A GIBBS SAMPLING APPROACH TO INDEPENDENT FACTOR ANALYSIS

Omolabake A. Adenle & William J. Fitzgerald

The Signal Processing & Communications Laboratory, Department of Engineering,
University of Cambridge, Cambridge CB2 1PZ

ABSTRACT

We present a Gibbs sampler for estimating parameters of the Independent Factor model. Independent Factor Analysis (IFA) is a generalization of Mixtures of Factor Analyzers, where instead we learn nonlinear subspaces in data. IFA can also be considered a method for blind source separation. The IFA generative model is hierarchical, with each factor modeled as an independent Mixture of Gaussians, each mixture component representing a factor state. Computing expectations over factors quickly becomes intractable with increasing number of factors as this requires summation over exponentially many state configurations, making parameter estimation via Expectation Maximization (EM) with an exact E-step infeasible. Unlike the Variational method that has been proposed, we take a simulation based approach to obtain exact parameter estimates. We define prior distributions and use a Gibbs sampler to obtain samples from the parameter posterior. Application to synthetic data demonstrates effectiveness of the method in estimating model parameters and robustness to model permutation invariance.

1. INTRODUCTION

Expectation Maximization (EM) is often ideal for performing unsupervised learning in models assuming latent data structure. Such models allow for efficient computation of the expectation over hidden data (E-step), e.g. Mixture of Factor Analyzers [1]. On the other hand, when the generative model is hierarchical, i.e. observations have a combinatorial dependence on parameters of hidden data-model, computing expectations of the hidden (data) factors then requires marginalizing over all possible configurations (states) of the underlying parameters. The computational expense of marginalization increases exponentially with the number of hidden factors. Some examples of hierarchical models include the Cooperative Vector Quantization, Factorial Hidden Markov [2] and, the model of interest in this paper, the Independent Factor (IF) models.

Both simulation-based and variational approaches have been applied to overcome intractability of computing expectations over hidden state configurations for IFA. Olshausen & Millman [3] applied a stochastic EM algorithm, replacing the exact E-step with a stochastic simulation and using steepest descent to perform the M-step. The gradient update requires computing an inverse Hessian matrix for each accepted state change, at each iteration of the algorithm. Also, there is a separate analytic form of the state distribution for each state (applied to the binary case). As the number of factors increases, there are exponetially many states, making practical implementation difficult. In [4], Utsugi applied a similar stochastic EM algorithm to the Ensemble of Independent Factors (EIF) model, a simplified version of IFA: the factors are not modeled as MoGs, but as Gaussians with zero mean and unit diagonal covariance matrices. Exact calculation of the E-step was replace with a Gibbs update. The stochastic E-step also does not generalize, suffering from the same problem as in [3]. A Variational EM algorithm was succesfully applied to obtain maximum likelihood estimates for IFA in [5], where the true likelihood was approximated by a factorial distribution.

We take a Bayesian approach to IFA, specifying priors over model parameters, and use MCMC to estimate parameters. In Section 2, we describe the IF model and exact factor estimation. We present a Gibbs sampler for estimating the IF parameters in Section 3. Finally, we apply the algorithm to simulated data and discuss implementation and results in Section 4.

2. INDEPENDENT FACTOR ANALYSIS

2.1. Independent Factor Model

A p-dimensional observation, which we denote $\mathbf{y} \in \mathbf{R}^p$ can be represented by an m-dimensional vector, $\mathbf{x} \in \mathbf{R}^m$ (factor), such that $m \ll p$. The factors are related to observations via a linear transformation plus additive noise:

$$\mathbf{y}_i = \mathbf{H}\mathbf{x}_i + \mathbf{u}_i, \qquad i = (1, ..., N) \tag{1}$$

where $\mathbf{H} \in \mathbf{R}^{m \times p}$ is a matrix of factor loadings and $\mathbf{u} \in \mathbf{R}^p$ is the noise vector such that $\mathbf{u} \sim N_p(\mathbf{0}, \Lambda)$, with diagonal noise covariance matrix $\Lambda = \mathrm{diag}(\lambda_{11}, \lambda_{22}, ..., \lambda_{pp})$. $N_c(\mathbf{a}, \mathbf{B})$ denotes a c-dimensional normal distribution with mean vector $\mathbf{a}$ and covariance matrix $\mathbf{B}$. We have N samples of the observation. Thus we write $\mathbf{Y} = [\mathbf{y}_1, ...\mathbf{y}_N] \in$

$\mathbf{R}^{p\times N}$ and $\mathbf{X} = [\mathbf{x}_1, ...\mathbf{x}_N] \in \mathbf{R}^{m\times N}$. Exclusive of the parametric model of the factors, this model is exactly the same as in Factor Analysis(FA). From (1), the conditional distribution of an observation is

$$p(\mathbf{y}|\mathbf{x}) = N(\mathbf{H}\mathbf{x}, \mathbf{\Lambda}), \tag{2}$$

also as in FA.

