An Infinite Factor Model Hierarchy Via a Noisy-Or Mechanism

Authored by:

Yoshua Bengio
Aaron C. Courville
Douglas Eck

Abstract

The Indian Buffet Process is a Bayesian nonparametric approach that models objects as arising from an infinite number of latent factors. Here we extend the latent factor model framework to two or more unbounded layers of latent factors. From a generative perspective, each layer defines a conditional emph{factorial} prior distribution over the binary latent variables of the layer below via a noisy-or mechanism. We explore the properties of the model with two empirical studies, one digit recognition task and one music tag data experiment.

1 Paper Body

The Indian Buffet Process (IBP) [5] is a Bayesian nonparametric approach that models objects as arising from an unbounded number of latent features. One of the main motivations for the IBP is the desire for a factorial representation of data, with each element of the data vector modelled independently, i.e. as a collection of factors rather than as monolithic wholes as assumed by other modeling paradigms such as mixture models. Consider music tag data collected through the internet service provider Last.fm. Users of the service label songs and artists with descriptive tags that collectively form a representation of an artist or song. These tags can then be used to organize playlists around certain themes, such as music from the 80?s. The top 8 tags for the popular band R ADIOHEAD are: alternative, rock, alternative rock, indie, electronic, britpop, british, and indie rock. The tags point to various facets of the band, for example that they are based in Britain, that they make use of electronic music and that their style of music is alternative and/or rock. These facets or features are not mutually exclusive properties but represent some set of distinct aspects of the band. Modeling such data with an IBP allows us to capture the latent factors that give rise to the tags, including inferring the number of factors characterizing the data. However the IBP assumes these latent features are independent across object instances. Yet in many situations, a more compact and/or accurate description of the data could be obtained if we were prepared to consider dependencies between latent factors. Despite there being a wealth of distinct factors that collectively describe an artist, it is clear that the co-occurrence of some features is more likely than others. For example, factors associated with the tag alternative are more likely to co-occur with those associated with the tag indie than those associated with tag classical. The main contribution of this work is to present a method for extending infinite latent factor models to two or more unbounded layers of factors, with upper-layer factors defining a factorial prior distribution over the binary factors of the layer below. In this framework, the upper-layer factors express correlations between lower-layer factors via a noisy-or mechanism. Thus our model may be interpreted as a Bayesian nonparametric version of the noisy-or network [6, 8]. In specifying the model and inference scheme, we make use of the recent stick-breaking construction of the IBP [10]. 1

For simplicity of presentation, we focus on a two-layer hierarchy, though the method extends readily to higher-order cases. We show how the complete model is amenable to efficient inference via a Gibbs sampling procedure and compare performance of our hierarchical method with the standard IBP construction on both a digit modeling task, and a music genre-tagging task.

2

Latent Factor Modeling

Consider a set of N objects or exemplars: x1:N = [x1 , x2 , . . . , xN]. We model the nth object with the distribution xn — zn,1:K , ? ? F (zn,1:K , ?1:K), with model parameters ?1:K = [?k]K k=1 (where ?k ? H indep. ?k) and feature variables zn,1:K = [znk]K k=1 which we take to be binary: znk ? {0, 1}. We denote the presence of feature k in example n as znk = 1 and its absence as znk = 0. Features present in an object are said to be active while absent features are inactive. Collectively, the features form a typically sparse binary N ? K feature matrix, which we denote as z1:N,1:K , or simply Z. For each feature k let ?k be the prior probability that the feature is active. The collection of K probabilities: ?1:K , are assumed to be mutually independent, and distributed according to a Beta(?/K, 1) prior. Summarizing the full model, we have (indep.?n, k): !? " ,1 xn — zn,1:K , ? ? F (zn,1:K , ?) znk — ?k ? Bernoulli(?k) ?k — ? ? Beta K According to the standard development of the IBP, we can marginalize over variables ?1:K and take the limit K ? ? to recover a distribution over an unbounded binary feature matrix Z. In the development of the inference scheme for our hierarchical model, we make use of an alternative characterization of the IBP: the IBP stick-breaking construction [10]. As with the stick-breaking construction of the Dirichlet process (DP), the IBP stick-breaking construction provides a direct characterization of the random latent feature probabilities via an unbounded sequence. Consider once again the finite latent factor model described above. Letting K ? ?, Z now possesses an unbounded number of columns with a corresponding unbounded set of random probabilities [?1 , ?2 , . . .]. Re-arranged in decreasing order: ?(1) ¿ ?(2) ¿ . . . , these factor probabilities can be # i.i.d expressed recursively as: ?(k) =

