JMLR: Workshop and Conference Proceedings 29:1–16, 2014 ACML 2014

Pòlya-Gamma augmentations for factor models

Arto Klami ARTO.KLAMI@CS.HELSINKI.FI
Helsinki Institute for Information Technology HIIT, Department of Computer Science
University of Helsinki

Abstract

Bayesian inference for latent factor models, such as principal component and canonical correlation analysis, is easy for Gaussian likelihoods with conjugate priors using both Gibbs sampling and mean-field variational approximation. For other likelihood potentials one needs to either resort to more complex sampling schemes or to specifying dedicated forms for variational lower bounds. Recently, however, it was shown that for specific likelihoods related to the logistic function it is possible to augment the joint density with auxiliary variables following a Pòlya-Gamma distribution, leading to closed-form updates for binary and over-dispersed count models. In this paper we describe how Gibbs sampling and mean-field variational approximation for various latent factor models can be implemented for these cases, presenting easy-to-implement and efficient inference schemas.

Keywords: Binary data, Count data, Latent factor models, Matrix factorization

1. Introduction

Bayesian formulation of latent factor models including principal component analysis (PCA) (Ilin and Raiko, 2010), factor analysis (FA) (Rowe, 2002), canonical correlation analysis (CCA) (Klami et al., 2013), and their various further generalizations is easy for Gaussian likelihoods. All of these models can be formulated as probabilistic low-rank matrix factorizations (Mnih and Salakhutdinov, 2007) with suitable priors for the factors and residual noise, and posterior inference consisting solely of closed-formed updates is efficient for fully observed data. For setups with missing data it is typically more efficient to resort for gradient-based optimization for the factors (Ilin and Raiko, 2010).

While the equivalent models for other likelihoods as also easy to specify, posterior inference becomes notably more challenging for non-Gaussian models. The existing approaches typically fall into one of two categories: Strategies for general exponential families, and strategies designed for specific likelihoods. The prime examples of the first category are exponential family PCA (Mohamed et al., 2009) and exponential family CCA (Klami et al., 2010) that use Hybrid Monte Carlo samplers for inference. These samplers are computationally heavy, as illustrated by the small-scale experiments of at most hundreds of data points presented by the authors. Even maximum a posterior estimation for such models requires approximations and general-purpose gradient-based solvers (Li and Tao, 2010).

The more practical tools for factor analysis of non-Gaussian data are based on dedicated inference schema for specific likelihoods, often via variational approximations that explicitly bound the non-conjugate parts of the model. Jaakkola (1997) presented a bound for logistic regression, Girolami and Rogers (2006) for probit transformations, and Bohning (1992) for

multinomial data. Even though these bounds were originally developed for regression, they generalize directly for factor models (Seeger and Bouchard, 2012; Khan et al., 2010) by simply treating the covariates as random variables.

The above techniques explicitly construct a variational lower bound for the likelihood and then estimate its parameters, whereas in this paper we consider posterior inference via model augmentation. Recently Polson et al. (2013) introduced an augmentation scheme for likelihood potentials of certain form, covering both logistic transformation for binary data and negative binomial likelihood for over-dispersed count data. They managed to make the conditional distribution of the regression weights Gaussian while retaining closed-form distribution for the augmentation variable itself, providing Gibbs inference for the whole model. The same augmentation leads also to closed-form mean-field variational updates. This observation has already been used for developing Bayesian logistic regression (Polson et al., 2013) and negative binomial regression models (Zhou et al., 2012).

This background naturally leads to latent factor models via similar constructs, as already hinted by Polson et al. (2013). Even though extending regression models to factor models is conceptually easy, the practical details have large effect on the efficiency and accuracy of the solution, and for truly working solution one needs to re-use recent advances in Gaussian factor models. In this paper we lay out the alternatives and demonstrate empirically the large differences in computational time and accuracy; the naive solutions are shown to be either an order of magnitude slower or inaccurate. We start by re-capping the inference for Gaussian models (much of the derivations can be re-used for the other likelihoods) and the necessary background on Pòlya-Gamma augmentations, before deriving the proposed models and illustrating them on artificial data.

2. Background: Gaussian latent factor models

The basic construct considered in this manuscipt is of the form

$$x_{ij} \sim \mathcal{N}(\sum_{k=1}^{K} u_{ik} v_{jk}, \tau^{-1}), \quad u_{ik} \sim \mathcal{N}(0, \beta_k^{-1}), \quad v_{jk} \sim \mathcal{N}(0, \alpha_k^{-1}), \quad \beta_k, \alpha_k, \tau \sim \mathcal{G}(a^0, b^0).$$

which corresponds to a latent factor model for $\mathbf{X} \in \mathbb{R}^{N \times D}$ with K factors $\mathbf{U} \in \mathbb{R}^{N \times K}$ and their loadings $\mathbf{V} \in \mathbb{R}^{N \times D}$, alternatively written in the matrix form as $\mathbf{X} = \mathbf{U}\mathbf{V}^T + \epsilon$, where ϵ is noise with precision τ. Throughout the paper i runs over the N samples and j over the D features. All of the factors are given normal priors with gamma priors on the precisions (assuming equal hyperparameters for notational simplicity). To simplify the notation we do not include explicit bias terms for $\mathbf{U}$ and $\mathbf{V}$ in the model; see Ilin and Raiko (2010) and Klami et al. (2014) for inference for different alternative assumptions for the bias terms, applicable also for the non-Gaussian likelihoods discussed later.

A wide range of standard models are subsumed by this formulation. By setting $\beta_k = 1$ we get principal component analysis (Tipping and Bishop, 1999; Ilin and Raiko, 2010), by setting $\beta_k = 1$ and letting τ depend on the dimension as τ_j we get factor analysis (Rowe, 2002). More complex models can be implemented with the exact same underlying construct by introducing more matrices that are suitably tied to each other. Canonical correlation analysis (CCA) (Klami et al., 2013) is obtained when $x_{ij} = \sum_k u_{ik} v_{jk}^{(x)}$ and $y_{ij} =$

$\sum_k u_{ik} v_{jk}^{(y)}$, where $\mathbf{U}$ with $\beta_k = 1$ is shared between the two input matrices while the $\mathbf{V}$ with arbitrary $\alpha_k^{(x)}$ and $\alpha_k^{(y)}$ are not. More general constructs that can be implemented with the same machinery include group factor analysis (Virtanen et al., 2012) and collective matrix factorization (Singh and Gordon, 2010). Even though these models consider simultaneous factorization of arbitrarily many matrices and they require specific forms of factor priors, they can be implemented using almost exactly the same formulas.