The j^{th} factor follows a mixture of n_j Gaussians, with $q_j = (1, ..., n_j)$ means μ_{j,q_j}, variances ν_{j,q_j}, and mixing proportions π_{j,q_j}. We assume a fixed value for n_j. Defining the parameter vector for the j^{th} factor as $\theta_j = (\mu_{j,q_j}, \nu_{j,q_j}, \pi_{j,q_j})$, the factor probability distribution is

$$p(x_i|\theta_j) = \sum_{q_j=1}^{n_j} \pi_{j,q_j} N(\mu_{j,q_j}, \nu_{j,q_j}), \tag{3}$$

As with standard mixture distributions, $0 \leq \pi_{j,q_j} \leq 1$ and $\sum_{q_j} \pi_{j,q_j} = 1$. The factor model uses the components of the Mixture of Gaussians (MoGs) as hidden states selected according to the state vector $\mathbf{q} = (q_j)_{j=1}^{p}$. Each state is selected with probability $p(q_j) = \pi_{j,q_j}$. Using (3), the joint probability of the factors is formed by taking the product over the MoGs of the m factors, the factors being independent:

$$\begin{aligned} p(\mathbf{x}|\theta) &= \prod_{j=1}^{m} p(x_i|\theta_j) \\ &= \prod_{j=1}^{m} \sum_{\mathbf{q}} \pi_{j,q_j} N(\mu_{j,q_j}, \nu_{j,q_j}), \end{aligned} \tag{4}$$

where $\mathbf{q} = (q_1, q_2, ..., q_m)$ and the summation is performed over all possible state configurations, $\mathbf{q}$. We summarise terms in (4) as,

$$\begin{aligned} \pi_{\mathbf{q}} &= \prod_{j=1}^{m} \pi_{j,q_j} = \pi_{j,q_1} \dots \pi_{p,q_m} \\ \mu_{\mathbf{q}} &= (\mu_{q,q_1}, \dots, \mu_{p,q_m}) \\ \mathbf{V}_{\mathbf{q}} &= \mathrm{diag}(\nu_{1,q_1}, \dots, \nu_{p,q_m}) \end{aligned}$$

The factor distribution is a factorial MoG and is similar to Co-operative Vector Quantization in the co-dependence of factor parameters across all the hidden states. For a given state configuration $\mathbf{q}$, the conditional distribution of the factors is

$$p(\mathbf{x}|\mathbf{q}) = N(\mu_{\mathbf{q}}, \mathbf{V}_{\mathbf{q}}).$$

In standard FA, the factor distribution would be zero mean with unit variance, whereas in IFA the factor mean and variance are determined by the state vector $\mathbf{q}$.

We define the vector over the collective model parameters as $\mathbf{W} = (\mathbf{H}, \mathbf{\Lambda}, \theta)$. The complete-data likelihood factorizes as

$$p(\mathbf{q}, \mathbf{x}, \mathbf{y}|W) = p(\mathbf{q})p(\mathbf{x}|\mathbf{q})p(\mathbf{y}|\mathbf{x}) \tag{5}$$

We refer the reader to Attias' paper [5] for a more detailed exposition of IFA.

2.2. Intractability of Factor Estimation

Estimating factors under the IF generative model is similar to FA. Integrating over the factors, the likelihood of the observed data is

$$p(\mathbf{y}|\mathbf{q}, W) = N(\mathbf{H}\mu_{\mathbf{q}}, \mathbf{H}\mathbf{V}_{\mathbf{q}}\mathbf{H}^{\mathbf{T}} + \mathbf{\Lambda}).$$

Starting from (5), we can derive the conditional distribution of the factors as

$$\begin{aligned} p(\mathbf{x}|\mathbf{q}, \mathbf{y}) &= N(\rho_{\mathbf{q}}(\mathbf{y}), \mathbf{\Sigma}_{\mathbf{q}}) && (6) \\ \mathbf{\Sigma}_{\mathbf{q}} &= (\mathbf{H}^{\mathbf{T}}\mathbf{\Lambda}\mathbf{H} + \mathbf{V}_{\mathbf{q}}^{-\mathbf{1}})^{-1} && (7) \\ \rho_{\mathbf{q}}(\mathbf{y}) &= \mathbf{\Sigma}_{\mathbf{q}}(\mathbf{H}^{\mathbf{T}}\mathbf{\Lambda}^{-\mathbf{1}}\mathbf{y} + \mathbf{V}_{\mathbf{q}}^{-\mathbf{1}}\mu_{\mathbf{q}})^{-1} && (8) \end{aligned}$$

