where $\mathbf{D}_{Z_j}^i$ denotes a diagonal matrix constructed from $Z_j^i$ and the auxiliary variables $t = \{t_j^l\}_{l=1}^{K^j+1}$ are sampled as below

$$p\left(\{t_j^l\}_{l=1}^{K^j+1} \mid \text{rest}\right) \propto \begin{cases} \mathbf{X}_j^{il} & t_j^{il} = 0, \forall i, l \in [1, K^j+1] \\ \prod_{k=1}^{K^j+1} \left(\Phi^j \mathbf{H}_j^k\right)^{t_j^{il}} & t_j^{il} > 0, \forall i, l \in [1, K^j+1] \end{cases}$$

(29)

Gibbs sampling of $\lambda_j$ remains similar to the variables $\Phi$ and $W_j$. The required posterior distributions are given as

$$p(\lambda_j \mid Z_{il}, W_{il}, \Phi, X_{il}, r) \propto \lambda_j \sum_{n=1}^{N_j} \exp\left\{-\left(b_{\lambda_j} + MN_j\right)\right\}$$

(30)

For each $i, l$, the auxiliary variables $r_j^l$ can be sampled as

$$r_j^l \mid \text{rest} \sim \text{Binomial}\left(\frac{\mathbf{X}_j^{il}}{\Phi^j \mathbf{H}_j^l}, \lambda_j + \lambda_j + \mu_j\right)$$

(31)

4.5 Sampling Parameters $\Phi, W_j, \sigma_{\phi_j}$ and $\sigma_{\eta_j}$ for NHFA-GGM model

Gibbs sampling updates for these parameters remains same as the updates described in [8].

4.6 Sampling $\alpha_j$

Once again, Gibbs sampling updates for $\alpha_j$ remains same as the updates described in [8]. The hyperparameter $\alpha_j$ controls the variation of $A_j$ (source-specific beta process) around $B$ (parent beta process). As $\alpha_j$ vary from a low to high value, its concentration on the random distribution $B$ increases. Since the random distribution $B$ is shared across different data sources, this increases the probability of sharing more and more factors.

4.7 Predictive Likelihood

Let us denote the test data from the $j$-th source by matrix $\mathbf{X}_j$ and the corresponding matrix factorization as $\mathbf{X}_j = \Phi (\mathbf{Z}_j \odot \mathbf{W}_j) + \mathbf{E}_j$, then the Monte Carlo approximation of predictive likelihood can be written as

$$p(\mathbf{X}_j \mid X_{il}, \Phi) \approx \frac{1}{LR} \sum_{r=1}^{R} \sum_{l=1}^{K^j} p\left(\mathbf{X}_j \mid \Phi^{[r]}, \mathbf{Z}_j^{[r]}, \mathbf{W}_j^{[r]}\right)$$

(32)

where $L$ is the number of training samples $\{\Phi^{[r]}\}$ and $R$ is the number of test samples $\{\mathbf{Z}_j^{[r]}, \mathbf{W}_j^{[r]}\}$.

5 Experiments

We carry out a variety of experiments to demonstrate the effectiveness of the proposed NHFA model. To illustrate the behavior of our model, we first perform experiments with a synthetic dataset, for which the true dimensionality of subspaces and other parameters are known. Through these experiments, we show the correct recovery of these parameters. Next, we demonstrate the usefulness of our model for two real-world tasks – text modeling and content based image retrieval. For both synthetic and real-world tasks, the priors for hyperparameters are chosen as the following: $\sigma_0 = 1, \sigma_j \sim \text{gamma}(1, 1)$. For NHFA-GGM, both the shape and the scale parameters of gamma priors for $\sigma_{\phi_j}$, $\sigma_{\eta_j}$ were set to 1. For NHFA-GGM, since we expect the results to be sparse we set the shape parameters of hyperparameters as $\alpha_0 = 1, a_{\eta_j} = 1$ whereas the scale parameters were sampled as: $b_0 \sim \text{gamma}(1, 1/\mu_0)$, $b_{\eta_j} \sim \text{gamma}(1, 1/\mu_{\eta_j})$, where $\mu_0 = \frac{1}{N} \sum_k \Phi^k$ and $\mu_{\eta_j} = \frac{1}{N} \sum_k \mathbf{W}_j^k$.

5.1 Experiments-I : Synthetic Data

We create a synthetic dataset similar to [8] so that we can show the benefits of our model vis-à-vis the model considered in [8]. For this dataset, we create twelve 100-D binary factors and distribute them across the two data sources termed as $\mathcal{D}_1$ and $\mathcal{D}_2$. The first four factors were used by the data points from $\mathcal{D}_1$, the next four factors were used by the data points from $\mathcal{D}_2$ while the last four factors were shared by data points across both $\mathcal{D}_1$ and $\mathcal{D}_2$. We generated the mixing configuration matrices $Z_1$ and $Z_2$ randomly with discrete support $\{0, 1\}$. The weight/coefficient matrices of the two sources, i.e. $W_1$ and $W_2$ were sampled from gamma distribution with shape and scale parameters 1 and 2 respectively. The using these parameters along with noise (with rate parameter 0.1 for both sources), the data for both $\mathcal{D}_1$ and $\mathcal{D}_2$ were generated from Poisson distributions according to the generative model described in section 3.

To verify the correctness of the inference, we run the slice sampler along with Gibbs updates as detailed in section 4 starting with the value of $K^j$ as one. We observe that the sampler converges to the true value of the number of factors (i.e. $K^j = 12$) in less than 100 iterations. However, we run the sampler longer to verify that the mode of the posterior over the number of factors remains at this true value. It can be seen from Figure 2 that the model correctly learns all the factors and their scores automatically.

To compare the computational efficiency of our slice sampler with the approximate Gibbs sampler of [8], we see that both methods need to sample $\{\beta, Z, \Phi, W\}$. Sampling $\Phi$ and $W$ remain identical in both cases. Only sampling of $\beta$ and $Z$ differ. In case of the approximate Gibbs sampler, $\beta$ conditioned on $\{m, 1, Z\}$ is drawn from a beta distribution whereas in the slice sampler, $\beta$ conditioned on $\{\rho, m, 1, Z\}$ uses adaptive rejection sampling (ARS) (see Eq (23)). Although not as fast as sampling from a beta distribution, ARS is quite efficient due to sampling in one dimensional space. When comparing sampling of $Z$, the slice sampler is more efficient than the approximate Gibbs sampler. The main difference lies in sampling the number of new factors.