

RESEARCH ARTICLE

Open Access

Bayesian inference of the number of factors in gene-expression analysis: application to human virus challenge studies

Bo Chen^{1†}, Minhua Chen^{1†}, John Paisley¹, Aimee Zaas², Christopher Woods², Geoffrey S Ginsburg², Alfred Hero III³, Joseph Lucas², David Dunson⁴, Lawrence Carin^{1*}

Abstract

Background: Nonparametric Bayesian techniques have been developed recently to extend the sophistication of factor models, allowing one to infer the number of appropriate factors from the observed data. We consider such techniques for sparse factor analysis, with application to gene-expression data from three virus challenge studies. Particular attention is placed on employing the Beta Process (BP), the Indian Buffet Process (IBP), and related sparseness-promoting techniques to infer a proper number of factors. The posterior density function on the model parameters is computed using Gibbs sampling and variational Bayesian (VB) analysis.

Results: Time-evolving gene-expression data are considered for respiratory syncytial virus (RSV), Rhino virus, and influenza, using blood samples from healthy human subjects. These data were acquired in three challenge studies, each executed after receiving institutional review board (IRB) approval from Duke University. Comparisons are made between several alternative means of performing nonparametric factor analysis on these data, with comparisons as well to sparse-PCA and Penalized Matrix Decomposition (PMD), closely related non-Bayesian approaches.

Conclusions: Applying the Beta Process to the factor scores, or to the singular values of a pseudo-SVD construction, the proposed algorithms infer the number of factors in gene-expression data. For real data the “true” number of factors is unknown; in our simulations we consider a range of noise variances, and the proposed Bayesian models inferred the number of factors accurately relative to other methods in the literature, such as sparse-PCA and PMD. We have also identified a “pan-viral” factor of importance for each of the three viruses considered in this study. We have identified a set of genes associated with this pan-viral factor, of interest for early detection of such viruses based upon the host response, as quantified via gene-expression data.

1. Background

When performing gene-expression analysis for inference of relationships between genes and conditions/phenotypes, one typically must analyze a small number of samples, each composed of expression values from tens of thousands of genes. In this setting the observed data is $\mathbf{X} \in \mathbb{R}^{p \times n}$, where each column corresponds to one of n samples, quantifying the associated gene-expression values for all p genes under investigation. We typically

must address the “large p , small n ” problem [1], in which often $n \ll p$. Therefore, to yield reliable inference, one must impose strong restrictions on the form of the model.

When developing regression and classification models for gene-expression data, a widely employed assumption (restriction) is that the model parameters are sparse, implying that only a small subset of the genes are important for prediction. If only a small set of genes ($\ll p$) are responsible for differences in disease groups, then reliable inference may often be performed even when $n \ll p$. Example approaches that have taken this viewpoint are lasso [2], the elastic net [3], and related Bayesian approaches [4]. In fact, sparse regression and

* Correspondence: lcarin@ee.duke.edu

† Contributed equally

¹Electrical and Computer Engineering Department, Duke University, Durham, NC, USA

Full list of author information is available at the end of the article

classification algorithms are widely used in many statistics and machine-learning applications, beyond gene analysis [5-7].

An important research direction for gene-expression analysis, and many other applications, involves the use of factor models [8-11]. To address the “large p , small n ” problem, sparseness is again imposed, now typically on the factor loadings. Specifically, in an unsupervised setting the data are assumed to satisfy

$$X = AS + E \quad (1)$$

where $A \in \mathbb{R}^{p \times r}$, $S \in \mathbb{R}^{r \times n}$ and $E \in \mathbb{R}^{p \times n}$; if covariates are available they may also be considered in the model [11], with none assumed here. Note that here and henceforth we assume that the gene-expression data are centered in advance of the analysis; otherwise, there should be an intercept added to the model. Considering the j th sample, x_j , corresponding to the j th column of X , the model states that $x_j = As_j + e_j$, where s_j and e_j are the j th columns of S and E , respectively.

The columns of A represent the factor “loadings”, and rows of S are often called factors. To address the fact that $n \ll p$, researchers have typically imposed a sparseness constraint on the columns of A [11], with the idea that each column of A should ideally (in the gene application) correspond to a biological “pathway”, which should be defined by a relatively small number of correlated genes. Within Bayesian formalisms, the sparse columns of A are typically imposed via spike-slab-like priors [1], [11], or alternatively via shrinkage (e.g., Student- t [6]) priors. Several non-Bayesian approaches have also been introduced, including sparse-PCA [12] and the related Penalized Matrix Decomposition (PMD) [13].

A problem that is receiving increased attention in factor-analysis-based approaches is a means of defining an appropriate number of factors (i.e., to infer r). The non-Bayesian approaches are often sequential, and one may infer r by monitoring the error $\|E\|_F$ as a function of iteration number [12], [13]. In many previous Bayesian approaches r has just been set [11], and presumably many non-biologically-relevant factor loadings are inferred. A computationally expensive reverse-jump MCMC approach has been developed [14], with computational efficiency improved in [15] while also considering a default robust prior specification. Perhaps the most widely employed approach [16-18] for choosing r is the Bayesian information criteria (BIC). A disadvantage is that conditioning on a fixed choice of the number of factors ignores uncertainty and the BIC is not well justified in hierarchical models, as the number of parameters is unclear.

There has been recent interest in applying nonparametric Bayesian methods [8], [9] to infer r (in fact, a

posterior distribution on r), based on the observed data X . An example of recent research in this direction employs the Indian Buffet Process (IBP) [19], [20]. In this paper we also consider the Beta Process (BP), recognizing that the BP and IBP are closely linked [21], [22].

For data sets with very large p (e.g., 10,000 or more), computational efficiency is of major practical importance. In previous use of nonparametric Bayesian methods to this problem, a Gibbs sampler has typically been employed [11]. The BP-based formulation admits a relatively simple variational Bayesian (VB) [23] approximation to the posterior, which is considerably faster than Gibbs sampling. An advantage of a VB analysis, in addition to speed, is that convergence is readily monitored (for the Gibbs sampler there are typically challenges when assessing convergence). We perform a comparison of the difference in inferred model parameters, based on VB and Gibbs analysis.

The specific data on which the models are employed correspond to gene-expression data from recent viral challenge studies. Specifically, after receiving institutional review board (IRB) approvals from Duke University, we performed three separate challenge studies, in which individuals were inoculated with respiratory syncytial virus (RSV), Rhino virus, and influenza. Informed consent was used in all studies. Using blood samples collected sequentially over time, we have access to gene-expression data at pre-inoculation, just after inoculation, and at many additional time points up to the point of full symptoms (such data were collected on all subjects, although not all became symptomatic). Using these data, we may investigate time-evolving factor scores of samples, to examine how the response to the virus evolves with time. Of particular importance is an examination of the factors of importance for individuals who became symptomatic relative to those who did not. In the factor analysis we consider data individually for each of the three viruses (at all times), as well as for all three viruses in a single analysis (seeking pan-viral factors). Results are generated based on nonparametric Bayesian approaches to factor analysis, employing the Beta Process, the Indian Buffet Process, and a related sparseness-constrained pseudo-SVD construction (a Bayesian construction of sparse-PCA [12]). We also make comparisons to the non-Bayesian Penalized Matrix Decomposition (PMD) [13].

II. Results

A. Brief summary of models

We first provide a brief intuitive explanation of the workings of the different Bayesian models considered. These models are built around the Indian buffet process (IBP) [19], so named for the following reason. In the factor model of (1), the columns of A represent factor

loadings in which the gene-expression values for sample j are expressed: $\mathbf{x}_i = \mathbf{A}\mathbf{s}_j + \mathbf{e}_j$. One construction of the IBP constitutes a set of candidate columns of \mathbf{A} , and these are termed “dishes” at an Indian “buffet”. Each of the n samples $\{\mathbf{x}_j\}_{j=1,n}$ correspond to “customers” at the buffet; each customer selects a subset of dishes from the buffet (*i.e.*, selects a subset of candidate columns of \mathbf{A}). The IBP is constructed such that the more a particular dish (column of \mathbf{A}) is used by a subset of customers $\{\mathbf{x}_j\}_{j=1,m}$, the more probable it is that it will be used by other customers. Thus, the IBP imposes the idea that many of the samples $\{\mathbf{x}_j\}_{j=1,n}$ will utilize the same subset of columns of \mathbf{A} , but each sample may also utilize idiosyncratic factor loadings, representing unique characteristics of particular samples. The IBP construction does not impose a total number of factors for the data $\{\mathbf{x}_j\}_{j=1,m}$, with this number inferred by the analysis. Thus, the IBP is a natural Bayesian method for inferring the number of factors appropriate for representing all observed data $\{\mathbf{x}_j\}_{j=1,m}$. A convenient means of implementing the IBP employs the Beta process (BP) [21].

There are multiple ways in which one may utilize the IBP/BP within the factor model, with three such methods considered here: (i) the BP is applied to the factor scores \mathbf{S} (termed below the BP construction), (ii) the IBP is employed on the factor loadings \mathbf{A} [8] (termed below the IBP construction), and (iii) a BP-like construction is employed to implement a Bayesian construction of a singular-value decomposition of \mathbf{X} (termed below the pseudo-SVD construction). To realize the approximate posterior density function for the parameters of these models, we have considered both MCMC and VB computational methods. The specifics of the BP, IBP and pseudo-SVD methods, as well as computational details, are provided in Section IV.

