Unsupervised Bayesian linear unmixing of gene expression microarrays

Cécile Bazot, Nicolas Dobigeon, Jean-Yves Tourneret, Aimee K. Zaas, Geoffrey S. Ginsburg, Alfred O. Hero

Unsupervised Bayesian linear unmixing of gene expression microarrays

Cécile Bazot[1*], Nicolas Dobigeon[1*], Jean-Yves Tourneret[1], Aimee K Zaas[2], Geoffrey S Ginsburg[2] and Alfred O Hero III[3]

Abstract

Background: This paper introduces a new constrained model and the corresponding algorithm, called unsupervised Bayesian linear unmixing (uBLU), to identify biological signatures from high dimensional assays like gene expression microarrays. The basis for uBLU is a Bayesian model for the data samples which are represented as an additive mixture of random positive gene signatures, called *factors*, with random positive mixing coefficients, called *factor scores*, that specify the relative contribution of each signature to a specific sample. The particularity of the proposed method is that uBLU constrains the factor loadings to be non-negative and the factor scores to be probability distributions over the factors. Furthermore, it also provides estimates of the number of factors. A Gibbs sampling strategy is adopted here to generate random samples according to the posterior distribution of the factors, factor scores, and number of factors. These samples are then used to estimate all the unknown parameters.

Results: Firstly, the proposed uBLU method is applied to several simulated datasets with known ground truth and compared with previous factor decomposition methods, such as principal component analysis (PCA), non negative matrix factorization (NMF), Bayesian factor regression modeling (BFRM), and the gradient-based algorithm for general matrix factorization (GB-GMF). Secondly, we illustrate the application of uBLU on a real time-evolving gene expression dataset from a recent viral challenge study in which individuals have been inoculated with influenza A/H3N2/Wisconsin. We show that the uBLU method significantly outperforms the other methods on the simulated and real data sets considered here.

Conclusions: The results obtained on synthetic and real data illustrate the accuracy of the proposed uBLU method when compared to other factor decomposition methods from the literature (PCA, NMF, BFRM, and GB-GMF). The uBLU method identifies an inflammatory component closely associated with clinical symptom scores collected during the study. Using a constrained model allows recovery of all the inflammatory genes in a single factor.

Background

Factor analysis methods such as principal component analysis (PCA) have been widely studied and can be used for discovering the patterns of differential expression in time course and/or multiple treatment biological experiments using gene or protein microarray samples. These methods aim at finding a decomposition of the observation matrix $\mathbf{Y} \in \mathbb{R}^{G\times N}$ whose rows (respectively columns) are indexed by gene index (respectively sample index). Typically, in gene expression analysis, the number N of samples is much less than the number G of genes. For example, in an Affymetrix HU133 gene chip, the number G of genes may range from ten to twenty thousand depending on the type of chip description file (CDF) processing used to translate the oligonucleotide fragments to gene labels whereas we only analyze about a hundred of samples.

This decomposition expresses each of the N samples as a particular linear combination of R characteristic gene expression signatures, also referred to as *factors*, with appropriate proportions (or contributions), called *factor scores*, following a linear mixing model

$$\mathbf{Y} = \mathbf{MA} + \mathbf{N} \tag{1}$$

*Correspondence: cecile.bazot@enseeiht.fr; nicolas.dobigeon@enseeiht.fr
[1] University of Toulouse, IRIT/INP-ENSEEIHT, 2 rue Camichel, BP 7122, 31071 Toulouse cedex 7, France
Full list of author information is available at the end of the article

where $\mathbf{M} \in \mathbb{R}^{G\times R}$ represents the *factor loading* matrix, $\mathbf{A} \in \mathbb{R}^{R\times N}$ the factor score matrix and $\mathbf{N} \in \mathbb{R}^{G\times N}$ is a matrix containing noise samples. Each sample $\mathbf{y}_i$ ($i = 1,\ldots,N$), corresponding to the i-th column of the observed gene expression matrix $\mathbf{Y}$, is a vector of G gene expression levels that can be expressed as

$$\mathbf{y}_i = \sum_{r=1}^{R} \mathbf{m}_r a_{r,i} + \mathbf{n}_i = \mathbf{M}\mathbf{a}_i + \mathbf{n}_i \tag{2}$$

where $\mathbf{m}_r$ is the r-th column of $\mathbf{M}$, $a_{r,i}$ denotes the (r,i)-th element of the matrix $\mathbf{A}$, $\mathbf{a}_i$ and $\mathbf{n}_i$ are the i-th column of $\mathbf{A}$ and $\mathbf{N}$ respectively. The number of factors R in the decomposition is usually much less than the number of samples N. Traditional factor analysis methods such as PCA require the experimenter to specify the desired number of factors to be estimated. However, some recent Bayesian factor analysis methods are totally unsupervised in the sense that the number of factors is directly estimated from the data [1-3].

The model (1) is identical to the standard factor analysis model [4] for which the columns of $\mathbf{M}$ are called *factors* and should correspond to biological signatures (or pathways). Note that the elements of the matrix $\mathbf{M}$ are referred to as *factor loadings*, and the columns of $\mathbf{A}$ are the *factor scores*. Approaches to fitting the model (1) to data include principal component analysis [5,6], least squares matrix factorization [7,8], finite mixture modeling [9,10], and Bayesian factor analysis [4,11,12].

This paper presents a new Bayesian factor analysis method called unsupervised Bayesian linear unmixing (uBLU), that estimates the number of factors and incorporates non-negativity constraints on the factors and factor scores, as well as a sum-to-one constraint for the factor scores. The uBLU method presented here differs from the BLU method, developed in [13] for hyperspectral imaging and applied to gene microarray expression analysis in [14]. Note that BLU requires user specification of the number of factors while uBLU determines the number of factors using Bayesian birth-death model. The positivity and sum-to-one constraints are natural in gene microarray analysis when the entries of the observation matrix are non-negative and when a proportional decomposition is desired. Thus each factor score corresponds to the concentration (or proportion) of a particular factor to a given sample. The advantage of this representation for gene expression analysis is twofold: i) the factor scores correspond to the strengths of each gene contributing to that factor; ii) for each gene chip the factor scores give the relative abundance of each factor present in the chip. For example, a gene having a large loading level (close to one) for a particular factor should have a small loading (close to zero) for all other factors. In this way, as opposed to other factor analysis methods, there is less multiplexing making it easier to associate specific genes to specific factors and vice versa.

A similar approach, based on NMR spectral imaging and called the *Bayesian decomposition* (BD), has been previously developed by Moloshok *et al.* and applied to gene expression data [11]. More recently, the coordinated gene activity in pattern sets method (CoGAPS), available as an open R-source [12], combines the GAPS-MCMC matrix factorization algorithm with a threshold-independent statistic to infer activity in specific gene sets. However, these approaches require cold restarts of the algorithm with different number of factors and with different random seeds to avoid the large number of local minima encountered in the process of fitting the matrix factorization model $\mathbf{MA}$ to the data $\mathbf{Y}$. In contrast, the proposed uBLU algorithm uses a judicious model to reduce sensitivity to local minima rather than using cold restarts. The novelty of the uBLU model is that it consists of: (1) a birth-death process to infer the number of factors; (2) a positivity constraint on the loading and score matrices $\mathbf{M}$, $\mathbf{A}$ to restrict the space of solutions; (3) a sum-to-one constraint on the columns of $\mathbf{A}$ to further restrict the solution space. The uBLU model is justified for non-negative data problems like gene expression analysis and produces an estimate of the non-negative factors in addition to their proportional representation in each sample.

Bayesian linear unmixing, traditionally used for hyperspectral image analysis (see [13] for example), is one of many possible factor analysis methods that could be applied to gene expression analysis. These methods include non-negative matrix factorization (NMF) [7,8], independent component analysis (ICA) [15], Bayesian decomposition (BD) [11], PCA [5], bi-clustering [16], penalized matrix decomposition (PMD) [2], Bayesian factor regression modeling (BFRM) [1], and more recently the gradient-based algorithm of Nikulin *et al.* for general matrix factorization (GB-GMF) [17]. Contrary to uBLU, the PCA, ICA, BFRM, GB-GMF, bi-clustering and PMD methods do not account for non-negativity of the factor loadings and factor scores. On the other hand, NMF does not account for sum-to-one constraints on the columns of the factor score matrix. Contrary to PCA and ICA, uBLU does not impose orthogonality or independence on the factors, as well as the GB-GMF algorithm. These relaxed assumptions might better represent what is known about the preponderance of overlap and dependency in biological pathways. Finally, uBLU naturally accommodates Bayesian estimation of the number of factors, like BFRM. Note that BFRM has been specifically developed for gene expression data [1].

In this paper we provide comparative studies that establish quantitative performance advantages of the proposed constrained model and its corresponding uBLU algorithm

with respect to PCA, NMF, BFRM and GB-GMF for time-varying gene expression analysis, using synthetic data with known ground truth. We also illustrate the application of uBLU to the analysis of a real gene expression dataset from a recent viral challenge study [18] in which several subjects were administered viral inoculum and gene expression time course data were collected over a period of several days. Using these data, we may infer relationships between genes and symptoms and examine how the human host response to viral infection evolves with time.

Methods

Mathematical constrained model

Let $\mathbf{y}_i$ represent a gene microarray vector of G gene expression levels. The elements of $\mathbf{y}_i$ have units of hybridization abundance levels with non-negative values. In the context of gene expression data, the starting point for Bayesian linear unmixing is the linear mixing model (LMM)

$$\mathbf{y}_i = \sum_{r=1}^{R} \mathbf{m}_r a_{r,i} + \mathbf{n}_i, \tag{3}$$

where R is the number of distinct gene signatures that can be present in the chip, $\mathbf{m}_r = \left[m_{1,r}, \ldots, m_{G,r}\right]^T$ is the r-th gene signature vector, $m_{g,r} \geq 0$ is the strength of the g-th gene ($g = 1, \ldots, G$) in the r-th signature ($r = 1, \ldots, R$), and $a_{r,i}$ is the relative contribution of the r-th signature vector to the i-th sample $\mathbf{y}_i$, where $a_{r,i} \in [0, 1]$ and $\sum_{r=1}^{R} a_{r,i} = 1$. Here $\mathbf{n}_i$ denotes the residual error of the LMM representation. For a matrix of N data samples $\mathbf{Y} = \left[\mathbf{y}_1, \ldots, \mathbf{y}_N\right] \in \mathbb{R}^{G \times N}$, the LMM can be rewritten with matrix notations

$$\mathbf{Y} = \mathbf{M}\mathbf{A} + \mathbf{N}, \tag{4}$$

where $\mathbf{M} = [\mathbf{m}_1, \ldots, \mathbf{m}_R] \in \mathbb{R}^{G \times R}$, $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_N] \in \mathbb{R}^{R \times N}$ and $\mathbf{N} = [\mathbf{n}_1, \ldots, \mathbf{n}_N] \in \mathbb{R}^{G \times N}$ represent the factor score matrix, the factor loading matrix and the noise matrix, respectively. The matrices $\mathbf{M}$, $\mathbf{A}$ satisfy positivity and sum-to-one constraints defined by

$$m_{g,r} \geq 0, \quad a_{r,i} \geq 0, \quad \text{and} \quad [1, \ldots, 1]\,\mathbf{A} = [1, \ldots, 1], \tag{5}$$

where $m_{g,r}$ denotes the (g, r)-th element of matrix $\mathbf{M}$. The constraints (5) arise naturally when dealing with positive data for which one is seeking the relative contribution of positive factors that have the same numerical characteristics as the data, i.e., the signature $\mathbf{m}_r$ is itself interpretable as a vector of hybridization abundances.

