranging from data compression to classification. The MFA model, by its construction and structure, is a rich and interesting extension of both the above models, and therefore has the potential for an even broader range of applications. The study of the MFA model is therefore a relevant Machine Learning topic. In fact, in recent years, the study of MFA has received considerable interest. The psychometrics community with its traditional interest in FA and related multivariate models has produced a good number of papers among which Yung (1997), Dolan and Van der Maas (1998), and Arminger, Stein, and Wittenberg (1999) all address the fitting of MFAs or closely related models to data, by various versions of Maximum Likelihood Estimation (MLE). From the Neural Computation community, Ghahramani and Hinton (1997) derived an EM algorithm for parameter estimation within the model. Ghahramani and Beal (2000) later considered a Bayesian treatment of MFA via a variational approximation. Ueda et al. (2000) applied their Split-and-Merge-EM (SMEM) algorithm to the MFA model, and obtained good results in such tasks as image compression and handwritten digits recognition. It is therefore fair to say that the MFA model is relevant for applications that are of interest to the Machine Learning community. From the mainstream statistics community, McLachlan and Peel (2000) presented a variant of the EM algorithm for a study of the MFA model with application to clustering and density estimation. They applied the resulting algorithm to data about wines,[1] hereinafter referred to as the *wine data set*, and obtained good results. One striking feature of the above developments is that only approximate techniques have been used to address the intractability of the functions of interest, from the Bayesian

perspective. While these techniques can be fast in producing reasonably good results, assessing the closeness of approximations to the true results still remains a rather complex problem.

To the best of our knowledge, the first attempt to use an "exact" (in the sense of not based on the use of approximations of the functional of interest) technique for inference about and estimation of the MFA model was presented by Fokoué (2000), who constructed an efficient sampling scheme for the posterior simulation of the distributions of interests. The derived Markov Chain Monte Carlo (MCMC) algorithm was essentially a straightforward adaptation of *Data Augmentation* (a two-stage Gibbs sampler) to the complete-data formulation of the MFA inferential task. There have since then been some other developments along the the lines of stochastic simulation for MFAs, namely in Fokoué and Titterington (2000a, 2000c).

MCMC algorithms are computationally intensive. They are relatively slow compared to their approximate competitors, and diagnosing their convergence is still a complex problem, especially in high-dimensional settings like that of the MFA model. However, as far as speed is concerned, we have applied our stochastic simulation algorithm to the moderate high-dimensional wine data set, and the results are encouraging. In other words, with the development of a variety of efficient hybrid MCMC algorithms and the construction of many other efficient sampling schemes, MCMC algorithms are fast becoming practical and useful alternatives to their approximate counterparts, and we hope to concentrate further developments of our approach on designing faster sampling schemes.

Despite their relatively lower speed of convergence compared to approximate techniques, MCMC methods have the advantage of providing essentially exact estimation of the posterior distributions of interest. Furthermore, unlike the EM algorithm which can be trapped into singularities (i.e situations where only one or indeed very few observations are allocated to a component), the use of hierarchical prior structures defining similar (but not equal) component covariance matrices allows the derivation of sampling schemes that can escape situations of near zero variances in the mixture of Gaussians setting. In many applications of factor analysis and its extensions, the meanings of parameters (especially factor loadings) are of paramount importance, and the ability to provide a unique set of parameter estimates becomes crucial. Since the matrix of factor loadings is invariant to orthogonal transformations, a naive technique can only produce estimates up to a rotation. This multiplicity of solutions is generally not useful when characterisation is the aim, and a unique solution is generally achieved by constraining some entries of the matrix of factor loadings to have preassigned values. Another clear advantage of stochastic simulation is that, unlike the EM algorithm, which can be complicated to apply when parameters must satisfy some restrictions (Dong & Taylor, 1995), constructing a sampling scheme under restrictions is much easier, and in fact straightforward in the case of factor analysis.

All the Markov Chains constructed in our sampling schemes are ergodic (irreducible and aperiodic), and therefore by virtue of irreducibility the convergence to the equilibrium distribution does not depend on initial guesses, unlike the EM algorithm whose convergence can be hindered by bad choices of initial parameter values.

The full conditional posterior distributions of the model parameters are all standard and familiar distributions that are easy to simulate, and this makes the implementation of the Gibbs sampler Markov Chain Monte Carlo (MCMC) algorithm efficient.

Last but not least, the output of the sampling scheme can be used for many other inferential tasks such as the computation of error estimates for the parameter of interest and also analysis of the goodness-of-fit of the proposed model, all this without extra computational effort.

A study of MFAs from a stochastic simulation perspective is therefore justified. However, it must be stressed that, while stochastic simulation offers many advantages and somehow overcomes some of the drawbacks of its approximate counterparts, it does require, as we shall see, a careful treatment of two fundamental identifiability problems: the label switching problem inherent to mixtures, and the rotation invariance problem of the underlying local factor analysers. As we said earlier, this becomes more crucial when the interpretation of model parameters is the main aim of the analysis.

The remainder of the paper is organised as follows. In Section 2, we present a brief review of both factor analysis and finite mixtures, and we introduce the mixture of factor analysers model and its building blocks along with some indications of possible areas of application. Section 3 introduces the Bayesian treatment of the MFA model, and explores the construction of our efficient MCMC algorithm for MFA with known model complexities, while also covering elements of hierarchical prior specification, along with some elements of solutions to the problems of label switching and rotation invariance. Section 4 covers the difficult issue of model selection. As it turns out, the rotation invariance of the matrix of factor loadings allows us to treat its columns (factor axes) as points in some high-dimensional space. Viewing the matrix of factor loadings as a point process (also known as *random configuration* with varying number of points) allows us to adapt ideas from Stephens's (2000b) birth-and-death MCMC (BDMCMC) method to estimate the number of factors in our underlying factor analysis models. We also construct a nested scheme for model selection in MFA, based on BDMCMC, after providing a justification of our preference for BDMCMC over Richardson and Green's (1997) Reversible Jump MCMC (RJMCMC) in this setting. Section 5 explores some synthetic and real life examples, while Section 6 concludes with a discussion.

2. What is a mixture of factor analysers?

2.1. Factor analysis

The traditional orthogonal factor analysis (FA) model assumes that a p-dimensional manifest (observed) variable $X \in \mathbb{R}^p$, made up of correlated attributes, can be reduced to a lower-dimensional latent (unobservable) vector $Z \in \mathbb{R}^q$, with $q < p$, of uncorrelated attributes. In other words, X is assumed to have been generated by a linear combination of the following form:

$$X = \Lambda Z + \boldsymbol{\mu} + \varepsilon, \tag{1}$$

where $\Lambda = (\lambda_{ij})$ is real-valued $p \times q$ matrix of parameters known as *factor loadings*, $\boldsymbol{\mu} \in \mathbb{R}^p$ is the marginal mean of X, and $\varepsilon \in \mathbb{R}^p$ is the independent disturbance vector. Orthogonal factor analysis further assumes that $Z \sim \mathcal{N}_q(0, \mathbf{I}_q)$, $\varepsilon \sim \mathcal{N}_p(0, \Sigma)$, where crucially $\Sigma = \mathsf{diag}(\boldsymbol{\sigma}_1^2, \ldots, \boldsymbol{\sigma}_p^2)$, and that ε and Z are independent. We therefore derive $\mathsf{cov}(X, Z) = \Lambda$. In

other words, Λ is the covariance matrix between X and Z, and therefore contains information (loadings) revealing how groups of observed attributes combine (linearly) together to form a single new attribute (factor) that is common to them. Throughout this paper, we use $\mathcal{H}$ to denote the entire latent space, and Θ to denote the entire parameter space. It is straightforward to show by direct manipulation of expectations that the marginal density $\mathbf{p}(\mathbf{x})$ of X is Gaussian with mean μ and covariance matrix $\Lambda\Lambda^{\mathsf{T}}+\Sigma$, and we write it as $\mathbf{p}(\mathbf{x}) = \mathcal{N}_p(\mathbf{x}; \boldsymbol{\mu}, \Lambda\Lambda^{\mathsf{T}} + \Sigma)$. While the marginal density of X is obviously of great importance, and since Z is unobservable and therefore missing, it is often more convenient to introduce the *missing data* (*incomplete-data*) formulation of the FA model in which each manifest variable X is assumed to have been generated by a specific but unknown vector Z of latent factors. This entails specifying the density $\mathbf{p}(\mathbf{z})$ of Z and then specifying the corresponding conditional density $\mathbf{p}(\mathbf{x} \mid \mathbf{z})$ of X given Z. With our assumptions, $\mathbf{p}(\mathbf{z}) = \mathcal{N}_q(\mathbf{z}; 0, \mathbf{I}_q)$, and $\mathbf{p}(\mathbf{x} \mid \mathbf{z}) = \mathcal{N}_p(\mathbf{x}; \boldsymbol{\mu} + \Lambda\mathbf{z}, \Sigma)$. This means that the diagonality of Σ is indeed one of the most crucial assumptions of FA since it implies that, conditional on the knowledge of the latent variables, the attributes of the manifest variable are essentially uncorrelated. Hence, the common factors explain all the dependence structure amongst the p observed attributes of the manifest variable. Since we assume $q < p$, the estimation of factor scores provides a representation of the manifest variable in a lower-dimensional space, thereby achieving the dimensionality reduction aim of factor analysis which seeks to explain a set of highly correlated observed scores by fewer uncorrelated common factors.[2]