B. Synthesized Data

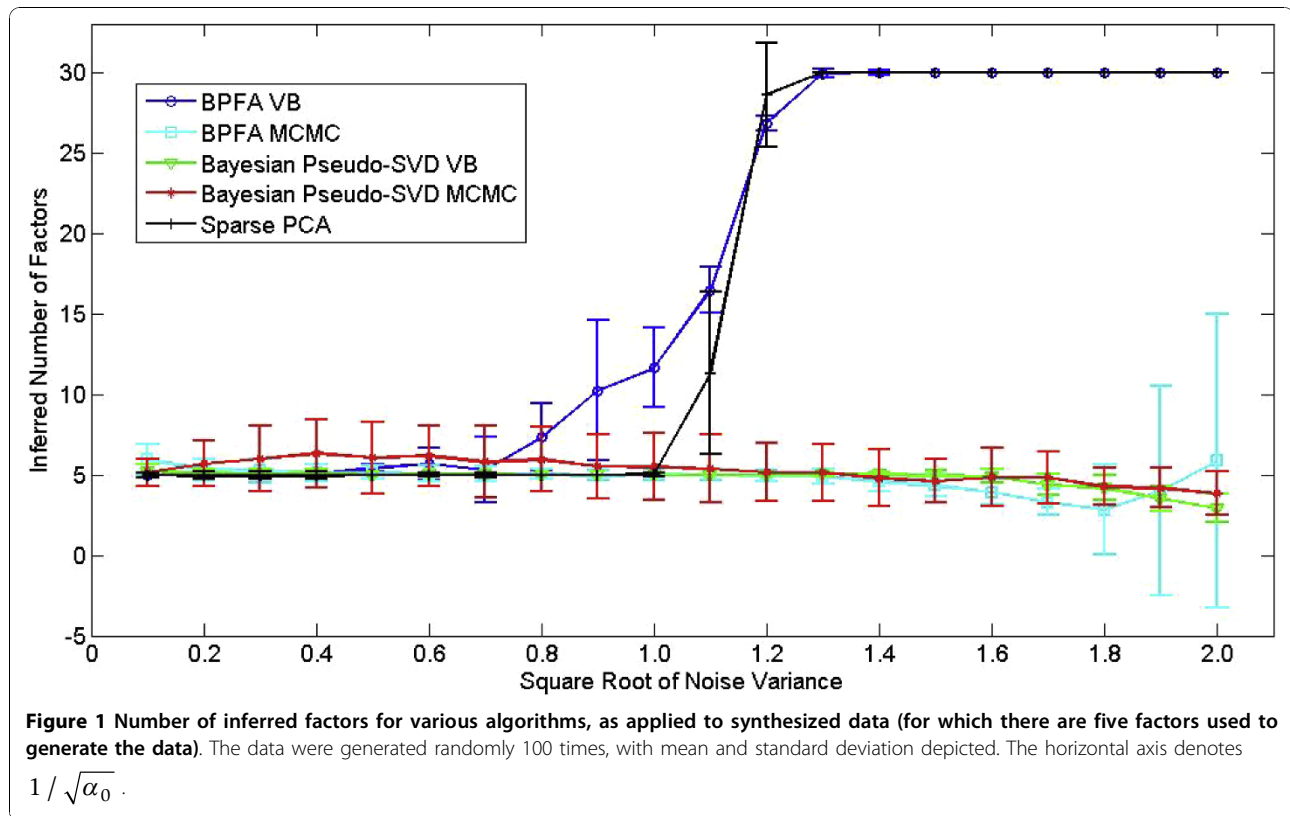
The first validation example we considered was taken from [8]. In this example the gene-factor connectivity matrix of an E-coli network is employed to generate a synthetic dataset having 100 samples of 50 genes and 8 underlying factors. The data had additive white Gaussian noise with a signal-to-noise-ratio of 10. For this very small-scale example we considered all three Bayesian methods (BP, IBP and pseudo-SVD); in each case we considered both MCMC and VB methods for inferring the posterior density function. We also considered the non-Bayesian PMD and sparse-PCA [13], [24]. All methods performed well in uncovering the proper number of factors, and in capturing the proper genes associated with each factor. For brevity we do not provide further details on this example. While it is worthy of consideration because it was considered in related published research [8], its small-scale nature (only 50 genes)

makes it less relevant for the large-scale real application we consider below. Therefore, in the next synthetic example we consider a much larger-scale problem, and consequently for that problem we were unable to test against the IBP method.

The synthetic data were generated as follows. A total of $p = 10,000$ features (“genes”) are employed, and the expression value for these p genes was constituted using five factors ($r = 5$) plus a noise term \mathbf{E} (*i.e.*, via the model in (1)). For each of the five factors, a unique set of 50 genes were selected and were given a factor-loading value of one. In addition, ten more genes were selected, with these shared among all five factors (again with unit-amplitude contribution to the factor loadings). Thus, a total of 260 genes contributed non-zero loadings to at least one of the five factors. For all other genes the factor-loading contribution was set to zero. The above construction defines the sparse matrix \mathbf{A} in (1). The components of $\mathbf{S} \in \mathbb{R}^{r \times n}$, for $n = 150$ samples, are drawn i.i.d. from $\mathcal{N}(0, 1)$. The elements of the noise matrix \mathbf{E} are drawn i.i.d. from $\mathcal{N}(0, \alpha_0^{-1})$. The data \mathbf{X} were then utilized within the various factor-analysis models, with the data-generation process repeated 100 independent times (100 different \mathbf{X}), with mean and standard-deviation results presented on the inferred model parameters (discussed below), based on the 100 runs.

We consider a range of noise variances $1/\alpha_0$ to constitute \mathbf{E} , to address model performance as a function of the signal-to-noise ratio (SNR). As one definition of SNR, one may consider the average energy contributed from a non-zero gene to a particular factor, relative to the energy in the noise contribution for that gene, from \mathbf{E} . Based on the fact that the non-zero components of \mathbf{A} have unit amplitude, and the components of \mathbf{S} are drawn from $\mathcal{N}(0, 1)$, on average (across many samples) the energy contributed by a non-zero gene to a particular factor is one. The average noise energy contributed to each gene is $1/\alpha_0$. Hence, the ratio of these two quantities, α_0 , may be considered as a measure of SNR. Other measures of SNR may be defined with respect to this model, each of which will be defined in terms of α_0 .

In Figure 1 are presented the average number of inferred factors and the associated standard deviation on this number, for the BP and pseudo-SVD models. We also compare to the sparse-PCA model in [12]. The integer K represents the truncation level in the models, defining the maximum number of columns of \mathbf{A} considered for analysis, from which $r \leq K$ columns are inferred as needed to represent the data \mathbf{X} . This is discussed in detail in Section IV. In these examples the models were each truncated to $K = 30$ factors. Consequently, when 30 factors are used, the models have effectively failed, since the true number of factors is 5 and 30 is the



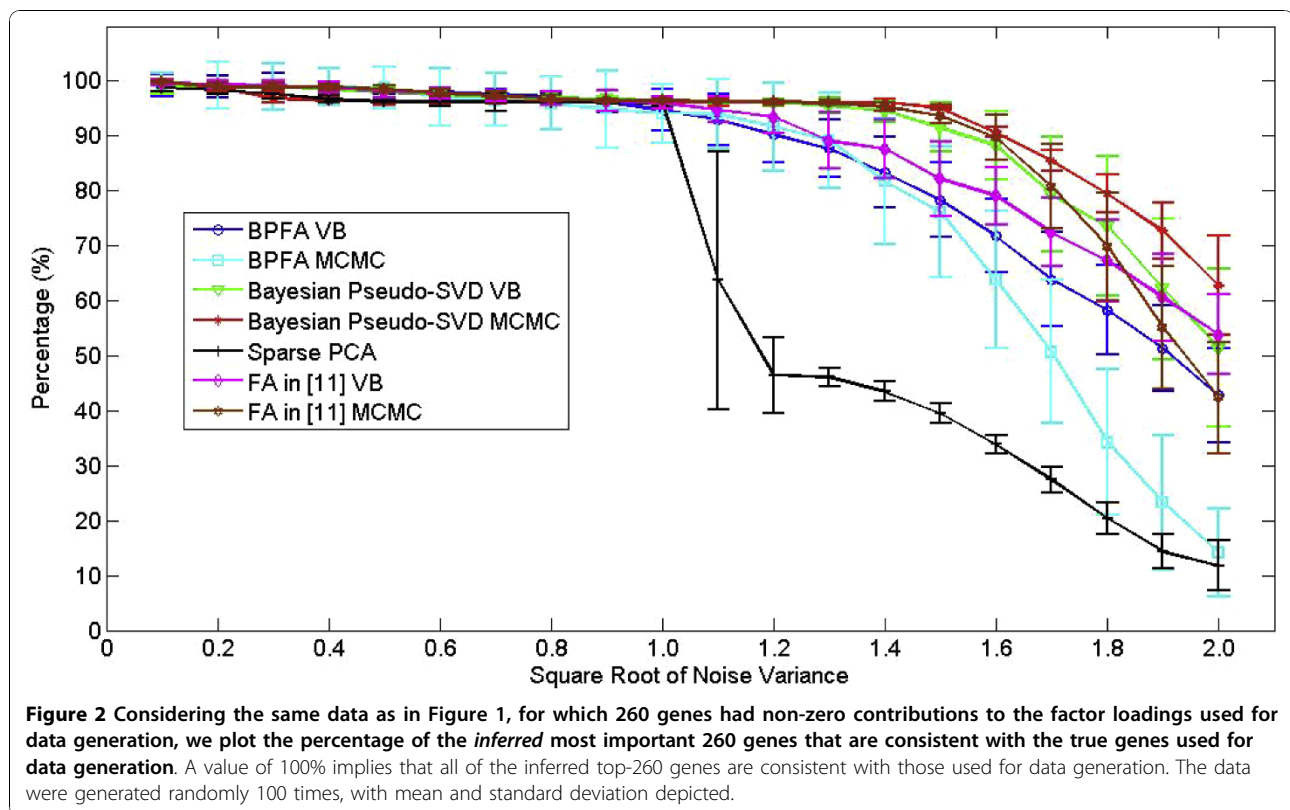
maximum allowed within the model, given the truncation level under consideration. The MCMC results are based upon 2000 burn-in iterations and 1000 collection iterations (the results are similar when 10,000 collection iterations are employed). Results are shown as a function of the standard deviation of the noise, $1 / \sqrt{\alpha_0}$. The sparse-PCA model works well up to the point that the noise variance equals the amplitude of the non-zero values in **A** (approximate SNR of one), while most of the Bayesian methods infer the proper number of factors to a higher level of noise.

In Figure 2 we examine how meaningful the inferred factor loadings are. Specifically, recall that the data are based upon 260 unique genes that contribute to the factor loadings. Based on the inferred factor loadings, we rank the genes based upon their strength in the loadings. We then rank the genes from 1 to 260, based on the above strength, and examine the percentage of the top 260 *inferred* genes are consistent with truth. Considering Figure 2, all of the Bayesian methods perform well in this task, up to a noise standard deviation of approximately 1.3, while sparse-PCA performs degrades quickly beyond standard deviations of one (for SNR values below one). Note that we also consider the Bayesian factor analysis model in [11]; we did not consider this

method in Figure 1 because it does not have a mechanism for estimating r -we simply set $r = K$ in this analysis, using the same $K = 30$ as employed for the other Bayesian methods. In [11] the authors only considered an MCMC implementation, where here we consider both MCMC and VB inference for this model; further, here we have employed a Student-t prior on the components of the factor loading matrix **A**, where in [11] a spike-slab prior was employed.

Concerning sparse-PCA [12] (and PMD, not shown), every effort was made to optimize the model parameters for this task. Our experience is that, while sparse-PCA and PMD [13] are very fast algorithms, and generally quite effective, they are not as robust to noise as the Bayesian methods (the Bayesian methods are also less sensitive to parameter settings). It is possible that the sparse-PCA and PMD results could be improved further if the model parameters are optimized separately for each noise level (and the Bayesian results may also be improved with such tuning). However, the model parameters were fixed for all noise variances considered (since the noise variance is often not known *a priori*, with the sparse-PCA carefully tuned to achieve the best results in such a circumstance).

We also performed additional simulated examples of the type discussed above, the details of which are omitted for brevity. In those experiments the different



genes did not have the same noise variance. The Bayesian methods, which as indicated above infer the noise variance separately for each gene, performed as well as in Figures 1 and 2. However, the sparse-PCA and PMD models performed relatively poorly in this case, since they assume the same noise variance for all genes. The assumption of a constant noise variance for each gene may not be as appropriate for real data.