The objective of linear unmixing is to simultaneously estimate the factor matrix $\mathbf{M}$ and the factor score matrix $\mathbf{A}$ from the available N data samples. The representation (1) is rank deficient since $\mathbf{A}$ has rank $N - 1$. This presents algorithmic challenges for solving the unmixing problem. Several algorithms have been proposed in the context of hyperspectral imaging to solve similar problems [6,19]. Most of these algorithms perform unmixing in a two step procedure where $\mathbf{M}$ is estimated first using an *endmember extraction algorithm* (EEA) followed by a constrained linear least squares step to estimate $\mathbf{A}$. A common (but restrictive) assumption in these algorithms is that some samples in the dataset are "pure" in the sense that the linear combination of (2) contains a unique factor, say $\mathbf{m}_r$, with factor score $a_{r,i}$. Recently, this assumption has been relaxed by applying a hierarchical Bayesian approach, called Bayesian linear unmixing (BLU). The resulting algorithm requires the number R of factors to be specified (see [13] for details). Here we extend BLU to a fully unsupervised algorithm, called unsupervised BLU (uBLU), that estimates R using a birth-death model and a Gibbs sampler. The Gibbs sampler produces an estimate of the entire joint posterior distribution of the model parameters, resulting in a fully Bayesian estimator of the number of factors R, the factor loadings $\mathbf{M}$, and the factor scores $\mathbf{A}$. The uBLU model is described in the next subsection and the Gibbs sampling algorithm is given in the Appendix. In the Results and discussion section below we demonstrate the performance advantages of uBLU as a factor analysis model for simulated and real gene expression data.

Unsupervised Bayesian linear unmixing algorithm

The BLU algorithm studied in [13] generates samples distributed according to the posterior distribution of $\mathbf{M}$ and $\mathbf{A}$ given the number R of factors for appropriate prior distributions assigned to the mixing parameters in (2). First, the residual errors $\mathbf{n}_i$ in (2) are assumed to be independent identically distributed (i.i.d.) according to zero-mean Gaussian distributions: $\mathbf{n}_i \sim \mathcal{N}\left(\mathbf{0}_G, \sigma^2 \mathbf{I}_G\right)$ for $i = 1, \ldots, N$, where $\mathbf{I}_G$ denotes the identity matrix of dimension $G \times G$.

The number of factors R to be estimated by the proposed uBLU algorithm is assigned a discrete uniform prior distribution on $[2, \ldots, R_{\max}]$

$$\mathrm{P}[R = k] = \frac{1}{R_{\max} - 1}, \quad \text{for } R = 2, \ldots, R_{\max}, \tag{6}$$

where $R_{\max}$ is the maximal number of factors present in the mixture.

Because of the constraints in (5), the data samples $\mathbf{y}_i$ ($i = 1, \ldots, N$) live in a lower-dimensional subspace of $\mathbb{R}^K$ (whose dimension is upper-bounded by $K - 1$) denoted as $\mathcal{V}_{K-1}$ ($R_{\max} - 1 \leq K \leq G$). This subspace can be identified by a standard dimension reduction procedure, such as PCA. Hence, instead of estimating the factor loadings $\mathbf{m}_r \in \mathbb{R}^G$ ($r = 1, \ldots, R$), we propose to estimate their

corresponding projections $\mathbf{t}_r \in \mathbb{R}^K$ onto this subspace. Specifically, these projections can be represented as

$$\mathbf{t}_r = \mathbf{P}(\mathbf{m}_r - \bar{\mathbf{y}}) \tag{7}$$

where $\bar{\mathbf{y}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{y}_i$ is the empirical mean of the data matrix $\mathbf{Y}$ and $\mathbf{P}$ is the $(K-1) \times G$ appropriate projection matrix that projects onto $\mathcal{V}_{K-1}$, which can be constructed from the principal eigenvectors of the empirical covariance matrix of $\mathbf{Y}$. This dimension reduction procedure allows us to work in a lower-dimensional subspace without loss of information, and reduces significantly the computational complexity of the Bayes estimator of the factor loadings. A multivariate Gaussian distribution (MGD) truncated on a subset $\mathcal{T}_r$ is chosen as prior distribution for the projected factors $\mathbf{t}_r$. The subset $\mathcal{T}_r$ is defined in order to ensure the non-negativity constraint on $\mathbf{m}_r$ (see [13])

$$\mathbf{t}_r \in \mathcal{T}_r \Leftrightarrow \{m_{g,r} \geq 0, \forall g = 1, \ldots, G\}. \tag{8}$$

More precisely, $\mathcal{T}_r$ is obtained by noting that $\mathbf{m}_r = \mathbf{P}^{-1}\mathbf{t}_r + \bar{\mathbf{y}}$ and by looking for the vectors $\mathbf{t}_r$ such that all components of $\mathbf{P}^{-1}\mathbf{t}_r + \bar{\mathbf{y}}$ are non-negative. To estimate the mean vectors $\mathbf{e}_r$ of these truncated MGDs, one can use a standard endmember extraction algorithm (EEA) common in hyperspectral imaging, e.g. N-FINDR [19]. To summarize, the prior distribution for the projected factor $\mathbf{t}_r$ is

$$\mathbf{t}_r | \mathbf{e}_r, s_r^2 \sim \mathcal{N}_{\mathcal{T}_r}\left(\mathbf{e}_r, s_r^2 \mathbf{I}_{R-1}\right) \tag{9}$$

where $\mathcal{N}_{\mathcal{T}_r}\left(\mathbf{e}_r, s_r^2 \mathbf{I}_{R-1}\right)$ denotes the truncated MGD with mean vector $\mathbf{e}_r$ and covariance matrix $s_r^2 \mathbf{I}_{R-1}$, with s_r^2 a fixed hyperparameter. Assuming the vectors $\mathbf{t}_r$, for $r = 1, \ldots, R$, are *a priori* independent, the prior distribution for the projected factor matrix $\mathbf{T} = [\mathbf{t}_1, \ldots, \mathbf{t}_R]$ is

$$f\left(\mathbf{T}|\mathbf{E}, \mathbf{s}^2, R\right) \propto \prod_{r=1}^{R} \exp\left[-\frac{\|\mathbf{t}_r - \mathbf{e}_r\|^2}{2s_r^2}\right] \mathbf{1}_{\mathcal{T}_r}(\mathbf{t}_r) \tag{10}$$

where $\propto$ stands for "proportional to", $\|\cdot\|$ is the standard l_2-norm, $\mathbf{1}_{\mathcal{X}}(\cdot)$ denotes the indicator function on the set $\mathcal{X}$, $\mathbf{E} = [\mathbf{e}_1, \ldots, \mathbf{e}_R]$ and $\mathbf{s}^2 = \left[s_1^2, \ldots, s_R^2\right]$.

The sum-to-one constraint for the factor scores $\mathbf{a}_i$, for each observed sample i $(i = 1, \ldots, N)$, allows this vector $\mathbf{a}_i$ to be rewritten as

$$\mathbf{a}_i = \begin{pmatrix} \mathbf{a}_{1:R-1,i} \\ a_{R,i} \end{pmatrix} \text{ with } \mathbf{a}_{1:R-1,i} = \left[a_{1,i}, \ldots, a_{R-1,i}\right]^T, \tag{11}$$

and $a_{R,i} = 1 - \sum_{r=1}^{R-1} a_{r,i}$. Note here that any component of $\mathbf{a}_i$ could be expressed as a function of the others, i.e., $a_{r,i} = 1 - \sum_{k \neq r} a_{k,i}$. The last component $a_{R,i}$ has been chosen here for notation simplicity. To ensure the positivity constraint, the subvectors $\mathbf{a}_{1:R-1,i}$ must belong to the simplex

$$\mathcal{S} = \{\mathbf{a}_{1:R-1,i} \mid \left\|\mathbf{a}_{1:R-1,i}\right\|_1 \leq 1 \text{ and } \mathbf{a}_i \geq \mathbf{0}\}, \tag{12}$$

where $\|\cdot\|_1$ is the l_1 norm ($\|\mathbf{a}_i\|_1 = \sum_{r=1}^{R} |a_{r,i}|$) and $\mathbf{a}_i \geq \mathbf{0}$ stands for the set of inequalities $\{a_{r,i} \geq 0\}_{r=1,\ldots,R}$. Following the model in [13], we propose to assign uniform distributions over the simplex $\mathcal{S}$ as priors for the subvectors $\mathbf{a}_{1:R-1,i}$ $(i = 1, \ldots, N)$, i.e.,

$$f\left(\mathbf{a}_{1:R-1,i}|R\right) = \mathbf{1}_{\mathcal{S}}\left(\mathbf{a}_{1:R-1,i}\right). \tag{13}$$

For the prior distribution on the variance σ^2 of the residual errors we chose a conjugate inverse-Gamma distribution with parameters $\nu/2$ and $\gamma/2$

$$\sigma^2|\nu, \gamma \sim \mathcal{IG}\left(\frac{\nu}{2}, \frac{\gamma}{2}\right). \tag{14}$$

The shape parameter ν is a fixed hyperparameter whereas the scale parameter γ will be adjustable, as in [13]. A non-informative Jeffreys' prior is chosen as prior distribution for the hyperparameter γ, i.e.,

$$f(\gamma) \propto \frac{1}{\gamma} \mathbf{1}_{\mathbb{R}^+}(\gamma). \tag{15}$$

The resulting hierarchical structure of the proposed uBLU model is summarized in the directed acyclic graph (DAG) presented in Additional file 1: Figure S1.

The model defined in (1) and the Gaussian assumption for the noise vectors $\mathbf{n}_1, \ldots, \mathbf{n}_N$ allow the likelihood of $\mathbf{y}_1, \ldots, \mathbf{y}_N$ to be determined

$$f(\mathbf{Y}|\mathbf{\Theta}) = \left(\frac{1}{2\pi\sigma^2}\right)^{\frac{GN}{2}} \exp\left[-\frac{\sum_{i=1}^{N} \left\|\mathbf{y}_i - \mathbf{M}\mathbf{a}_i\right\|^2}{2\sigma^2}\right]. \tag{16}$$

Multiplying this likelihood by the parameter priors defined in (10), (13), (14) and (6), and integrating out the nuisance parameter γ, the posterior distribution of the unknown parameter vector $\mathbf{\Theta} = \{\mathbf{M}, \mathbf{A}, \sigma^2, R\}$ can be expressed as

$$\begin{aligned} f(\mathbf{\Theta}|\mathbf{Y}) &= \int f(\mathbf{\Theta}, \gamma|\mathbf{Y})\, d\gamma \\ &\propto \int f(\mathbf{Y}|\mathbf{\Theta}) f(\mathbf{\Theta}|\gamma) f(\gamma)\, d\gamma. \end{aligned} \tag{17}$$

Considering the parameters to be *a priori* independent, the following result can be obtained

$$f(\mathbf{\Theta}|\gamma) = f(\mathbf{A}|R) f\left(\mathbf{T}|\mathbf{E}, \mathbf{s}^2, R\right) f\left(\sigma^2|\nu, \gamma\right) f(R) \tag{18}$$

where $f(\mathbf{A}|R)$, $f\left(\mathbf{T}|\mathbf{E}, \mathbf{s}^2, R\right)$ and $f\left(\sigma^2|\nu, \gamma\right)$ are respectively the prior distributions of the factor score matrix $\mathbf{A}$, the projected factor matrix $\mathbf{T}$ and the noise variance σ^2 previously defined.

Due to the constraints enforced on the data, the posterior distribution $f(\mathbf{M}, \mathbf{A}, R|\mathbf{Y})$ obtained from the proposed hierarchical structure is too complex to derive analytical expressions of the Bayesian estimators, e.g., the minimum mean square (MMSE) and maximum a posteriori (MAP) estimators. In such case, it is natural to use Markov chain Monte Carlo (MCMC) methods [20] to generate samples $\mathbf{M}^{(t)}$, $\mathbf{A}^{(t)}$ and $R^{(t)}$ asymptotically distributed according to $f(\mathbf{M}, \mathbf{A}, R|\mathbf{Y})$. However, the dimensions of the factor loading matrix $\mathbf{M}$ and the factor score matrix $\mathbf{A}$ depend on the unknown number R of signatures to be identified. As a consequence, sampling from $f(\mathbf{M}, \mathbf{A}, R|\mathbf{Y})$ requires exploring parameter spaces of different dimensions. To solve this dimension matching problem, we include a birth/death process within the MCMC procedure. Specifically, a birth, death or switch move is chosen at each iteration of the algorithm (see the Appendix and [21]). This birth-death model differs from the classical reversible-jump MCMC (RJ-MCMC) (as defined in [21]) in the sense that, for the birth-death model, each move is accepted or rejected at each iteration using the likelihood ratio between the current state and the new state proposed by the algorithm. The factor matrix $\mathbf{M}$, the factor score matrix $\mathbf{A}$ and the noise variance σ^2 are then updated, conditionally upon the number of factors R, using Gibbs moves.