Essentially, there are three mains goals in FA, namely: (a) *model selection* which consists of the determination or estimation of the adequate number of factors that can be used to represent the original manifest variables without much loss of information; (b) *parameter estimation* which consists of estimating the parameters (especially the factor loadings) of the postulated model in order to interpret and characterise the covariance (association) structure of the manifest variables; and (c) *prediction* which consists of *estimating factor scores* for future unseen observations for such purposes as data-reduction. Estimated factor scores can be used in image compression to store high-dimensional images that are later reconstructed. Estimated factor scores can also be used for data visualisation in the plane (2-factor model) to explore group structures in the observed population of interest. Psychometricians, sociologists and educationalists are generally interested in factor loadings as a way to explain or at least interpret the associations (correlations) amongst some designed variables (tests grades, monthly expenses, etc.) and their relationships to some hypothesised latent (non directly measurable) concepts like intelligence, social class or aptitude.

However, the estimation of factor scores requires a set of model parameters, which in turn requires some knowledge (estimate) of the number of factors. Estimating the number of factors is therefore central, and we address it in Section 4. For now, we assume the number of factors known and fixed, and we focus our attention on parameter estimation.

Parameter estimation presupposes the existence of a unique set of parameters that characterise the proposed model, so that the objective of the estimation task is to determine that unique set of parameters.[3] Unfortunately, as we shall see, our model as specified by Eq. (1) is indeterminate (unidentifiable), and does not provide a unique set of parameters, but a multiplicity of parameter sets, each related to the other by an orthogonal transformation. In fact, Λ has pq free parameters (factor loadings). The diagonal matrix Σ has p

free parameters (specific variances). We therefore have $p(q+1)$ variance-covariance parameters to be estimated. Now, given a sample of observations $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, our objective is to use the $\frac{1}{2}p(p+1)$ items of information provided by the sample covariance matrix to estimate our $p(q+1)$ unknown free parameters. As we see, in most cases, we will have $p(q+1) > \frac{1}{2}p(p+1)$, and the sample will therefore not provide enough information to allow the estimation of a unique set of Λ and Σ. If Γ is an orthogonal matrix and $\tilde{Z} = \Gamma Z$, then we have $\Gamma\Gamma^{\mathsf{T}} = \Gamma^{\mathsf{T}}\Gamma = \mathbf{I}_q$, $\tilde{Z} \sim \mathcal{N}_q(\mathsf{0}, \mathbf{I}_q)$ and $X = \Lambda\Gamma^{\mathsf{T}}\tilde{Z} + \mu + \mathbf{e}$, and the covariance matrix of X in the new coordinate system would be $\Lambda\Gamma^{\mathsf{T}}(\Lambda\Gamma^{\mathsf{T}})^{\mathsf{T}} + \Sigma = \Lambda\Gamma^{\mathsf{T}}\Gamma\Lambda^{\mathsf{T}} + \Sigma = \Lambda\Lambda^{\mathsf{T}} + \Sigma$. In other words, Λ and $\Lambda\Gamma^{\mathsf{T}}$ are two different matrices of factor loadings that produce exactly the same covariance matrix $\Lambda\Lambda^{\mathsf{T}} + \Sigma$. We can therefore obtain an infinite number of equivalent matrices of factor loadings by simply applying successive orthogonal transformations to an initial one. Geometrically speaking, the columns of Λ can be viewed as defining the axes of the lower-dimensional latent space (coordinate system) of factors. Since a rotation is a non-singular orthogonal transformation, and a permutation of columns is particular type of rotation, we say that a factor solution is invariant to permutations of axes. This feature will be useful in Section 4, where we address the estimation of the number of factors. In practice, a unique solution is guaranteed by imposing some constraints on Λ so that the only valid solution is the one that satisfies the constraints. For estimability of parameters, constraints are imposed in such a way that the number of parameters to be estimated is at most equal to the number of items of information provided by the sample. Traditionally, there are two types of constraint that are equivalent:

1. Constrain Λ to be such that $\Lambda^{\mathsf{T}}\Lambda$ is diagonal. Since, $\Lambda^{\mathsf{T}}\Lambda \in \mathbb{R}^{q\times q}$ is symmetric and diagonal, $\frac{1}{2}q(q-1)$ of its elements are all zeros. This means that $\frac{1}{2}q(q-1)$ elements do not need to be estimated by the parameter estimation procedure. This approach is used when estimation is done via a deterministic optimisation algorithm.
2. A second approach equivalent to the above consists of preassigning values to some entries of Λ as in Eq. (2). This particular lower diagonal form[4] of Λ reduces the number of parameters to be estimated by $\frac{1}{2}q(q-1)$ as above. This is the form of constraints that we use in the Bayesian sampling framework, since its application is straightforward.

$$\Lambda = \begin{pmatrix} \boldsymbol{\lambda}_{11} & 0 & 0 & \cdots & 0 & 0 \\ \boldsymbol{\lambda}_{21} & \boldsymbol{\lambda}_{22} & 0 & \cdots & 0 & 0 \\ \boldsymbol{\lambda}_{31} & \boldsymbol{\lambda}_{32} & \boldsymbol{\lambda}_{33} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{\lambda}_{q-1,1} & \boldsymbol{\lambda}_{q-1,2} & \boldsymbol{\lambda}_{q-1,3} & \cdots & \boldsymbol{\lambda}_{q-1,q-1} & 0 \\ \boldsymbol{\lambda}_{q,1} & \boldsymbol{\lambda}_{q,2} & \boldsymbol{\lambda}_{q,3} & \cdots & \boldsymbol{\lambda}_{q,q-1} & \boldsymbol{\lambda}_{q,q} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{\lambda}_{p,1} & \boldsymbol{\lambda}_{p,2} & \boldsymbol{\lambda}_{p,3} & \cdots & \boldsymbol{\lambda}_{p,q-1} & \boldsymbol{\lambda}_{p,q} \end{pmatrix} \tag{2}$$

Both the above approaches provide an upper bound on the number of factors that can be included in a model. In fact, to guarantee a unique solution under our constraints, all we

need is to determine q such that $p(q+1) - \frac{1}{2}q(q-1) \leq \frac{1}{2}p(p+1)$, which means

$$(p+q) \leq (p-q)^2. \tag{3}$$

Note. It must be said that there are situations where solutions satisfying constraint (3) might not provide an adequate fit for the data. In fact, given a data set, a fundamental question without an obvious answer is whether there exists a matrix of factor loadings Λ such that the model in Eq. (1) adequate fits the data. An exploration of this issue and many other related topics of FA can be found in such references as Bartholomew (1987), Everitt (1984), Krzanowski and Marriott (1994, 1995), and Press (1972) amongst others.