U(k) ?(k?1) = (l) U(l) , where U(k) ? Beta(?, 1).

3 A Hierarchy of Latent Features Via a Noisy-OR Mechanism

In this section we extend the infinite latent features framework to incorporate interactions between multiple layers of unbounded features. We begin by defining a finite version of the model before considering the limiting process. We consider here the simplest hierarchical latent factor model consisting of two layers of binary latent features: an upper-layer binary latent feature matrix Y with elements ynj , and a lower-layer binary latent feature matrix Z with elements znk . The probability distribution over the elements ynj is defined as previously in the limit construction of the IBP: ynj — ?j ? Bernoulli(?j), with ?j — ?? ? Beta(?? /J, 1). The lower binary variables znk are also defined as Bernoulli distributed random quantities: $ znk — yn,: , V:,k ? Bernoulli(1 ? (1 ? ynj Vjk)) indep.?n, k. (1) j

However, here the probability that znk = 1 is a function of the upper binary variables yn,: and the kth column of the weight matrix V , with probabilities Vjk ? [0, 1] connecting ynj to znk . The crux of the model is how ynj interacts with znk via a noisy-or mechanism defined in Eq. (1). The binary ynj modulates the involvement of the Vjk terms in the product, which in turn modulates P (znk = 1 — yn,: , V:,k). The noisy-or mechanism interacts positively in the sense that changing an element ynj from inactive to active can only increase P (znk = 1 — yn: , V:k), or leave it unchanged in the case where Vjk = 0. We interpret the active yn,: to be possible causes of the activation of the individual znk , ?k. Through the weight matrix V , every element of Yn,1:J is connected to every element of Zn,1:K , thus V is a random matrix of size J ? K. In the case of finite J and K, an i.i.d

obvious choice of prior for V is: Vjk ? Beta(a, b), ?j, k. However, looking ahead to the case where J ? ? and K ? ?, the prior over V will require some additional structure. Recently, [11] introduced the Hierarchical Beta Process (HBP) and elucidated the relationship between this and the Indian Buffet Process. We use a variant of the HBP to define a prior over V : ?k ? Beta(?? /K, 1) Vjk — ?k ? Beta(c?k , c(1 ? ?k) + 1) indep.?k, j, (2) 2

AM

Mj

AN Nk

Kmd H

Jmd

Ql

ynj Vjk

!k

xn N

?

i.i.d

Beta(?? , 1), ?j =

j $

Ql

l

k $
?
Beta(?? , 1), ?k =
Vjk ynj
? ?
znk
?
Beta(c?k , c(1 ? ?k) + 1) Bern(?j) $ Bern(1 ? (1 ? ynj Vjk)).
Rl znk
i.i.d
Rl
l
j

Figure 1: Left: A graphical representation of the 2-layer hierarchy of infinite binary factor models. Right: Summary of the hierarchical infinite noisy-or factor model in the stick-breaking parametrization.

where each column of V (indexed by k) is constrained to share a common prior. Structuring the prior this way allows us to maintain a well behaved prior over the Z matrix as we let K ? ?, grouping the values of Vjk across j while E[?k] ? 0. However beyond the region of very small ?k (0 ¡ ?k ¡¡ 1), we would like the weights Vjk to vary more independently. Thus we modify the model of [11] to include the +1 term to the prior over Vjk (in Eq. (2)) and we limit c ? 1. Fig. 1 shows a graphical representation of the complete 2-layer hierarchical noisy-or factor model, as J ? ? and K ? ?.

