Multi-HDP: A Non Parametric Bayesian Model for Tensor Factorization

Ian Porteous and **Evgeniy Bart** and **Max Welling**
Dept. of Computer Science
UC Irvine
Irvine, CA 92697
{iporteou,welling}@ics.uci.edu, bart@caltech.edu

Abstract

Matrix factorization algorithms are frequently used in the machine learning community to find low dimensional representations of data. We introduce a novel generative Bayesian probabilistic model for unsupervised matrix and tensor factorization. The model consists of several interacting LDA models, one for each modality. We describe an efficient collapsed Gibbs sampler for inference. We also derive the nonparametric form of the model where interacting LDA models are replaced with interacting HDP models. Experiments demonstrate that the model is useful for prediction of missing data with two or more modalities as well as learning the latent structure in the data.

Introduction

Matrix factorization refers to the problem of representing a given input matrix as the product of two lower rank matrices. In this paper we focus on Matrix factorization for the purpose of learning the latent or hidden structure of the data and prediction of missing elements of the input matrix. Collaborative filtering is an example of a problem where Matrix factorization has been applied to achieve these goals. In Collaborative filtering we assume a set of N users 1,...,N, a set of M items 1,...,M, and a set of V discrete rating values 1,...,V. For example we might have N movie patrons (users) and M movies, for which we have sparse data on how some users rated some movies. We can represent this data as a matrix X with N rows and M columns most of which are unknown. Our goal may be to predict the missing values of X, (i.e. how much a user will enjoy a movie they have not seen), or we may want to find groups of users with similar tastes or movies with similar themes. If we factor X into the product of UWV^T we can generate values for the unknown elements of the input data. Furthermore, if each row of U is a reduced dimensionality description of a user, and each row of V is a reduced dimensionality representation of a movie, we may learn hidden structure in the data by examining the rows of U and V.

Non probabilistic models for Matrix factorization such as NMF and SVD have been applied to collaborative filtering and work well for predicting missing values, or classification based on the low rank representation. However, generative probabilistic models have a number of advantages that these methods lack such as predictive *distributions* from which confidence intervals can be inferred, the possibility to include prior knowledge into the generative process and a principled framework to select model structure. Moreover, the inferred latent structure is directly tied to the generative process and therefore often easy to interpret.

Probabilistic Models such as PLSA and their Bayesian Extensions such as LDA (Blei, Ng, & Jordan 2003), have been proposed as text models in the bag-of-words representation. These models have the benefit of learning a latent structure for the data and providing a probability distribution for the missing data predictions. Extensions of PLSA (Hofmann 2004) and LDA (Marlin 2003) to the collaborative filtering case have also been proposed. However, using movie ratings as an example, these models treat users and movies differently. In particular, they discover hidden structure for users but not for movies.

Our model fits into the class of models known as "blockmodels" (Airoldi *et al.* 2008). The parametric, two modality version of our model, Bi-LDA, is similar to (Airoldi *et al.* 2008) with multinomial instead of Bernoulli mixture components. The extension to nonparametric methods has also been considered before with other blockmodels (Kemp *et al.* 2006) and (Mansinghka *et al.* 2006). However these nonparametric block models have objects (users, movies) belonging to only a single group. In our model objects can belong to several groups (i.e. one for each rating).

(Meeds *et al.* 2007) is another Bayesian matrix factorization method which does treat two modalities symmetrically but assumes a different structure for the data. They use a 'factorial learning' model that encourages a representation where an object is described by a diverse set of features. Our model on the other hand encourages a succinct representation where an object belongs to few groups akin to soft multi-modality clustering. Specifically, in our model we have that the more an object belongs to one group, the less it belongs to other groups. Another essential difference is that our model combines factors/topics in the probability domain, while the model of (Meeds *et al.* 2007) combines factors in the log-probability domain.

Bi-LDA.

Bi-LDA consists of two interacting LDA models, one LDA model for the movie patrons (users) and one for the movies.

LDA is a Bayesian generative model originally proposed for text modeling where documents are represented as a vector of word counts. LDA has the following generative model for words and documents. For each each of N words in document d, sample a topic $z_n \sim \text{Multinomial}(\pi_d)$ then sample a word $w_n \sim p(w_n|z_n, \phi_{z_n})$. π and ϕ are given Dirichlet priors to complete the model.

We can naively apply LDA to movie ratings by ignoring users and treating movies as documents and ratings as words. In the same way we could ignore movies and treat users as documents and movies as words. However, unlike in documents where LDA successfully discovers topics, because we only have a small vocabulary of ratings, we are unlikely to discover interesting groups of movie or user rating patterns.