After a sufficient number of iterations ($\mathrm{N_{mc}}$ iterations, including a burn-in period of $\mathrm{N_{bi}}$ iterations), the traditional Bayesian estimators (e.g., MMSE and MAP) can be approximated using the generated samples $\mathbf{M}^{(t)}$, $\mathbf{A}^{(t)}$ and $R^{(t)}$. First, the generated samples are used to approximate the MAP estimator of the number of factors

$$\begin{aligned}\widehat{R}_{\mathrm{MAP}} &= \underset{k\in\{2,\ldots,R_{\max}\}}{\operatorname{argmax}} \ \mathrm{P}[R=k|\mathbf{Y}] \\ &\approx \underset{k\in\{2,\ldots,R_{\max}\}}{\operatorname{argmax}} \ \frac{N_k}{\mathrm{N_r}}\end{aligned} \tag{19}$$

where N_k is the number of generated samples $R^{(\mathrm{N_{bi}}+1)}, \ldots, R^{(\mathrm{N_{mc}})}$ satisfying $R^{(t)} = k$ and $\mathrm{N_r} = \mathrm{N_{mc}} - \mathrm{N_{bi}}$. Then, conditioned on $\widehat{R}_{\mathrm{MAP}}$, the joint MAP estimator $\left(\widehat{\mathbf{M}}_{\mathrm{MAP}}, \widehat{\mathbf{A}}_{\mathrm{MAP}}\right)$ of the factor and factor score matrices is determined as follows

$$\left(\widehat{\mathbf{M}}_{\mathrm{MAP}}, \widehat{\mathbf{A}}_{\mathrm{MAP}}\right) \approx \underset{t=\mathrm{N_{bi}}+1,\ldots,\mathrm{N_{mc}}}{\operatorname{argmax}} f\left(\mathbf{M}^{(t)}, \mathbf{A}^{(t)}|\mathbf{Y}, R=\widehat{R}_{\mathrm{MAP}}\right). \tag{20}$$

Results and discussion

The proposed method consists of estimating simultaneously the matrices $\mathbf{M}$ and $\mathbf{A}$ defined in (1), under the positivity and sum-to-one constraints mentioned previously, in a fully unsupervised framework, i.e., the number of factors R is also estimated from the data. A Gibbs sampler algorithm is designed that generates samples distributed according to the posterior distribution associated to the proposed uBLU model. For more details about the Gibbs sampling strategy, see the Appendix.

Simulations on synthetic data

To illustrate the performance of the proposed Bayesian factor decomposition, we first present simulations conducted on synthetic data. More extensive simulation results are reported in the Additional file 1.

Simulation scenario

Several synthetic datasets $\mathcal{D}_1, \ldots, \mathcal{D}_4$ were generated. The experiments presented here correspond to the expression values of $G = 512$ genes (for datasets $\mathcal{D}_1$, $\mathcal{D}_3$ and $\mathcal{D}_4$) or $G = 12000$ genes (for dataset $\mathcal{D}_2$) with $N = 128$ samples. Each sample is composed of exactly $R = 3$ factors mixed using the linear mixing model in (1). The factors of the first dataset $\mathcal{D}_1$ have been generated so that only a few genes affect each factor. For the second dataset $\mathcal{D}_2$, realistic factors have been extracted from real genetic datasets. The third dataset $\mathcal{D}_3$ has been generated enforcing the factors to be orthogonal but not necessarily positive whereas in the forth dataset, $\mathcal{D}_4$, factors are orthogonal and positive. These simulation conditions are summarized in Table 1.

In each case, the $R = 3$ factors were mixed in random proportions (factor scores), with positivity and sum-to-one constraints. All synthetic datasets were corrupted by an i.i.d. Gaussian noise sequence. The signal-to-noise ratio is $\mathrm{SNR}_i = 20$ dB where $\mathrm{SNR}_i = G^{-1}\sigma^{-2}\left\|\sum_{r=1}^{R}\mathbf{m}_r a_{r,i}\right\|^2$ for each sample i ($i = 1, \ldots, N$).

Proposed method (uBLU)

The first step of the algorithm consists of estimating the number of factors R involved in the mixture, and hence determining the dimensions of the matrices $\mathbf{M}$ and $\mathbf{A}$, using the *maximum a posteriori* (MAP) estimator $\widehat{R}_{\mathrm{MAP}}$. The second step of the algorithm consists of estimating the unknown model parameters ($\mathbf{M}$, $\mathbf{A}$ and σ^2) given $\widehat{R}_{\mathrm{MAP}}$. The estimated posterior distributions of the unknown model parameters are given in Additional file 1: Figure S5 and validate the proposed Bayesian model.

The burn-in period and number of Gibbs samples were determined using quantitative methods described in the Additional file 1: Section "Convergence diagnosis".

Table 1 Synthetic datasets $\mathcal{D}_1, \ldots, \mathcal{D}_4$

$\mathcal{D}_1$	Peaky factors
$\mathcal{D}_2$	Realistic factors
$\mathcal{D}_3$	Orthogonal factors
$\mathcal{D}_4$	Orthogonal and positive factors

Comparison to other methods

The performance of the proposed uBLU algorithm is compared with other existing factor decomposition methods including PCA, NMF, BFRM and GB-GMF by using the following criteria, which are common measures used to compare factor analysis algorithms,

- the factor mean square errors (MSE)

$$\mathrm{MSE}_r^2 = \frac{1}{G}\left\|\hat{\mathbf{m}}_r - \mathbf{m}_r\right\|^2,\ r = 1,\ldots,R$$

  where $\hat{\mathbf{m}}_r$ is the estimated r-th factor loading vector,
- the global MSE of factor scores

$$\mathrm{GMSE}_r^2 = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{a}_{r,i} - a_{r,i}\right)^2,\ r = 1,\ldots,R$$

  where $\hat{a}_{r,i}$ is the estimated proportion of the r-th factor in the i-th sample,
- the reconstruction error (RE)

$$\mathrm{RE} = \frac{1}{NG}\sum_{i=1}^{N}\left\|\mathbf{y}_i - \hat{\mathbf{y}}_i\right\|^2 \tag{21}$$

  where $\hat{\mathbf{y}}_i = \sum_{r=1}^{R}\hat{\mathbf{m}}_r\hat{a}_{r,i}$ is the estimate of $\mathbf{y}_i$,
- the spectral angle distance (SAD) between $\mathbf{m}_r$ and its estimate $\hat{\mathbf{m}}_r$ for each factor $r = 1,\ldots,R$

$$\mathrm{SAD}_r = \arccos\left(\frac{\hat{\mathbf{m}}_r^T\mathbf{m}_r}{\left\|\hat{\mathbf{m}}_r\right\|\left\|\mathbf{m}_r\right\|}\right)$$

  where $\arccos(\cdot)$ is the inverse cosine function,
- the global spectral angle distance (GSAD) between $\mathbf{y}_i$ (the i-th observation vector) and $\hat{\mathbf{y}}_i$ (its estimate)

$$\mathrm{GSAD} = \frac{1}{N}\sum_{i=1}^{N}\arccos\left(\frac{\hat{\mathbf{y}}_i^T\mathbf{y}_i}{\left\|\hat{\mathbf{y}}_i\right\|\left\|\mathbf{y}_i\right\|}\right),$$

- the computational time.

The proposed uBLU algorithm, the PCA, NMF and GB-GMF methods were implemented in Matlab 7.8.0 (R2009a). The BFRM software (version 2.0) was downloaded from [22] and implemented with default values for the parameters. All methods were implemented on an Intel(R) Core(TM)2 Duo processor.

Simulation results are reported in Tables 2, 3, 4 and 5. Note that the positivity and sum-to-one constraints that are enforced on the data for the proposed uBLU algorithm avoid the scale ambiguity inherent to any factor decomposition problem. Conversely, for the other factor decomposition methods (PCA, NMF, BFRM and GB-GMF), if $\{\mathbf{M}, \mathbf{A}\}$ is an admissible solution, $\{\mathbf{MB}, \mathbf{B}^T\mathbf{A}\}$ is also admissible for any scaling and permutation matrix $\mathbf{B}$. Hence a re-scaling is required to identify appropriate permutations before computing MSEs and GMSEs. Moreover, when PCA, NMF, BFRM and GB-GMF methods are run for $R = 4$, we only considered the 3 factors yielding the 3 smallest SADs values.

These results show that the uBLU method is more flexible since it provides better unmixing performance across all of the considered synthetic datasets $\mathcal{D}_1,\ldots,\mathcal{D}_4$ as compared to other existing factorization methods (PCA, NMF, BFRM and GB-GMF). Moreover, uBLU has the following advantages: i) it is fully unsupervised and does not require the number of factors to be specified as a prior knowledge, ii) due to the constraints, the factors and factor scores are estimated without scale ambiguity. The disadvantage is the execution time: uBLU requires more computation due to the Gibbs sampling.

Evaluation on gene expression data

Here the proposed algorithm is illustrated on a real time-evolving gene expression data from recent viral challenge studies on influenza A/H3N2/Wisconsin. The data are available at GEO, accession number GSE30550.

Details on data collection

We briefly describe the dataset. For more details the reader is referred to [14,18]. H3N2 dataset consists of the gene expression levels of $N = 267$ Affymetrix chips collected on 17 healthy human volunteers experimentally infected with influenza A/Wisconsin/67/2005 (H3N2). A clinical symptom score was assigned to each sample by clinicians who participated in the study. Nine of the 17 subjects (those labeled Z01, Z05, Z06, Z07, Z08, Z10, Z12, Z13, and Z15 in Figure 1c) became clinically ill during the study. These labels are only used as ground truth to quantify performance and are not available to the uBLU algorithm. The challenge consists of inoculating intranasally a dose of 10^6 TCID_{50} Influenza A manufactured and processed under current good manufacturing practices (cGMP) by Baxter BioScience. Peripheral blood microarray analysis was performed at multiple time instants corresponding to baseline (24 hours prior to inoculation with virus), then at 8 hour intervals for the initial 120 hours and then 24 hours for two further days. Each sample consisted of over $G = 12000$ gene expression values after standard microarray data normalization with RMA using the custom brain array cdf [14]. No other preprocessing was applied prior to running the five unsupervised methods (uBLU, PCA, NMF, BFRM, and GB-GMF).

Application of the proposed uBLU algorithm

The uBLU algorithm was run with $\mathrm{N_{mc}} = 10000$ Monte Carlo iterations, including a burn-in period of $\mathrm{N_{bi}} = 1000$ iterations. uBLU allows the posterior distribution of the number of factors R, depicted in Figure 1a, to be estimated. The results show that the MAP estimate of the

Table 2 Simulation results for dataset $\mathcal{D}_1$

			(a) $R = 2$		
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
$\text{MSE}_r^2(\times 10^{-2})$	**0.39**	N/A	N/A	205.99	267.42
	0.60	6.04	61.12	N/A	N/A
	0.54	0.97	9.78	325.58	67.14
$\text{GMSE}_r^2(\times 10^{-3})$	**0.04**	N/A	N/A	64.39	226.58
	0.04	2.00	2.00	N/A	N/A
	0.05	0.30	0.28	75.87	41.33
$\text{SAD}_r^2(\times 10^{-1})$	**0.46**	N/A	N/A	21.69	12.48
	0.29	3.49	3.50	N/A	N/A
	0.28	1.49	1.50	23.24	27.43
GSAD $(\times 10^{-2})$	**3.39**	20.38	20.38	24.04	37.35
RE	**0.18**	9.12	9.12	1.94	9.16
Time (*s*)	1.24×10^3	**0.03**	0.71	47.15	0.39×10^3
			(b) $R = 3$		
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
$\text{MSE}_r^2(\times 10^{-2})$	**0.39**	6.01	0.48	212.30	40.27
	0.60	6.53	**0.45**	681.42	147.74
	0.54	5.86	**0.28**	137.22	94.90
$\text{GMSE}_r^2(\times 10^{-3})$	**0.04**	6.62	0.19	76.09	45.29
	0.04	2.40	**0.01**	142.72	17.37
	0.05	0.84	0.05	76.22	33.78
$\text{SAD}_r^2(\times 10^{-1})$	**0.46**	1.86	0.53	10.68	11.86
	0.29	1.18	0.31	15.18	12.50
	0.28	1.36	**0.26**	5.33	13.96
GSAD $(\times 10^{-2})$	**3.37**	3.39	3.38	24.23	33.38
RE	**0.18**	**0.18**	0.18	1.84	0.18
Time (*s*)	1.24×10^3	**0.10**	0.95	53.60	0.56×10^3
			(c) $R = 4$		
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
$\text{MSE}_r^2(\times 10^{-2})$	**0.39**	6.02	87.78	205.66	195.89
	0.60	6.53	**0.45**	247.96	101.34
	0.54	8.03	**0.26**	330.01	68.69
$\text{GMSE}_r^2(\times 10^{-3})$	**0.04**	23.82	26.56	64.59	57.58
	0.04	11.70	0.23	114.02	3.10
	0.05	6.37	18.04	75.47	27.72
$\text{SAD}_r^2(\times 10^{-1})$	**0.46**	1.86	6.14	9.74	8.84
	0.29	1.18	0.31	22.15	26.80
	0.28	1.36	**0.26**	8.17	27.32
GSAD $(\times 10^{-2})$	3.39	**3.34**	3.36	28.62	29.23
RE	**0.18**	**0.18**	0.18	2.08	0.18
Time (*s*)	1.24×10^3	**0.11**	0.96	63.88	0.70×10^3

MSEs, GMSEs, SADs, GSADs, REs and computational times between the proposed uBLU algorithm and PCA, NMF, BFRM and GB-GMF methods.