C. Details on Data Collections for Three Viral Challenge Studies

We considered three cohorts of healthy volunteers experimentally infected with either rhinovirus, respiratory syncytial virus (RSV) or influenza A; these three challenges were performed separately, with no overlap in the subjects. All exposures were approved by the Duke University institutional review board and conducted according to the Declaration of Helsinki. The three challenges are briefly summarized here, with further details provided in [25].

Human Rhinovirus cohort

We recruited 20 healthy volunteers via advertisement to participate in the rhinovirus challenge study through an active screening protocol at the University of Virginia (Charlottesville, VA). On the day of inoculation, 10^6 TCID₅₀ GMP rhinovirus (Johnson and Johnson) was inoculated intranasally. Subjects were admitted to the

quarantine facility for 48 hours following rhinovirus inoculation and remained in the facility for 48 hours following inoculation. Blood was sampled into PAXGene™ blood collection tubes at pre-determined intervals post inoculation. Nasal lavage samples were obtained from each subject daily for rhinovirus titers to accurately gauge the success and timing of the rhinovirus inoculation. Following the 48th hour post inoculation, subjects were released from quarantine and returned for three consecutive mornings for sample acquisition and symptom score ascertainment.

Human RSV cohort

A healthy volunteer intranasal challenge with RSV A was performed in a manner similar to the rhinovirus intranasal challenge. The RSV challenge was performed at Ret-roscreen Virology, Ltd (Brentwood, UK) using 20 pre-screened volunteers who provided informed consent. On the day of inoculation, a dose of 10^4 TCID₅₀ respiratory syncytial virus (RSV; serotype A) manufactured and processed under current good manufacturing practices (cGMP) by Meridian Life Sciences, Inc. (Memphis, TN USA) was inoculated intranasally per standard methods. Blood and nasal lavage collection methods were similar to the rhinovirus cohort, but continued throughout the duration of the quarantine. Due to the longer incubation period of RSV A, subjects were not released from quarantine until after the 165th hour and

were negative by rapid RSV antigen detection (Binax-Now Rapid RSV Antigen; Inverness Medical Innovations, Inc).

Influenza cohort

A healthy volunteer intranasal challenge with influenza A/Wisconsin/67/2005 (H3N2) was performed at Retro-screen Virology, LTD (Brentwood, UK), using 17 pre-screened volunteers who provided informed consent. On the day of inoculation, a dose of 106 TCID₅₀ Influenza A manufactured and processed under current good manufacturing practices (cGMP) by Bayer Life Sciences, Vienna, Austria was inoculated intranasally per standard methods at a varying dose (1:10, 1:100, 1:1000, 1:10000) with four to five subjects receiving each dose. Due to the longer incubation period of influenza as compared to rhinovirus, subjects were not released from quarantine until after the 216th hour. Blood and nasal lavage collection continued throughout the duration of the quarantine. All subjects received oral oseltamivir (Roche Pharmaceuticals) 75 mg by mouth twice daily prophylaxis at day 6 following inoculation. All patients were negative by rapid antigen detection (BinaxNow Rapid Influenza Antigen; Inverness Medical Innovations, Inc) at time of discharge.

For each viral challenge, subjects had samples taken 24 hours prior to inoculation with virus (baseline), immediately prior to inoculation (pre-challenge) and at set intervals following challenge. For the rhinovirus challenge, peripheral blood was taken at baseline, then at 4 hour intervals for the first 24 hours, then 6 hour intervals for the next 24 hours, then 8 hour intervals for the next 24 hours, and then 24 hour intervals for the remaining 3 days of the study. For the RSV and influenza challenges, peripheral blood was taken at baseline, then at 8 hour intervals for the initial 120 hours, and then 24 hours for the remaining 2 days of the study. All results presented here are based on gene-expression data from blood samples. For the RSV and Rhino virus cases not all blood samples were converted to gene expression values, as a cost-saving measure. Hence, for these two cases the gene expression data are not sampled as finely in time as are the influenza data.

In the statistical analysis, the matrix X in (1) has columns that correspond to the n samples; $n = n_s n_t$, with n_s representing the number of subjects and n_t the number of sample time points. We do not impose a prior on the time-dependence of the factors scores, and uncover this time dependence via the inferred posterior distribution of factor scores S .

D. Analysis of influenza data

The gene-expression data consisted of over $p = 12,000$ genes, and consequently we found that the IBP approach developed in [8] was computationally

intractable. We found that the VB and MCMC results were generally in good agreement for this real data, and therefore the two very distinct computational tools served to cross-validate each other. The VB and MCMC computations also required similar CPU time (for the number of Gibbs iterations considered); while the VB analysis required far fewer iterations to converge, each iteration is significantly more expensive than that associated with the Gibbs sampler.

For brevity, we here focus exclusively on MCMC solutions when considering Bayesian analysis. Results are presented using the BP and pseudo-SVD methods, as well as via PMD [13] (similar results were computed using sparse-PCA [24]). We note that the design of each the experiments involves samples from the *same* subjects observed at multiple time points (with different subjects for the three viruses). Therefore, the assumption within the models that the samples at different times are statistically independent may warrant reconsidering in future studies. This subject has been considered in related work [26], although that research assumes a known factor structure and Gaussian latent factors.

We first consider results based on the BP as applied to the factor scores. In these results we set $K = 30$ (recall this is the truncation level on the number of factors), and inferred approximately $r = 13$ important factors (see Figure 3); although only approximately $r = 13$ factors are used, we show the factor scores for all $K = 30$ possible factors such that the sparseness of the unused factors is evident, as inferred via the posterior. The results in Figure 3 correspond to one example (representative) collection sample from the Gibbs sampler; Factor 1, which is most closely tied to the symptomatic/asymptomatic response, is employed by all data, while other factors are used more idiosyncratically (e.g., Factors 3 and 14 are only used by a small subset of the data samples; see the detailed discussion of the model in the Methods section).

At each time point, there are data from 17 subjects (the same individuals were sampled at a sequence of times). The horizontal axis in Figure 3 corresponds to a sequence of *groups* of data, proceeding in time from inoculation, with generally 17 samples per time point (all data will be released for other investigators to experiment with). The blue points correspond to samples of individuals who eventually became symptomatic, and the red points to asymptomatic individuals.

The vertical axis in these plots corresponds to the factor score associated with the respective sample. We observe in Figure 3 that Factor 1 (the factor indexing is arbitrary) provides a clear discriminator of those who will become symptomatic, particularly as time proceeds (note that the model is completely unsupervised, and

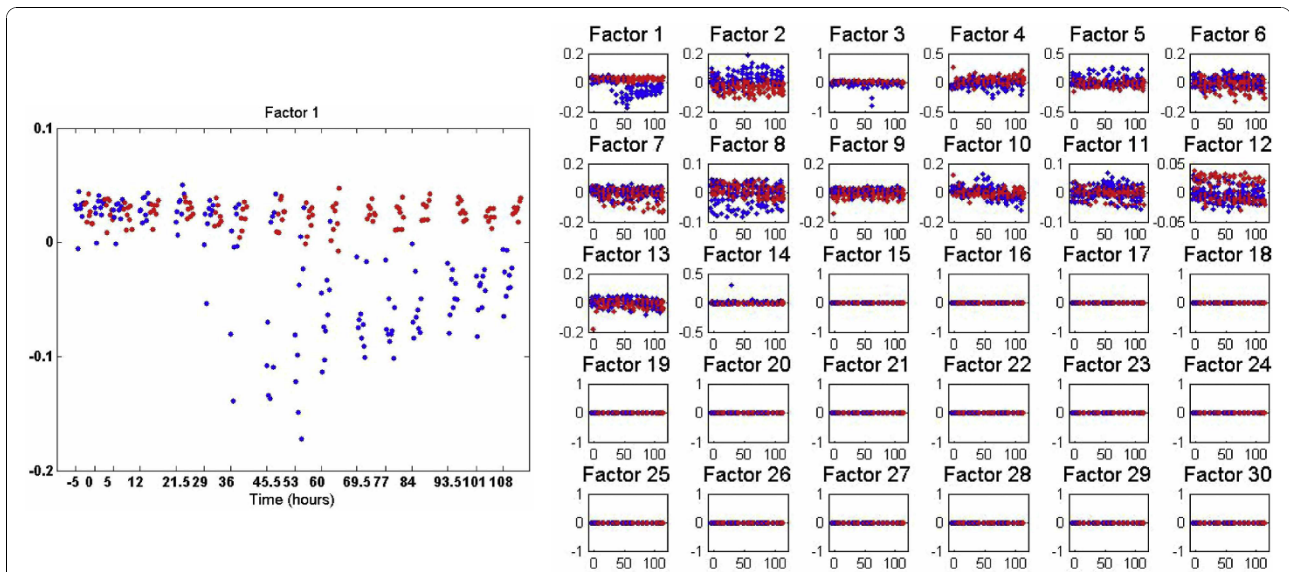


Figure 3 Factor-analysis of Flu data with BP applied within design of factor scores, as discussed in Section IV-B. The MCMC inference was based on 2000 burn-in iterations and 500 collection iterations, and factor scores are depicted for one (typical) collection sample from the Gibbs sampler. Approximately thirteen factors were inferred with non-zero factor scores (shown at right), and at left is a blow-up of the factor that most separates symptomatic (blue) from asymptomatic (red) samples. The horizontal axis denotes time in hours. The data were collected in groups, at discrete times; the results at a given time are shifted slightly along the horizontal axis with respect to one another, to enhance readability.

therefore this discriminating power was uncovered without using label information).

Having introduced the form of the data presentation, we now present results using the pseudo-SVD method and PMD; for the pseudo-SVD method we again show one (typical) sample from the Gibbs collection samples, while for PMD the results are the single solution. In

Figures 4 and 5 we present results, respectively, for the Bayesian pseudo-SVD model and for PMD [13]. For the Bayesian methods we again set $K = 30$. Both methods uncover a relatively small (less than K) number of relevant factors.