2.2. *Finite mixtures of distributions*

Finite mixture models provide another rich class of models that are heavily used in statistical modelling, and that have been extensively studied in recent years by both the Neural Computation and the Machine Learning community for a variety of practical applications. The use of finite mixture models is particularly relevant to applications where the input space is assumed to be nonhomogeneous (heteregeneous), so that it would be unrealistic to use a single density to model the distribution of the data. We herein only give a brief review of issues related to this vast topic, and refer the reader to such references as Titterington, Smith, and Makov (1985), Everitt and Hand (1981), and McLachlan and Peel (2000) for more detailed presentations. Given an observation X, a finite mixture model assumes that X was generated by one of k subpopulations, each containing a proportion π_j of elements of the wider population. Obviously, $\pi_j > 0$ and $\sum \pi_j = 1$ for $j = 1, \ldots, k$. Each subpopulation is also known as a component of the mixture, and can be viewed as a cluster in the input space. It is usually convenient to define a discrete random latent variable Y that identifies the component or cluster from which the observation originated. This means that $\mathsf{Pr}(Y = j) = \pi_j$ for $j = 1, \ldots, k$. It is easy to see that Y has a multinomial distribution, $Y \sim \mathsf{Mn}(1; \pi_1, \ldots, \pi_k)$. The π_j's are called mixing proportions or weights. In each subpopulation, X has a specific class conditional density (also called component density) given by $\mathbf{p}(\mathbf{x} \mid Y = j)$. The marginal density of X is therefore

$$\mathbf{p}(\mathbf{x}) = \sum_{j=1}^{k} \mathsf{Pr}(Y = j)\mathbf{p}(\mathbf{x} \mid Y = j) = \sum_{j=1}^{k} \pi_j \mathbf{p}(\mathbf{x} \mid Y = j). \tag{4}$$

Example. When $\mathbf{p}(\mathbf{x} \mid Y = j)$ is the Gaussian density, we have a Mixture of Gaussians, arguably the most extensively used subclass of finite mixture models, since so many applications can be adequately modelled by mixtures of Gaussians.

Essentially, when using finite mixtures of distributions, some of the main modelling goals are: (a) clustering with applications to classification and pattern recognition; (b) density estimation; (c) parameter estimation and (d) model selection.

As we shall see later, parameter estimation via Bayesian sampling requires a careful treatment of a phenomenon known as label switching that arises because of the fact that the posterior distribution of the parameters is invariant to permutations of component labels.

2.3. Mixtures of factor analysers (MFA)

Before describing the building blocks of the MFA model, it is good to look at some points that justify the relevance of such a model:

- *Factor Analysis in nonhomogeneous input space.* The Factor Analysis model is essentially *linear* and may perform poorly when the input space is nonhomogeneous. Combining a finite number of local factor analysers results in a globally nonlinear model that is theoretically more flexible and therefore better able to capture the complexity of the data.
- *Mixture of Gaussians with structured component covariance matrices.* When used for density estimation, finite mixtures of Gaussians can be prone to *overfitting* in high-dimensional spaces. In fact, as the number of mixture components increases, density estimation is greatly improved. However, this increase in the number of components leads to a significant increase in the number of free model parameters when full covariance matrices are used, and this naturally leads to overfitting in the event of small samples. MFAs control or avoid overfitting by using the *intrinsic dimensionalities* of local factor analysers to control the number of model parameters. This is a good trade-off between the use of restrictive isotropic covariance matrices and the use of full covariance matrices.

Closely related to the MFA model are Mixtures of Probabilistic Principal Component Analysers, studied by Tipping and Bishop (1999) via the EM algorithm, and to some extent (although purely univariate[5] at this stage), Mixtures of Regressions studied from a stochastic simulation perspective by Hurn, Justel, and Robert (2000).

2.3.1. Ingredients of the MFA model. The generative equation of the MFA model is a combination of elements from both factor analysis and mixture distributions, and we use the following notation:

$$X = \Lambda_j Z + \boldsymbol{\mu}_j + \varepsilon, \quad j = 1, \ldots, k. \tag{5}$$

We assume our disturbance term ε to be measurement error, and therefore not component-dependent. In other words, we assume ε to have the same distribution in all the components. Here, we take $Z \sim \mathcal{N}_q(0, \mathbf{I}_q)$ across the clusters. A possible generalisation of this would be to assume that intrinsic dimensionalities vary across the clusters. That would mean taking

$$X = \Lambda_j Z_j + \boldsymbol{\mu}_j + \varepsilon, \quad j = 1, \ldots, k. \tag{6}$$

with $Z_j \sim \mathcal{N}_{q_j}(0, \mathbf{I}_{q_j})$, q_j being the intrinsic dimensionality of the j-th cluster. For now, we restrict ourselves to the simple case where q is the same across the clusters as described in Eq. (5).

If we keep the same assumptions as used above for both the factor analysis model and the mixture of distributions model, it is easy to show that $\mathbf{p}(\mathbf{x} \mid Y = j) = \mathcal{N}_p(\mathbf{x}; \boldsymbol{\mu}_j, \Lambda_j \Lambda_j^{\mathsf{T}} + \Sigma)$, and that the MFA model is nothing but a finite mixture of multivariate Gaussians with the marginal density of X given by

$$\mathbf{p}(\mathbf{x}) = \sum_{j=1}^{k} \boldsymbol{\pi}_j \mathcal{N}_p\big(\mathbf{x}; \boldsymbol{\mu}_j, \Lambda_j \Lambda_j^{\mathsf{T}} + \Sigma\big). \tag{7}$$

For convenience, we define a new version of our discrete categorical latent variable Y as a vector $\mathbf{y} = (y_1, \ldots, y_k)^{\mathsf{T}}$ of indicator variables: $y_j = 1$ if $Y = j$ and $y_j = 0$ if $Y \neq j$. $\mathbf{y}$ is obviously a k-dimensional vector with $y_j \in \{0, 1\}$ for $j = 1, \ldots, k$, and $\sum_{j=1}^{k} y_j = 1$. The complete-data density is therefore given by

$$\mathbf{p}(\mathbf{x}, \mathbf{y}, \mathbf{z}) \propto \prod_{j=1}^{k} \boldsymbol{\pi}_j^{y_j} [\mathcal{N}_p(\mathbf{x}; \boldsymbol{\mu}_j + \boldsymbol{\Lambda}_j \mathbf{z}, \boldsymbol{\Sigma})]^{y_j}. \tag{8}$$

With $\boldsymbol{\pi} = \{\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_k\}$, $\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k\}$ and $\boldsymbol{\Lambda} = \{\Lambda_1, \ldots, \Lambda_k\}$, we define our complete collection of model parameters as $\boldsymbol{\theta} \equiv \{\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Lambda}, \Sigma\}$.

3. Estimation and inference via Bayesian sampling

3.1. The data augmentation algorithm for MFA

Being a typical incomplete-data problem, the inferential task inherent in the MFA model naturally lends itself to two-stage iterative algorithms where the first stage imputes values to the missing (unobserved) data while the second stage performs the estimation on the complete-data. The EM algorithms derived by Ghahramani and Hinton (1997) and McLachlan and Peel (2000) provide a deterministic likelihood-based application of such a two-stage procedure. In fact, the EM algorithm for MFA simply combines elements of the now standard EM algorithm solutions for Gaussian mixtures on the one hand and Factor Analysis on the other hand. From a stochastic simulation perspective, Tanner and Wong's (1987) Data Augmentation (two-stage Gibbs sampler) algorithm turns out to be the natural[6] way to construct an efficient sampling scheme for the MFA model. In fact, Diebolt and Robert (1994), Robert (1996a, 1996b), Richardson and Green (1997), and Escobar and West (1995), among others, have applied the method to mixtures (mostly of essentially univariate data), while Lopes and West (1999) and many others have used it for Factor Analysis. As we shall see, the Data Augmentation algorithm, sometimes referred to as the Imputation-Posterior algorithm, is a stochastic analogue of the EM algorithm in that it solves an incomplete-data problem by iteratively solving a complete-data version of it until some convergence criterion is satisfied. To specify the details of the corresponding sampling scheme, we use the complete-data density function of Eq. (8), and the corresponding *complete-data likelihood* function for the entire sample can be written as

follows:

$$\mathcal{L}(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}) \propto \prod_{i=1}^{n} \left(\prod_{j=1}^{k} \boldsymbol{\pi}_j^{y_{ij}} [\mathcal{N}_p(\mathbf{x}_i; \boldsymbol{\mu}_j + \Lambda_j \mathbf{z}_i, \Sigma)]^{y_{ij}} \right) \tag{9}$$

where $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ denote the total sets of observed data, component indicators and factor scores respectively. With the above likelihood function (9), we can write the joint posterior density of both the parameters and the latent variables as

$$\mathbf{p}(\boldsymbol{\theta}, \mathbf{Y}, \mathbf{Z} \mid \mathbf{X}) \propto \left[\prod_{i=1}^{n} \left(\prod_{j=1}^{k} \boldsymbol{\pi}_j^{y_{ij}} [\mathcal{N}_p(\mathbf{x}_i; \boldsymbol{\mu}_j + \Lambda_j \mathbf{z}_i, \Sigma)]^{y_{ij}} \right) \right] \mathbf{p}(\boldsymbol{\theta}) \tag{10}$$

Essentially, the steps of the derived sampling scheme are the following:

Algorithm 1: The Data Augmentation Algorithm for MFA.