Table 3 Simulation results for dataset $\mathcal{D}_2$

			(a) $R = 2$		
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
MSE_r^2	**0.09**	1.97	N/A	N/A	N/A
	0.14	N/A	1.06	37.67	58.75
	0.14	**0.12**	26.68	52.09	150.09
$GMSE_r^2 (\times 10^{-1})$	0.34	**0.01**	N/A	N/A	N/A
	0.15	N/A	1.12	1.17	22.37
	0.09	0.94	6.24	0.62	1.18
$SAD_r^2 (\times 10^{-1})$	**0.39**	0.44	N/A	N/A	N/A
	0.48	N/A	1.32	16.53	13.34
	0.47	**0.44**	3.72	15.21	18.14
GSAD ($\times 10^{-2}$)	1.51	**1.02**	1.53	37.99	129.40
RE ($\times 10^{-2}$)	**0.64**	1.62	1.65	0.65	5.47
Time (*s*)	22.06×10^3	**0.29**	32.02	4.07×10^3	9.24×10^3
			(b) $R = 3$		
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
MSE_r^2	**0.09**	1.97	14.87	24.41	61.00
	0.14	**0.01**	20.53	50.59	58.31
	0.14	**0.09**	14.02	35.89	65.11
$GMSE_r^2 (\times 10^{-1})$	0.34	**0.03**	0.34	1.41	4.80
	0.15	**0.02**	2.44	0.65	9.40
	0.09	**0.05**	0.92	1.19	5.40
$SAD_r^2 (\times 10^{-1})$	**0.39**	0.44	2.84	14.35	13.72
	0.48	**0.12**	4.75	15.47	13.62
	0.47	**0.37**	4.00	17.50	15.82
GSAD ($\times 10^{-2}$)	**1.02**	1.02	1.49	29.29	129.29
RE ($\times 10^{-2}$)	0.64	**0.63**	1.55	0.75	1.62
Time (*s*)	22.06×10^3	**0.28**	45.91	5.37×10^3	16.59×10^3
			(c) $R = 4$		
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
MSE_r^2	**0.09**	1.97	13.13	24.25	64.90
	0.14	**0.01**	20.53	50.52	64.09
	0.14	**0.09**	14.02	28.32	69.99
$GMSE_r^2 (\times 10^{-1})$	0.34	**0.09**	0.20	1.42	15.12
	0.15	0.48	1.00	0.65	9.55
	0.09	**0.05**	0.44	1.31	7.73
$SAD_r^2 (\times 10^{-1})$	**0.39**	0.44	2.54	14.74	14.53
	0.48	**0.13**	5.52	15.45	14.55
	0.47	**0.37**	4.79	16.45	16.17
GSAD ($\times 10^{-2}$)	1.02	**1.01**	1.06	40.36	129.29
RE ($\times 10^{-2}$)	0.64	**0.63**	0.69	0.86	1.50
Time (*s*)	22.06×10^3	**0.54**	55.86	5.59×10^3	16.59×10^3

MSEs, GMSEs, SADs, GSADs, REs and computational times between the proposed uBLU algorithm and PCA, NMF, BFRM and GB-GMF methods.

Table 4 Simulation results for dataset $\mathcal{D}_3$

		(a) $R = 2$			
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
$\text{MSE}_r^2(\times 10^{-3})$	**0.01**	0.83	0.82	N/A	1.14
	0.85	**0.80**	0.92	1.34	2.30
	1.15	N/A	N/A	1.36	N/A
$\text{GMSE}_r^2(\times 10^{-2})$	7.75	**7.29**	7.72	N/A	8.94
	7.76	**0.47**	0.48	12.30	11.86
	9.84	N/A	N/A	11.05	N/A
$\text{SAD}_r^2(\times 10^{-1})$	**0.59**	7.09	7.04	N/A	15.55
	7.13	**6.71**	7.19	8.41	16.43
	8.71	N/A	N/A	**8.54**	N/A
GSAD $(\times 10^{-1})$	3.23	**2.58**	2.59	6.59	15.26
RE $(\times 10^{-4})$	3.11	0.70	0.70	**0.47**	2.50
Time (*s*)	1.59×10^3	**0.01**	0.70	42.02	0.40×10^3
		(b) $R = 3$			
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
$\text{MSE}_r^2(\times 10^{-3})$	**0.01**	0.15	0.15	1.74	1.20
	0.85	1.02	**0.76**	1.76	2.26
	1.15	1.57	**1.03**	1.55	2.40
$\text{GMSE}_r^2(\times 10^{-2})$	7.75	14.89	**2.80**	11.40	14.09
	7.76	**0.11**	0.40	12.11	12.33
	9.84	**0.11**	0.30	10.94	12.76
$\text{SAD}_r^2(\times 10^{-1})$	**0.59**	2.60	2.47	11.34	15.76
	7.13	7.16	**6.59**	9.45	16.40
	8.71	8.80	**7.67**	9.06	15.66
GSAD $(\times 10^{-1})$	3.23	**2.58**	1.71	6.88	15.20
RE $(\times 10^{-4})$	3.11	**0.27**	0.29	0.49	2.44
Time (*s*)	1.59×10^3	**0.10**	1.24	59.72	0.54×10^3
		(c) $R = 4$			
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
$\text{MSE}_r^2(\times 10^{-3})$	**0.01**	0.02	1.43	1.43	1.19
	0.85	1.48	5.49	3.92	2.06
	1.15	1.68	**0.90**	1.88	2.33
$\text{GMSE}_r^2(\times 10^{-2})$	**7.75**	13.78	20.56	16.66	13.15
	7.76	**4.35**	12.36	15.34	11.75
	9.84	3.99	**2.67**	11.25	13.29
$\text{SAD}_r^2(\times 10^{-1})$	**0.59**	0.97	10.27	10.24	15.97
	7.13	7.93	15.78	16.45	14.92
	8.71	8.66	**6.93**	10.98	15.89
GSAD $(\times 10^{-1})$	3.23	**1.17**	1.20	5.51	15.98
RE $(\times 10^{-4})$	3.11	**0.16**	0.16	0.41	2.45
Time (*s*)	1.59×10^3	**0.13**	1.15	67.71	0.69×10^3

MSEs, GMSEs, SADs, GSADs, REs and computational times between the proposed uBLU algorithm and PCA, NMF, BFRM and GB-GMF methods.

Table 5 Simulation results for dataset $\mathcal{D}_4$

			(a) $R = 2$		
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
$\text{MSE}_r^2(\times 10^{-2})$	**0.02**	N/A	5.12	N/A	N/A
	1.61	**0.01**	3.59	15.35	18.69
	0.05	0.44	N/A	14.42	19.20
$\text{GMSE}_r^2(\times 10^{-1})$	**0.28**	N/A	3.23	N/A	N/A
	0.87	**0.02**	2.65	0.33	1.62
	0.69	0.76	N/A	**0.50**	1.30
$\text{SAD}_r^2(\times 10^{-1})$	**0.34**	N/A	4.25	N/A	N/A
	3.08	**0.17**	3.71	14.90	14.89
	0.51	0.68	N/A	15.59	15.70
GSAD ($\times 10^{-2}$)	**4.97**	5.24	5.25	157.09	156.19
RE ($\times 10^{-4}$)	**4.49**	4.88	4.89	19.34	8.48
Time (*s*)	1.61×10^3	**0.02**	1.36	35.29	0.40×10^3
			(b) $R = 3$		
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
$\text{MSE}_r^2(\times 10^{-2})$	0.02	**0.01**	6.18	18.38	21.63
	1.61	**0.01**	4.79	16.10	19.55
	0.05	0.09	4.21	15.04	19.85
$\text{GMSE}_r^2(\times 10^{-1})$	0.28	**0.05**	1.67	1.44	1.29
	0.87	**0.05**	1.01	0.37	1.75
	0.69	**0.05**	0.94	0.26	1.17
$\text{SAD}_r^2(\times 10^{-1})$	0.34	**0.27**	4.12	15.21	15.65
	3.08	**0.17**	4.09	15.26	15.90
	0.51	**0.32**	4.16	16.07	15.36
GSAD ($\times 10^{-2}$)	4.97	**4.95**	4.99	157.08	154.80
RE ($\times 10^{-4}$)	4.49	**4.34**	4.36	25.00	8.48
Time (*s*)	1.61×10^3	**0.10**	1.78	41.05	0.55×10^3
			(c) $R = 4$		
	uBLU	**PCA**	**NMF**	**BFRM**	**GB-GMF**
$\text{MSE}_r^2(\times 10^{-2})$	0.02	**0.01**	6.98	17.51	21.60
	1.61	**0.01**	7.30	15.07	19.03
	0.05	0.07	4.27	14.55	19.14
$\text{GMSE}_r^2(\times 10^{-1})$	0.28	**0.22**	0.65	0.75	1.29
	0.87	**0.51**	0.91	0.77	1.18
	0.69	**0.05**	0.56	0.56	1.33
$\text{SAD}_r^2(\times 10^{-1})$	0.34	**0.27**	4.41	15.61	15.51
	3.08	**0.19**	4.81	16.31	14.77
	0.51	**0.33**	4.00	15.84	15.26
GSAD ($\times 10^{-2}$)	4.97	**4.91**	4.94	156.76	162.63
RE ($\times 10^{-4}$)	4.49	**4.30**	4.33	13.48	8.29
Time (*s*)	1.61×10^3	**0.16**	1.56	48.22	0.70×10^3

MSEs, GMSEs, SADs, GSADs, REs and computational times between the proposed uBLU algorithm and PCA, NMF, BFRM and GB-GMF methods.