Note that in each case there appears to be one factor that clearly distinguishes symptomatic vs. asymptomatic,

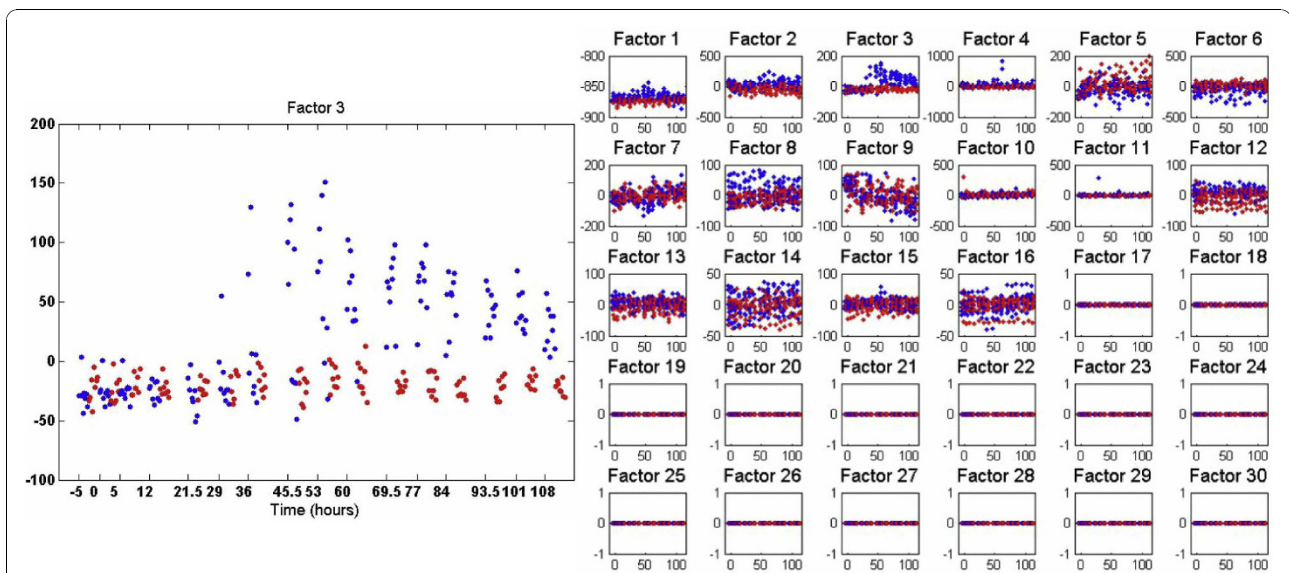
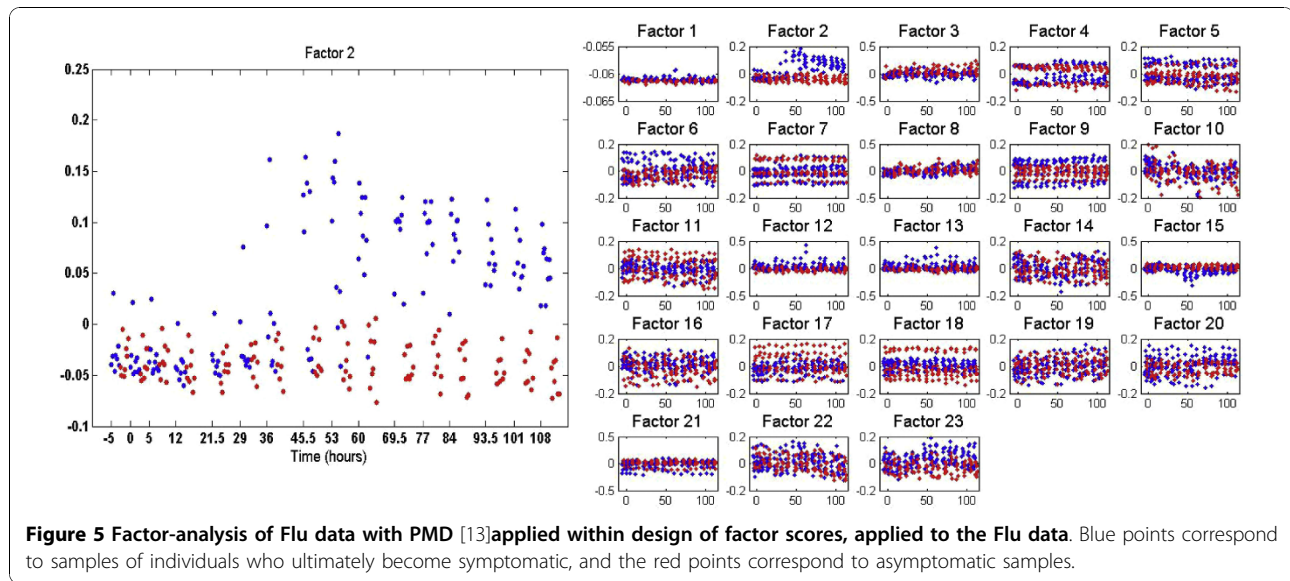


Figure 4 Factor-analysis of Flu data with Bayesian pseudo-SVD applied within design of factor scores, applied to the Flu data. Results are presented in the same form as Figure 3.



particularly as time increases. Upon examining the important genes in each of these factors, one recognizes a high level of overlap (suggesting consistency between the different methods). Further discussion of the associated genes and their biological significance is provided in [25].

E. Pan-viral factors

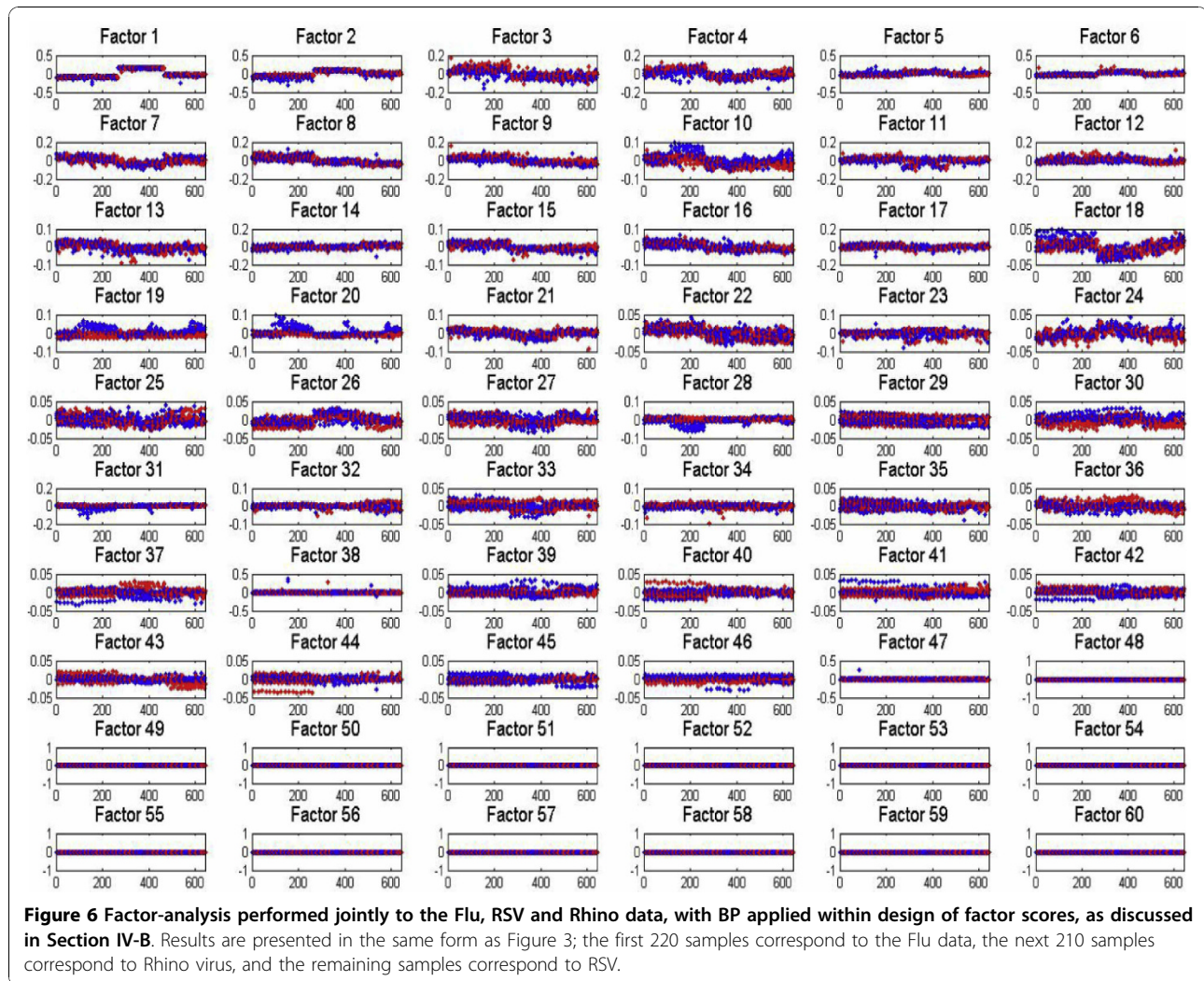
We now consider a “pan-viral” analysis, in which data from all three viruses are analyzed *jointly*. For further conciseness, for this example we only present results for the BP applied to the factor scores; similar results were obtained with the Bayesian pseudo-SVD framework and by PMD.

Since three viruses are now considered jointly, we have increased K to $K = 60$ in this example, and now approximately 46 factors were inferred (with non-zero factor scores). Considering Figure 6, we note that Factor 20 provides good discrimination between the symptomatic (blue) and asymptomatic (red) samples, with this factor examined more closely in Figure 7. This same factor is able to distinguish the samples of each virus, at sufficient time after inoculation (a *single* “pan-viral” factor has been inferred, able to separately distinguish symptomatic vs. asymptomatic for each of the three viruses considered). Factor 19 in Figure 6 also appears to provide separation between symptomatic and asymptomatic samples; however, this is manifested because it contains two genes that are highly discriminative (SERPING1 and TNFAIP6), with most of the other genes in Factor 19 not discriminative. When addressing biological significance in [25], the focus is on Factor 20 in Figure 6, as it contains numerous discriminative genes. In these figures we are again showing one (typical) sample from the Gibbs collection.

It is also of interest to consider Factors 1 and 2 in Figure 6. Each of the samples from the individual viruses is offset by a distinct amplitude, almost entirely *independent* of whether the sample was symptomatic or asymptomatic. This phenomenon associated with Factors 1 and 2 in Figure 6 is attributed to challenge-study-dependent offsets in the data (the gene-expression values were obtained separately for each of these studies, and the data normalized separately), which account for different normalizations of the data between the three distinct viral challenges. This underscores that not all factors have biological significance, with some a consequence of the peculiarities of gene-expression data (study-dependent offsets in normalization). The other factor-analysis methods (omitted here for brevity) produced very similar normalization-related factors.

In Figure 8 are depicted the important genes associated with the discriminative pan-viral Factor 20 in Figure 6. It is a subject of further research, but based on the data analyzed thus far, it appears the FA model applied to gene-expression data cannot distinguish well *between* the different viruses. However, we have applied FA jointly to our pan-virus data and to bacterial data available from related but distinct studies [27]. From that analysis we are able to distinguish between viral-based phenotypes and bacteria-based phenotypes; this is discussed in greater detail in [25].

We have here identified many genes that are inferred to be connected with the viruses under study. It has been observed, by the medical doctors on our research team, that the inferred genes are closely aligned with relevant known pathways, with this discussed in detail in [25].