Assuming the current values on the chain to be $\boldsymbol{\theta}^{(t)}$, $\mathbf{Y}^{(t)}$, $\mathbf{Z}^{(t)}$,
Imputation step: Impute some values for the missing latent variables.
Simulate $\mathbf{Y}^{(t+1)} \sim \mathbf{p}(\mathbf{Y} \mid \boldsymbol{\theta}^{(t)}, \mathbf{X}, \mathbf{Z}^{(t)})$
Simulate $\mathbf{Z}^{(t+1)} \sim \mathbf{p}(\mathbf{Z} \mid \boldsymbol{\theta}^{(t)}, \mathbf{X}, \mathbf{Y}^{(t+1)})$
Posterior step: Draw new parameter values given the augmented data.
Simulate $\boldsymbol{\theta}^{(t+1)} \sim \mathbf{p}(\boldsymbol{\theta} \mid \mathbf{X}, \mathbf{Y}^{(t+1)}, \mathbf{Z}^{(t+1)})$

Note. In the spirit of the Gibbs sampler, the equilibrium distribution reached by Algorithm 1 should provide samples from posterior marginals $\mathbf{p}(\boldsymbol{\theta} \mid \mathbf{X})$, $\mathbf{p}(\mathbf{z} \mid \mathbf{X})$ and $\mathbf{p}(\mathbf{y} \mid \mathbf{X})$ that can then be used to obtain parameters estimates, estimated factor scores and data clustering respectively. In fact, as it turns out, the use of conjugate priors leads to full conditional posteriors that are all standard and easy to simulate, making the sampling scheme efficient.

3.2. Elements of prior specification and trapping states

The natural approach to prior specification in this context would be to use the standard hierarchical prior structure as given by

$$\mathbf{p}(\boldsymbol{\theta}) = \mathbf{p}(\boldsymbol{\pi} \mid \delta)\mathbf{p}(\boldsymbol{\mu} \mid \xi, \kappa)\mathbf{p}(\Sigma \mid \alpha, \tau)\mathbf{p}(\boldsymbol{\Lambda} \mid \eta, \Omega), \tag{11}$$

where $\delta, \xi, \kappa, \alpha, \tau, \eta, \Omega$ are hyperparameters. However, as reported by Robert and Casella (1999) and also noticed in our simulations, the use of the standard hierarchical prior structure for mixtures of Gaussians often leads to situations where a given component is allocated a very small number of observations, resulting in an almost zero probability for that component to be allocated more observations or to have some of its few observations allocated to any other component. In fact, it turns out that these almost-absorbing states in MCMC are the analogues of the singularities encountered in the MLE approach. In other words, if one

of the component covariance matrices $\Lambda_j \Lambda_j^{\mathsf{T}} + \Sigma$ is allowed to become extremely small (i.e have terms of very small magnitude) at any given sample point, then that component of the MFA will be allocated that single point, with no chance of having any other point allocated to it, since the fixed hyperparameter will obviously never change the state of the chain. Constraining all the covariance matrices to be equal is one of the solutions to this problem. However, such a constraint is clearly unrealistic in the majority of cases. A solution to the problem along the lines of Richardson and Green (1997) consists of adding an extra layer to the hierarchical prior structure in order to allow the hyperparameters of the covariance matrices of the components of the mixture to be stochastic quantities. Such an extension allows the covariance matrices to be *similar* without constraining them to be equal, and effectively allows the sampling scheme to explore extensively at least the current modal region of the posterior surface thereby increasing the chance escaping from trapping states.[7] In other words, instead of using the standard hierarchical prior structure of Eq. (11), we use the extended structure of Eq. (12), where an extra layer allows the component covariance matrices to be explored at least locally through the stochasticity of the hyperparameters of Λ:

$$\mathbf{p}(\boldsymbol{\theta}) = \mathbf{p}(\boldsymbol{\pi} \mid \delta)\mathbf{p}(\boldsymbol{\mu} \mid \xi, \kappa)\mathbf{p}(\Sigma \mid \alpha, \tau)\mathbf{p}(\boldsymbol{\Lambda} \mid \eta, \Omega)\mathbf{p}(\Omega \mid g, h), \tag{12}$$

where g and h are the hyperparameters of the hyperprior Ω. For both the parameters and the latent variables, we use conjugate priors. Details of the corresponding full conditional posterior distributions are given in the Appendix. As far as hyperparameters are concerned, Richardson and Green (1997) used data-dependent hyperpriors for mixtures of univariate normals. We simply extend and adapt some of their ideas to our multivariate context.

3.3. Dealing with label switching

Combining the use of a symmetric Dirichlet prior for the mixing weights with a likelihood that is invariant to permutations of labels, we end up with a posterior distribution that is also invariant to relabelling. This means that, for a k-component mixture, the posterior essentially has $k!$ modes of equal importance. During the MCMC iterative sampling procedure, samples of parameters drawn from the stationary (equilibrium) distribution are therefore likely to have originated from one of those $k!$ modal regions of the posterior surface.

Ideally, for parameters estimates to be meaningful, the samples used to estimate them have to have been drawn from the same modal region. While label switching is desirable in that it is an indication of good mixing and therefore good exploration of the posterior surface, a careless treatment of its effect would lead to meaningless parameter estimates.

In practice, as we noticed throughout our simulations, label switching does not happen very often when the Gibbs sampler is used, since the Gibbs sampler is not very good at jumping between different modal regions of the posterior surface of the parameters. In a sense, this might be good for parameter estimation for reasons given above, but can lead to very poor density estimation.

The use in this context of sampling strategies like simulated tempering (Celeux, Hurn, & Robert, 2000) allows better exploration of all the modal regions of the posterior surface, which results in good mixing and therefore many occurrences of label switching.

Many strategies have been used to address the difficult issue of label switching. The most natural approach, tested by Diebolt and Robert (1994), Richardson and Green (1997), Fokoué (2000), and Fokoué and Titterington (2000a) and many others, consists of imposing an ordering a priori to make sure that all the samples of the Markov chain come from the same mode of the posterior. In practice, one may decide to accept only samples satisfying the constraint $\pi_1 < \pi_2 < \cdots < \pi_k$ or in the univariate setting to impose an ordering on the means of the Gaussians, e.g. $\mu_1 < \mu_2 < \cdots < \mu_k$. Despite its intuitive nature, this approach leads to a poor representation of the geometry of the posterior surface. Besides, it cannot be easily extended to the multivariate setting and, worst of all, it leads to a high rejection rate (especially in multivariate settings) and considerably retards the sampler.

Some other solutions to this problem based on k-means-like clustering algorithms and the use of loss functions have been explored by Celeux (1998), Celeux, Hurn, and Robert (2000), and Stephens (2000a/b), and tested by Hurn, Justel, and Robert (2000) and Fokoué and Titterington (2000c). In this paper, we use an online clustering algorithm (Celeux, 1998; Celeux, Hurn, & Robert 2000) that consists of isolating one of the $k!$ modes (the mode of reference). The reader is referred to the cited references for details of the methods.

4. Stochastic model selection for MFA

While there are many cases in practice where k and q are known and/or fixed, as we assumed earlier, it must be said that these values are very often unknown in many real-life applications, and the study of the MFA model therefore needs to address their uncertainty.

At the root of model determination for finite mixtures lies the difficult question of *what makes a component a separate and homogeneous entity*? Isn't there always the possibility of a hierarchy of clusters with a given cluster being made up of its own inner clusters? In Factor Analysis, a similar problem arises in that it is hard to find principled methods to determine what makes a particular factor important. For instance, in exploratory factor analysis heavily used by psychologists, sociologists and psychometricians, ad hoc methods based on the eigenvalues of the sample correlation matrix were used quite satisfactorily until the development of more sophisticated methods based on the likelihood and information criteria. In fact, for both FA and finite mixtures, this difficult problem of model determination has been one of the burning issues over the years, captivating the interests of researchers from both the likelihood-based and Bayesian perspectives. From a likelihood-based standpoint, many versions of Akaike's AIC and various adaptations of likelihood ratio tests have been used. Many variants of BIC have also been used. The reader is referred to such references as Krzanowski and Marriott (1994, 1995), and Press (1972) for detailed coverage of these approaches.