Again the only difference between IFA and FA is that the factor means and variances are determined by $\mathbf{q}$. To obtain estimates of the factors, we must marginalize $p(\mathbf{x}, \mathbf{q}|\mathbf{y})$ over $\mathbf{q}$:

$$p(\mathbf{x}|\mathbf{y}) = \sum_{\mathbf{q}} p(\mathbf{x}|\mathbf{q}, \mathbf{y})p(\mathbf{q}|\mathbf{y})$$

Marginalization (E-step) requires $\prod_j n_j$ sums at each iteration of EM, quickly becoming intractable as the number of underlying factors (and their states) increases. To overcome this, we present a Gibbs sampler to generate samples from the posterior distribution of $\mathbf{W}$. The complexity of the algorithm is linear in the number of factors.

3. ESTIMATING PARAMETERS VIA GIBBS SAMPLING

We use the Gibbs sampler to obtain samples from the posterior distribution of the collective parameter set $\mathbf{W}$. The algorithm can be described in two stages [6]: (i) sample hidden data: state vector $\mathbf{q}$ and factors $\mathbf{x}$ according to (6), and (ii) sample $\mathbf{W}$ from the parameter posterior. To obtain a sample of the state vector, we perform one cycle of a Gibbs Sampler with invariant distribution $p(\mathbf{q}|\mathbf{y}, W) \propto p(\mathbf{y}|\mathbf{q}, W)p(\mathbf{q})$, sampling the j^{th} state according to $p(q_j) \sim p(q_{k\neq j}|\mathbf{y}, W)$.

3.1. Parameter Priors

We use conjugate priors, for convenient sampling from the parameter posterior. For the factors, we use a hierarchical prior structure and data dependent initialisation for MoGs similar to [7]. As the factors are independent and may have widely different supports, we use separate priors for each factor.

$$p(\theta_j) = p(\mu_{j,q_j}|\nu_j, \kappa_j)p(\pi_q|\gamma)p(\nu_{j,q_j}|\alpha_j, \beta_j)p(\beta_j).$$

The conjugate priors for the j^{th} factor are

$$\begin{aligned} \mu_{j,q_j} &\sim N(\eta_j, \kappa_j^{-1}) \qquad \nu_{j,q_j}|\beta \sim \mathrm{IG}(2\alpha_j, 2\beta_j) \\ \pi_{\mathbf{q}} &\sim \mathrm{Dir}(\gamma) \qquad \beta_j \sim \mathrm{Gam}(2g, (2h)^{-1}) \end{aligned}$$

where β_j is a hyperparameter and $(\eta_j, \alpha_j, \gamma_j, g, h)$ are scalars. IG, Gam and Dir denote the Inverse Gamma, Gamma and Dirichlet distributions, respectively.

To enforce sparsity on the mixing matrix $\mathbf{H}$, we use an Automatic Relevance Determination (ARD) [8] prior on the columns, $\mathbf{H}_{.i}$, of the mixing matrix. Thus, we treat Δ as a hyperparameter with gamma distribution: $p(\Delta) \sim \mathrm{Gam}(a, b)$, where (a, b) are parameters of the hyperprior on Δ.

$$p(H_{ij}) = N(0, \delta_{ij}^{-1}) \qquad p(\delta_{ij}|a, b) = \mathrm{Gam}(a, b),$$

Finally, we use a Gamma prior on the elements of the noise covariance matrix $\lambda_{ij}^{-1} \sim \mathrm{Gam}(\frac{\alpha}{2}, \frac{\tau}{2})$

3.2. The Algorithm

Parameter estimation can be described in two stages: we first draw samples of the factors and the state configuration and then draw samples from the resulting factor model. After initialising parameters and hyperparameters $(W^{(0)}, \beta^{(0)})$, the t^{th} iteration of the Gibbs sampler with n_j, $(j = 1...m)$ mixture components for each factor proceeds as follows.