Finally, we augment the model with an additional random matrix A with multinomial elements Ank , assigning each instance of znk = 1 to an index j corresponding to the active upper-layer unit ynj responsible for causing the event. The probability that Ank = j is defined via a familiar stick-breaking scheme. By enforcing an (arbitrary) ordering over the indices j = [1, J], we can view the noisy-or mechanism defined in Eq. (1) as specifying, for each znk , an ordered series of binary trials (i.e. coin flips). For each znk , we proceed through the ordered set of elements, {Vjk , ynj }j=1,2,... , performing random trials. With probability yn,j ? Vj ? ,k , trial j ? is deemed a ?success? and we set znk = 1, Ank = j ? , and no further trials are conducted for {n, k, j ¿ j ? }. Conversely, with probability (1 ? ynj ? Vj ? k) the trial is deemed a ?failure? and we move on to trial j ? + 1. Since all trials j associated with inactive upper-layer features are failures with probability one (because ynj = 0), we need only consider the trials for which ynj = 1. If, for a given znk , all trials j for which ynj = 1 (active) are failures, then we set znk = 0 with probability one. The probability associated with the event znk = 0 is therefore given by the product of the failure #J probabilities for each of the J trials: P (znk = 0 — yn,: , V:,k) = j=1 (1 ? ynj Vjk), and with P (znk = 1 — yn,: , V:,k) = 1 ? P (znk = 0 — yn,: , V:,k), we arrive at the noisy-or mechanism given in Eq. (1). This process is similar to the sampling process associated with the Dirichlet process stick-breaking construction [7]. Indeed, the process described above specifies a stick-breaking construction of a generalized Dirichlet distribution [1] over the multinomial probabilities corresponding to the Ank . The generalized Dirichlet distribution defined in this way has the important property that it is conjugate to multinomial sampling. With the generative process specified as above, we can define the posterior distribution over the weights V given the assignment matrix A and the latent feature matrix Y . Let Mjk = %N was a success for z:,k (i.e. the number of n=1 I(Ank = j) be the number of times that the jth trial %N times ynj caused the activation of znk) and let Njk = n=1 ynj I(Ank ¿ j), that is the number of times that the j-th trial was a failure for znk despite ynj being active. Finally, let us also denote the %N number of times y:,j is active: Nj = n=1 ynj . Given these quantities, the posterior distributions for the model parameters ?j and Vjk are given by: ?j — Y ? Beta(?? /J + Nj , 1 + N ? Nj) Vjk — Y, A ? Beta(c?k + Mjk , c(1 ? ?k) + Njk + 1)

(3) (4)

These conjugate relationships are exploited in the Gibbs sampling procedure described in Sect. 4. By integrating out Vjk , we can recover (up to a constant) the posterior distribution over ?k : 3

? /K?1

p(?k — A:,k) ? ?k ?

J $?(c?k + Mjk) ?(c(1 ? ?k) + Njk + 1) ?(c?k) ?(c(1 ? ?k) + 1) j=1

(5)

One property of the marginal likelihood is that wholly inactive elements of Y , which we denote as y:,j " = 0, do not impact the likelihood as Nj " ,k = 0, Mj " ,k = 0. This becomes particularly important as we let J ? ?.

Having defined the finite model, it remains to take the limit as both K ? ? and J ? ?. Taking the limit of J ? ? is relatively straightforward as the upper-layer factor model naturally tends to an IBP: Y ? IBP, and its involvement in the remainder of the model is limited to the set of active elements of Y , which remains finite for finite datasets. In taking K ? ?, the distribution over the unbounded ?k converges to that of the IBP, while the conditional distribution over the noisy-or weights Vjk remain simple beta distributions given the corresponding ?k (as in Eq. (4)).