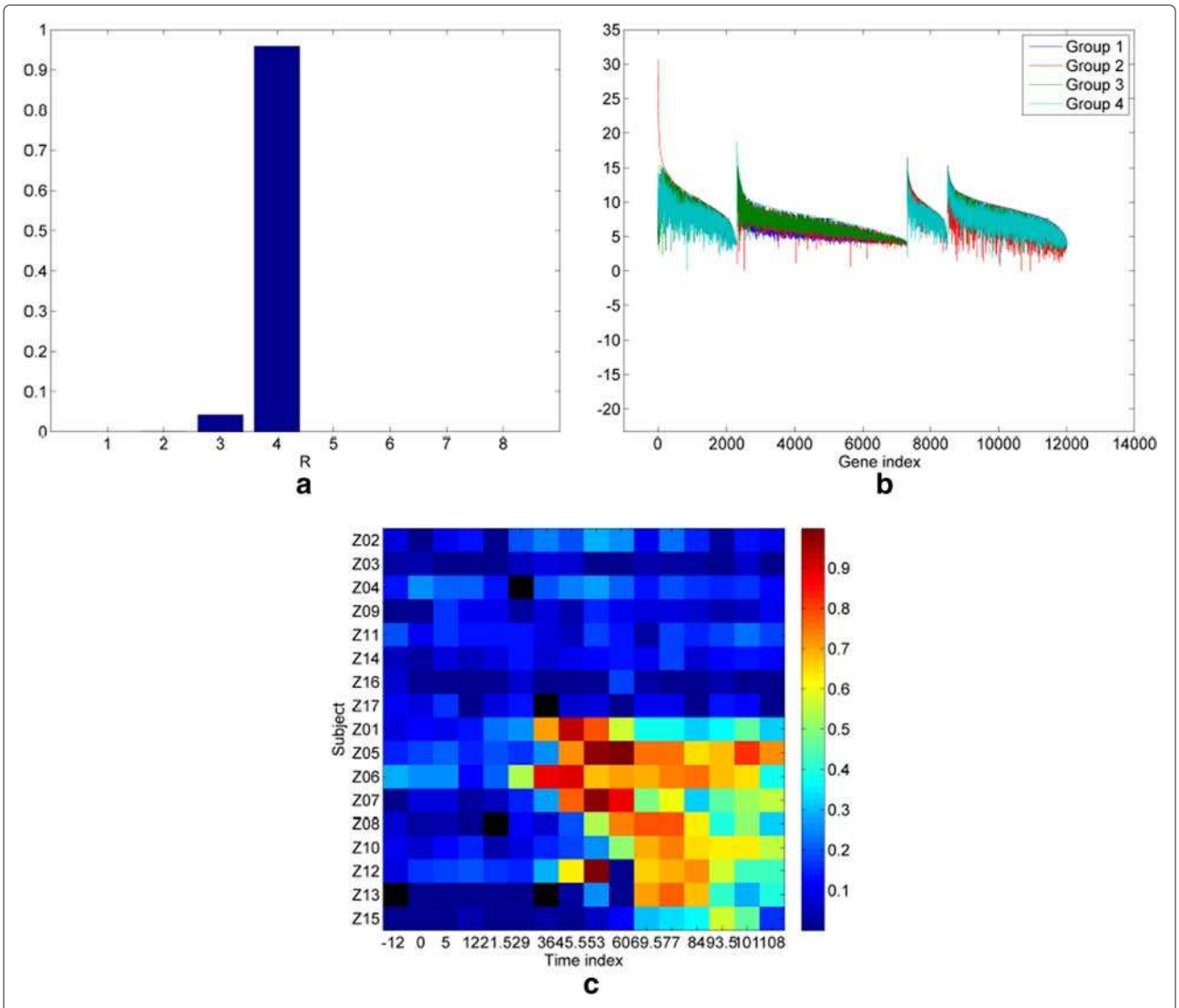

Figure 1 Experimental results on the H3N2 viral challenge dataset of gene expression profiles. (**a**) Estimated posterior distribution of the number of factors *R*. (**b**) Factor loadings ranked by decreasing dominance. (**c**) Heatmap of the factor scores of the inflammatory component clearly separates symptomatic subjects (bottom 9 rows) and the time course of their molecular inflammatory response. The five black colored pixels indicate samples that were not assayed.

number of factors is $\widehat{R}_{\mathrm{MAP}} = 4$ (more than 90% of the generated Gibbs samples of the number of factors were equal to 4).

Figure 2 shows the reconstruction error $\mathrm{RE}^{(t)}$ as a function of the number of iterations ($t = 1, \ldots$). The reconstruction errors are computed from the observed gene expression data matrix and the estimates of the factor and factor score matrix **M** and **A** at each iteration. Figure 2 also indicates that the number of burn-in and Monte Carlo samples $\mathrm{N_{bi}} = 1000$ and $\mathrm{N_{mc}} = 10000$ are sufficient.

The different factors are depicted in Figure 1b where the *G* genes have been reordered so that the dominant genes are grouped together in each factor. Factors are then ordered with respect to their maximum loading. Specifically, the *k*-th sharp peak in the figure occurs at the gene index that has maximal loading in factor *k*. Genes to the right of this dominant gene up to the $(k + 1)$-st peak also dominate in this *k*-th factor, but at a lower degree. uBLU identifies a strong factor (the first factor, in red) by virtue of its significantly larger proportion of highly dominant genes. Many of the genes in this strong factor are recognizable as immune response genes that regulate pattern recognition, interferon, and inflammation pathways in respiratory viral response. A very similar factor was found in a different analysis [14,18] of this dataset and here we call it the "*inflammatory component*".

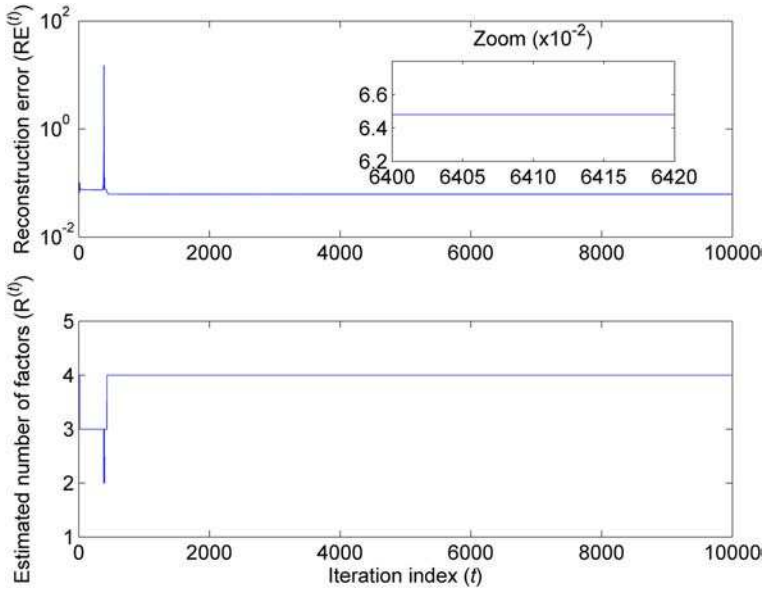

Figure 2 Reconstruction error and estimated number of factors as a function of the number of iterations (H3N2 challenge data). Top: Reconstruction error ($RE^{(t)}$) computed from the observation matrix $\mathbf{Y}$ and the estimated matrices $\mathbf{M}^{(t)}$ and $\mathbf{A}^{(t)}$ as a function of the iteration index t. Bottom: Estimated number of factors $R^{(t)}$ as a function of the iteration number t.

The factor scores corresponding to this inflammatory component are shown in Figure 1c, where they are rendered as an image whose columns (respectively rows) index the subjects (respectively the different time sampling instants). Figure 1c shows that uBLU clearly separates the samples of subjects exhibiting symptoms (associated with the last 9 rows) from those who remain asymptomatic (associated with the first 8 rows), when the estimated number of factors is $\hat{R} = 4$. Moreover, this symptom factor can be used to segment the data matrix into 3 states: pre-onset-symptomatic (before significant symptoms occur), post-onset-symptomatic and asymptomatic.

Furthermore, this inflammatory factor identified by the proposed uBLU algorithm is most highly represented in those samples associated with acute flu symptoms, as measured by modified Jackson scores (see [14], Figure 1B). The dominant gene contributors to this inflammatory component correspond to well-known transcription factors controlling immune response, inflammatory response and antigen presentation – see Table 6. The reader is referred to [14,18] for more details on clinical determination of symptom scores and biological significance of the inflammatory component genes.

For comparison we applied a supervised version of the proposed uBLU algorithm to the H3N2 dataset. This was

Table 6 NCI-curated pathway associations of group of genes contributing to uBLU inflammatory component

Pathway name	Genes	P-value
IFN-gamma pathway	CASP1, CEBPB, IL1B, IRF1, IRF9, PRKCD, SOCS1, STAT1, STAT3	1.34e-09
PDGFR-beta signaling pathway	DOCK4, EIF2AK2, FYN, HCK, LYN, PRKCD, SLA, SRC, STAT1, STAT3, STAT5A, STAT5B	3.26e-08
IL23-mediated signaling events	CCL2, CXCL1, CXCL9, IL1B, STAT1, STAT3, STAT5A	2.18e-07
Signaling events mediated by TCPTP	EIF2AK2, SRC, STAT1, STAT3, STAT5A, STAT5B, STAT6	6.38e-07
Signaling events mediated by PTP1B	FYN, HCK, LYN, SRC, STAT3, STAT5A, STAT5B	2.40e-06
GMCSF-mediated signaling events	CCL2, LYN, STAT1, STAT3, STAT5A, STAT5B	3.70e-06
IL12-mediated signaling events	HLA-A, IL1B, SOCS1, STAT1, STAT3, STAT5A, STAT6	1.32e-05
IL6-mediated signaling events	CEBPB, HCK, IRF1, PRKCD, STAT1, STAT3	1.80e-05

NCI-curated pathway associations of group of genes contributing to uBLU inflammatory component, whose factor scores are shown in Figure 1 (Source: NCI pathway interaction database http://pid.nci.nih.gov). Genes in uBLU factor are significantly better represented in the NCI-curated pathways than the genes in NMF (compare p-values here to those in Table 8).

implemented by setting the number of factors to $R = 4$ and using the algorithm [13] to jointly estimate $\mathbf{M}$ and $\mathbf{A}$. The inflammatory component found by the supervised algorithm was virtually identical to the one found by the proposed algorithm (uBLU) that automatically selects $R = 4$.

Comparison to other methods

The uBLU algorithm is compared with four matrix factorization algorithms, i.e. PCA, NMF, BFRM and GB-GMF methods.

Figure 3 depicts the different factors, ordered so that the inflammatory group is the leftmost one (in red). The factor loadings obtained with NMF or PCA reveal the inflammatory component. However, there are fewer highly dominant genes in the NMF and PCA loadings for this factor as compared to uBLU. The BFRM and GB-GMF methods found four pathways, several overlapping with those of uBLU, NMF and PCA.

The factor scores of the five matrix factorization methods corresponding to the inflammatory component are depicted in Figure 4. This figure shows that the uBLU and the NMF methods are better able to attain a high contrast separation between the acutely symptomatic samples and the other samples. This is confirmed by the evaluation of the Fisher criteria (22) between these two regions (see Table 7). Indeed, denote by $(\mu_{\text{pos}}, \sigma^2{}_{\text{pos}})$ the empirical mean and variance of the scores associated with the N_{pos} samples in the acute symptomatic state (bright colored samples in the lower right rectangle of Figure 1c). Denote

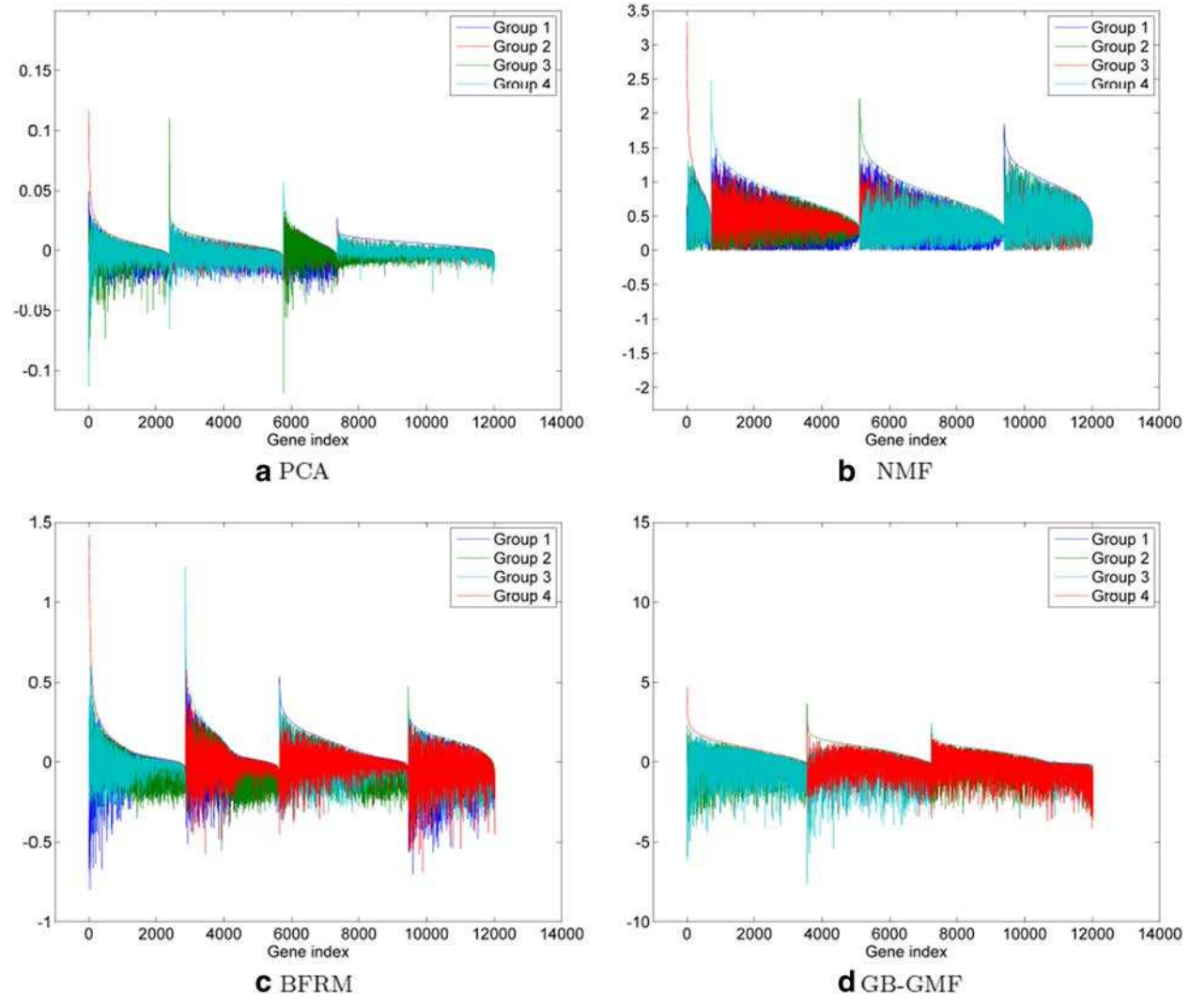

Figure 3 Factor loadings ranked by decreasing dominance for H3N2 challenge data. uBLU shows a particularly strong component (Figure 1b), the group ♯1, that corresponds to the well-known inflammatory pathway. NMF and PCA algorithms also reveal an inflammatory component, but it includes fewer relevant genes than uBLU. See Figure 4 for the corresponding factor scores.