For all models mentioned above, efficient Bayesian inference is possible via both Gibbs sampling and variational approximation, using the same set of basic updates. The main goal of this paper is to demonstrate that all of these factorization models can be easily modified to work for binary and over-dispersed count data by suitable augmentation. In the following we present the derrivations for the most straightforward case of PCA; the other models require merely changes in book-keeping and priors. The necessary general notation is presented in Klami et al. (2014), with a brief recap in Section 4.4 of this paper.

2.1. Gibbs sampling

The full likelihood of the PCA model is given by

$$\mathcal{G}(\tau|a^0, b^0) \prod_{i,j} \left[\mathcal{N}(x_{ij}| \sum_k u_{ik} v_{jk}, \tau^{-1}) \right] \prod_k \left[\mathcal{G}(\alpha_k|a^0, b^0) \prod_i \mathcal{N}(u_{ik}|0,1) \prod_j \mathcal{N}(v_{jk}|0, \alpha_k^{-1}) \right]. \tag{1}$$

Straightforward Gibbs sampler is obtained by deriving the following conditionals:

$$u_i|- \sim \mathcal{N}(\mu_i, \Sigma_i), \qquad v_j|- \sim \mathcal{N}(m_j, S_j),$$
$$\alpha_k|- \sim \mathcal{G}(a_{\alpha_k}, b_{\alpha_k}), \qquad \tau|- \sim \mathcal{G}(a_\tau, b_\tau),$$

where $\cdot|-$ refers to conditioning on all other parameters, and u_i denotes the ith row of $\mathbf{U}$. The terms corresponding to the factor updates are given by

$$\Sigma_i = (I + \tau \sum_j v_j^T v_j)^{-1}, \qquad \mu_i = \tau \sum_j v_j x_{ij} \Sigma_i,$$
$$S_j = (\alpha + \tau \sum_i u_i^T u_i)^{-1}, \qquad m_j = \tau \sum_i u_i x_{ij} S_j.$$

These could also be written in matrix form, but these explicit summations are directly applicable also for missing data so that the sums go only over the observed entries. For the precision parameters the conditionals are defined by

$$a_\tau = a^0 + ND/2, \qquad b_\tau = b^0 + \sum_{i,j} (x_{ij} - \sum_k u_{ik} v_{jk})^2/2,$$
$$a_{\alpha_k} = a^0 + D/2, \qquad b_{\alpha_k} = b^0 + \frac{1}{2} \sum_j v_{jk}^2.$$

In practice the automatic relevance determination prior turns unnecessary factors off more effectively if we marginalize over v_{jk} in the last step, resulting in $b_{\alpha_k} = b^0 + \frac{1}{2} \sum_j (m_{jk}^2 + S_j[k,k])$, where $S_j[k,k]$ is the kth element on the diagonal of S_j.

A notable observation is that for fully observed data Σ_i and S_j do not depend on the data point or feature, and hence only need to be computed and inverted once per iteration. For partially observed data they need to be re-computed for each sample, since the summations in Σ_i and S_j are only over the observed entries for each row/column.

2.2. Variational approximation

Variational approximation (Jaakkola, 1997) approximates the posterior distribution $p(\theta|\mathbf{X})$ with a factorized distribution $q(\theta) = \prod_l q(\theta_l)$, so that the variational lower bound $\mathcal{L} = p(\mathbf{X}) - \mathrm{KL}(q|p) = \mathrm{const} + \langle \log p(x,\theta) - \log q(\theta) \rangle$ is maximized. Here $\langle \cdot \rangle$ denotes expectation over $q(\theta)$ and $\mathrm{KL}(q|p)$ is the Kullback-Leibler divergence. The mean-field updates for individual terms are obtained as $q(\theta_l) = e^{\langle \log p(x,\theta) \rangle}$, where the expectation is over all other terms.

Ilin and Raiko (2010) presented two alternative variational approximations for model (1) that have later been used also for other factorization models. The *naive approximation*

$$q(\theta) = q(\tau|\hat{a}_\tau, \hat{b}_\tau) \prod_k q(\alpha_k|\hat{a}_{\alpha_k}, \hat{b}_{\alpha_k}) \prod_i q(u_i|\hat{\mu}_i, \hat{\Sigma}_i) \prod_j q(v_j|\hat{m}_j, \hat{S}_j)$$

assumes the row and column latent variables to be independent, but models each of them as a K-dimensional multivariate distribution. Since the model is fully conjugate, the functional forms match the priors and the updates are

$$\begin{aligned}
\hat{S}_j &= (\langle \alpha \rangle + \langle \tau \rangle \sum_i \langle u_i^T u_i \rangle)^{-1} = (\langle \alpha \rangle + \langle \tau \rangle \sum_i (\hat{\mu}_i^T \hat{\mu}_i + \hat{\Sigma}_i))^{-1}, \\
\hat{\Sigma}_i &= (I + \langle \tau \rangle \sum_j \langle v_j^T v_j \rangle)^{-1} = (I + \langle \tau \rangle \sum_j (\hat{m}_j^T \hat{m}_j + \hat{S}_j))^{-1}, \\
\hat{m}_j &= \langle \tau \rangle \sum_i x_{ij} \langle u_i \rangle \hat{S}_v = \langle \tau \rangle \sum_i x_{ij} \hat{\mu}_i \hat{S}_v, \quad \hat{\mu}_i = \langle \tau \rangle \sum_j x_{ij} \langle v_j \rangle \hat{\Sigma}_i = \langle \tau \rangle \sum_j x_{ij} \hat{m}_j \hat{\Sigma}_i, \\
\hat{a}_\tau &= a^0 + ND/2, \quad \hat{a}_{\alpha_k} = a^0 + D/2, \\
\hat{b}_\tau &= b^0 + \sum_{i,j} \langle (x_{ij} - u_i v_j^T)^2 \rangle / 2 \\
&= b^0 + \sum_{i,j} \left((x_{ij} - \hat{\mu}_i \hat{m}_j^T)^2 + \hat{\mu}_i \hat{S}_v \hat{\mu}_i^T + \hat{m}_j \hat{\Sigma}_u \hat{m}_j^T + \mathrm{Tr}[\hat{\Sigma}_i \hat{S}_j] \right), \text{ and} \\
\hat{b}_{\alpha_k} &= b^0 + \langle (\sum_j v_{jk}^2)/2 \rangle = b^0 + \sum_j (\hat{m}_{jk}^2 + \hat{S}_j[k,k]))/2,
\end{aligned}$$

where we have already written out all of the expectations using the variational parameters, except for the gamma terms for which $\langle \tau \rangle = a_\tau / b_\tau$ and similarly for α_k. Again computation is efficient for fully observed data since $\hat{\Sigma}_i$ and $\hat{S}_j$ do not depend on the data point or feature. For partially observed data they need to be re-computed for each case.