4 Inference In this section, we describe an inference strategy to draw samples from the model posterior. The algorithm is based jointly on the blocked Gibbs sampling strategy for truncated Dirichlet distributions [7] and on the IBP semi-ordered slice sampler [10], which we employ at each layer of the hierarchy. Because both algorithms are based on the strategy of directly sampling an instantiation of the model parameters, their use together permits us to define an efficient extended blocked Gibbs sampler over the entire model without approximation. To facilitate our description of the semi-ordered slice sampler, we separate ?1:? into two subsets: + o + ?+ 1:J + and ?1:? , where ?1:J + are the probabilities associated with the set of J active upper-layer + + factors Y (those that appear at least once in the dataset, i.e. ?i : yij " = 1, 1 ? j $? J +) and ?o1:? are associated with the unbounded set of inactive features Y o (those not appearing in the dataset). + o + Similarly, we separate ?1:? into ?1:K and inactive + and ?1:? , and Z into corresponding active Z o + Z where K is the number of active lower-layer factors. 4.1

Semi-ordered slice sampling of the upper-layer IBP

+ The IBP semi-ordered slice sampler maintains an unordered set of active y1:N,1:J + with correspond+ ing ?1:J + and V1:J + ,1:K , while exploiting the IBP stick-breaking construction to sample from the distribution of ordered inactive features, up to an adaptively chosen truncation level controlled by an auxiliary slice variable sy .

Sample sy . The uniformly distributed auxiliary slice variables, sy controls the truncation level of the upper-layer IBP, where ?? is defined as the smallest probability ? corresponding to an active feature: & ' + sy — Y, ?1:? ? Uniform(0, ??), ?? = min 1, min ? (6) j" . " + 1?j ?J

As discussed in [10], the joint distribution is given by p(sy , ?1:? , Y) = p(Y, ?1:?) ? p(sy — Y, ?1:?), where marginalizing over sy preserves the original distribution over Y and ?1:? . However, given sy , the conditional distribution p(ynj " = 1 — Z, sy , ?1:?) = 0 for all n, j $ such that ?j " ¡ sy . This is the crux of the slice sampling approach: Each sample sy adaptively truncates the model, with ?1:J ¿ sy . Yet by marginalizing over sy , we can recover samples from the original non-truncated distribution p(Y, ?1:?) without approximation. Sample ?o1:J o . For the inactive features, we use adaptive rejection sampling (ARS) [4] to sequentially draw an ordered set of J o posterior feature probabilities from the distribution: * (N) 1 o n o o o (1 ? ?j) ? (?oj)?? ?1 (1 ? ?oj)N I(0 ? ?oj ? ?oj?1), p(?j — ?j?1 , y:,?j = 0) ? exp ?? n n=1

until ?oJ o +1 ¡ sy . The above expression arises from using the IBP stick-breaking construction to marginalize over the inactive elements of ?: [10]. For each of the J o inactive features drawn, the 4

o o corresponding features y1:N,1:J o are initialized to zero and the corresponding weight V1:J o ,1:K are sampled from their prior in Eq. (2). With the probabilities for both the active and a truncated set of inactive features sampled, the set of features are re-integrated into a set of J = J + + J o features + + o o Y = [y1:N,1:J + , y1:N,1:J o] with probabilities ?1:J = [?1:J + , ?1:J o], and corresponding weights + T T o T V = [(V1:J + ,1:K) , (V1:J o ,1:K)].

Sample Y . Given the upper-layer feature probabilities ?1:J , weight matrix V , and the lower-layer binary feature values znk , we update each ynj as follows: p(ynj = 1 — ?j , zn,: , ??) ?

K ?j $ p(znk — ynj = 1, yn,?j , V:,k) ??

(7)

k=1

The denominator ?? is subject to change if changing ynj induces a change in ?? (as defined in Eq. (6)); yn,?j represents all elements yn,1:J except ynj The # conditional probability of the lower-layer binary variables is given by: p(znk — yn,: , V:,k) = (1 ? j (1 ? ynj Vjk)).