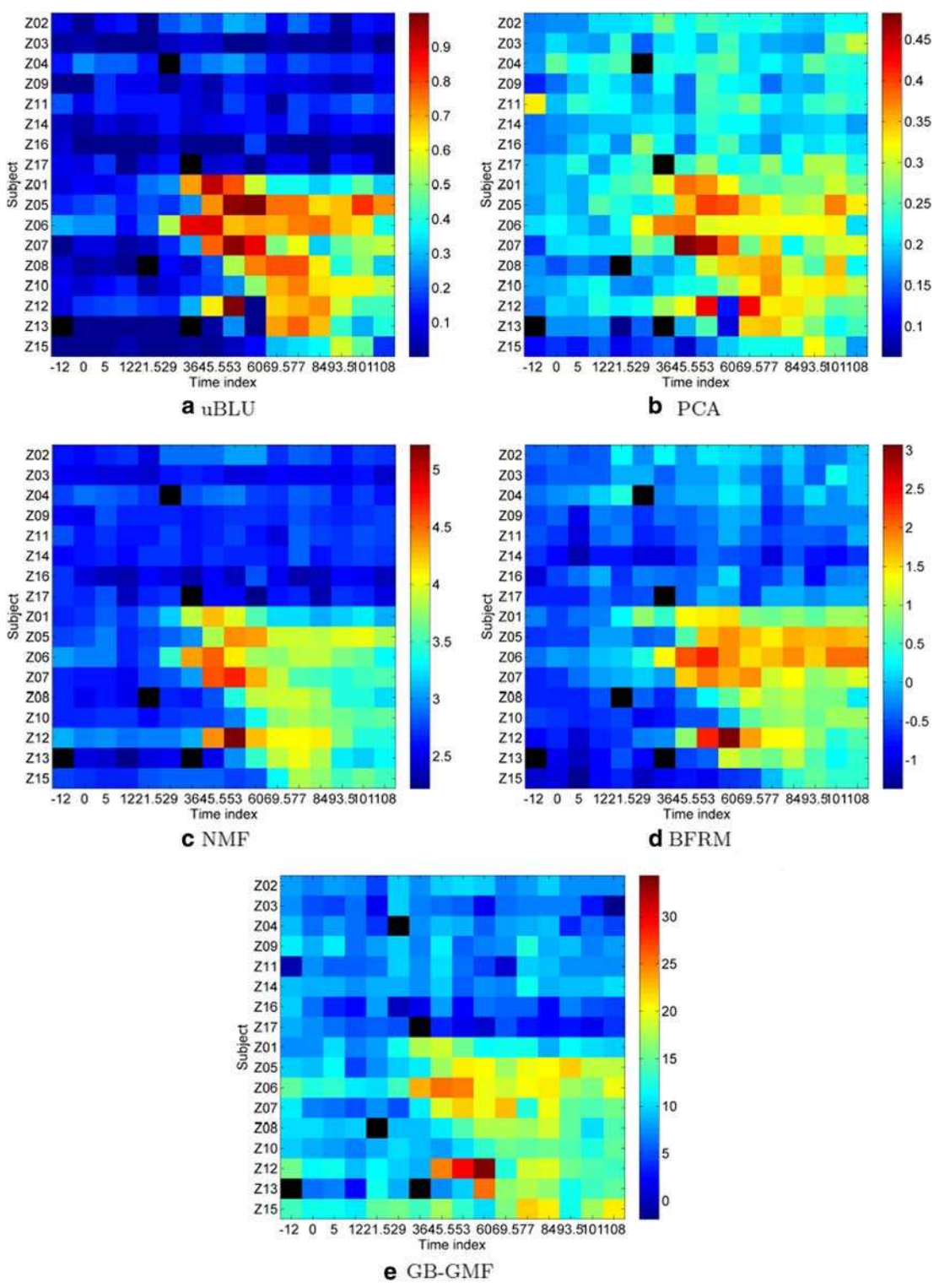

Figure 4 Heatmaps of the factor scores of the inflammatory component for H3N2 challenge data. The inflammatory factor determined by the proposed uBLU method (**a**) shows higher contrast between symptomatic and asymptomatic subjects than the other methods. The five black colored pixels of the heatmaps indicate samples that were not assayed.

Table 7 Simulation results for real H3N2 dataset

	uBLU	PCA	NMF	BFRM	GB-GMF
Fisher criteria ($\times 10^{-2}$) (22)	**6.20**	2.03	6.17	4.68	2.30
RE	$\mathbf{6.48.10^{-2}}$	4.89	$7.31.10^{-2}$	4.82	$9.51.10^{-2}$
Time	$\approx 12\ h$	**1.5 *s***	116 *s*	$\approx 47\ min$	$\approx 10\ h$
Number of iterations	10 000	N/A	5 000	10 000	500

Measure of the Fisher linear discriminant measure ([23], p. 119) between post-onset symptomatic samples and the other samples on heatmaps (Figure 4), reconstruction error (RE) between the observed data and the MAP estimators, computational times (for an implementation in MATLAB 7.8.0 (R2009a) on a 3 GHz Intel(R) Core(TM)2 Duo processor), and corresponding number of iterations.

by $\left(\mu_{\overline{\text{pos}}}, \sigma^2{}_{\overline{\text{pos}}}\right)$ the same parameters for the remaining samples. The Fisher linear discriminant measure ([23], p. 119) is defined as

$$\frac{\left(\mu_{\text{pos}} - \mu_{\overline{\text{pos}}}\right)^2}{N_{\text{pos}}\sigma^2{}_{\text{pos}} + (N - N_{\text{pos}})\sigma^2{}_{\overline{\text{pos}}}}. \tag{22}$$

To compare the biological relevance of the inflammatory genes found by uBLU to those found by the other methods we performed gene enrichment analysis (GEA). Here we only report GEA comparisons between uBLU and NMF. Tables 6 and 8 show the pathway enrichment associated with the top 200 genes found by uBLU and NMF, respectively, using NCI pathway interaction database (http://pid.nci.nih.gov). The uBLU genes are significantly better associated with the NCI-curated pathways than the NMF genes. In particular, the two most enriched pathways, IFN-gamma and PDGFR beta signaling, associated with the uBLU genes have much higher statistical significance (lower p-value) than any of the pathways associated with NMF.

Figure 5 shows how the factor scores of the dominant factor can be used as features to cluster samples. Euclidean multidimensional scaling (MDS) [24] is used to map the factor score vector for each sample as a coordinate in the plane. Each sample is embedded with a color and a size, denoting the state of the subject (asymptomatic subjects in blue, symptomatic subjects in red) and the time stamp, respectively. These figures show that uBLU can separate sick and healthy subjects, as well as or better than NMF, BFRM and GB-GMF.

One can conclude from these comparisons that, when applied on the H3N2 dataset, the proposed uBLU algorithm outperforms PCA, NMF, BFRM, and GB-GMF algorithms in terms of finding genes with higher pathway enrichment and achieving higher contrast of the acute symptom states.

The computational times required by the five considered matrix factorization methods, including the proposed uBLU algorithm, when applied to this real dataset, are reported in Table 7. The GB-GMF algorithm is slightly faster than the proposed algorithm but does not identify the inflammatory component or achieve good contrast of the acute symptom states in the H3N2 challenge study.

Conclusions

This paper proposes a new Bayesian unmixing algorithm for discovering signatures in high dimensional biological data, and specifically for gene expression microarrays. An interesting property of the proposed algorithm is that it provides positive factor loadings to ensure positivity as well as sum-to-one constraints for the factor scores.

Table 8 NCI-curated pathway associations of group of genes contributing to NMF inflammatory component

Pathway name	Genes	P-value
IL23-mediated signaling events	CCL2, CXCL1, CXCL9, IL1B, JAK2, STAT1, STAT5A	2.18e-07
IL12-mediated signaling events	GADD45B, IL1B, JAK2, MAP2K6, SOCS1, STAT1, STAT5A, STAT6	1.10e-06
IFN-gamma pathway	CASP1, IL1B, IRF9, JAK2, SOCS1, STAT1	1.07e-05
Signaling events mediated by TCPTP	EIF2AK2, PIK3R2, STAT1, STAT5A, STAT5B, STAT6	1.07e-05
IL27-mediated signaling events	IL1B, JAK2, STAT1, STAT2, STAT5A	1.22e-05
CXCR3-mediated signaling events	CXCL10, CXCL11, CXCL13, CXCL9, MAP2K6, PIK3R2	1.23e-05
GMCSF-mediated signaling events	CCL2, JAK2, STAT1, STAT5A, STAT5B	6.24e-05
PDGFR-beta signaling pathway	EIF2AK2, JAK2, PIK3R2, ARAP1, DOCK4, STAT1, STAT5A, STAT5B	1.38e-04

NCI-curated pathway associations of group of genes contributing to NMF inflammatory component, whose factor scores are shown in Figure 4 (Source: NCI pathway interaction database http://pid.nci.nih.gov). Genes in uBLU factor are significantly better represented in the NCI-curated pathways than the genes in NMF (compare p-values here to those in Table 6).