To improve the efficiency for the partially observed case, Ilin and Raiko (2010) proposed a *fully factorized approximation*

$$q(\theta) = q(\tau|\hat{a}_\tau, \hat{b}_\tau) \prod_k \left[q(\alpha_k|\hat{a}_{\alpha_k}, \hat{b}_{\alpha_k}) \prod_i q(u_{ik}|\hat{\mu}_{ik}, \hat{\sigma}_{ik}^2) \prod_j q(v_{jk}|\hat{m}_{jk}, \hat{s}_{jk}^2) \right].$$

Even though introducing more independencies is generally unadviced, here it does not notably influence the accuracy of the approximation since the goal is to learn latent factors that eventually are independent of each other. Again all of the parameters can be updated in closed form, with the only changes being for the way the expectations are computed. However, updating $\hat{\mu}_{ik}$ and $\hat{m}_{jk}$ one element at a time would be extremely inefficient due to heavy correlations between the elements. Instead, one should perform gradient-based optimization; the gradient of the variational lower bound with respect to $\hat{m}_{jk}$ is

$$\frac{\delta \mathcal{L}}{\delta \hat{m}_{jk}} = -\langle \alpha_k \rangle \hat{m}_{jk} - \langle \tau \rangle \sum_i \left[-(x_{ij} - \hat{\mu}_i \hat{m}_j^T)\hat{\mu}_{ik} + \hat{\sigma}_{ik}^2 \hat{m}_{jk} \right],$$

the diagonal Hessian equals $\hat{s}_{jk}^{-2}$, and analogous equations are obtained for $\hat{\mu}$. Efficient Newton-Rhapson -style optimization is hence possible; for details see Ilin and Raiko (2010).

Compared to the first approximation this solution has the advantage that no KxK matrices need to be inverted, with the disadvantage that gradient-based updates are used instead of closed-form updates. The gradients, however, allow updating the whole $\mathbf{U}$ and $\mathbf{V}$ at once instead of doing it for each row at a time. In practice the two approximations often have comparable computational cost per iteration (except for very large K that makes the naive one slow) and require roughly as many iterations for convergence. For setups setups with missing data the factorized one is clearly faster; then the naive one needs to invert ND covariance matrices for each iteration, whereas the factorized one retains its speed.

3. Background: Pòlya-Gamma augmentation

To proceed towards efficient inference for binary and over-dispersed count data, we next introduce the data augmentation scheme for logistic transformations.

Polson et al. (2013) proved the following equality

$$\frac{(e^{\psi})^a}{(1+e^{\psi})^b} = 2^{-b} e^{\kappa \psi} \int_0^{\infty} e^{-\omega \psi^2/2} p(\omega) d\omega, \tag{2}$$

where $\kappa = a - b/2$ and $p(\omega) = PG(\omega|b, 0)$ is the Pòlya-Gamma distribution

$$PG(\omega|b,c) = \cosh^b(c/2) \frac{2^{b-1}}{\Gamma(b)} \sum_{n=0}^{\infty} (-1)^n \frac{\Gamma(n+b)\Gamma(2n+b)}{\Gamma(n+1)\sqrt{2\pi\omega^3}} e^{-\frac{(2n+b)^2}{8\omega}} e^{-\frac{c^2}{2}\omega},$$

and the $\cosh^b(c/2)$ and $e^{-\frac{c^2}{2}\omega}$ terms simplify out for the special case PG(b,0) required for the identity. Even though the density function is complicated, the moments of $PG(\omega|b,c)$ can be computed in closed form; we will need in particular the equation $\langle \omega \rangle = \frac{b}{2c}\tanh(c/2)$. In addition, Polson et al. (2013) provides an efficient algorithm for sampling from PG(b,0), whereas Zhou et al. (2012) showed that samples from the general case PG(b,c) can be drawn by truncating and bias-correcting an infinite sum of suitably weighted gamma random variables, with good accuracy obtained already with very low truncation levels.

The practical significance of the construct in (2) is seen by noting that both Bernoulli and negative binomial likelihoods of logistic parameters can be written in that form. If we

denote $p = \text{logistic}(\psi) = (1 + e^{-\psi})^{-1}$ then the Bernoulli likelihood is given by

$$p(x|p) = p^x(1-p)^{1-x} = \frac{(e^\psi)^x}{1+e^\psi},$$

which matches (2) with $a = x$, $b = 1$, and $\kappa = x - 1/2$. Similarly, the negative binomial likelihood $\text{NB}(x|r,p)$ using the same logistic transformation for the p parameter is

$$p(x|r,p) = \frac{\Gamma(r+x)}{x!\Gamma(r)} p^x(1-p)^r \propto \frac{(e^\psi)^x}{(1+e^\psi)^{x+r}},$$

corresponding to $a = x$, $b = x + r$, and $\kappa = (x - r)/2$.

In practice we use (2) to implement factor models for non-Gaussian data by explicitly representing ω as a random variable. Then both of the above likelihoods can be written as

$$p(x, \omega|, -) = p(x|\omega, -)PG(\omega|b, 0) \propto 2^{-b} e^{\kappa\psi - \omega\psi^2/2} PG(\omega|b, 0).$$

We see that conditional on ω the likelihood depends quadratically on ψ and is hence Gaussian. This suggests efficient inference techniques for models where ψ has normal priors, assuming inference for ω conditional on the data and ψ is easy. As shown by Polson et al. (2013), the conditional distribution $p(\omega|\psi, x)$ is obtained by exponential tilting of the prior $PG(\omega|b, 0)$ and equals $PG(\omega|b, \psi)$. In other words, it is known in closed form and we can easily compute expectations (and other moments) of it. Using these observations Polson et al. (2013) derived a Gibbs sampler for logistic regression, and Zhou et al. (2012) provided both Gibbs sampler and variational approximation for negative binomial regression.

4. Latent factor models with polya-gamma augmentation

Next we will derive efficient posterior inference schemes for arbitrary matrix factorization models with normal priors on the factors and either Bernoulli or negative binomial likelihood on data. The detailed derivations are shown for the special case of PCA model with Bernoulli likelihood, but in Sections 4.3 and 4.4 we will show the necessary modifications for negative binomial likelihood and other factor models.

We define the PG augmented factor model for binary data as

$$p_{ij} = \text{logistic}(\psi_{ij}), \quad x_{ij} \sim \text{Bernoulli}(p_{ij}),$$

where ψ is either low-rank or low-rank with additive Gaussian noise:

$$\psi_{ij} = \sum_k u_{ik} v_{jk} \quad \text{or} \quad \psi_{ij} = \sum_k u_{ik} v_{jk} + \epsilon_{ij}.$$

In the former ψ is not a random variable, but merely a convenience notation. The priors for $\mathbf{U}$ and $\mathbf{V}$ are as in (1), and $\epsilon_{ij} \sim N(0, \tau^{-1})$.