This paper adopts a stochastic simulation approach based on the construction of an ergodic Markov chain having the posterior distribution of the complete collection of all the unknowns (including k, q and the latent variables) as its equilibrium distribution.

When the dimension of the parameter space is known and fixed, traditional MCMC algorithms like the Gibbs sampler or the Metropolis-Hastings and their hybrid versions are used to construct the ergodic Markov chains of interest. However, if this dimension is allowed to vary throughout the MCMC iterative procedure, the classical algorithms

mentioned earlier are no longer valid, and they have to be replaced by birth-and-death type algorithms capable of jumping between spaces of different dimensions.

In the Bayesian framework, Green's (1995) Reversible Jump Markov Chain Monte Carlo (RJMCMC) algorithm is one such algorithm. These algorithms make transitions based on extended versions of the classical MCMC detailed balanced requirement that take into account the varying dimensionality of the support of the parameters. Richardson and Green (1997) offer the most detailed and comprehensive presentation of the application of RJMCMC to the Bayesian analysis of mixtures of univariate distributions with an unknown number of components. Lopes and West (1999) applied an adaptation of RJMCMC to the factor analysis model with an unknown number of common factors, and obtained good results on both synthetic and real-life problems. More recently, Stephens (2000b), using ideas from stochastic geometry and spatial statistics, developed an alternative to RJMCMC based on the simulation of a continuous-time birth-and-death Markov marked point process. He applied the derived Birth-and-Death MCMC (BDMCMC) method to mixtures of univariate and bivariate Gaussians with unknown number of components, and obtained good results.

Despite the fact that RJMCMC is based on a discrete-time Markov process while BDMCMC is based on a continuous time Markov process, the two methods are essentially equivalent in that they both successfully construct ergodic Markov chains in spaces of varying dimensions. In fact, BDMCMC can be thought of as a limit of RJMCMC. However, for practical reasons and to a certain extent for computational convenience, this paper adopts the BDMCMC approach. To the best of our knowledge, no one has treated the MFA model in this way before.

The first reason is the modularity and portability of the BDMCMC scheme: we note that, unlike with RJMCMC, ideas developed in BDMCMC and applied to mixtures can be readily adapted to model selection in FA, making it a very appealing scheme for MFA. The fact that RJMCMC makes use of the latent variables is not appealing in our context in that it would be very complicated to apply it to our nested scheme. From a computational point of view, we find death rates calculated on the basis of the "importance" of the component easier to interpret than RJMCMC's birth-and-death moves occurring uniformly. Finally, while RJMCMC has been extensively used in the analysis of mixtures of univariate distributions, its extension to mixtures of multivariate distributions still poses many difficulties, such as the complexity of the Jacobian calculations, and this prevents it from being a good candidate method for an essentially multivariate model like the MFA. Since the BDMCMC scheme treats parameters as points (no ordering) in a point process, it does not make use of such identifiability constraints as ordering, and its extension to multivariate distributions such as the one we have in the MFA model is therefore straightforward. Moreover, in contrast to the RJMCMC, the method requires very little mathematical sophistication and is easy to implement and interpret.

The central idea behind this approach is to view each component parameter of the model as a point in the parameter space, and adapt the methodology of point process simulation to help construct a Markov chain with the posterior distribution of the parameters as its equilibrium distribution. The method developed is therefore general and applicable to every context where parameters can be treated as point processes. Ideas used in the BDMCMC scheme are similar to those developed by Grenander and Miller (1994) and Phillips and

Smith (1996) who approached this same problem of Bayesian model comparison via *jump diffusions*. However, it is fair to point out that the implementation of the schemes developed by Grenander and Miller (1994) and Phillips and Smith (1996) is more complicated than Stephens's (2000b) BDMCMC.

Definition. The mathematical definition of a point process[8] on $\mathbb{R}^d$ is as a random variable[9] taking values in a measurable space of families of all sequences $\varphi = \{\nu_1, \nu_2, \ldots, \nu_d\}$ of points in $\mathbb{R}^d$ satisfying two regularity conditions:

1. the sequence φ is locally finite (each bounded subset of $\mathbb{R}^d$ must contain only a finite number of points of φ),
2. the sequence is simple (with elements such that $\nu_i \neq \nu_j$ if $i \neq j$).

As we discussed earlier, FA and Finite Mixture models have in common the fact that they both yield posterior distributions that are invariant to permutations of the order of their parameters. In both cases, the collection of parameters can be viewed as a random configuration or point process. Throughout this section, we assume that the number of common factors varies across the clusters. In other words, each local factor analyser has its own internal dimension, q_j, $j = 1, \ldots, k$, and we define the k-dimensional vector $\mathbf{q} = \{q_1, \ldots, q_k\}$. The complete collection of our model parameters is now given by $\boldsymbol{\theta} = \{k, \mathbf{q}, \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Lambda}, \Sigma\}$. If we assume that $\mathbf{q}$ and k are unknown a priori, our aim in parameter estimation from a stochastic simulation perspective now extends to the construction of an ergodic Markov chain with the joint posterior distribution $\mathbf{p}(k, \mathbf{q}, \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Lambda}, \Sigma \mid \mathbf{X})$ as its equilibrium distribution. In the previous section, we constructed a Markov chain with $\mathbf{p}(\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Lambda}, \Sigma \mid k, \mathbf{q}, \mathbf{X})$ as its equilibrium distribution. Now, we must accommodate the two new *counting* random variables k and $\mathbf{q}$. Intuitively, we are in the presence of a two-level hierarchical counting process:

- *Within a factor analyser*: simulate a birth-death Markov point process to estimate the number of common factors in each local factor analyser.
- *Between factor analysers*: simulate a birth-death Markov point process to estimate the number of components k.

A pseudocode for the overall sampling scheme would look like the following:

Algorithm 2: Stochastic model selection for MFA.

Assuming a current set $(k^{(t)}, \mathbf{q}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{\mu}^{(t)}, \boldsymbol{\Lambda}^{(t)})$ of parameters,
Simulate $k^{(t+1)}$ through a run of BDMCMC for mixtures.
For $j = 1, \ldots, k^{(t+1)}$
  Simulate $q_j^{(t+1)}$ through a run of BDMCMC for FA.
End
Set $\mathbf{q}^{(t+1)} = (q_1^{(t+1)}, \ldots, q_{k^{(t+1)}}^{(t+1)})$
Simulate $(\boldsymbol{\pi}^{(t+1)}, \boldsymbol{\mu}^{(t+1)}, \boldsymbol{\Lambda}^{(t+1)})$ via Algorithm 1 given $(k^{(t+1)}, \mathbf{q}^{(t+1)})$

The simulation of the type of birth-and-death process that we use in this paper has been extensively studied and applied in recent years, and the reader is referred to references like Stoyan, Kendall, and Mecke (1995) and Barndorff-Nielsen, Kendall, and van Lieshout, (1999) for comprehensive coverage of applications of such sampling schemes in stochastic geometry and spatial statistics. Baddeley (1994) and van Lieshout (1994) also provide very useful insights into other aspects of such sampling schemes. Stephens (2000b) provides a comprehensive coverage of his application of BDMCMC to mixtures. We therefore refer the reader to his paper for details. Here we focus our adaptation of BDMCMC to factor analysis.

4.1. BDMCMC for factor analysis

From our previous arguments, the number of common factors is nothing but the number of columns of Λ. We showed earlier that $\Lambda\Lambda^{\mathsf{T}} + \Sigma$ is invariant to permutations of axes in Λ. From a Bayesian perspective, Λ is therefore a "random configuration" or point process. For simplicity, we adopt a vector notation for Λ, say $\mathcal{C} = \{\Lambda_{.1}, \Lambda_{.2}, \ldots, \}$. We also simplify further and use $\mathbf{p}(\mathcal{C})$ in place of $\mathbf{p}(\cdot \mid \mathbf{X})$.