+ Sample ?+ with prob1:J + . Once again we separate Y and ?1:? into a set of active features: Y + o o abilities ?1:J + ; and a set of inactive features Y with ?1:? . The inactive set is discarded while the + + active set of ?+ 1:J + are resampled from the posterior distribution: ?j — y:,j ? Beta(Nj , 1+N ?Nj). At this point we also separate the lower-layer factors into an active set of

K+ factors Z + with cor+ + + responding ?1:K + , V1:J + ,1:K + and data likelihood parameters ? ; and a discarded inactive set.

4.2

Semi-ordered slice sampling of the lower-layer factor model

Sampling the variables of the lower-layer IFM model proceeds analogously to the upper-layer IBP. However the presence of the hierarchical relationship between the ?k and the V:,k (as defined in Eqs. (3) and (4)) does require some additional attention. We proceed by making use of the marginal distribution over the assignment probabilities to define a second auxiliary slice variable, sz . Sample sz . The auxiliary slice variable is sampled according to the following, where ? ? is defined as the smallest probability corresponding to an active feature: & ' + ? ? sz — Z, ?1:? ? Uniform(0, ?), ? = min 1, min ? " . " + k 1?k ?K

o Sample ?1:K Given sz and Y , the random probabilities over the inactive lower-layer binary o. o features, ?1:? , are sampled sequentially to draw a set of K o feature probabilities, until ?K o +1 ¡ sz . The samples are drawn according to the distribution:

p(?ko
—
o o ?k?1 , Y + , z:,?k
= 0)
?
I (0 ?
?ko
exp ??
?
o ?k?1) (?ko)?? ?1
J Y
j=1
?(c) ?(c + Nj)
J Y ?(c(1 ? ?ko) + Nj) ?(c(1 ? ?ko)) j=1
N1 +???+NJ
X i=0
!
i X 1 wi c (1 ? ?ko)l l i
l=1
?
!
?
(8)

Eq. (8) arises from the stick-breaking construction of the IBP and from the expression for o P (z:,¿k = 0 — ?ko , Y +) derived in the supplementary material [2]. Here we simply note that the wi are weights derived from the expansion of a product of terms involving unsigned Stirling numbers of the first kind. The distribution over the ordered inactive features is log-concave in log ?k , and is therefore amenable to efficient sample via adaptive rejection sampling (as was done in sampling ?o1:J o). Each of the K o inactive features are initialized to zero for every data object, Z o = 0, while the corresponding V o and likelihood parameters ?o are drawn from their priors. Once the ?1:K o are drawn, both the active and inactive features of the lower-layer are re-integrated into the set of + o K = K + + K o features Z = [Z + , Z o] with probabilities ?1:K = [?1:K + , ?1:K o] and corresponding + o + o weight matrix V = [V1:J + ,1:K + , V1:J + ,1:K o] and parameters ? = [? , ?]. 5

Sample Z. Given Y + and V we use Eq. (1) to specify the prior over z1:N,1:K ? . Then, conditional on this prior, the data X and parameters ?, we sample sequentially for each znk : p(znk

0 1 J+ Y 1 + + — yn,: , V:,k , zn,?k , ?, ? ?) = ? @1 ? (1 ? ynj Vjk)A f (xn — zn,: , ?), ? j=1

where f (xn — zn,: , ?) is the likelihood function for the nth data object.

+ Sample A. Given znk , yn,: and V:,k , we draw the multinomial variable Ank to assign responsibil+ ity, in the event zik = 1, to one of the upper-layer features ynj , + , V:,k) = Vjk p(Ank = j — znk = 1, yn,:

”j?1 Y i=1

#

(9)

#j ? ?1

+ (1 ? yni Vik) to ensure

+ (1 ? yni Vik) ,

+ $? ? + and if yn,j — znk = 1, yn,: , V:,k) = ” = 0, ?j ¿ j , then p(Ank = j normalization of the distribution. If znk = 0, then P (Ank = ?) = 1.

i=1

+ Sample V and ?1:K Conditional on Y + , Z and A, the weights V are resampled from Eq. (4), +. following the blocked Gibbs sampling procedure of [7]. Given the assignments A, the posterior of ?k+ is given (up to a constant) by Eq. (5). This distribution is log concave in ?k+ , therefore we can once again use ARS to draw samples of the posterior of ?k+ , 1 ? k ? K + .