We call the first alternative *direct approach* and the latter *explicit noise approach*. The direct approach has the advantage of tying the data directly with the low rank parameters, whereas the latter results in more efficient updates for fully observed data but (as shown later) slower convergence due to the intermediate random variable ψ. Next we will present the two alternatives in detail, providing both Gibbs and mean-field variational approximations for both.

4.1. Model 1: Direct approach

The full likelihood of the direct approach in the augmented form is

$$\prod_i \prod_j \left[e^{\kappa_{ij}\psi_{ij} - \omega_{ij}\psi_{ij}^2/2} PG(\omega_{ij}|b,0) \right] \prod_k \left[\mathcal{G}(\alpha_k|a^0,b^0) \prod_i \mathcal{N}(u_{ik}|0,1) \prod_j \mathcal{N}(v_{jk}|0,\alpha_k^{-1}) \right].$$

4.1.1. Gibbs sampler

For the factors we re-use the conditionals Polson et al. (2013) provided for logistic regression

$$S_j = (\alpha + \sum_i \omega_{ij} u_i^T u_i)^{-1}, \quad m_j = \sum_i u_i \kappa_{ij} S_j,$$

with analogous equations for μ_i and Σ_i, whereas the conditionals for α_k equal the Gaussian case. The remaining update for ω_{ij} is simply element-wise sampling from $PG(1,\psi_{ij}) = PG(1,\sum_k u_{ik}v_{jk})$, based on the exponential tilting argument in Section 3.

These updates can be contrasted with the Gaussian case in (1), to see a very close relationship. The data is replaced by $\kappa_{ij} = x_{ij} - 1/2$ which is merely a constant shift, and ω_{ij} plays a role of element-wise noise precision when computing the covariance. An important observation, however, is that ω_{ij} is not used for scaling the mean in the same way as τ acts as a multiplier for m_j in the Gaussian case. Finally, we see that Σ_i and S_j depend on i and j due to ω_{ij} and hence this sampler is not efficient for large data.

4.1.2. Variational approximation

The naive variational approximation can be written as

$$q(\theta) = \prod_k q(\alpha_k|\hat{a}_{\alpha_k},\hat{b}_{\alpha_k}) \prod_i q(u_i|\hat{\mu}_i,\hat{\Sigma}_i) \prod_j q(v_j|\hat{m}_j,\hat{S}_j) \prod_{i,j} q(\omega_{ij}|\hat{\gamma}_{ij},\hat{\eta}_{ij}), \tag{3}$$

where the single $q(\tau)$ factor of the Gaussian case is replaced with element-wise factors $q(\omega_{ij})$.

The updates for $q(\alpha_k)$ are identical to the Gaussian case, and the updates for the factors are very closely related as

$$\hat{S}_j = (\langle\alpha\rangle + \sum_i \langle\omega_{ij}\rangle\langle u_i u_i^T\rangle)^{-1} = (\langle\alpha\rangle + \sum_i \langle\omega_{ij}\rangle(\hat{\mu}_i\hat{\mu}_i^T + \hat{\Sigma}_i))^{-1},$$
$$\hat{m}_j = \sum_i \langle u_i\rangle\kappa_{ij}\hat{S}_j = \sum_i \hat{\mu}_i\kappa_{ij}\hat{S}_j,$$

with analogous updates for $q(u_i)$. The remaining update for $q(\omega_{ij})$ can also be derived in closed form. According to standard mean-field procedure we get

$$\log q(\omega_{ij}) = \langle \log p(x_{ij},\omega_{ij},-)\rangle = -\frac{1}{2}\omega_{ij}\langle\psi_{ij}^2\rangle + \log PG(\omega_{ij}|b,0).$$

This is regognized as exponential tilting of the PG distribution with $\sqrt{\langle\psi_{ij}^2\rangle}$, which implies

$$\hat{\gamma}_{ij} = b, \quad \hat{\eta}_{ij} = \sqrt{\langle\psi_{ij}^2\rangle} = \sqrt{(\hat{\mu}_i\hat{m}_j^T)^2 + \hat{\mu}_i\hat{S}_j\hat{\mu}_i^T + \hat{m}_j\hat{\Sigma}_i\hat{m}_j^T + \mathrm{Tr}(\hat{\Sigma}_i\hat{S}_j)},$$

where we used the fact that $\psi_{ij} = u_i v_j^T$ when computing the expectation. Finally, we need $\langle \omega_{ij} \rangle$ for updating the factors:

$$\langle \omega_{ij} \rangle = \frac{b}{2\sqrt{\langle \psi_{ij}^2 \rangle}} \tanh(\sqrt{\langle \psi_{ij}^2 \rangle}/2). \tag{4}$$

In summary, we get closed-form updates by only slightly modifying the Gaussian rules, and the only computational overhead is for the hyperbolic tangent function since $\langle \psi_{ij}^2 \rangle$ is needed for updating τ in the Gaussian case. However, we again have the problem that Σ_i and S_j depend on i and j, which makes the algorithm slow in practice.

For the Gaussian case switching from the naive factorization to the fully factorized one with gradient-based optimization helped avoiding the computationally expensive repeated matrix inversion. The same can be done here by adding the gradient of the $\langle e^{\kappa\psi} \rangle$ term, replacing τ with ω_{ij}, and noting that ψ_{ij} is centered around zero instead of the data:

$$\frac{\delta \mathcal{L}}{\delta \hat{m}_{jk}} = -\langle \alpha_k \rangle \hat{m}_{jk} - \sum_i \left[(\kappa_{ij} - \langle \omega_{ij} \rangle \hat{\mu}_i \hat{m}_j^T) \hat{\mu}_{ik} + \langle \omega_{ij} \rangle \hat{\sigma}_{ik}^2 \hat{m}_{jk} \right].$$

4.2. Model 2: Explicit noise approach

For the direct model the Gibbs sampling equations and the naive factorization led to inefficient algorithms because ω_{ij} make the covariances depend on the sample and feature. We can get rid of this problem by explicitly instantiating ψ_{ij} as random variables with additive Gaussian noise. Then the full likelihood becomes

$$\mathcal{G}(\tau | a^0, b^0) \prod_i \prod_j \left[e^{\kappa_{ij}\psi_{ij} - \omega_{ij}\psi_{ij}^2/2} e^{-\tau/2(\psi_{ij} - \sum_k u_{ik} v_{ik})^2} PG(\omega_{ij} | b, 0) \right]$$
$$\prod_k \left[\mathcal{G}(\alpha_k | a^0, b^0) \prod_i \mathcal{N}(u_{ik} | 0, 1) \prod_j \mathcal{N}(v_{jk} | 0, \alpha_k^{-1}) \right].$$

4.2.1. Gibbs sampler

Now the Gibbs updates for the factors no longer depend on the data, but instead only on ψ_{ij}. Hence, they are exactly equivalent to the Gaussian case, as are the updates for α_k and τ, but replacing x_{ij} with ψ_{ij}. Sampling of ω_{ij}, in turn, is exactly as in the direct approach. The only new conditional is for ψ_{ij}, given by

$$\sigma_{\psi_{ij}}^2 = (\omega_{ij} + \tau)^{-1} \quad \mu_{\psi_{ij}} = \sigma_{\psi_{ij}}^2 (\tau \sum_k u_{ik} v_{jk} + \kappa_{ij}).$$

This is easy to sample, and hence the only notable computational overhead compared to the Gaussian case is the sampling of ω_{ij}.