Our aim is to construct a Markov chain with $\mathbf{p}(q, \Lambda, \mu, \Sigma \mid \mathbf{X})$ as its stationary distribution. As shown by Stephens (2000a/b), such a Markov chain can be constructed by simulating a birth-and-death process satisfying the detailed balance equation

$$(q+1)d(\mathcal{C}; \nu)\mathbf{p}(\mathcal{C} \cup \{\boldsymbol{\nu}\}) = \beta(\mathcal{C})b(\mathcal{C}; \nu)\mathbf{p}(\mathcal{C}), \tag{13}$$

where $\mathbf{p}(\mathcal{C} \cup \{\boldsymbol{\nu}\})$ represents the posterior density of a configuration with $q+1$ points, $b(\mathcal{C}; \nu)$ and $d(\mathcal{C}; \nu)$ represent the birth and the death density functions respectively, while $\beta(\mathcal{C}) = \beta$ is a constant birth rate. Intuitively, Eq. (13) means that, under the equilibrium distribution $\mathbf{p}(\cdot|\mathbf{X})$, transitions from $\mathcal{C}$ into $\mathcal{C} \cup \{\boldsymbol{\nu}\}$ are exactly matched by transitions from $\mathcal{C} \cup \{\boldsymbol{\nu}\}$ into $\mathcal{C}$. If we use a truncated Poisson prior for q with hyperparameter α, that is, $\mathbf{p}(q) = Po(\alpha)$, the resulting sampling scheme is as follows:

Algorithm 3: Birth-death point process for FA.

Choose $\beta(\mathcal{C}) = \beta$, and set $t_{fa} = 0$ and $q = q^{(t-1)}$
Repeat
  Compute $\delta_j(\mathcal{C}) = \dfrac{\mathcal{L}(\mathcal{C}\backslash\Lambda_{.j})}{\mathcal{L}(\mathcal{C})}\dfrac{\beta}{\alpha}$ for $j = 1, \ldots, q$
  Compute $\delta(\mathcal{C}) = \sum_{j=1}^{q} \delta_j(\mathcal{C})$
  Simulate $s \sim \mathsf{Exp}(1/(\beta(\mathcal{C}) + \delta(\mathcal{C}))$ and Set $t_{fa} = t_{fa} + s$
  If (Ber$(\beta(\mathcal{C})/(\beta(\mathcal{C}) + \delta(\mathcal{C})) = 1)$ /* It is a birth */
    Set $q = q + 1$
    Simulate $\Lambda_{.q} \sim \mathcal{N}(0, \mathbf{I})$
    Set $\mathcal{C} = \mathcal{C} \cup \{\Lambda_{.q}\}$
  Else /* It is a death */
    Simulate $j' =$ Mn$(\delta_1(\mathcal{C})/\delta(\mathcal{C}), \ldots, \delta_q(\mathcal{C})/\delta(\mathcal{C}))$
    Set $\mathcal{C} = \mathcal{C}\backslash\{\Lambda_{.j'}\}$
    Set $q = q - 1$
Until $(t_{fa} \geq \rho)$

Ber, Exp and Mn represent the Bernouilli, the exponential and the multinomial distributions respectively. While $\mathcal{C}\backslash\{\Lambda_{.j'}\}$ represents the deletion of column $\Lambda_{.j'}$ from the current configuration $\mathcal{C}$. Our simulations reveal that the above algorithm is insensitive to initial conditions and always infer the right number of common factors regardless of whether we start the chain with one factor ($q = 1$) or with many factors ($q = q_{\max}$).

5. Implementation and results

All our simulations are written in Matlab 6.0 for Unix. We are currently optimising the codes and including pieces of C code wherever we encounter loops that cannot be easily "vectorised" and that slow down the program. In this section, we present two simulations, one based on the real-life and moderately high-dimensional ($p = 13$) wine data set, and the other based on a synthetic dataset that we generated to illustrate our methods.

5.1. Example 1: Analysis of the wine data set

For the wine data set, $p = 13$ and the hypothesised number of classes is $k = 3$. In this analysis, our aims were clustering, estimation of the intrinsic dimensionality of the underlying factor analysers, and density estimation. In fact, with $p = 13$, the full covariance matrix for each component would have 91 free parameters to be estimated, an estimation that would be very inefficient and prone to overfitting with the small sample of only $n = 178$ observations. The use of the MFA model is therefore justified for this task. We assumed that all three hypothetical classes (components) had the same[10] intrinsic dimensionality q, and we therefore used one separate BDMCMC simulation to estimate q, using $T_o = 9500$ burn-in iterations, $\beta = 0.618$ as our overall constant birth-rate, and $M = 2500$ useful final MCMC samples. Figure 1 (right) shows the histogram of the relative frequencies of values of q as produced by the sample path of the Markov chain obtained from the BDMCMC. This

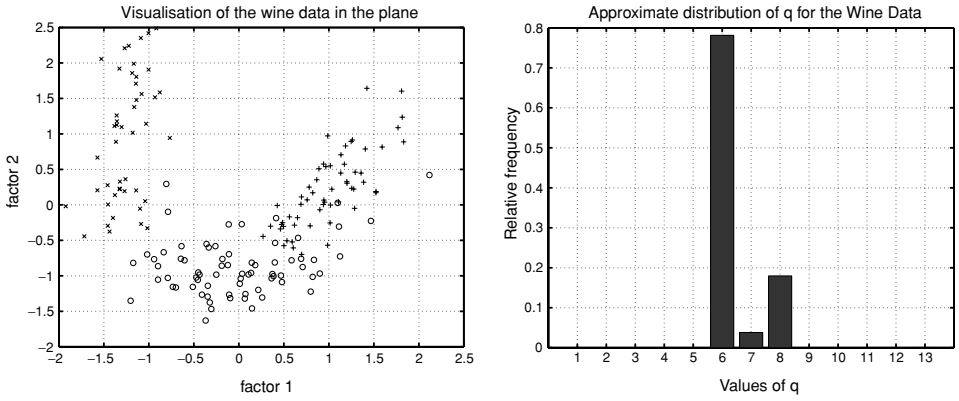

Figure 1. (Left) Visualisation of the 2-factor projection of the wine data set. (Right) Histogram of the approximate distribution of q for the wine data set.

unequivocally suggests that $q = 6$ would be the intrinsic dimensionality of the wine data. The good news is that the result obtained here by stochastic simulation is consistent with the one obtained by McLachlan and Peel (2000) through the use of sequential likelihood ratio tests. As far as clustering is concerned, we ran Algorithm 1 with $k = 1$ and $q = 2$ to project the data on to the plane and therefore explore its group structure. We plotted the estimated factor scores as shown in figure 1 (left). At least in the plane, the figure seems to agree with the hypothesis that there could be 3 classes of wines. We then ran the algorithm with $k = 3$, using different values of q, and the overall clustering performance was good, ranging from 95% to 98%. The highest loglikelihood was obtained in this case with $q = 6$, suggesting that the best density estimates would come from an MFA with $q = 6$. However, the best clustering performance came from $q = 2$ and $q = 3$.

5.2. Example 2: Analysis of simulated data

This second example is purely illustrative, and is based on simulated data. We generated a synthetic data set from a Mixture of Factor Analysers with $k = 3$, $p = 9$ and $q = 2$. With $T_o = 12000$ burn-in iterations, $M = 2000$ useful MCMC final samples, the algorithm easily inferred $q = 2$ as shown in figure 2 (left). Running the algorithm for the estimation of k, we also obtained accurate inference. Not surprisingly in this toy problem is the fact that the algorithm easily achieves 100% rate of good clustering and accurate parameter estimates, somehow confirming the perfect separation of the clusters in figure 2 (right).

6. Discussion

We have developed a stochastic simulation based algorithm for the analysis of the Mixture of Factor Analysers model. Our experiments show that our approach performs well in

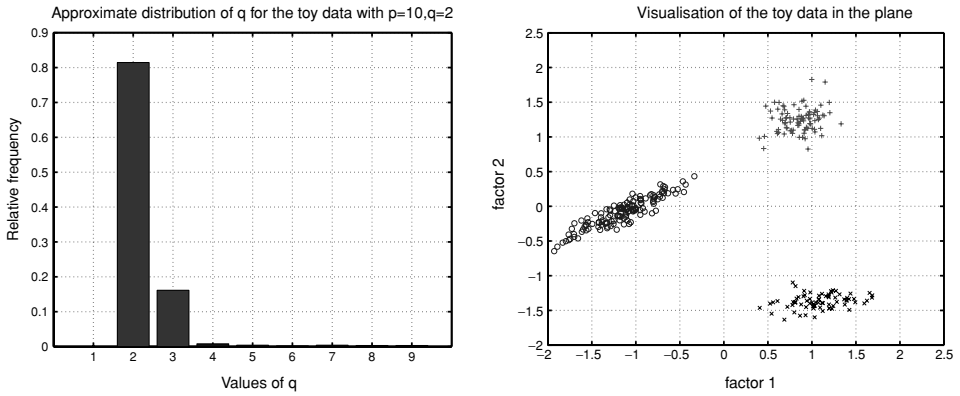

Figure 2. (Left) Histogram of the approximate distribution of q for the synthetic data ($q = 2$) as produced by the sample path of the Markov chain. (Right) Visualisation of the 2-factor projection of the 3-component synthetic task with known and fixed $q = 2$. The perfect separation is certainly due to the fact that the data are projected on to the intrinsic dimensionality.