5 Experiments In this section, we present two experiments to highlight the properties and capabilities of our hierarchical infinite factor model. Our goal is to assess, in these two cases, the impact of including an additional modeling layer. To this end, and in each experiment, we compare our hierarchical model to the equivalent IBP model. In each case, hyperparameters are specified with respect to the IBP (using cross-validation by evaluating the likelihood of a holdout set) and held fixed for the hierarchical factor model. Finally all hyperparameters of the hierarchical model that were not marginalized out were held constant over all experiments, in particular c = 1 and ?? = 1. 5.1

Experiment I: Digits

In this experiment we took examples of images of hand-written digits from the MNIST dataset. Following [10], the dataset consisted of 1000 examples of images of the digit 3 where the handwritten digit images are first preprocessed by projecting onto the first 64 PCA components. To model MNIST digits, we augment both the IBP and the hierarchical model with a matrix G of the same size as Z and with i.i.d. zero mean and unit variance elements. Each data object, xn is modeled 2 as: xn — Z, G, ?, ?x2 ? N ((zn,: + gn,:)?, ?X I) where + is the Hadamard (element-wise) product. The inclusion of G introduces an additional step to our Gibbs sampling procedure, however the rest of the hierarchical infinity factor model is as described in Sect. 3. In order to assess the success of our hierarchical IFM in capturing higher-order factors present in the MNIST data, we consider a de-noising task. Random noise (std=0.5) was added to a post-processed test set and the models were evaluated in its ability to recover the noise-free version of a set of 500 examples not used in training. Fig. 2 (a) presents a comparison of the log likelihood of the (noise-free) test-set for both the hierarchical model and the IBP model. The figure shows that the 2-layer noisy-or model gives significantly more likelihood to the pre-corrupted data than the IBP, indicating that the noisy-or model was able to learn useful higher-order structure from MNIST data. One of the potential benefits of the style of model we propose here is that there is the opportunity for latent factors at one layer to share features at a lower layer. Fig. 2 illustrates the conditional mode of the random weight matrix V (conditional on a sample of the other variables) and shows that there is significant sharing of lowlevel features by the higher-layer factors. Fig. 2 (d)-(e) compare the features (sampled rows of the ? matrix) learned by both the IBP and by the hierarchical noisy-or factor model. Interestingly, the sampled features learned in the hierarchical model appear to be slightly more spatially localized and sparse. Fig. 2 (f)-(i) illustrates some of the marginals that arise from the Gibbs sampling inference process. Interestingly, the IBP model infers a greater number of latent factors that did the 2-layer 6

5000

?2.5

2000 1000

?3

0.5

1

150 100 50

140 160 180 num. active features

0

200

0

10 20 30 num. active features

1.5 2 2.5 3 3.5 A MCMC iterations

4

4.5 x10 4

40

G

8000

600 500

6000

num. of objects

2?layer Noisy?Or model

?4 ?4.5 0

200
F
IBP
?3.5
IBP Hierarchical
250
3000
0 120
num. MCMC iterations
log likelihood
?2
300 IBP Hierarchical
4000
num. of objects
num. MCMC iterations
4 ?1.5 x 10
4000
2000
400 300 200 100
0 20
25 30 35 num. active features
H
40
0
1
2
3 4 num. active features
5
I
0.9 5
0.8 0.7
10
0.6 0.5
15
0.4 0.3
20
0.2 0.1
25 10 20
30 40 50 60 70 80 90 100
B
0
C
D
E