III. Conclusions

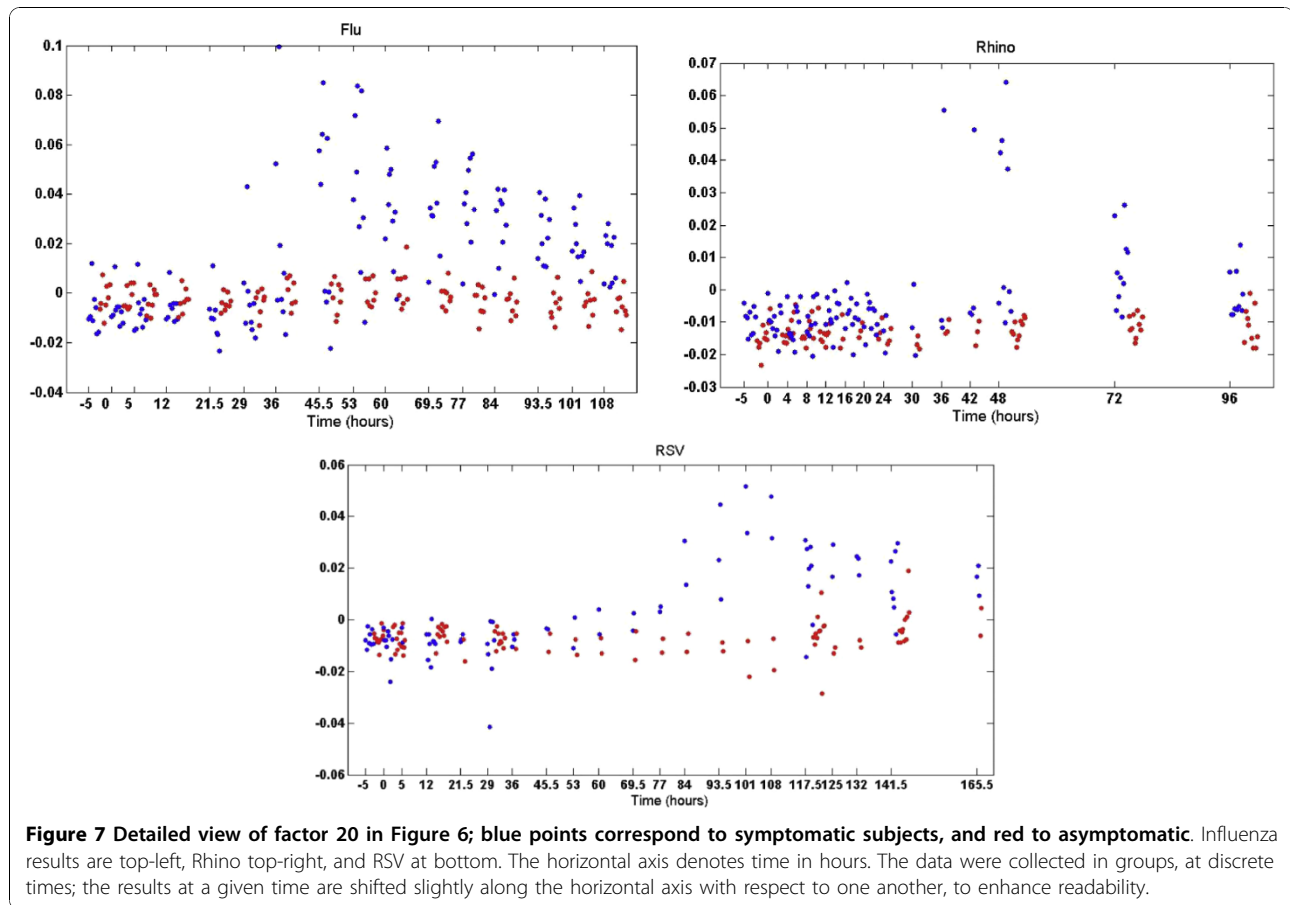
We have examined two distinct but related objectives. First, in the context of Bayesian factor analysis, we have examined three ways of inferring an appropriate number of factors. Each of these methods is based on a different means of leveraging the utility of the Beta Process, and the closely related Indian Buffet Process (IBP). In the context of such models, we have examined inference based on variational Bayesian analysis, and based on a Gibbs sampler. We have also compared these Bayesian approaches to state-of-the-art non-Bayesian factor models.

The second contribution of this paper is the introduction of a new set of gene-expression data, from three time-evolving viral challenge studies. These data allow one to examine the time-evolution of Rhino virus, RSV and Influenza-A. In addition to the gene-expression data, we have also recorded clinical symptom scores, to which the gene-expression analysis may be compared.

With the limited space available here, we have presented results on the Influenza data alone, and for all three viruses together (a “pan-viral” analysis).

Based on this study, we may make the following observations. For the number of Gibbs iterations deemed necessary, the VB and MCMC inference approaches required comparable computation time (VB was slightly faster, but not substantially). Although VB requires far fewer iterations (converges typically in 50 iterations), each VB iteration is significantly more expensive than that associated with MCMC. The advantage of using these two very distinct computational methods on the models considered is that they serve to cross-validate each other (providing confidence in the results, when these two very different methods agree, as they generally did in the studies considered).

Of the three methods of inferring the number of factors, the IBP applied to the factor loadings works well for small-scale problems, but it is computationally



intractable for the large-scale viral data considered here. Applying the Beta Process to the factor scores, or to the singular values of a pseudo-SVD construction, yields reliable and high-quality results.

It is not our purpose to provide a detailed (perhaps philosophical) discourse on the relative merits of Bayesian and non-Bayesian approaches. However, we observed that the non-Bayesian Penalized Matrix Decomposition (PMD) yielded very high-quality results, as long as the model parameters were set carefully via cross-validation; very similar phenomenon was observed for the closely related sparse-PCA. Both PMD and sparse-PCA infer an appropriate number of factors, but one must very carefully set the stop criterion. Since PMD and sparse-PCA are much faster than the Bayesian approaches, perhaps a good compromise is to use the output of these models to initialize the Gibbs sampler in a Bayesian solution (this is a subject for future research).

Concerning the viral data, it was observed that all methods were able to infer a factor that was capable of distinguishing those subjects who would become symptomatic from those who would not. It was possible to infer a “pan-viral” factor, that was discriminative for all viruses considered.

The evolution of the factor scores tracked well the recorded clinical symptom scores. Further, for the discriminative factor, there was a good association between the genes inferred as important and the associated biology [25] (with interpretation provided by the medical doctors on our research team).

IV. Methods

A. Basic sparse factor model

Recall the factor model in (1); r defines the number of factors responsible for the data \mathbf{X} , and it is not known in general, and must be inferred. Within the analysis we will consider K factors (K columns of \mathbf{A}), with K set to a value anticipated to be large relative to r . We then infer the number of columns of \mathbf{A} needed to represent the observed data \mathbf{X} , with this number used as an estimate of r . Since we will be performing a Bayesian analysis, we will infer a posterior density function on r . Henceforth we assume \mathbf{A} has K columns, with the understanding that we wish to infer the $r < K$ columns that are actually needed to represent the data.

Let \mathbf{a}_k represent the k th column of \mathbf{A} , for $k = 1, \dots, K$, and \mathbf{e}_j and \mathbf{s}_j represent respectively the j th columns of \mathbf{E} and \mathbf{S} (with $j = 1, \dots, n$). Within the imposed

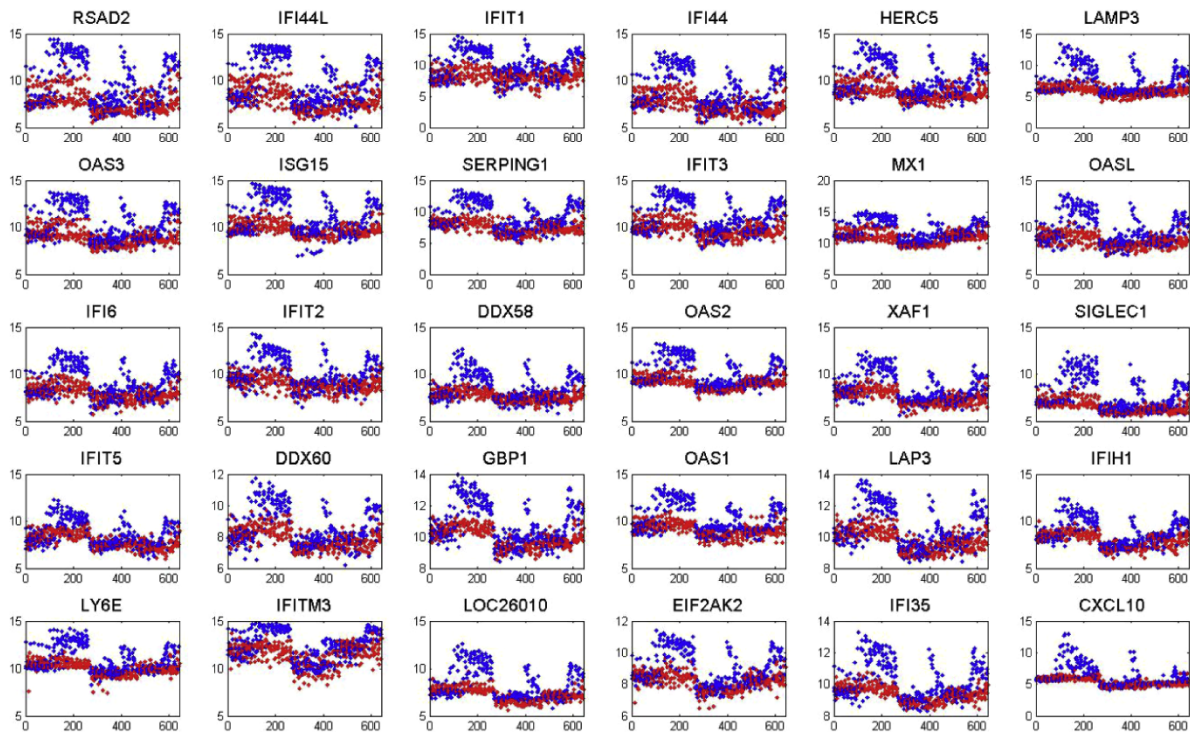


Figure 8 Set of important genes inferred for combined analysis of Flu, RSV and Rhino data, associated Factor 20 from the BP applied to the factor scores (Figure 6). Blue points correspond to samples of individuals who ultimately become symptomatic, and the red points correspond to asymptomatic samples. The first 220 samples correspond to the Flu data (encompassing a total of 108 hrs), the next 210 samples correspond to Rhino virus (encompassing a total of 96 hrs), and the remaining samples correspond to RSV (encompassing a total of 165.5 hrs).

prior, vectors e_j and s_j are generated as $s_j \sim \mathcal{N}(0, I_K)$, and $e_j \sim \mathcal{N}(0, \text{diag}(\psi_1^{-1}, \dots, \psi_p^{-1}))$; I_K is the identity matrix and the precisions (ψ_1, \dots, ψ_p) are all drawn i.i.d. from a gamma prior.