4.2.2. Variational approximation

Now the naive VB approximation ((3), with additional terms for $q(\psi_{ij}) = N(\hat{\mu}_{\psi_{ij}}, \hat{\sigma}_{\psi_{ij}}^2)$) also leads to updates where the covariances do not depend on i or j. Again we can re-use

the updates for the Gaussian case for $q(u_i)$, $q(v_j)$ and $q(\alpha_k)$, and the updates for the direct approach for $q(\omega_{ij})$ (but noting that now the expectation $\langle\psi_{ij}^2\rangle$ is simply $\mu_{\psi_{ij}}^2 + \sigma_{\psi_{ij}}^2$). The remaining updates for $q(\tau)$ and $q(\psi_{ij})$ are given by

$$\hat{a}_\tau = a^0 + ND/2, \quad \hat{b}_\tau = b^0 + \sum_{i,j} \langle (\psi_{ij} - u_i v_j^T)^2 \rangle / 2$$

$$\hat{\sigma}_{\psi_{ij}} = (\langle \omega_{ij} \rangle + \langle \tau \rangle)^{-1}, \quad \hat{\mu}_{\psi_{ij}} = \sigma_{\psi_{ij}}^2 (\langle \tau \rangle \sum_k \hat{u}_{ik} \hat{v}_{jk} + \kappa_{ij}).$$

Here we assumed $q(\psi_{ij})$ to factorize over the samples. One could also consider factorizing $q(\psi)$ only over the rows or the columns. These would still be tractable, but requiring inversion of $D \times D$ or $N \times N$ matrices, which is only feasible for small problems.

For cases with missing data it also makes sense to consider a fully factorized approximation, especially since the gradient updates of the Gaussian case can be readily re-used by replacing the data x_{ij} and its square with the expectations $\mu_{\psi_{ij}}$ and $\mu_{\psi_{ij}}^2 + \sigma_{\psi_{ij}}^2$.

4.3. Negative binomial models

For fixed parameter r of the negative binomial likelihood, the equations above can be directly re-used for implementing a negative binomial latent factor model, by simply replacing $\kappa_{ij} = x_{ij} - \frac{1}{2}$ with $\kappa_{ij} = (x_{ij} - r)/2$ and $PG(1, .)$ with $PG(x_{ij} + r, .)$ in both the samplers and the variational approximations. However, in practice we want to perform inference over r as well. There are four alternatives on how to formulate the model, corresponding to constant r, one r_i for each row, one r_j for each column, and separate r_{ij} for each element. The last choice is unlikely to work well in practice since both r_{ij} and ψ_{ij} would compete on modeling the same entry. In the following we show how the inference is done for the choice of parameters r_j controlling the magnitudes of individual features.

For inference on r_j we turn to the solutions provided by Zhou and Carin (2012), based on compound-Poisson augmentation of the negative binomial distribution. We do not have space to repeat their details, but they show that with augmentation variables l_{ij} that correspond to the number of tables occupied by x_{ij} customers in a Chinese restaurant process with a concentration parameter r_j and gamma prior on r_j, there are closed-form expressions on the conditional distributions: $r_j | l_{ij}$ is a gamma distribution and $l_{ij} | r_j, x_{ij}$ is a sum of x_{ij} Bernoulli variables b_n with probabilities $r_j/(n - 1 + r_j)$. These constructs provide both Gibbs sampling and variational updates. As a practical note, we observe that $\sum_i \langle l_{ij} \rangle$ needed for updating $q(r_j)$ only requires $\langle l_{ij} \rangle$ for each unique count in $\mathbf{X}$ multiplied by their number; this is often much faster than directly computing $\langle l_{ij} \rangle$ for all ND elements.

Finally, it is no longer obvious that $q(\omega_{ij})$ can be obtained via exponential tilting of $PG(b, 0)$. As in the binary case, we have

$$\log q(\omega_{ij}) = -\frac{1}{2} \omega_{ij} \langle \psi_{ij}^2 \rangle + \langle \log PG(\omega_{ij} | x_{ij} + r, 0) \rangle,$$

where the latter expectation over r_j is now tricky since it is over a logarithm of an infinite sum. Straightforward numerical comparison, however, reveals that the closed-form expression $q(\omega_{ij}) = PG(x_{ij} + \langle r_j \rangle, \sqrt{\langle \psi_{ij}^2 \rangle})$ still holds. Finally, the variational updates for $q(r_j)$ require computing the expectation $\langle \log(1 - \text{logistic}(\psi_{ij})) \rangle$ which cannot be done in closed form but requires Monte carlo integration.

4.4. Other matrix factorization models

Like mentioned earlier, the update rules and conditional densities presented here for the case of binary PCA are directly applicable for wide range of other factor models. The functional forms remain identical, requiring only marginal changes in implementation. For a practical example, we briefly mention here the necessary changes for implementing a collective matrix factorization model for a typical recommender engine scenario of three matrices. Here $\mathbf{X}_1$ is users times items, $\mathbf{X}_2$ is users times user-features, and $\mathbf{X}_3$ is items times item-features:

$$\mathbf{X}_1 \approx \mathbf{U}_1\mathbf{U}_2^T, \quad \mathbf{X}_2 \approx \mathbf{U}_1\mathbf{U}_3^T, \quad \mathbf{X}_3 \approx \mathbf{U}_2\mathbf{U}_4^T,$$

with separate gamma priors for the precisions of the columns of each u_c and the approximation

$$q(\theta) = \prod_{c,k} q(\alpha_{ck}|\hat{a}_{\alpha_{ck}}, \hat{b}_{\alpha_{ck}}) \prod_c \prod_i q(u_i|\hat{\mu}_i, \hat{\Sigma}_i) \prod_{m,i,j} q(\omega_{ij}^{(m)}|\hat{\gamma}_{ij}^{(m)}, \hat{\eta}_{ij}^{(m)}).$$

The updates for $q(\alpha_{ck})$ are exactly as in the PCA case, done separately for each of the $C = 4$ matrices $\mathbf{U}_c$. Similarly the updates for $\omega_{ij}^{(m)}$ are done independently for each of the $M = 3$ observation matrices $\mathbf{X}_m$.