parameter estimation, clustering, density estimation and model selection. We have not yet tested our sampling scheme on very high-dimensional tasks like handwritten digits recognition or image reconstruction, but we are actively working on devising faster sampling schemes that should handle such tasks in practically acceptable computing times. One of the main drawbacks of Data Augmentation in this very high-dimensional context is the fact the sample paths of the latent variables are so far stored throughout the iterations. This easily becomes explosive even for problems with latent spaces of moderately high dimensions. At this stage of the development of our approach, we strongly believe that, for the method to be fully applicable to real-life tasks, we will have to rely more on online versions of our algorithm, or marginalise over some of the latent variables in order to avoid storage. In fact, our results on the wine data were obtained using an online version of our method. Our simulations reveal that the use of an extra layer in the hierarchical prior structure effectively eliminates singularities and therefore achieves an advantage over the EM algorithm, for which we noticed many occurrences of singularities. However, despite escaping singularities, we still noticed rather poor mixing of the chains when k and q were known and fixed. An improvement on this might come from the use of tempered transitions, and we are exploring a simulated tempering version of our algorithm to achieve better exploration of the posterior surface. We only used vague conjugate priors throughout our study. We did this partly for computationally convenience, but also because these priors have produced excellent results in similar contexts (Richardson & Green, 1997; Diebolt & Robert, 1994) and have somehow become standard. It would be nice to be fully Bayesian and consider the use of more informative priors, but their incorporation in the sampling scheme could be very difficult and could destroy some nice properties of the Markov chains. Our adaptation of BDMCMC to Factor Analysis is probably the aspect of our proposed method that does not require much extra work. It works very well so far on both synthetic and real-life tasks. The nested scheme, however, requires some improvements, especially on the derivation of an adaptive birth rate that would evolve dynamically as a likelihood related function, allowing only likely models to be born. We are exploring ideas from van Lieshout (1994), Stoyan, Kendall, and Mecke (1995), and Barndorff-Nielsen, Kendall, and van Lieshout (1999) to find solutions to this problem. Overall, our results suggest that the method we have proposed is a good alternative to the EM algorithm and Variational approximations. We believe that a careful study of the limitations noticed so far would lead to better sampling schemes that would then be fully applicable to truly high-dimensional Machine Learning tasks. Finally, we acknowledge the very recent appearance of Utsugi and Kumagai (2001), who set out basic MCMC principles for MFAs along the lines also reported by Fokoué (2000).

Appendix

A.1. Imputation step for MFA

This step consists in simulating samples from the conditional posterior distributions of the latent variables. It can be easily shown that Y has a multinomial conditional posterior distribution, denoted here by Mn, and that Z has Gaussian distribution. More specifically,

for $j = 1, \ldots, k$, we have that

$$[\mathsf{y}_i | \cdots] \sim \mathsf{Mn}(1, \boldsymbol{\pi}^*_{1i}, \ldots, \boldsymbol{\pi}^*_{ki}) \text{ with } \boldsymbol{\pi}^*_{ji} \propto \boldsymbol{\pi}_j \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_j + \Lambda_j \mathbf{z}_i, \Sigma)$$
$$[\mathbf{z}_{i:\mathsf{y}_i=j} | \cdots] \sim \mathcal{N}\big(\big(\mathbf{I} + \Lambda_j^\mathsf{T} \Sigma^{-1} \Lambda_j\big)^{-1} \Lambda_j^\mathsf{T} \Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_j), \big(\mathbf{I} + \Lambda_j^\mathsf{T} \Sigma^{-1} \Lambda_j\big)^{-1}\big).$$

A.2. Full conditional posteriors

For $\boldsymbol{\pi}$, we use a symmetric Dirichlet prior distribution, $\boldsymbol{\pi} \sim \mathsf{Di}(\delta, \ldots, \delta)$, which gives a Dirichlet full conditional posterior:

$$[\boldsymbol{\pi} | \cdots] \sim \mathsf{Di}(\delta + n_1, \ldots, \delta + n_k),$$

where $n_j = \sharp\{i : \mathsf{y}_i = j\}$, for $j = 1, \ldots, k$, denotes the number of observations currently allocated to component j of the mixture. For each μ_j, we use the Gaussian prior $\mu_j \sim \mathcal{N}(\xi, \kappa)$ which gives a Gaussian full conditional posterior:

$$[\mu_j | \cdots] \sim \mathcal{N}((n_j \Sigma^{-1} + \kappa^{-1})^{-1} (n_j \Sigma^{-1} \bar{\bar{\mathbf{x}}}_j + \kappa^{-1} \xi), (n_j \Sigma^{-1} + \kappa^{-1})^{-1}),$$

where $\bar{\bar{\mathbf{x}}}_j = \frac{1}{n_j} \sum_{i:\mathsf{y}_i=j}^{n} (\mathbf{x}_i - \Lambda_j \mathbf{z}_i)$, for $j = 1, \ldots, k$.

Since $\Sigma^{-1} = \mathsf{diag}(\sigma_1^{-2}, \ldots, \sigma_p^{-2})$, we use independent Gamma conjugate priors for each σ_r^{-2}, namely $\sigma_r^{-2} \sim \mathsf{Ga}(\alpha, \tau)$, for $r = 1, \ldots, p$. We also define the matrix $S = \sum_{j=1}^{k} \sum_{i:\mathsf{y}_i=j}^{n} (\mathbf{x}_i - \Lambda_j \mathbf{z}_i - \mu_j)(\mathbf{x}_i - \Lambda_j \mathbf{z}_i - \mu_j)^\mathsf{T}$. From all that, we easily derive a Gamma full conditional conjugate posterior of the following form:

$$\big[\sigma_r^{-2} \big| \cdots\big] \sim \mathsf{Ga}(\alpha + n/2, \tau + S_{rr}/2).$$

We define the column vector $\Lambda_{jr.} \in \mathbb{R}^q$, made up of the r-th row of the j-th matrix of factor loadings. We use the zero mean Gaussian prior $\Lambda_{jr.} \sim \mathcal{N}(0, \Omega)$ for $j = 1, \ldots, k$ and $r = 1, \ldots, p$. This gives a Gaussian full conditional posterior:

$$[\Lambda_{jr.} | \cdots] \sim \mathcal{N}\big(\big(\Omega^{-1} + \sigma_r^{-2}\big(\mathbf{Z}_j^\mathsf{T} \mathbf{Z}_j\big)\big)^{-1} \big(\sigma_r^{-2} \mathbf{Z}_j^\mathsf{T} \ddot{\mathbf{X}}_{\cdot jr}\big), \big(\Omega^{-1} + \sigma_r^{-2}\big(\mathbf{Z}_j^\mathsf{T} \mathbf{Z}_j\big)\big)^{-1}\big),$$

where $\ddot{\mathbf{X}}$ is the data matrix obtained from $\ddot{\mathbf{x}}_j = \mathbf{x} - \mu_j$. We assume Ω to be diagonal, and more precisely $\Omega^{-1} = \mathsf{diag}(\omega_1^{-2}, \ldots, \omega_q^{-2})$. We also define $B = \sum_{j=1}^{k} \sum_{r=1}^{p} \Lambda_{jr.} \Lambda_{jr.}^\mathsf{T}$. Since each $\Lambda_{jr.}$ has a Gaussian distribution, we use an independent Gamma conjugate prior for each ω_c^{-2}, for $c = 1, \ldots, q$. Finally, with $\omega_c^{-2} \sim \mathsf{Ga}(g, h)$, we easily derive a Gamma full conditional conjugate posterior distribution for ω_c^{-2}:

$$\big[\omega_c^{-2} \big| \cdots\big] \sim \mathsf{Ga}(g + kp/2, h + B_{cc}/2).$$

Acknowledgments

The first author wishes to thank Dr. Agostino Nobile, the associate editors and the referees for their constructive and helpful comments and suggestions.