One may consider many alternative means of defining sparseness on the a_k , with the choice often dictated by convenience; we discuss two such methods here. In one approach [11] one may employ a spike-slab prior:

$$\begin{aligned} A_{lk} &\sim w_{lk}\delta_0 + (1 - w_{lk})\mathcal{N}(0, \alpha_k^{-1}), \\ w_{lk} &\sim \text{Beta}(a, b), \alpha_k \sim \text{Gamma}(c, d) \end{aligned} \quad (2)$$

where (a, b) are selected as to strongly favor $w_{lk} \rightarrow 1$, δ_0 is a distribution concentrated at zero, and $l = 1, \dots, p$. The advantage of (2) is that sparseness is imposed explicitly (many components of a_k are *exactly* zero).

An alternative to (2) is to employ a Student-t prior [6], implemented via the hierarchical construction

$$A_{lk} \sim \mathcal{N}(0, \alpha_{lk}^{-1}), a_{lk} \sim \text{Gamma}(e, f) \quad (3)$$

but now with (e, f) selected as to constitute a Student-t sharply peaked about zero. One may employ a similar construction to impose a double-exponential (Laplace) sparseness-promoting prior [4].

B. Beta process for inferring number of factors

The Beta Process (BP) was first developed by Hjort for survival data [28], and more recently it has found many other applications and extensions [19-21]. We here seek to provide a simple discussion of how this construction may be of interest in inferring an appropriate number of factors in factor modeling [22]. Our goal is to use the BP construction, which is closely related to the Indian buffet process (IBP) [19-21], to infer the number of factors r based on the observed data X .

Consider a measure drawn $H \sim \text{BP}(\alpha, \beta, H_0)$ and constructed as

$$H = \sum_{k=1}^K \pi_k \delta_{a_k}, \pi_k \sim \text{Beta}(\alpha / K, \beta(K-1) / K), \alpha_k \sim H_0 \quad (4)$$

We seek to link our construction explicitly to the factor model, and therefore a_k is the k th candidate factor loading (column of A), and H_0 is defined by the construction in (2) or (3), depending upon which model is used. The expression π_k represents the probability that a_k is used to represent any particular data sample, defined by the columns of X . The expression δ_{a_k} represents a unit point measure concentrated at a_k .

The BP is closely linked with a *Bernoulli Process* BeP (H) [21]. Specifically, for the j th column of \mathbf{X} , we perform a draw from the Bernoulli process

$$B_j = \sum_{k=1}^K z_{kj} \delta_{\mathbf{a}_k}, z_{kj} \in \{1, 0\}, z_{kj} \sim \text{Bernoulli}(\pi_k), j = 1, \dots, n \quad (5)$$

where the H in BeP(H) is drawn $H \sim \text{BP}(\alpha, \beta, H_0)$, as defined in (4). As discussed further below, if $z_{kj} = 1$ then \mathbf{a}_k is used as a factor loading to represent \mathbf{x}_j , the j th column of \mathbf{X} ; if $z_{kj} = 0$, \mathbf{a}_k is not used to represent \mathbf{x}_j . In other words, B_j is a sum of point measures ($\delta_{\mathbf{a}_k}$ is a unit point measure concentrated at \mathbf{a}_k), and the binary variables z_{kj} denote whether specific $\delta_{\mathbf{a}_k}$ are employed within B_j . More details on such constructions may be found in [21].

To make a connection to the introductory comments in Section II-A, and to relate the model to the IBP [19], we consider the above construction in the limit $K \rightarrow \infty$. Further, we marginalize (integrate) out the probabilities (π_1, \dots, π_K) used to constitute the BP draw H ; we retain the K candidate factor loadings $\{\mathbf{a}_k\}_{k=1, K}$ used to define \mathbf{A} , as drawn from the BP. Recall that \mathbf{x}_j represents the j th data sample (j th column of \mathbf{X}). We assume that the data samples (“customers”) select from among “dishes” at a “buffet”, with the dishes defined by $\{\mathbf{a}_k\}_{k=1, K}$. Data sample \mathbf{x}_1 enters the buffet first, and selects the first v_1 dishes $\mathbf{a}_1, \dots, \mathbf{a}_{v_1}$, where v_1 is a random variable drawn from $\text{Poisson}(\alpha/\beta)$. Therefore, the first column of \mathbf{S} has the first v_1 elements as non-zero, with the remaining elements in that column set to zero. The second “customer” \mathbf{x}_2 then enters the buffet, and selects from among the first v_1 dishes; the probability that \mathbf{x}_2 selects \mathbf{a}_k , for each of $k \in \{1, \dots, v_1\}$, is $1/(\beta + 1)$; i.e., $z_{k2} \sim \text{Bernoulli}(1/(\beta + 1))$, for $k \in \{1, \dots, v_1\}$. Customer \mathbf{x}_2 also selects v_2 new dishes $\{\mathbf{a}_{v_1+1}, \dots, \mathbf{a}_{v_1+v_2}\}$, with $v_2 \sim \text{Poisson}(\alpha/(\beta + 1))$. Hence, $z_{k2} = 1$ for $k \in \{v_1 + 1, \dots, v_1 + v_2\}$, and unless stated explicitly otherwise, all other components of \mathbf{z}_j are zero. This process continues sequentially, with each \mathbf{x}_j entering the buffet in ascending order of j . Sample \mathbf{x}_J , with $J \in \{1, \dots, n\}$ selects dishes as follows. Let $C_{J-1} = \sum_{j=1}^{J-1} v_j$ represent the cumulative number of dishes selected off the buffet, among the previous customers $\{\mathbf{x}_1, \dots, \mathbf{x}_{J-1}\}$. Further, let $m_{J-1,k} \geq 1$ represent the total number of times dish \mathbf{a}_k has been selected by previous customers $\{\mathbf{x}_1, \dots, \mathbf{x}_{J-1}\}$, for $k \in \{1, \dots, C_{J-1}\}$. Then \mathbf{x}_J selects dish \mathbf{a}_k , $k \in \{1, \dots, C_{J-1}\}$, with probability $m_{J-1,k}/(\beta + J - 1)$; i.e.,

$$z_{k,J} \sim \text{Bernoulli}\left(\frac{m_{J-1,k}}{\beta + J - 1}\right) \text{ for } k \in \{1, \dots, C_{J-1}\}.$$

Note that the more “popular” \mathbf{a}_k among the previous $J - 1$ customers (i.e., larger $m_{J-1,k}$), the more probable it is

that it will be selected by \mathbf{x}_J . Additionally, \mathbf{x}_J selects new dishes \mathbf{a}_k for $k \in \{C_{J-1} + 1, \dots, C_{J-1} + v_J\}$, where $v_J \sim \text{Poisson}\left(\frac{\alpha}{\beta + J - 1}\right)$. Therefore we have $z_{k,J} = 1$

for $k \in \{C_{J-1} + 1, \dots, C_{J-1} + v_J\}$. Thus, each new customer selects from among the dishes (factor loadings) already selected by at least one previous customer, and the more “popular” one of these dishes is, the more probable it is that the new customer will select it. Further, a new customer will also select additional dishes (factor loadings) not selected by any of the previous customers. However, note that as J increases, the draws $v_J \sim \text{Poisson}\left(\frac{\alpha}{\beta + J - 1}\right)$ are likely to be

decreasing in size, since $\frac{\alpha}{\beta + J - 1}$ is getting smaller with increasing J . Therefore, although $K \rightarrow \infty$, a finite subset of the candidate dishes (factor loadings) $\{\mathbf{a}_k\}_{k=1, K}$ will be used among the n customers, defined by the columns of \mathbf{X} , thereby imposing sparseness in the use of factor loadings. This model is also fully exchangeable, in that the order of the columns of \mathbf{X} may be permuted, with no change in the properties of the prior [19]. The model imposes that many of the n samples will share the same set of factors, but the model is flexible enough to allow idiosyncratic (sample-dependent) factor usage.

In practice K is finite, and therefore it is also of interest to consider the properties of this prior for finite K . For finite K , one may show that the number of non-zero components of \mathbf{z}_j is drawn from $\text{Binomial}(K, \alpha/(\alpha + \beta(K - 1)))$, and therefore one may set α and β to impose prior belief on the number of factors that will be important. The expected number of non-zero components in \mathbf{z}_j is $\alpha K/(\alpha + \beta(K - 1))$.

To complete the model specifications, note that \mathbf{a}_k from the Beta-Bernoulli construction above defines the k th column of the factor-loading matrix \mathbf{A} . The factor-score matrix \mathbf{S} utilizes the binary vectors $\mathbf{z}_j = (z_{1j}, \dots, z_{Kj})^T$ defined in (5), for $j \in \{1, \dots, n\}$. Specifically, we define the j th column of \mathbf{S} as $\mathbf{s}_j = \hat{\mathbf{s}}_j \circ \mathbf{z}_j$ (\circ represents a point-wise, or Hadamard product), with $\hat{\mathbf{s}}_j \sim \mathcal{N}(0, \mathbf{I}_K)$.

The vector product $\hat{\mathbf{s}}_j \circ \mathbf{z}_j$ selects a subset of the components in $\hat{\mathbf{s}}_j$, setting the rest to zero, and therefore the columns of \mathbf{S} are sparse.

C. Sparse factor modeling with BP/IBP placed on factor loadings

In the above discussion the Beta-Bernoulli and IBP processes were presented for a specific construction of the

factor-analysis model, with the goal of making the connection to the model explicit and hence clearer. However, there are alternative ways of utilizing the IBP for design of factor models. Specifically, rather than using the binary vectors to construct \mathbf{S} , as above, they may alternatively be used to define \mathbf{A} , with factor scores designed as in traditional factor models. This approach was considered in [8], using an Indian Buffet Process (IBP) construction (explicitly using the marginalization discussed above). A limitation of this approach is that one must perform p draws from the IBP to construct \mathbf{A} , and typically p is very large for the gene-expression problems of interest. When presenting results in Section II-B, we discuss our experience with this model on small-scale problems, although this approach was found computationally intractable for the motivating virus studies considered in Section II-D.

D. Constructing pseudo singular values

The final Bayesian construction considered for inferring r is closely related to the non-Bayesian sparse-PCA [12] and penalized matrix decomposition (PMD) [13] models. We generate the vectors $\{\mathbf{a}_k\}_{k=1,K}$ as before, using a sparseness-promoting prior like that discussed in Section IV-A. Further, the factor scores ξ_k for factor loading k is drawn $\xi_k \sim \mathcal{N}(0, \mathbf{I}_n)$, for $k=1, \dots, K$; ξ_k^T constitutes the k th row of \mathbf{S} , and we consider K such rows, for large K (relative to the anticipated r). Finally, the vector of pseudo singular values $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_K)$ is generated

$$\begin{aligned} \boldsymbol{\lambda} &= \mathbf{z} \circ \mathbf{w} \\ z_k &\sim \text{Bernoulli}(\pi_k), k=1, \dots, K \\ \pi_k &\sim \text{Beta}(\alpha / K, \beta K / (K - 1)), k=1, \dots, K \\ \mathbf{w} &\sim \mathcal{N}(0, \mathbf{I}_K) \end{aligned} \quad (6)$$

The matrix product \mathbf{AS} in (1) is now constituted as $\sum_{k=1}^K \lambda_k \mathbf{a}_k \xi_k^T$. The non-zero components of $\boldsymbol{\lambda}$ select the columns of \mathbf{A} used across all columns of \mathbf{X} . As discussed in Section IV, the number of non-zero components of $\boldsymbol{\lambda}$ is drawn Binomial($K, \alpha/(\alpha + \beta(K - 1))$), and the posterior on the number of such components provides desired information on the number of factors r . Note that this construction is *like* the Beta-Bernoulli process discussed above, in that it utilizes $\pi_k \sim \text{Beta}(\alpha/K, \beta K/(K - 1))$ and the Bernoulli distribution; however, it only draws the binary vector \mathbf{z} *once*, and therefore there is not the idea of multiple “customers”, as in the two IBP-related formulations discussed above.