The only updates that change are the gradients for the latent factors that become sums over the observation matrices $\mathbf{X}_m$ influenced by each low-rank term $\mathbf{U}_c$. In the example setup above the gradient for $\mathbf{U}_1$ is the sum of the gradients for two separate PCA models for $\mathbf{X}_1$ and $\mathbf{X}_2$. While some of the other gradients would involve also matrix transposes, all updates re-use the same pieces that were required for the PCA case, complemented with suitable looping over the correct matrices, as presented by Klami et al. (2014).

5. Methods summary

Above we presented two Gibbs samplers and four variational approximations for both likelihood potentials. In terms of practical computation these fall into two categories: The Gibbs sampler and the naive variational approximations for the direct approach are considerably slower than the rest of the variants due to inverting a $K \times K$ matrix for each sample and feature. These are hence only feasible for small problems. The explicit noise variants, however, are roughly as efficient as the Gaussian case (for binary data) – the computational complexity is identical, but the constant factor is larger. This, however, comes at the expense of considerably looser the link between the actual data and the factors, which often means poor convergence.

The most interesting variants are the gradient-based fully factorized variational approximations. For both direct and explicit noise cases these are again slower than the Gaussian case only by a constant factor (for the binary case; the negative binomial case is slower due to somewhat heavy inference for r_j), and support for missing data is trivially achieved exactly as in the Gaussian case. These are hence the most likely candidates to become useful practical factor models for binary and over-dispersed count data.

5.1. Relationship with previous work

The models provided in this work are most closely related to the Pòlya-Gamma augmented regression models by Polson et al. (2013) and Zhou et al. (2012). Except for the efficient

updating of the Chinese restaurant table expectations for the negative binomial model and the closed-form updates for $q(\omega_{ij})$, the technical elements are for the augmentation are borrowed from their derivations. The fully factorized variational updates are, however, novel compared to their solutions, and are directly applicable for speeding up practical inference also for regression tasks, especially in presence of missing covariates. Pólya-Gamma augmentations have also recently been used for tensor factorizations by Rai et al. (2014), using only binary data and Gibbs sampling; the variational solutions and support for count data developed here could be extended also for the tensor case.

The most closely related sampling approaches for latent factor models are the Bayesian exponential family PCA by Mohamed et al. (2009) and Bayesian exponential family projections for coupled data sources by Klami et al. (2010). Even though their samplers are formulated for general exponential families, most practical applications (including the experiments in their papers) are on binary data. Compared to these approaches, our samplers have closed-form Gibbs updates, which makes implementation considerably easier. For factor models on count data we are not aware of earlier samplers, except for the Poisson factor analysis model of Zhou and Carin (2012) that is more closely related to topic models; it provides discrete latent variables instead of continuous ones as in our case.

The variational approximations provided here for the binary case can be directly contrasted with earlier approaches, for example those of Jaakkola (1997) and Seeger and Bouchard (2012). They explicitly construct bounds for the logistic function, whereas here the PG augmentation allows direct closed-form mean-field updates. This makes the approximation conceptually easier, using the same basic principle for updating all of the terms. The resulting updates, however, have interesting relationships with the earlier bounds.

First, we consider the relationships between the Bohning bound used for binary matrix factorizations by Seeger and Bouchard (2012). Their bound results in Gaussian updates wih fixed noise precision 0.25, applied on pseudo-data computed iteratively as $\tilde{x}_{ij} = (x_{ij} - \text{logistic}(u_i v_j^T))/4 + u_i v_j^T$. In effect, it models the data with constant variance but attempts to move the pseudo-data points further away from zero for high/low probabilities. Our bounds, in turn, have element-wise precision $\langle \omega_{ij} \rangle$ bounded above by 0.25, applied directly on the original data. In practice our bound is more accurate in modeling very high/low probabilities, as will be demonstrated in the next section, whereas for the linear regime of the logistic transformation the methods are very close.

The relationship between our bound and the bound Jaakkola (1997) presented for the binary regression case is even more interesting. Their bound results in, using our notation, factor precisions of $\langle \alpha \rangle + \sum_i \lambda_i(\xi) \langle u_i^T u_i \rangle$ where $\lambda_i(\xi)$ is updated iteratively for each sample as $\lambda_i(\xi) = \frac{1}{2\xi} \tanh(\xi/2)$ and $\xi^2 = \langle (\sum_k u_{ik} v_{jk})^2 \rangle$. We immediately see the close relationship with $\langle \omega_{ij} \rangle$ in (4); both bounds use the same hyperbolic tangent mapping. An important difference, however, is that the bound by Jaakkola (1997) has one parameter for each data point, whereas our bound naturally leads to separate values for each entry of $\mathbf{X}$.

6. Experiments

In this section we illustrate the approximations on synthetic data, to highlight the accuracy-efficiency tradeoffs of the alternatives. We compare the proposed methods against gaussian models followed by truncation of the parameter values into the correct domain, and against

the variational factor models presented by Klami et al. (2014) for both binary and count data, using Bohning bounds as presented by Seeger and Bouchard (2012). This is not to be considered as a fully-fledged comparison, but instead as a way of illustrating how the proposed solutions differ from characteristic examples of earlier work. For both the proposed variant and the comparison methods we se set the number of factors to $1.5 \times K_{true}$ and let ARD prune out the unnecessary factors; all methods do this with sufficient accuracy.

Since we compare also against models using different likelihood potentials, we evaluate the methods by computing the mean absolute error of the learned parameters. For binary data we measure the error for $p(x_{ij} = 1) = \langle \text{logistic}(\psi_{ij}) \rangle$, and for count data for the mean $\langle x_{ij} \rangle = \langle r_j \rangle \langle e^{\psi_{ij}} \rangle$. We evaluate these quantities with Monte Carlo integration and compare them against the true values used for creating the data.

To avoid cluttering the presentation we present the results for four representative variants: (i) Direct approach with naive approximation (PG-Direct-Naive), (ii) Direct approach with factorized approximation (PG-Direct-F), (iii) Explicit noise approach with factorized approximation (PG-Noise-F), and (iv) Direct approach with Gibbs sampling. The comparison between (ii) and (iii) shows the difference between the approaches, whereas the comparison between (i) and (ii) illustrates the dramatic difference in computational speed between the naive and factorized approximations. For the gradient-based variants we use a conservative choice of step length of 0.3 times the diagonal Hessian, and we run all methods for equal number of iterations ($3,000$ for binary data, $6,000$ for count data).

6.1. Comparison on binary data

Earlier binary factor models (Mohamed et al., 2009; Li and Tao, 2010) were demonstrated on repeated binary patterns corrupted by flipping noise. Such data would be too simple to highlight the differences of our model variants, but it is worth mentioning that all of our variants find the correct structure for the data of Mohamed et al. (2009) in a few iterations and in a matter of seconds, in contrast to the $4,000$ iterations required by their sampler.