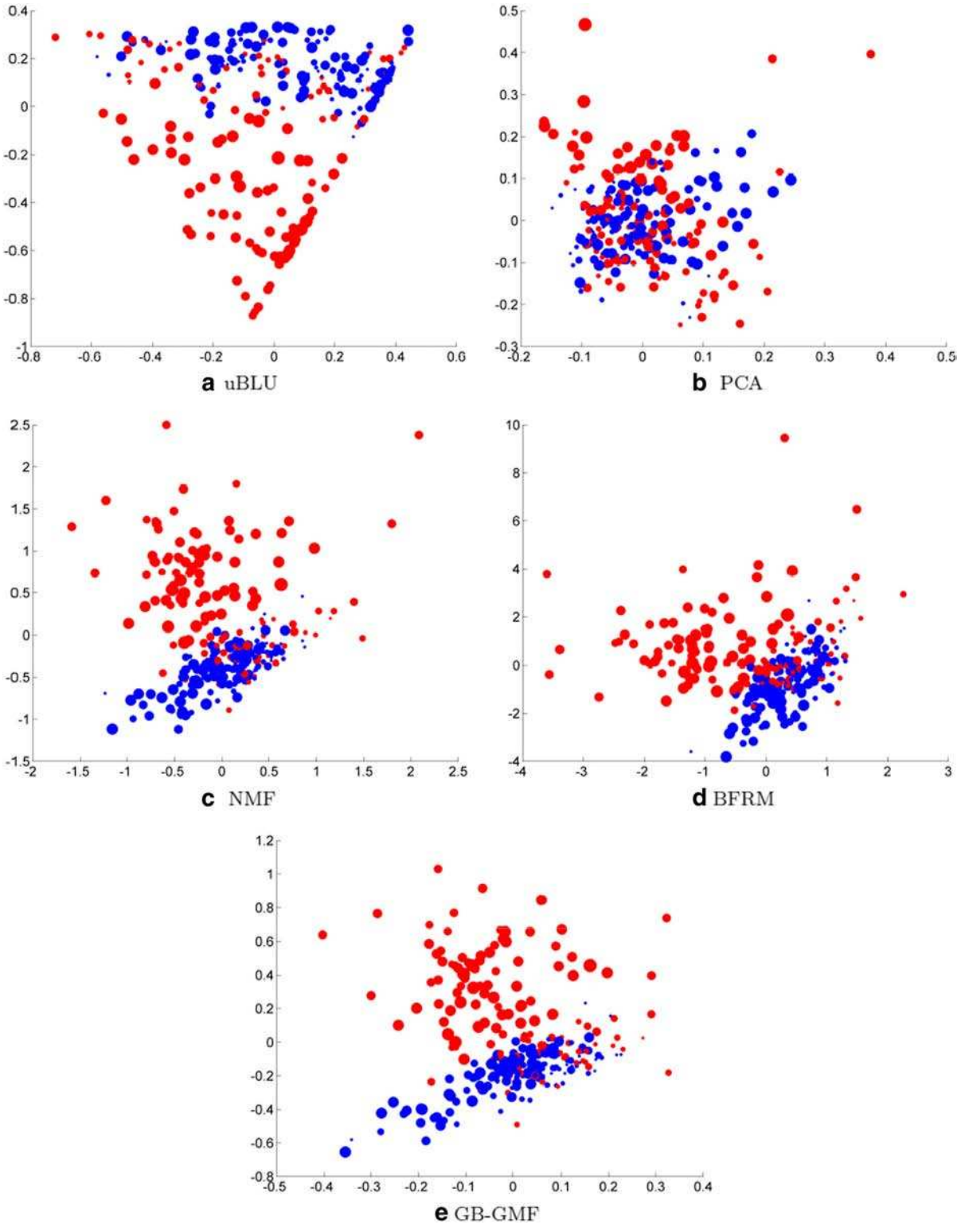

Figure 5 Chip clouds after demixing for H3N2 challenge data. These figures show the scatter of the four dimensional factor score vectors (projected onto the plane using MDS) for each algorithm that was compared to uBLU. uBLU, NMF and BFRM obtain a clean separation of samples of symptomatic (red points) and asymptomatic (blue points) subjects whereas the separation is less clear with PCA. In these scatter plots the size of a point is proportional to the time at which the sample was taken during challenge study.

The advantages of these constraints are that they lead to better discrimination between sick and healthy individuals, and they recover the inflammatory genes in a unique factor, the inflammatory component. The proposed algorithm is fully unsupervised in the sense that it does not depend on any labeling of the samples and that it can infer the number of factors directly from the observation data matrix. Finally, as any Bayesian algorithm, the Monte Carlo-based procedure investigated in this study provides point estimates as well as confidence intervals for the unknown parameters, contrary to many existing factor decomposition methods such as PCA or NMF.

Simulation results performed on synthetic and real data demonstrated significant improvements. Indeed, when applied to real time-evolving gene expression datasets, the uBLU algorithm revealed an inflammatory factor with higher contrast between subjects who would become symptomatic from those who would remain asymptomatic (as determined by comparing to ground truth clinical labels).

In this study, the time samples were modeled as independent. Future works include extensions of the proposed model to account for time dependency between samples.

Appendix A: Gibbs sampler

This appendix provides more details about the Gibbs sampler strategy to generate samples $\{\mathbf{M}^{(t)}, \mathbf{A}^{(t)}, \sigma^{2^{(t)}}, R^{(t)}\}$ distributed according to the joint distribution $f\left(\mathbf{M}, \mathbf{A}, \sigma^2, R|\mathbf{Y}\right)$ (the reader is referred to [25] for more details about the Gibbs sampler and MCMC methods). This joint distribution can be obtained by integrating out the hyperparameter γ from $f(\mathbf{\Theta}, \gamma|\mathbf{Y})$ defined in (18) and can be written

$$\begin{aligned} f\left(\mathbf{M}, \mathbf{A}, \sigma^2, R|\mathbf{Y}\right) \propto & f\left(\mathbf{Y}|\mathbf{M}, \mathbf{A}, \sigma^2, R\right) \\ & \times f\left(\mathbf{T}|\mathbf{E}, \mathbf{s}^2, R\right) \\ & \times f\left(\mathbf{A}|R\right) \\ & \times f\left(\sigma^2\right) f\left(R\right) \end{aligned} \tag{23}$$

where the dimensions of the matrices $\mathbf{M}$, $\mathbf{T}$, and $\mathbf{A}$ depend on the unknown number of factors R and the priors have been defined in the Section "Methods".

The different steps of the Gibbs sampler are detailed below.

Inference of the number of factors

The proposed unsupervised algorithm includes a birth/death process for inferring the number of factors R, i.e., it generates samples R in addition to $\mathbf{M}$ and $\mathbf{A}$. More precisely, at iteration t of the algorithm, a birth, death or switch move is randomly chosen with probabilities $b_{R^{(t)}}$, $d_{R^{(t)}}$ and $s_{R^{(t)}}$. The *birth* and *death* moves consist of increasing or decreasing by 1 the number R of factors using a reversible jump step (see [21] for more details), whereas the *switch* move does not change the dimension of R and requires the use of a Metropolis-Hastings acceptance procedure. Let consider a move, at iteration index t, from the state $\{\mathbf{M}^{(t)}, \mathbf{A}^{(t)}, R^{(t)}\}$ to the new state $\{\mathbf{M}^\star, \mathbf{A}^\star, R^\star\}$. The birth, death and switch moves are defined as follows, similar to those used in [26] (Algorithms 3, 4 and 5).

- ***Birth* move:** When a birth move is proposed, a new signature $\mathbf{m}^\star$ is randomly generated to build $\mathbf{M}^\star = \left[\mathbf{M}^{(t)}, \mathbf{m}^\star\right]$. The new corresponding space is checked so that the signatures are sufficiently distinct and separate from one another. Then, a new factor score coefficient is drawn, for each vector $\mathbf{a}_i$ $(i = 1, \ldots, N)$, from a Beta distribution $\mathcal{B}\left(1, R^{(t)}\right)$, and the new factor score matrix, denoted as $\mathbf{A}^\star$, is re-scaled to sum to one.
- ***Death* move:** When a death move is proposed, one of the factors of $\mathbf{M}^{(t)}$, and its corresponding factor score coefficients, are randomly removed. The remaining factor scores are re-scaled to ensure the sum-to-one constraint.
- ***Switch* move:** When a switch move is proposed, a signature $\mathbf{m}^\star$ is randomly chosen and replaced with another signature randomly generated. If the new signature is too close to another, its corresponding factor scores are proportionately distributed among its closest factors. Indeed, the switch move consists of creating a new signature (birth move) and deleting another one (death move) in a faster single step.

Each move is then accepted or rejected according to an empirical acceptance probability: the likelihood ratio between the actual state and the proposed new state. The factor matrix $\mathbf{M}$, the factor score matrix $\mathbf{A}$ and the noise variance σ^2 are then updated, conditionally upon the number of factors R, using the following Gibbs steps.

Generation of samples according to $f\left(\mathbf{T}|\mathbf{A}, \sigma^2, R, \mathbf{Y}\right)$

Sampling from the joint conditional $f(\mathbf{T}|\mathbf{A}, \sigma^2, R, \mathbf{Y})$ is achieved by updating each column of $\mathbf{T}$ using Gibbs moves. Let denote $\mathbf{T}_{\setminus r}$ the matrix $\mathbf{T}$ whose r-th column has been removed. The posterior distribution of $\mathbf{t}_r$ is the following truncated multivariate Gaussian distribution (MGD)

$$\mathbf{t}_r|\mathbf{T}_{\setminus r}, \mathbf{a}_r, \sigma^2, \mathbf{Y} \sim \mathcal{N}_{\mathcal{T}_r}\left(\boldsymbol{\tau}_r, \mathbf{\Gamma}_r\right) \tag{24}$$

where

$$\begin{aligned}\boldsymbol{\Gamma}_r &= \left[\sum_{i=1}^{N} a_{r,i}^2 \mathbf{P}\Sigma^{-1}\mathbf{P}^T + \frac{1}{s_r^2}\mathbf{I}_R\right]^{-1},\\ \boldsymbol{\tau}_r &= \boldsymbol{\Gamma}_r\left[\sum_{i=1}^{N} a_{r,i}\mathbf{P}\Sigma^{-1}\boldsymbol{\epsilon}_{r,i} + \frac{1}{s_r^2}\mathbf{e}_r\right],\\ \boldsymbol{\epsilon}_{r,i} &= \mathbf{y}_i - a_{r,i}\bar{\mathbf{y}} - \sum_{j\neq r} a_{r,i}\mathbf{m}_j.\end{aligned} \tag{25}$$

For more details on how we generate realizations from this truncated distribution, see [13].

Generation of samples according to $f\left(\mathbf{a}_{1:R-1,i}|\mathbf{T},\sigma^2,R,\mathbf{Y}\right)$
Straightforward computations lead to the posterior distribution of each element of $\mathbf{a}_{1:R-1,i}$

$$f\left(\mathbf{a}_{1:R-1,i}|\mathbf{T},\sigma^2,R,\mathbf{Y}\right) \propto \exp\left[-\frac{1}{2}\bar{\mathbf{a}}_{1:R-1,i}^T\Sigma_{1:R-1,i}^{-1}\bar{\mathbf{a}}_{1:R-1,i}\right] \times \mathbf{1}_{\mathcal{S}}\left(\mathbf{a}_{1:R-1,i}\right) \tag{26}$$

where

$$\begin{aligned}\bar{\mathbf{a}}_{1:R-1,i} &= \mathbf{a}_{1:R-1,i} - \mu_{1:R-1,i},\\ \Sigma_{1:R-1,i} &= \left[\overline{\mathbf{M}}_{\setminus R}^T\Sigma^{-1}\overline{\mathbf{M}}_{\setminus R}\right]^{-1},\\ \mu_{1:R-1,i} &= \Sigma_{1:R-1,i}\left[\overline{\mathbf{M}}_{\setminus R}^T\Sigma^{-1}\overline{\mathbf{M}}_{\setminus R}\right],\\ \overline{\mathbf{M}}_{\setminus R} &= \mathbf{M}_{\setminus R} - \mathbf{m}_R\mathbf{1}_{R-1}^T,\end{aligned} \tag{27}$$

$\mathbf{1}_{R-1} = [1,\ldots,1] \in \mathbb{R}^{R-1}$ and $\mathbf{M}_{\setminus R}$ denotes the factor loading matrix $\mathbf{M}$ whose R-th column has been removed.