E. Computational issues, model quality and hyper-parameter settings

The MCMC results presented here correspond to using 5000 collection samples, after a burn-in of 2000 iterations. However, with 2000 burn-in iterations and 500 collection samples, the average results of the factor scores and factor loadings are almost identical to those found with 5000. For all MCMC results, we employed a singular value decomposition (SVD) of the data matrix to initialize the factor loading and factor score matrix in the FA model, as well as the right-and left-singular matrix in the matrix decomposition model. For each iteration of the Gibbs sampler a particular number of factors r are employed, and based upon all collection samples one may infer an approximate posterior distribution for r . Running on a typical modern PC, the computation times are summarized in Table 1 for the different models, as applied to the influenza data (using 100 VB iterations).

To be explicit, we provide detailed hyper-parameter settings for the model in (7)-(14); the other models are set similarly. Specifically, $\alpha = 1$, $\beta = 1$, $c = 1$, and $d = g = h = e = f = 10^{-6}$. These parameters were not optimized, and were set in the same manner for all experiments. Although the PMD model is a non-Bayesian method, it also has parameter settings that must be addressed carefully; two hyper-parameters need adjusting: the sparseness threshold and the stop condition [13]. In all PMD experiments, we set the sparseness threshold as 4, and the PMD iterations were terminated when the reconstruction error was smaller than 5%.

All calculations were performed on PCs with Intel Pentium Dual E2200 processors and 2.00 GB memory, and all software was written in Matlab. For the large-scale analysis performed on the real data discussed above, MCMC required approximately 4 hours of CPU, while VB required 3 hours (per analysis).

Appendix: Gibbs and Variational Bayesian Analysis

We here provide a concise summary of the inference methods applied to one of the Bayesian FA models discussed above, with this representative of the analysis applied to the rest. Specifically, we consider the model

Table 1 Relative CPU times of the different models, implemented on a pc, as applied to the influenza data. the pmc method required a few minutes

	CPU Time VB (hours)	CPU Time MCMC (hours)
BPFA	0.5	4.87
Bayesian Pseudo-SVD	0.11	3.47
FA in [11]	0.11	4.87

discussed in Section IV-B, in which the BP is applied within the factor-score matrix. The complete model may be expressed as

$$\mathbf{x}_i \sim \mathcal{N}(A(z_i \circ s_i), \text{diag}(\psi_1^{-1}, \dots, \psi_p^{-1})) \quad (7)$$

$$z_{ki} \sim \text{Bernoulli}(\pi_k) \quad (8)$$

$$\pi_k \sim \text{Beta}(\alpha / K, \beta(K-1) / K) \quad (9)$$

$$A_{jk} \sim \mathcal{N}(0, \gamma_{jk}^{-1}) \quad (10)$$

$$s_i \sim \mathcal{N}(0, \delta^{-1}I_K) \quad (11)$$

$$\gamma_{jk} \sim \text{Gamma}(c, d) \quad (12)$$

$$\psi_j \sim \text{Gamma}(g, h) \quad (13)$$

$$\delta \sim \text{Gamma}(e, f) \quad (14)$$

where $i = 1, \dots, n, j = 1, \dots, p$ and $k = 1, \dots, K$.

Gibbs sampler

The full likelihood of the model is

$$\begin{aligned} p(\mathbf{X}, \mathbf{Z}, \mathbf{A}, \mathbf{S}, \psi) &= \prod_{i=1}^n \mathcal{N}(x_i; (A(z_i \circ s_i)), \text{diag}(\psi)^{-1}) \mathcal{N}(s_i; 0, \delta^{-1}I_K) \\ &\times \prod_{j=1}^p \prod_{k=1}^K \mathcal{N}(A_{jk}; 0, \gamma_{jk}^{-1}) \text{Gamma}(\gamma_{jk}; c, d) \\ &\times \prod_{i=1}^n \prod_{k=1}^K \text{Bernoulli}(z_{ki}; \pi_k) \text{Beta}(\pi_k; \frac{\alpha}{K}, \frac{\beta(K-1)}{K}) \\ &\times \prod_{j=1}^p \text{Gamma}(\psi_j; g, h) \times \text{Gamma}(\delta; e, f) \end{aligned}$$

The sequential update equations of the Gibbs sampler are as follows.

• Sample each entry of the binary matrix, z_{ki} . The probability of $z_{ki} = 1$ is expressed as

$$\begin{aligned} p(z_{ki} = 1 | \mathbf{X}, \mathbf{Z}_{-ki}, \mathbf{A}, \mathbf{S}, \psi) &\propto \ln(\pi_k) \\ &- \frac{1}{2} (A_k^T \text{diag}(\psi) A_k s_{ki}^2 - 2A_k^T \text{diag}(\psi) X_i^{-k} s_{ki}). \end{aligned}$$

• Sample π_k from $p(\pi_k | -) = \text{Beta}(\pi_k; \alpha', \beta')$ where $\alpha' = \sum_{i=1}^n z_{ki} + \frac{\alpha}{K}$ and $\beta' = n + \frac{\beta(K-1)}{K} - \sum_{i=1}^n z_{ki}$.

• Sample each entry of factor loading matrix, A_{jk} from $p(A_{jk} | -) = \mathcal{N}(A_{jk}; \mu_{jk}, \Sigma_{jk})$ where

$$\begin{aligned} \Sigma_{jk} &= \left[\sum_{i=1}^n \psi_j s_{ki}^2 z_{ki}^2 + \gamma_{jk} \right]^{-1}, \mu_{jk} = \Sigma_{jk} \left(\sum_{i=1}^n \psi_j z_{ki} s_{ki} X_{ji}^{-k} \right), \text{ and} \\ X_{ji}^{-k} &= x_{ji} - \sum_{l=1, l \neq k}^K A_{jl} z_{li} s_{li}. \end{aligned}$$

• Sample each column of factor score matrix, s_i , from $p(s_i | -) = \mathcal{N}(s_i; \xi_i, \Lambda_i)$ where $\Lambda_i = [(A^T \circ \tilde{Z}_i) \text{diag}(\psi) (A \circ \tilde{Z}_i^T) + \delta I_K]^{-1}$, $\xi_i = \Lambda_i (A \circ \tilde{Z}_i) \text{diag}(\psi) x_i$, and $\tilde{Z}_i = [z_i, \dots, z_i]$ with the K -dimensional vector, z_i , repeated p times, $1 \leq i \leq n$.

• Sample ψ_j from $p(\psi_j | -) = \text{Gamma}(\psi_j; g', h')$

where $g' = g + \frac{n}{2}$ and

$$h' = h + \frac{1}{2} \sum_{i=1}^n (\|x_{ji} - A_j(z_i \circ s_i)\|^2).$$

• Sample γ_{jk} from $p(\gamma_{jk} | -) = \text{Gamma}(\gamma_{jk}; c', d')$ where $c' = c + 1/2$ and $d' = d + \frac{1}{2} A_{jk}^2$.

• Sample δ from $p(\delta | -) = \text{Gamma}(\delta; e', f')$ where $e' = e + nK/2$ and $f' = f + \frac{1}{2} \sum_{i=1}^n (s_i^T s_i)$. In the above equations

expressions of the form $p(\gamma_{jk} | -)$ represent the probability of γ_{jk} conditioned on all other parameters.

Variational Bayesian inference

We seek a distribution $Q(\Theta; \Gamma)$ to approximate the exact posterior $p(\Theta | \mathbf{X})$, where in $\Theta \equiv \{\mathbf{A}, \mathbf{S}, \mathbf{Z}, \alpha, \pi, \psi, \gamma, \delta\}$. Our objective is to optimize the parameters Γ in the approximation $Q(\Theta; \Gamma)$. Toward that end, consider the variational expression

$$\tilde{F}(\Gamma) = \int d\Theta Q(\Theta; \Gamma) \ln \frac{Q(\Theta; \Gamma)}{p(\mathbf{X})p(\Theta | \mathbf{X})} = -\ln p(\mathbf{X}) + \text{KL}(Q(\Theta; \Gamma) || p(\Theta | \mathbf{X})) \quad (15)$$

Note that the term $p(\mathbf{X})$ is a constant with respect to Γ , and therefore $\tilde{F}(\Gamma)$ is maximized when the Kullback-Leibler divergence $\text{KL}(Q(\Theta; \Gamma) || p(\Theta | \mathbf{X}))$ is minimized. However, we cannot explicitly compute the KL divergence, since $p(\Theta | \mathbf{X})$ is unknown. However, the denominator term in $\tilde{F}(\Gamma)$ may be computed, since $p(\mathbf{X})p(\Theta | \mathbf{X}) = p(\mathbf{X} | \Theta)p(\Theta)$, and the prior $p(\Theta)$ and likelihood function $p(\mathbf{X} | \Theta)$ are available. To make computation of $\tilde{F}(\Gamma)$ tractable, we assume $Q(\Theta | \Gamma)$ has a factorized form $Q(\Theta; \Gamma) = \prod_i Q_i(\Theta_i; \Gamma_i)$. With appropriate choice of Q_i , the variational expression $\tilde{F}(\Gamma)$ may be evaluated analytically. The update equations are as follows.

• For z_{ki} we have $Q(z_{ki}) = \text{Bernoulli}(z_{ki}; \rho'_{ki})$ where ρ'_{ki} is the probability of $z_{ki} = 1$. We consider the following two conditions:

discussion below, the symbol $\langle \bullet \rangle$ represents the expectation of the argument.

$$\ln Q(z_{ki} = 1) \propto \zeta_1 = \langle \ln \pi_k \rangle - \left[\frac{1}{2} (-2 \langle X_i^{-k} \rangle^T \text{diag}(\langle \psi \rangle \langle A_k \rangle \langle w_{ki} \rangle + \langle w_{ki}^2 \rangle \langle A_k^T \text{diag}(\psi) A_k)) \right]$$

$$\ln Q(z_{ki} = 0) \propto \zeta_2 = \langle \ln(1 - \pi_k) \rangle \quad \text{where}$$

$$\langle \ln(1 - \pi_k) \rangle = \Psi \left(\frac{\beta(K-1)}{K} + n - \sum_{i=1}^n \langle z_{ki} \rangle \right) - \Psi \left(\frac{\alpha + \beta(K-1)}{K} + n \right), \Psi(x) = \frac{\partial}{\partial x} \ln \Gamma(x),$$

$$\langle \ln(1 - \pi_k) \rangle = \Psi \left(\frac{\beta(K-1)}{K} + n - \sum_{i=1}^n \langle z_{ki} \rangle \right) - \Psi \left(\frac{\alpha + \beta(K-1)}{K} + n \right), \Psi(x) = \frac{\partial}{\partial x} \ln \Gamma(x)$$

and $\Gamma^1(x) = \int_0^\infty d\tau \tau^{x-1} e^{-\tau}$. Therefore, we can calculate

$\rho'_{ki} = \frac{\exp(\zeta_1)}{\exp(\zeta_1) + \exp(\zeta_2)}$. Above, and in the discussion below, the symbol $\langle \bullet \rangle$ represents the expectation of the argument.