Notes

1. This data set is available at the Machine Learning repository of the University of California, Irvine (Merz, Murphy, & Aha, 1997).
2. Oblique factor analysis deals with cases where factors are assumed to be correlated, but we restrict ourselves here to simpler structures with uncorrelated factors.
3. In some applications, the meanings of parameters are not relevant. In such cases, FA is just a means to an end, so that all that is needed is any set of factor scores providing a valid reduced representation of the manifest vector. In such situations, one need not worry about identifiability.
4. We assume Λ to be full rank, so we constrain its "diagonal" elements to be nonzero.
5. A good understanding of MFAs should form a good starting point for estimating Mixtures of Multivariate Regressions.
6. Celeux, Hurn, and Robert (2000), and Hurn, Justel, and Robert (2000) have used versions of Langevin Metropolis and random walk for mixtures in univariate settings without any significant gain in performance. Besides, it is not easy to extend them to multivariate settings.
7. It is fair to point out that, while this is feasible via the use of an extended prior structure, such a flexible and principled solution is not available in the deterministic setting of the EM algorithm.
8. See Stoyan, Kendall, and Mecke (1995) for a detailed version.
9. Grenander and Miller (1994) used the term *random configurations* to refer to point processes.
10. We could well have used the nested model selection algorithm for MFA, but, because the small size ($n = 178$) of the data does not allow each component to have enough items of information for efficient estimation of each separate q_j, we thought it more sensible to use the available data for estimating only one q.

References

Arminger, G., Stein, P., & Wittenberg, J. (1999). Mixtures of conditional mean and covariance structure models. *Psychometrika, 64,* 475–494.

Baddeley, A. J. (1994). Discussion on *representation of knowledge in complex systems* by Grenander and Miller. *Journal of the Royal Statistical Society, Series* B*, 56*, 584–585.

Barndorff-Nielsen, O. E., Kendall, W. S., & van Lieshout, M. N. M. (1999). *Stochastic geometry: Likelihood and computation*. Monographs on Statistics and Applied Probability. London: Chapman and Hall.

Bartholomew, D. J. (1987). *Latent variable models and factor analysis*, Griffin's Statistical Monographs and Courses. Charles Griffin & Company Limited.

Celeux, G. (1998). Bayesian inference for mixtures: The label switching problem. In J. Payne, & P. Green (Eds.), *Proceedings in Computational Statistics 1998*. pp. 227–232, Physica-Verlag.

Celeux, G., Hurn, M., & Robert, C. P. (2000). Computational and inferential difficulties with mixture posterior distributions. *Journal of the American Statistical Association, 95*, 957–970.

Diebolt, J., & Robert, C. P. (1994). Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society, Series* B*, 56*, 363–375.

Dolan, C. V., & Van der Maas, H. L. J. (1998). Fitting multivariate normal finite mixtures subject to structural equation modelling. *Psychometrika, 63*, 227–253.

Dong, K. K., & Taylor, J. M. G. (1995). The restricted EM algorithm for maximum likelihood estimation under linear restrictions on the parameters. *Journal of the American Statistical Association, 90*, 707–716.

Escobar, M. D., & West, M. (1995). Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association, 90*, 577–588.

Everitt, B. S. (1984). *An introduction to latent variable models*. Monographs on Statistics and Applied Probability. Chapman and Hall.

Everitt, B. S., & Hand, D. J. (1981). *Finite mixture distributions*. Monographs on Statistics and Applied Probability. Chapman and Hall.

Fokoué, E. (2000). A Markov chain Monte Carlo (MCMC) approach to the Bayesian analysis of mixtures of factor analysers. In W. Jansen, & J. G. Bethlehem (Eds.), *Proceedings in Computational Statistics 2000, Short Communication and Posters* (pp. 29–30), Statistics The Netherlands.

Fokoué, E., & Titterington, D. M. (2000a). Bayesian sampling for mixtures of factor analysers. Technical Report No. 00-3, Department of Statistics, University of Glasgow, Glasgow, G12 8QW, Scotland, UK.

Fokoué, E., & Titterington, D. M. (2000b). Mixtures of factor analysers with fixed observed covariates. Technical Report No. 00-4, Department of Statistics, University of Glasgow, Glasgow, G12 8QW, Scotland, UK.

Fokoué, E., & Titterington, D. M. (2000c). Stochastic model selection for mixtures of factor analysers. Technical Report No. 00-5, Department of Statistics, University of Glasgow, Glasgow, G12 8QW, Scotland, UK.

Gelman, A., Meng, S. L., & Stern, H. (1996). Posterior predictive assessment of model fitness via realised discrepancies. *Statistica Sinica, 6*, 733–759.

Ghahramani, Z., & Hinton, G. E. (1997). The EM algorithm for mixtures of factor analysers. Technical Report CRG-TR-96-1, Department of Computer Science, University of Toronto, 6 King's College Road, Toronto, Canada, M5S 1A4.

Ghahramani, Z., & Beal, M. (2000). Variational inference for Bayesian mixture of factor analysers. In S. A. Solla, T. K. Leen, & K.-R. Müller (Eds.) *Advanced in neural information processing systems* 12. Cambridge, MA: MIT Press.

Gilks, W. R., Richardson, S., & Spiegelhalter, D. J. (1996). *Markov chain Monte Carlo in practice*. Interdisciplinary Statistics. Chapman and Hall.

Grenander, U., & Miller, M. I. (1994). Representation of knowledge in complex systems. *Journal of the Royal Statistical Society, Series* B*, 56*, 549–603.

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

Hurn, M., Justel, A., & Robert, C. P. (2000). Estimating mixtures of regressions. Technical Report, Department of Mathematical Sciences, University of Bath, UK.

Krzanowski, W. J., & Marriott, F. H. C. (1994). *Multivariate analysis, Part 1: Distribution, ordination and inference*, Kendall's Library of Statistics 1. Arnorld.

Krzanowski, W. J., & Marriott, F. H. C. (1995). *Multivariate analysis, Part 2: Classification, covariance structures and repeated measurements*, Kendall's Library of Statistics 2. Arnorld.

van Lieshout, M. N. M. (1994). Discussion on *representation of knowledge in complex systems* by Grenander and Miller. *Journal of the Royal Statistical Society, Series* B*, 56*, 585.

Lopes, H. F., & West, M. (1999). Model uncertainty in factor analysis. Technical Report, Institute of Statistics and Decision Sciences, Duke University, USA.

McLachlan, G., & Peel, D. (2000). *Finite mixture models*, Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons.

Phillips, D. B., & Smith, A. F. M. (1996). Bayesian model comparison via jump diffusions. In W. R. Gilks, S. Richardson, & D. J. Spiegelhalter (Eds.), *Markov chain Monte Carlo in practice*. Interdisciplinary Statistics (ch. 13, pp. 215–239). Chapman and Hall.

Press, S. J. (1972). *Applied multivariate analysis*. Holt, Rinehart and Winston.

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

Robert, C. P. (1996a). Mixtures of distributions: Inference and estimation. In W. R. Gilks, S. Richardson, & D. J. Spiegelhalter (Eds.), *Markov chain Monte Carlo in practice*, Interdisciplinary Statistics (ch. 24, pp. 441–464). Chapman & Hall.

Robert, C. P. (1996b). *Méthodes de Monte Carlo par Chaines de Markov*, Statistique Mathématique et Probabilité. Economica.

Robert, C. P., & Casella, G. (1999). *Monte Carlo statistical methods*. Springer.

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

Stephens, M. (2000b). Bayesian analysis of mixtures models with an unknown number of components-an alternative to reversible jump methods. *Annals of Statistics, 28*, 40–74.

Stoyan, D., Kendall, W. S., & Mecke, J. (1995). *Stochastic geometry and its applications*, 2nd edn., Wiley Series in Probability and Statistics. John Wiley and Sons.

Tanner, M. A., & Wong, W. H. (1987). The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association, 82*, 529–550.

Tipping, M., & Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society*, B*, 61,* 611–622.

Titterington, D. M., Smith, A. F. M., & Makov, U. E. (1985). *Statistical analysis of finite mixture distributions*, Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons.

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

Utsugi, A., & Kumagai, T. (2001). Bayesian analysis of mixture of factor analysers. *Neural Computation, 13*, 993–1002.

Yung, Y. F. (1997). Finite mixtures in confirmatory factor analysis models. *Psychometrika, 62*, 297–330.

Received August 2, 2000
Revised April 18, 2001
Accepted August 17, 2001
Final manuscript August 26, 2001