For properly evaluating the proposed variants we create data from the underlying model, sampling $u_{ij}, v_{jk} \sim N(0, 1)$ and the data from Bernoulli distribution with logistic-transformed $\psi_{ij} = s u_i v_j^T + b$ rate. Here s controls the extremity of the probabilities so that small s makes all probabilities close to 0.5 whereas large s makes most probabilities close to 0 or 1. The shift parameter b, in turn, controls the ratio of ones; negative values make 1 less likely. We run the experiments for various values of s and b to illustrate how the methods work in different conditions relevant for real-world data analysis: (i) balanced data with probabilities near the linear region of the logistic function ($s = 0.5$, $b = 0$), (ii) balanced data with extreme probabilities ($s = 2.5$, $b = 0$), and (iii) imbalanced data with only few ones ($s = 2$, $b = -4$). The first setup is easy for all methods since most probabilities are near the linear region of the logistic function, the second one is harder, and the last one is particularly difficult for the earlier techniques. For all cases we use $N = 1000$, $D = 100$ and $K = 10$, but similar results would be obtained for a wide range of matrix sizes.

The errors for these three setups are presented in Figure 1. For the first scenario all methods are roughly as accurate, whereas for the rest the PG-Direct variants outperform the rest, with the Gibbs sampler being superior for the hardest case. The naive variational approximation and the Gibbs sampler are, however, dramatically slower than the rest of the

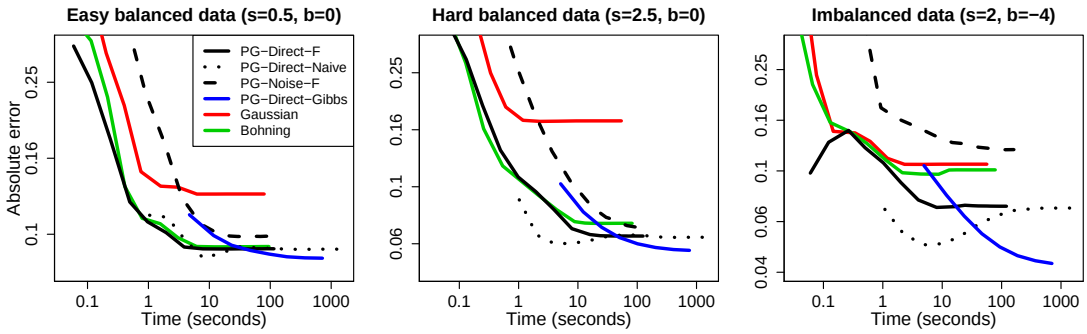

Figure 1: Illustration on binary data, showing the absolute error against wall-clock time in log scale. On the easy setup (left) the comparison method of Klami et al. (2014) using Bohning bound (green line) is as fast and accurate as the proposed methods (black and blue lines), but on the harder setups (middle and right) the proposed model variants with the direct approach are clearly the best. The Gibbs sampler (blue line) is the most accurate method but slow compared to the factorized variational approximations. The noisy variant (PG-Noise-F) and the Gaussian baseline (red line) are inaccurate.

methods; they require minutes to converge compared to just seconds for the other methods. The faster variants are not notably slower than the Gaussian model, which makes them easily applicable also for large scale setups. While the data matrix here has only $100,000$ samples, the computational times would be roughly the same also for much larger setups that have comparable number of observed entries.

The accuracy differences are best understood by looking at the extreme probabilities near zero and one. For the second setup the mean absolute error for the entries that are either below 0.05 or above 0.95 is 0.034 for PG-Direct-F and 0.044 for Bohning. This difference explains almost completely the deviation in the overall accuracies. Similarly, for the third setup the mean error for the entries with probability above 0.95 is 0.32 for PG-Direct-F and 0.49 for Bohning. Since the Bernoulli likelihood emphasizes the extreme probabilities, this difference would also be clearly visible if using likelihood for measuring accuracy. For the second setup the average likelihood per entry is -0.155 for PG-Direct-F and -0.168 for Bohning. For the third setup the difference is even bigger, -0.129 vs -0.185.

6.2. Comparison on count data

We run experiments on two kinds of count data. The first setup uses Poisson data having equal mean and variance, which we generate using rate $\lambda_{ij} = e^{su_i v_j^T + b}$. The second setup uses over-dispersed data from the proposed model, with variance larger than the mean. We generate it with $r_j \sim \mathcal{G}(5, 1)$ and $p_{ij} = \text{logistic}(su_i v_j^T + b)$. For both setups we sample data sets with small and large counts by varying s, setting $N = 500$, $D = 50$ and $K = 10$.

The accuracies are visualized in Figure 2. For negative binomial data the three direct alternatives are the most accurate ones, with PG-Direct-F being an order of magnitude

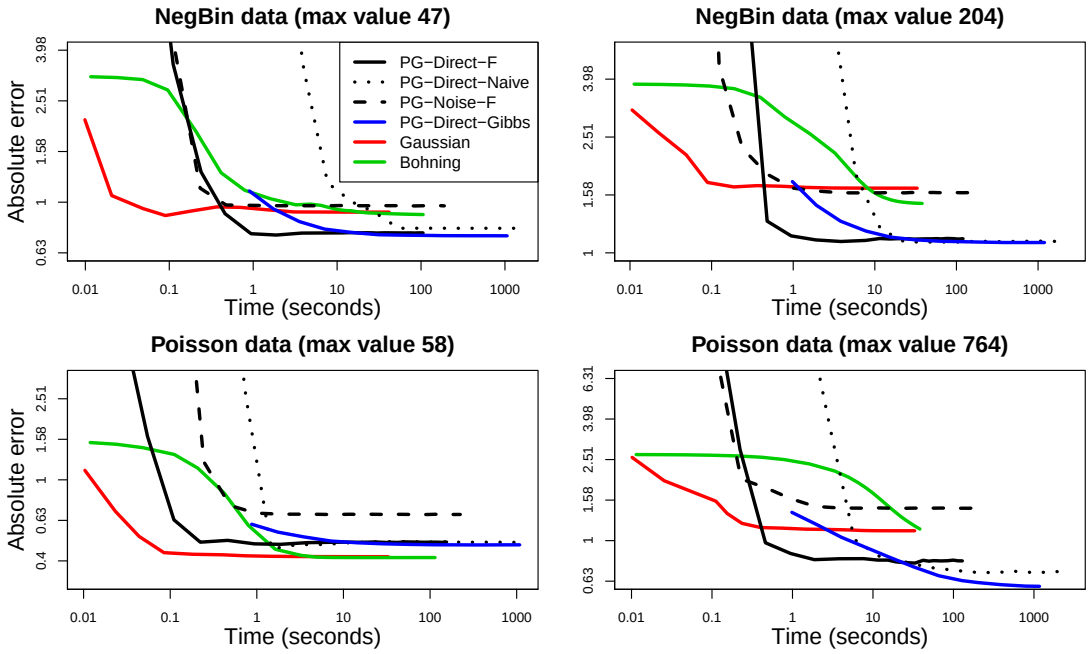

Figure 2: Illustration on count data, with negative binomial data in the top row and Poisson data in the bottom row. The direct variants are clearly the best on all but the small-scale Poisson data (bottom left), and even for that setup PG-Direct-F converges faster than the comparison method using Bohning bounds (green line). The explicit noise variant (PG-Noise-F) has poor accuracy, as does the Gaussian comparison method (red line) for all but the small-scale Poisson data.