Figure 2: (a) The log likelihood of a de-noised testset. Corrupted (with 0.5-std Gaussian noise) versions of

test examples were provided to the factor models and the likelihood of the noise-free testset was evaluated for both an IBP-based model as well as for the 2-layer noisy-or model. The two layer model shown substantial improvement in log likelihood. (b) Reconstruction of noisy examples. The top row shows the original values for a collection of digits. The second row shows their corrupted versions; while the third and fourth row show the reconstructions for the IBP-based model and the 2 layer noisy-or respectively. (c) A subset of the V matrix. The rows of V are indexed by j while the columns of V are indexed by k. The vertical striping pattern is evidence of significant sharing of lower-layer features among the upper-layer factors. (d)-(e) The most frequent 64 features (rows of the ? matrix) for (d) the IBP and for (e) the 2-layer infinite noisy-or factor model. (f) A comparison of the distributions of the number of active elements between the IBP and the noisy-or model. (g) A comparison of the number of active (lower-layer) factors possessed by an object between the IBP and the hierarchical model. (h) the distribution of upper-layer active factors and (i) the number of active factors found in an object.

noisy-or model (at the first layer). However, the distribution over factors active for each data object is nearly identical. This suggests the possibility that the IBP is maintaining specialized factors that possibly represent a superposition of frequently co-occurring factors that the noisy-or model has captured more compactly. 5.2 Experiment II: Music Tags Returning to our motivating example from the introduction, we extracted tags and tag frequencies from the social music website Last.fm using the Audioscrobbler web service. The data is in the form of counts1 of tag assignment for each artist. Our goal in modeling this data is to reduce this often noisy collection of tags to a sparse representation for each artist. We will adopt a different approach to the standard Latent Dirichlet Allocation (LDA) document processing strategy of modeling the document ? or in this case tag collection ? as having been generated from a mixture of tag multinomials. We wish to distinguish between an artist that everyone agrees is both country and rock versus an artist that people are divided whether they are rock or country. To this end, we can again make use of the conjugate noisy-or model to model the count data in the form of binomial probabilities, i.e. to the model defined in Sect. 3, we add the random weights i.i.d W #kt ? Beta(a, b), ?k.t connecting Z to the data X via the distribution: Xnt ? Binomial(1 ? k (1 ? znk W), C) where C is the limit on the number of possible counts achievable. This would correspond to the number of people who ever contributed a tag to that artist. In the case of the Last.fm data C = 100. Maintaining conjugacy over W will require us to add an assignment parameter 1

The publicly available data is normalized to maximum value 100.

7

800

300

2000

600 500

200 150 100

200

1500 num. objects
400
MCMC iterations
num. objects
MCMC iterations
250 600
1000
100 120 140 num. active features
A
160
0
300 200
500
50 0 80
400
100 0
2
4 6 num. active features
0 20
8
30
40 50 60 num. active features
70
0
0
1
2 3 num. active features
C
B
4
D

Figure 3: The distribution of active features for the noisy-or model at the (a) lower-layer and (c) the upperlayer. The distribution over active features per data object for the (b) upper-layer and (d) lower-layer.

Bnt whose role is analogous to Ank . With the model thus specified, we present a dataset of 1000 artists with a vocabulary size of 100 tags representing a total of 312134 counts. Fig. 3 shows the result running the Gibbs sampler for 10000 iterations. As the figure shows, both layers are quite sparse. Generally, most of the features learned in the first layer are dominated by one to three tags. Most features at the second layer cover a broader range of tags. The two most probable factors to emerge at the upper layer are associated with the tags (in order of probability): 1. electronic, electronica, chillout, ambient, experimental 2. pop, rock, 80s, dance, 90s The ability of the 2-layer noisy-or model to capture higher-order structure in the tag data was again assessed though a comparison to the standard IBP using the noisy-or observation model above. The model was also compared against a more standard latent factor model with the latent representation ?nk modeling the data through a generalized linear model: Xnt ? Binomial(Logistic(?n,: O:,t), C), where the function Logistic(.) is the logistic sigmoid link function and the latent representation ?nk ? N (0, ??) are normally distributed. In this case, inference is performed via a MetropolisHastings MCMC method that mixes readily. The test data was missing 90% of the tags and the models were evaluated by their success in imputing the missing data from the 10% that remained. Here again, the 2-Layer Noisy-Or model achieved superior performance, as measured by the marginal log likelihood on a hold out set of 600 artist-tag collections. Interestingly both sparse models ? the IBP and the noisy-or model ? dramatically out performed the generalized latent linear model. Method Gen. latent linear model (Best Dim = 30) IBP 2-Layer Noisy-Or IFM