- `Sample factors and state vector` $\mathbf{q}$
  $\mathbf{x}^t \sim N_m(\Sigma_q(H^T\Lambda^{-1}y + V_q\mu_q), \Sigma_q)$
  $\mathbf{q}^t \sim p(\mathbf{q}^{(t-1)}|\mathbf{y}, W^{(t-1)})$

- `Sample factor parameters` $W^{(t)}$ `and` $\beta_j^{(t)}$

  - `For j=1:m, i=1:p,`
    $\beta_j^{(t)} \sim \mathrm{Gam}(2g+2m\alpha, (2h+2\sum_{j=1}^{m}\nu_{j,q_j})^{-1,(t-1)})$
    $p(z_{nj}^{(t)} = i) \propto \pi_{j,q_j}^{(t-1)}\mathrm{N}(\mu_{j,q_j}^{(t-1)}, \nu_{j,q_j}^{(t-1)})$,
    ${\pi_{\mathbf{q}}}^{(t)} \sim \mathrm{Dir}(\gamma + n_{i1}, ..., \gamma + n_{im})$
    $\nu_{j,q_j}^{(t)} \sim$
    $\mathrm{IG}(2\alpha+n_{ij}, 2\beta^{(t)}+\sum_{j:z_j^{(t)}=i}(x_{ji}^{(t-1)}-\mu_{j,q_j}^{(t-1)})^2)$
    $\delta_{ji} \sim \mathrm{Gam}(a+\frac{1}{2}, b+\frac{Hji^{(t-1),2}}{2})$
  - `For j=1:m,`
    $H_{.j}^{(t)} \sim \mathrm{N}(m_j, R_j|\Delta_{.j})$
    $\lambda_{jj}^{(t),2} \sim \mathrm{Gam}(\frac{N+\alpha}{2}, \frac{S_{jj}+\tau}{2})$

$z_{nj}^{(t)}$ is the state of the n^{th} sample of the j^{th} factor and n_{ij} the number of factor samples allocated to the i^{th} state of the j^{th} factor. Rows of the hyperparamter Δ on $\mathbf{H}$ are obtained from the prior as $\Delta_i = \mathrm{diag}(\delta_{ij})_{j=1}^m$ and $i = 1, ..., p$. Factors are sampled from Normal distributions with mean m_j and covariance matrix R_j such that

$$\begin{aligned} m_j &= (\lambda_{ii}, \Delta_i^{-1} + X'X)^{-1}YX_{.i} \\ R_j &= (\lambda_{ii} + \Delta_i^{-1}(X'X))^{-1} \end{aligned}$$

Elements λ_{ii} of noise covariance matrix Λ are sampled from an Inverse Gamma distribution dependent on $S_{ii} = \mathrm{diag}(S)$, where

$$S = \sum_{i=1}^{N}(\mathbf{y}_i - H\mathbf{x}_i)(\mathbf{y}_i - H\mathbf{x}_i)^T$$

After a *burn in* period, we obtain samples from the posterior over model parameters. Since a Gibbs sampler allows sampling from distributions known only up to a constant of proportionality, we do not have to bear the expense of computing normalizing constants for $p(\mathbf{q}|\mathbf{y}, W)$. To marginalise over states, we only consider samples $\mathbf{x}^{(t)}$. Each iteration is linear in the number of hidden states and is much less computationally demanding than the exponential dependence of EM with an exact E-step.

4. DISCUSSION AND IMPLEMENTATION

As in FA and ICA, IFA suffers from scaling and permutation invariance. To remove scale invariance (up to a factor of ± 1), we assume the factors are of unit variance and postprocess parameter samples to ensure factor estimates are also of unit variance:

$$\mu_{j,q_j} \rightarrow \frac{\mu j, q_j}{\sigma_j} \qquad \nu_{j,q_j} \rightarrow \frac{\nu j, q_j}{\sigma_j^2} \qquad H_{ij} \rightarrow H_{ij}\sigma_j$$

In [5], the structure of the true mixing matrix was used to overcome permutation ambiguities in estimates of $\mathbf{H}$. In trying to maintain a blind approach to parameter estimation, we do not utilise knowledge of the known mixing matrix. Instead, we model factors with more mixture components than required (assuming prior knowledge of the n_j's). However, in practice the algorithm did not exhibit permutations since Gibbs samplers do not mix well across modes.

We tested the algorithm on data with factors $\mathbf{x}_i \in \mathbf{R}^3$ mixed with $\mathbf{H}$ in Table 1 to obtain $N = 200$ sample observations $\mathbf{y}_i \in \mathbf{R}^{10}$. The factors were trimodal ($n_j = 3$), bimodal($n_j = 2$) and heavy tailed($n_j = 3$); all factors were modeled with three states. Additive noise was sampled from $N_{10}(\mathbf{0}, .1\mathbf{I})$. The initial factors were chosen as the first 3 principal components of the observations.