• For π_k we have $Q(\pi_k) = \text{Beta}(\pi_k; \alpha'_k, \beta'_k)$ where

$$\alpha'_k = \sum_{i=1}^n \langle z_{ki} \rangle + \frac{\alpha}{K} \quad \text{and}$$

$$\beta'_k = n + \frac{\beta(K-1)}{K} - \sum_{i=1}^n \langle z_{ki} \rangle.$$

• For A_{jk} we have $Q(A_{jk}) = \mathcal{N}(A_{jk}; \mu_{jk}, \Sigma_{jk})$ with

$$\Sigma_{jk} = \left[\sum_{i=1}^n \langle \psi_j \rangle \langle s_{ki}^2 \rangle \langle z_{ki} \rangle + \langle \gamma_{jk} \rangle \right]^{-1} \quad \text{and}$$

$$\mu_{jk} = \Sigma_{jk} \left(\sum_{i=1}^n \langle \psi_j z_{ki} s_{ki} X_{ji}^{-k} \rangle \right)$$

$$\text{, where } X_{ji}^{-k} = x_{ji} - \sum_{l=1; l \neq k}^K A_{jl} z_{li} s_{li}.$$

• For s_i we have $Q(s_i) = \mathcal{N}(s_i; \xi_i, \Lambda_i)$, with

$$\Lambda_i = [(\mathbf{A}^T \circ \tilde{\mathbf{Z}}_i) \text{diag}(\langle \psi \rangle) (\mathbf{A} \circ \tilde{\mathbf{Z}}_i^T)]^{-1} \quad \text{and}$$

$$\xi_i = \Lambda_i ((\mathbf{A}^T \circ \tilde{\mathbf{Z}}_i) \text{diag}(\langle \psi \rangle) x_i) \quad \text{, where}$$

$\tilde{\mathbf{Z}}_i = [z_i, \dots, z_i]$ is a K -dimensional vector of all z_i repeated p times. In order to exactly calculate the expectation,

$$\mathbf{B} = \langle (\mathbf{A}^T \circ \tilde{\mathbf{Z}}_i) \text{diag}(\psi) (\mathbf{A} \circ \tilde{\mathbf{Z}}_i^T) \rangle$$

, we have to consider it as two parts. Specifically, the off-diagonal elements of \mathbf{B} are $\langle (\mathbf{A}^T \circ \tilde{\mathbf{Z}}_i) \text{diag}(\langle \psi \rangle) (\mathbf{A} \circ \tilde{\mathbf{Z}}_i^T) \rangle$, and the diagonal

elements, $B_{kk} = \left(\sum_{j=1}^p \langle (A_{jk})^2 \rangle + \Sigma_{jk} \langle \psi_j \rangle \right) \langle z_{ki} \rangle$, since

$\langle (A_{jk}^2) \rangle = \langle A_{jk} \rangle^2 + \Sigma_{jk}$ and $\langle (A_{jk}^2) \rangle = \langle A_{jk} \rangle^2 + \Sigma_{jk}$, where $1 \leq k \leq K, 1 \leq j \leq p$ and $1 \leq i \leq n$.

• For ψ_j we have $Q(\psi_j) = \text{Gamma}(\psi_j; g'_j, h'_j)$,

$$\text{where } g'_j = g + \frac{n}{2}, h'_j = h + \frac{1}{2} \sum_{i=1}^n \langle (x_{ji} - A_j(z_i \circ s_i))^2 \rangle.$$

• For γ_{jk} we have $Q(\gamma_{jk}) = \text{Gamma}(c'_{jk}, d'_{jk})$, with

$$c'_{jk} = c + 1/2 \quad \text{and} \quad d'_{jk} = d + \frac{1}{2} \langle (A_{jk}^2) \rangle.$$

• For δ we have $Q(\delta) = \text{Gamma}(e', f')$, where $e' = e + Kn/2$ and $f' = f + \frac{1}{2} \sum_{i=1}^n \langle s_i^T s_i \rangle$.

Acknowledgements

The research reported here was supported under the DARPA PHD program. The research and results presented here are the contributions of the authors, and the results were not influenced in any way by the sponsor.

Author details

¹Electrical and Computer Engineering Department, Duke University, Durham, NC, USA. ²Institute for Genome Sciences & Policy, Department of Medicine Duke University, Durham, NC, USA. ³Electrical & Computer Engineering Department, University of Michigan, Ann Arbor, MI, USA. ⁴Statistics Department, Duke University, Durham, NC, USA.

Authors' contributions

The following authors performed the statistical analysis: BC, MC, JP, AH, JL, DD and LC. The following authors executed the three viral challenge studies, and performed all biological interpretation of the results: AZ, CW and GSG. All authors read and contributed to writing this paper.

Received: 1 September 2009 Accepted: 9 November 2010
 Published: 9 November 2010

References

- West M: "Bayesian factor regression models in the "large p, small n" paradigm". In *Bayesian Statistics 7*. Edited by: Bernardo JM, Bayarri M, Berger J, Dawid A, Heckerman D, Smith A, West M. Oxford University Press; 2003:723-732.
- Tibshirani R: "Regression shrinkage and selection via the lasso". *Journal of Royal Statistical Society Ser. B* 1996, **58**:267-288.
- Zou H, Hastie T: "Regularization and variable selection via the elastic net". *Journal of Royal Statistical Society Ser. B* 2005, **67**:301-320.
- Park T, Casella G: "The Bayesian Lasso". *Journal of the American Statistical Association* 2008, **103**:681-686.
- Cristianini N, Shawe-Taylor J: *An Introduction to Support Vector Machines*. Cambridge University Press; 2000.
- Tipping M: "Sparse Bayesian learning and the relevance vector machine". *Journal of Machine Learning Research* 2001, **1**:211-244.
- Ji S, Xue Y, Carin L: "Bayesian compressive sensing". *IEEE Transactions on Signal Processing* 2008, **56**.
- Rai P, Daum'e H III: "The infinite hierarchical factor regression model". *Proc Conf Neural Information Proc Systems (NIPS), Vancouver, Canada* 2008.
- Knowles D, Ghahramani Z: "Infinite sparse factor analysis and infinite independent components analysis". *7th International Conference on Independent Component Analysis and Signal Separation* 2007.
- Meeds E, Ghahramani Z, Neal R, Roweis S: "Modeling dyadic data with binary latent factors". *Advances in Neural Information Processing Systems* 2007, **977-984**.

11. Carvalho C, Chang J, Lucas J, Nevins JR, Wang Q, West M: "High-dimensional sparse factor modelling: Applications in gene expression genomics,". *Journal of the American Statistical Association* 2008, **103**:1438-1456.
12. Zou H, Hastie T, Tibshirani R: "Sparse principal component analysis,". *Journal of Computational and Graphical Statistics* 2006, **15**:2004.
13. Witten D, Tibshirani R, Hastie T: "A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis,". *Biostatistics* 2009, **10**:515-534.
14. Lopes H, West M: "Bayesian model assessment in factor analysis,". *Statistica Sinica* 2004, **14**:41-67.
15. Ghosh J, Dunson D: "Bayesian model selection in factor analytic models,". In *Random Effect and Latent Variable Model Selection*. Edited by: Dunson D. John Wiley 2008.
16. Berger J, Ghosh J, Mukhopadhyay N: "Approximation and consistency of Bayes factors as model dimension grows,". *J. Statist. Plann. Inference* 2003, **112**:241258.
17. Press S, Shigemasu K: "A note on choosing the number of factors,". *Comm Statist Theory Methods* 1999, **28**:1653-1670.
18. Lee S, Song X: "Bayesian selection on the number of factors in a factor analysis model,". *Behaviormetrika* 2002, **29**:2339.
19. Griffiths T, Ghahramani Z: "Infinite latent feature models and the indian buffet process,". *Advances in Neural Information Processing Systems* 2005, 475-482.
20. Doshi-Velez F, Miller K, Gael JV, The Y: "Variational inference for the indian buffet process,". *AISTATS* 2009.
21. Thibaux R, Jordan M: "Hierarchical beta processes and the Indian buffet process,". *International Conference on Artificial Intelligence and Statistics* 2007.
22. Paisley J, Carin L: "Nonparametric factor analysis with beta process priors,". *Int Conf Machine Learning* 2009.
23. Beal M: "Variational algorithms for approximate bayesian inference,". Ph. D. dissertation, Gatsby Computational Neuroscience Unit, University College London; 2003.
24. Zou H, Hastie T, Tibshirani R: "Sparse principal component analysis,". *Technical Report, Statistics Department, Stanford University* 2004.
25. Zaas AK, Chen M, Lucas J, Veldman T, Hero AO, Varkey J, Turner R, Oien C, Kingsmore S, Carin L, Woods CW, Ginsburg GS: "Peripheral blood gene expression signatures characterize symptomatic respiratory viral infection,". *Cell Host & Microbe* 2009, **6**:207-217.
26. Dunson D: "Dynamic latent trait models for multidimensional longitudinal data,". *J. Am. Statistical Ass* 2003, **98**:555-563.
27. Ramilo O, Allman W, Chung W, Mejias A, Ardura M, Glaser C, Wittkowski KM, Piqueras B, Banchereau J, Palucka AK, Chaussabel D: "Gene expression patterns in blood leukocytes discriminate patients with acute infections,". *Blood* 2007, **109**:2066-2077.
28. Hjort NL: "Nonparametric bayes estimators based on beta processes in models for life history data,". *Annals of Statistics* 1990, **18**(3):1259-1294.

doi:10.1186/1471-2105-11-552

Cite this article as: Chen et al.: Bayesian inference of the number of factors in gene-expression analysis: application to human virus challenge studies. *BMC Bioinformatics* 2010 **11**:552.

Submit your next manuscript to BioMed Central
and take full advantage of:

- Convenient online submission
- Thorough peer review
- No space constraints or color figure charges
- Immediate publication on acceptance
- Inclusion in PubMed, CAS, Scopus and Google Scholar
- Research which is freely available for redistribution

Submit your manuscript at
www.biomedcentral.com/submit