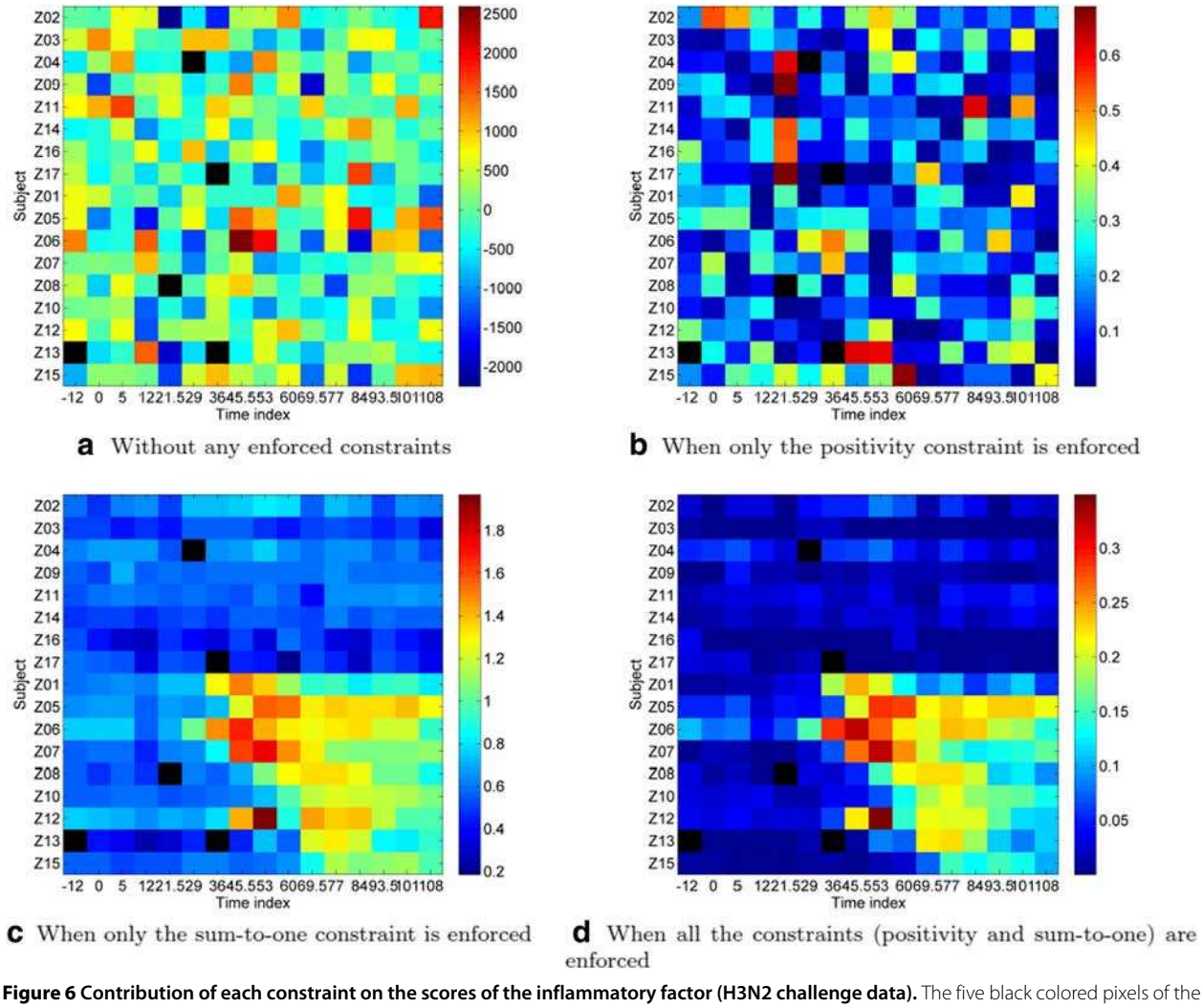

Figure 6 Contribution of each constraint on the scores of the inflammatory factor (H3N2 challenge data). The five black colored pixels of the heatmaps indicate samples that were not assayed. Note that when only the sum-to-one constraint is applied, non-negativity is not guaranteed. However, for this dataset the sum-to-one factor scores turn out to take on non-negative values for the inflammatory factor (but not for the other factors).

Table 9 Contribution of each of uBLU's constraints

	Without constraints	Positivity	Sum-to-one	Positivity and sum-to-one
P-value of the "IFN-gamma pathway"	$6.00.10^{-2}$	$2.05.10^{-2}$	$2.17.10^{-1}$	$\mathbf{1.34.10^{-9}}$
P-value of the "IL23-mediated signaling events"	$2.60.10^{-1}$	$8.37.10^{-2}$	$2.28.10^{-2}$	$\mathbf{2.18.10^{-7}}$

Benefit of constraints in uBLU in terms of gene enrichment in the NCI-curated IFN-gamma and IL23-mediated pathways. As in Tables 6 and 8, the top 200 genes in the inflammatory components, whose scores are shown in Figures 6(a-d), were analyzed using the NCI Pathway Interaction Database. Both positivity and sum-to-one constraints are necessary for uBLU to reveal these two pathways with the high significance (p-value less than 10^{-6}).

Equation (26) shows that the factor score distribution is an MGD truncated on the simplex $\mathcal{S}$ defined in (12).

Generation of samples according to $f\left(\sigma^2|\mathbf{T}, \mathbf{A}, R, \mathbf{Y}\right)$

Using (14) and (16), one can show that the conditional distribution $f(\sigma^2|\mathbf{M}, \mathbf{A}, \mathbf{Y})$ is the following inverse-Gamma distribution

$$\sigma^2|\mathbf{M}, \mathbf{A}, \mathbf{Y} \sim \mathcal{IG}\left(\frac{GN}{2}, \frac{1}{2}\sum_{i=1}^{N}\left\|\mathbf{y}_i - \mathbf{M}\mathbf{a}_i\right\|^2\right). \quad (28)$$

Appendix B: Contribution of each of uBLU's constraints

To illustrate the advantage of enforcing non-negativity and sum-to-one constraints on the factors and on the factor scores, as detailed in the Methods section, we evaluated the effect of successively stripping out these constraints from uBLU. In particular we implemented uBLU under the following conditions: i) without any constraints, ii) with only the positivity constraints on the factors and the scores, iii) with only the sum-to-one constraint on the scores, and iv) with both positivity and sum-to-one constraint on factors and scores as proposed in (5).

Figures 6 display heatmaps of the factor scores of the inflammatory component. The segmentation into two main regions (post-symptomatic samples and asymptomatic samples) becomes apparent only when the sum-to-one constraint is enforced on the scores. To quantify the benefit that is visible in Figure 6 we performed a GEA analysis, reported in Table 9, on the top 200 genes found in each of the inflammatory components found by uBLU implemented with no constraints, positivity constraints, sum-to-one constraints, and both constraints. The table shows that both constraints are necessary to obtain the best enrichment scores (lowest possible p-values).

Additional file

Additional file 1: Supplementary materials on algorithm details and performance validation. Directed acyclic graph (DAG) of the model and flowchart of the proposed algorithm are provided in this additional file. More results on synthetic datasets are also presented to validate the proposed Bayesian algorithm, including a convergence diagnosis.

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
CB, ND, JYT and AH performed the statistical analysis. GG and AZ designed the Flu challenge experiment that generated the data used to compare the methods. All authors contributed to the manuscript and approved the final version.

Acknowledgements
This work was supported in part by DARPA under the PHD program (DARPA-N66001-09-C-2082). The views, opinions, and findings contained in this article are those of the authors and should not be interpreted as representing the official views or policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the Department of Defense. Approved for Public Release, Distribution Unlimited.

Author details
[1]University of Toulouse, IRIT/INP-ENSEEIHT, 2 rue Camichel, BP 7122, 31071 Toulouse cedex 7, France. [2]Department of Medicine and Institute for Genome Sciences and Policy, Duke University, Durham, North Carolina, USA. [3]Center for Computational Biology and Bioinformatics and EECS Department, University of Michigan, 1301 Beal Avenue, Ann Arbor, MI 48109-2122, USA.

Received: 5 December 2012 Accepted: 24 January 2013
Published: 19 March 2013

References

1. Carvalho CM, Chang J, Lucas JE, Nevins JR, Wang Q, West M: **High-dimensional sparse factor modelling: applications in gene expression genomics.** *J Am Stat Assoc* 2008, **103**(484):1438–1456.
2. Paisley J, Carin L: **Nonparametric factor analysis with beta process priors.** In *Proc 26th Annual Int Conf on Machine Learning. ICML 2009.* Montreal, Quebec, Canada; 2009:777–784.
3. Chen B, Chen M, Paisley J, Zaas A, Woods C, Ginsburg GS, Hero AO, Lucas J, Dunson D, Carin L: **Bayesian inference of the number of factors in gene-expression analysis: application to human virus challenge studies.** *BMC Bioinformatics* 2010, **11:**552.
4. West M: **Bayesian Factor regression models in the "Large p, Small n" paradigm.** In *Bayesian Statistics 7*: Oxford University Press; 2003:723–732.
5. Yeung KY, Ruzzo WL: **Principal component analysis for clustering gene expression data.** *Bioinformatics* 2001, **17**(9):763–774.
6. Nascimento JM, Bioucas-Dias JM: **Vertex component analysis: A fast algorithm to unmix hyperspectral data.** *IEEE Trans Geosci Remote Sensing* 2005, **43**(4):898–910.
7. Lee DD, Seung HS: **Algorithms for non-negative matrix factorization.** *Proc Neural Info Process Syst* 2000, **13:**556–562.
8. Fogel P, Young SS, Hawkins DM, Ledirac N: **Inferential, robust non-negative matrix factorization analysis of microarray data.** *Bioinformatics* 2007, **23:**44–49.
9. McLachlan GJ, Bean RW, Peel D: **A mixture model-based approach to the clustering of microarray expression data.** *Bioinformatics* 2002, **18**(3):413–422.
10. Baek J, McLachlan GJ: **Mixtures of common t-factor analyzers for clustering high-dimensional microarray data.** *Bioinformatics* 2011, **27:**1269–1276.

11. Moloshok TD, Klevecz RR, Grant JD, Manion FJ, Speier WF, Ochs MF: **Application of Bayesian decomposition for analysing microarray data.** *Bioinformatics* 2002, **18:**566–575.
12. Fertig EJ, Ding J, Favorov AV, Parmigiani G, Ochs MF: **CoGAPS: an R/C++ package to identify patterns and biological process activity in transcriptomic data.** *Bioinformatics* 2010, **26:**2792–2793.
13. Dobigeon N, Moussaoui S, Coulon M, Tourneret JY, Hero AO: **Joint Bayesian endmember extraction and linear unmixing for hyperspectral imagery.** *IEEE Trans Signal Process* 2009, **57**(11):4355–4368.
14. Huang Y, Zaas AK, Rao A, Dobigeon N, Woolf PJ, Veldman T, Oien NC, McClain MT, Varkey JB, Nicholson B, Carin L, Kingsmore S, Woods CW, Ginsburg GS, Hero A: **Temporal dynamics of host molecular responses differentiate symptomatic and asymptomatic influenza A infection.** *PLoS Genet* 2011, **8**(7):e1002234.
15. Hyvärinen A, Karhunen J, Oja E: *Independent Component Analysis*. New York: John Wiley; 2001.
16. Dueck D, Morris QD, Frey BJ: **Multi-way clustering of microarray data using probabilistic sparse matrix factorization.** *Bioinformatics* 2005, **21:**144–151.
17. Nikulin V, Huang TH, Ng SK, Rathnayake S, McLachlan GJ: **A very fast algorithm for matrix factorization.** *Stat Probability Lett* 2011, **81**(7):773–782.
18. Zaas AK, Chen M, Varkey J, Veldman T, Hero AO, Lucas J, Huang Y, Turner R, Gilbert A, Lambkin-Williams R, Øien NC, Nicholson B, Kingsmore S, Carin L, Woods CW, Ginsburg GS: **Gene expression signatures diagnose influenza and other symptomatic respiratory viral infections in humans.** *Cell Host Microbe* 2009, **6**(3):207–217. [http://www.ncbi.nlm.nih.gov/pubmed/19664979]
19. Winter ME: **N-FINDR: an algorithm for fast autonomous spectral end-member determination in hyperspectral data.** In *Imaging Spectrometry V Proc SPIE 3753*; 1999:266–275.
20. Gilks WR, Richardson S, Spiegelhalter DJ: *Markov Chain Monte Carlo in Practice*. London: Chapman and Hall; 1996. (ISBN: 0-412-05551-1).
21. Green PJ: **Reversible jump Markov chain Monte Carlo computation and Bayesian model determination.** *Biometrika* 1995, **82**(4):711–732.
22. **BFRM Software: bayesian factor regression modelling.** [http://www.stat.duke.edu/research/software/west/bfrm/download.html]
23. Duda RO, Hart PE, Stork DG: *Pattern Classification*. 2nd edition. New York: Wiley-Interscience; 2000.
24. Cox TF, Cox MAA: *Multidimensional Scaling*. London: Chapman and Hall; 1994.
25. Robert CP, Casella G: *Monte Carlo Statistical Methods*. 1edition. New York: Springer-Verlag; 1999.
26. Dobigeon N, Tourneret JY, Chang CI: **Semi-supervised linear spectral unmixing using a hierarchical Bayesian model for hyperspectral imagery.** *IEEE Trans Signal Process* 2008, **56**(7):2684–2695.

doi:10.1186/1471-2105-14-99
Cite this article as: Bazot *et al.*: Unsupervised Bayesian linear unmixing of gene expression microarrays. *BMC Bioinformatics* 2013 **14**:99.