Table 1 and Figure 4 show parameter estimates obtained from the last 150 samples after 300 iterations. The first factor was learned up to a -1 scale. We found convergence of

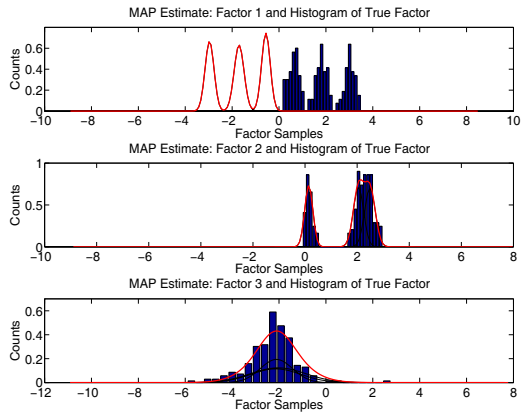

Fig. 1. Estimated factors and histogram of true factors; Note that factor 1 was learnt to a -1 scale

the algorithm was sensitive to the initialization of the factors and the chosen latent space dimensionality. Initialization to the required number of prinicpal components of the observations was found to provide suitable convergence. While inclusion of extra mixture components may make the algorithm robust to permutations, it may also "confuse" the algorithm. When the chosen dimensionality is greater than the true one, the algorithm tends to exhibit overfitting. Underspecification also leads to poor exploration of the latent factors. In spite overspecifying the latent dimensionality of the second factor, the algorithm was still able to attain good estimates of the true factors.

5. FUTURE WORK

Modeling factors by more mixture components wastes computational effort when not all the components are needed. Most methods assume knowledge of the number of mixture components n_j, although some work has been done using ARD to estimate n_j. We are currently investigating *jump diffusion* methods for performing model selection.

6. REFERENCES

[1] Zoubin Ghahramani and Geoffrey E. Hinton, "The EM algorithm for mixtures of factor analyzers," Tech. Rep. CRG-TR-96-1, 21 1996.

[2] Zoubin Ghahramani and Michael I. Jordan, "Factorial hidden Markov models," in *Proc. Conf. Advances in Neural Information Processing Systems, NIPS*, David S. Touretzky, Michael C. Mozer, and Michael E. Hasselmo, Eds. 1995, vol. 8, pp. 472–478, MIT Press.

[3] B.A. Olshausen and K.J. Millman, "Learning sparse codes with a mixture-of-gaussians prior," in *Advances in Neural Information Processing Systems*. 2000, vol. 12, pp. 841–847, MIT Press, Cambridge, MA.

[4] Akio Utsugi, "Ensemble of independent factor analyzers with application to natural image analysis," *Neural Processing Letters*, vol. 14, no. 1, pp. 49–60, 2000.

[5] Hagai Attias, "Independent factor analysis," *Neural Computation*, vol. 11, no. 4, pp. 803–851, 1999.

[6] Jean Diebolt and Christian P. Robert, "Estimation of finite mixture distributions through bayesian sampling," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 56, no. 2, pp. 363–375, 1994.

[7] Matthew Stephens, "Bayesian analysis of mixture components-an alternative to reversible jump methods," *The Annals of Statistics*, vol. 28, no. 1, pp. 40–74, February 2000.

[8] Radford M. Neal, *Bayesian Learning for Neural Networks*, Springer, 1996.

$$\mathbf{H} = \begin{bmatrix} .8 & .8 & .8 & .8 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .8 & .8 & .8 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & .8 & .8 & .8 \end{bmatrix}^T$$

$$\hat{\mathbf{H}} = \begin{bmatrix} -.81 & -.83 & -.81 & -.81 & -.029 & .016 & -.021 & .01 & .04 & -.00 \\ .03 & .05 & .00 & .04 & .78 & .83 & .77 & -.08 & -.06 & -.05 \\ .00 & .02 & -.02 & .00 & -.02 & -.01 & -.03 & .83 & .81 & .88 \end{bmatrix}^T$$

	μ_j	ν_j	π_j	$\hat{\mu}_j$	$\hat{\nu}_j$	$\hat{\pi}_j$
$\mathbf{x}_1$	(2.98,1.79,.60)	(.06,.06,.06)	(.33,.33.33)	(-2.96,-.56,-1.70)	(.04,.03,.04)	(.32,.35,.33)
$\mathbf{x}_2$	(.15,2.24)	(.02,.09)	(.33,.67)	(.14,2.49,2.05)	(.03,.04,.04)	(.29,.36,.35)
$\mathbf{x}_3$	(-2.20,-2.20,-2.20)	(.77,.19,1.93)	(.33,.33,.33)	(-2.13,-2.11,-2.03)	(1.12,.47,1.38)	(.30,.33,.36)

Table 1. Summary of Parameter Estimates, $\hat{W}$