faster than the naive variant and Gibbs sampler. Even though the proposed methods are not designed for Poisson data, they are still the most accurate ones even for that when the scale gets large. For the large-scale Poisson data the Gibbs sampler is ultimately the most accurate method, but PG-Direct-F reaches good accuracy in a few seconds whereas the sampler takes more than ten minutes to converge and half a minute to reach the accuracy of PG-Direct-F. An interesting observation is that here PG-Direct-F is also considerably faster than the comparison method of Klami et al. (2014), despite the additional computation needed for updating $q(r_j)$. This is because the comparison method converges very slowly due to too high noise precision, fixed to $0.17 \max(x_{ij})$ in that algorithm.

7. Conclusion

In this paper we laid out the details for implementing matrix factorization models for binary and over-dispersed count data using Pòlya-Gamma augmentations. Even though many of the inference details follow from what Polson et al. (2013) and Zhou et al. (2012) presented for regression models, alternative ways of implementing the factor models lead to big differences in accuracy and computational cost. The main result of this paper is that

the fully factorized variational approximation (Ilin and Raiko, 2010) for the direct approach is both accurate (only losing to the Gibbs sampler in some cases) and the most efficient method, and hence the recommended alternative for large data sets. It learns factorizations for fairly large matrices in a matter of seconds and is directly applicable for a wide range of matrix factorization models besides PCA, including canonical correlation analysis (Klami et al., 2013), group factor analysis (Virtanen et al., 2012), and collective matrix factorization (Singh and Gordon, 2010; Klami et al., 2014). The main advantage compared to earlier variational models is better accuracy for extreme probabilities in the binary case and support for over-dispersed count data. The model was also shown to outperform the comparison methods on imbalanced binary data.

The samplers presented here are accurate but not very efficient; the direct approach is slow due to needing to invert $K \times K$ matrix for each sample, whereas the explicit noise approach converges slowly. Nevertheless, they are good choices for small data sets. More efficient solutions should be possible by gradient-based sampling for the direct approach.

In the course of deriving the variational approximations we also clarified some practical details on variational inference for Pòlya-Gamma augmentations in general. Zhou et al. (2012) used Monte Carlo integration for $\langle \omega_{ij} \rangle$, whereas we showed that it can be computed in closed form. We also showed how the Chinese restaurant process expectation required for inferring $q(r_j)$ can be efficiently implemented with low computational cost. Finally, we showed an interesting relationship between the explicit bound of Jaakkola (1997) and the update rules of the PG augmentation for binary data.

Acknowledgments

We acknowledge funding from Academy of Finland (grants 251170 and 266969), TEKES DIGILE SHOK programme D2I, and Xerox University Affairs Committee.

References

D. Bohning. Multinomial logistic regression algorithm. *Annals of the Institute of Statistical Mathematics*, 44:197–200, 1992.

Mark Girolami and Simon Rogers. Variational bayesian multinomial probit regression with gaussian process priors. *Neurocomputing*, 18(8):1790–1817, 2006.

Alexander Ilin and Tapani Raiko. Practical approaches to principal component analysis in the presence of missing data. *Journal of Machine Learning Research*, 11:1957–2000, 2010.

Tommi S. Jaakkola. Variational methods for inference and estimation in graphical models. Phd thesis, Massachusetss Institute of Technology, 1997.

Mohammad E. Khan, Guillaume Bouchard, Benjamin M. Marlin, and Kevin P. Murphy. Variational bounds for mixed-data factor analysis. In *Advances in Neural Information Processing Letters 24*, 2010.

Arto Klami, Seppo Virtanen, and Samuel Kaski. Bayesian exponential family projections for coupled data sources. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence (2010)*, pages 286–293, 2010.

Arto Klami, Seppo Virtanen, and Samuel Kaski. Bayesian canonical correlation analysis. *Journal of Machine Learning Research*, 14:965–1003, 2013.

Arto Klami, Guillaume Bouchard, and Abhishek Tripathi. Group-sparse embeddings in collective matrix factorization. In *Proceedings of the International Conference on Representation Learning*, 2014.

Jun Li and Dacheng Tao. Simple exponential family PCA. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, pages 453–460, 2010.

Andriy Mnih and Ruslan Salakhutdinov. Probabilistic matrix factorization. In *Advances in neural information processing systems 20*, pages 1257–1264, 2007.

Shakir Mohamed, Katherine Heller, and Zoubin Ghahramani. Bayesian exponential family PCA. In *Advances in Neural Information Processing Systems 21*, pages 1089–1096, 2009.

Nicholas G. Polson, James G. Scott, and Jesse Windle. Bayesian inference for logistis models using Polya-Gamma latent variables. *Journal of the American Statistical Association*, 108 (504):1339–1349, 2013.

Piyush Rai, Yingjian Wang, Shengbo Guo, Gary Chen, David Dunson, and Lawrence Carin. Scalable Bayesian low-rank decomposition of incomplete multiway tensors. In *Proceedings of the 31st International Conference on Machine Learning*. JMLR, 2014.

Daniel B. Rowe. *Multivariate Bayesian statistics: models for source separation and signal unmixing*. CRC Press, 2002.

Matthias Seeger and Guillaume Bouchard. Fast variational Bayesian inference for non-conjugate matrix factorization models. In *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics*, volume 22, pages 1012–1018. JMLR, 2012.

Ajit P. Singh and Geoffrey J. Gordon. A Bayesian matrix factorization model for relational data. In *Uncertainty in Artificial Intelligence*, 2010.

M. Tipping and C. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society B*, 21(3):611–622, 1999.

Seppo Virtanen, Arto Klami, Suleiman A. Khan, and Samuel Kaski. Bayesian group factor analysis. In *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics*, volume 22 of *JMLR W&CP*, pages 1269–1277. JMLR, 2012.

Mingyuan Zhou and Lawrence Carin. Augment-and-conquer negative binomial processes. In *Advances in Neural Information Processing Systems 25*, 2012.

Mingyuan Zhou, Lingbo Li, David Dunson, and Lawrence Carin. Lognormal and gamma mixed negative binomial regression. In *Proceedings of the 29th International Conference on Machine Learning*, 2012.