6

NLL 8.7781e05 ? 0.02e05 5.638e05 ? 0.001e05 5.542e05 ? 0.001e05

Discussion

We have defined a noisy-or mechanism that allows one infinite factor model to act as a prior for another infinite factor model. The model permits high-order structure to be captured in a factor model framework while maintaining an efficient sampling algorithm. The model presented here is similar in spirit to the hierarchical Beta process, [11] in the sense that both models define a hierarchy of unbounded latent factor models. However, while the hierarchical Beta process can be seen as a way to group objects in the data-set with similar features, our model provides a way to group features that frequently co-occur in the data-set. It is perhaps more similar in spirit to the work of [9] who also sought a means of associating latent factors in an IBP, however their work does not act directly on the unbounded binary factors as ours does. Recently the question of how to define a hierarchical factor model to induce correlations between lower-layer factors was addressed by [3] with their IBPIBP model. However, unlike our model, where the dependencies induced by the upper-layer factors via an noisy-or mechanism, the IBP-IBP model models correlations via an AND construct through the interaction of binary factors. Acknowledgments The authors acknowledge the support of NSERC and the Canada Research Chairs program. We also thank Last.fm for making the tag data publicly available and Paul Lamere for his help in processing the tag data. 8

2 References

[1] Robert J. Connor and James E. Mosimann. Concepts of independence for proportions with a generalization of the Dirichlet distribution. Journal of the American Statistical Association, 64(325):194?206, 1969. [2] Aaron C. Courvile, Douglas Eck, and Yoshua Bengio. An infinite factor model hierarchy via a noisy-or mechanism: Supplemental material. Supplement to the NIPS paper. [3] Finale Doshi-Velez and Zoubin Ghahramni. Correlated nonparametric latent feature models. In Proceedings of the 25 th Conference on Uncertainty in Artificial Intelligence, 2009. [4] W. R. Gilks and P. Wild. Adaptive rejec-

tion sampling for Gibbs sampling. Applied Statistics, 41(2):337?348, 1992. [5] Tom Griffiths and Zoubin Ghahramani. Infinite latent feature models and the indian buffet process. In Advances in Neural Information Processing Systems 18, Cambridge, MA, 2006. MIT Press. [6] Max Henrion. Practical issues in constructing a bayes? belief network. In Proceedings of the Proceedings of the Third Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-87), page 132?139, New York, NY, 1987. Elsevier Science. [7] Hemant Ishwaran and Lancelot F. James. Gibbs sampling methods for stick-breaking priors. American Statistical Association, 96(453):161?173, 2001. [8] Michael Kearns and Yishay Mansour. Exact inference of hidden structure from sample data in noisy-or networks. In Proceedings of the 14 th Conference on Uncertainty in Artificial Intelligence, pages 304?310, 1998. [9] Piyush Rai and Hal Daum?e III. The infinite hierarchical factor regression model. In Daphne Koller, Dale Schuurmans, Yoshua Bengio, and L?eon Bottou, editors, Advances in Neural Information Processing Systems 21, 2009. [10] Yee Whye Teh, Dilan G?or?ur, and Zoubin Ghahramani. Stick-breaking construction for the indian buffet process. In Proceedings of the Eleventh International Conference on Artifical Intelligence and Statistics (AISTAT 2007)., 2007. [11] Romain Thibaux and Michael I. Jordan. Hierarchical beta process and the indian buffet process. In Proceedings of the Eleventh International Conference on Artifical Intelligence and Statistics (AISTAT 2007)., 2007.

9