Bi-LDA treats users and movies symmetrically, learning both groups of users and groups of movies. In Bi-LDA ratings are generated in the following way:

1. Choose $J \times K$ distributions over ratings $\phi^m_{jk} \sim \text{Dir}(\beta)$
2. Choose a distribution over K user groups for each user $\pi^{user}_u \sim \text{Dir}(\alpha^{user})$
3. Choose a distribution over J movie groups for each movie $\pi^{movie}_m \sim \text{Dir}(\alpha^{movie})$
4. For each movie-user pair (mu)
   (a) Choose a user group $z^{user}_{mu} \sim \text{Multinomial}(\pi^{user}_u)$
   (b) Choose a movie group $z^{movie}_{mu} \sim \text{Multinomial}(\pi^{movie}_m)$
   (c) Choose a rating $r_{mu} \sim p(r_{mu}|z^{user}_{mu}, z^{movie}_{mu}, \phi^m_{z^{movie}_{mu}, z^{user}_{mu}})$

Φ_{jk} is the distribution over values $1...V$ for cluster j, k and has a Dirichlet prior with parameter β. π^{movies}_m and π^{users}_u are distributions over movie and user groups with Dirchlet priors using parameters $\alpha^{movies}, \alpha^{users}$ respectively. We also introduce indicator variables z^{movies}_{um} and z^{users}_{um} for each modality representing the group chosen for each movie-user pair. So z^{movie}_{um} represents the group that movie m picked for the rating given by user u. X_{mu} is the rating observed for user u and movie m. To reduce clutter m, u will be used instead of $movie, user$ in the superscript to indicate the modality ($\pi^u_u \equiv \pi^{user}_u$)

Putting everything together we obtain the joint distribution for the Bi-LDA model.

$$P(X, \mathbf{z}^m, \mathbf{z}^u, \mathbf{\Phi}, \boldsymbol{\pi}^m, \boldsymbol{\pi}^u) = \quad (1)$$
$$P(X|\mathbf{z}^m, \mathbf{z}^u, \mathbf{\Phi}) P(\mathbf{\Phi}|\boldsymbol{\beta})\ P(\mathbf{z}^m|\boldsymbol{\pi}^m) \times$$
$$P(\mathbf{z}^u|\boldsymbol{\pi}^u)\ P(\boldsymbol{\pi}^m|\boldsymbol{\alpha}^m) P(\boldsymbol{\pi}^u|\boldsymbol{\alpha}^u)$$

Where bold variables represent the collection of individual variables, $\mathbf{z}^m \equiv \{z^m_{11}, z^m_{12}, .., z^m_{UM}\}$ (U is the number of users, and M is the number of movies). Now our goal is to perform inference on (1) to learn the posterior distribution of $\{\mathbf{z}^m, \mathbf{z}^u\}$ and other variables of interest given the input data.

Gibbs Sampling

Although exact inference is intractable in Bi-LDA as it is in LDA, we can derive an efficient collapsed Gibbs sampler analogous to the one derived for LDA (Griffiths & Steyvers 2002). If we then run the Gibbs sampler long enough, we will produce samples from the correct posterior distribution which we can use for inference. The basic idea is to analytically marginalize out all the conjugate distributions $\mathbf{\Phi}, \boldsymbol{\pi}^u, \boldsymbol{\pi}^m$ in (1) and obtain an expression for the joint probability $P(X, \mathbf{z}^m, \mathbf{z}^u)$. From this joint probability one can compute the conditional distributions necessary for Gibbs sampling.

We will need the following counts: $N_{uk} = \sum_m \mathbb{I}[z^u_{um} = k]$, $N_{mj} = \sum_u \mathbb{I}[z^m_{um} = j]$ and $N^v_{jk} = \sum_{um} \mathbb{I}[X_{mu} = v]\mathbb{I}[z^m_{mu} = j]\mathbb{I}[z^u_{mu} = k]$. Where N^v_{jk} represents the number of entries in the entire data-array that are assigned to user factor k and movie factor j, for which the rating has the value v. N_{uk} is the number of ratings for user u assigned to factor k. N_{mj} is the number of ratings for movie m assigned to factor j. Also $N_{jk} = \sum_v N^v_{jk}$. We will use the superscript $^{\neg(um)}$ to denote that data-entry (um) is subtracted from the counts. In terms of these we find the following conditional distribution for movie indicator variables $\mathbf{z}^m$,

$$P(z^m_{um} = j|\mathbf{z} \backslash z^m_{um}, X) \propto$$
$$\left(\frac{N^{v,\neg(um)}_{jk} + \beta^v}{N^{\neg(um)}_{jk} + \beta}\right)\left(N^{\neg(um)}_{mj} + \frac{\alpha^m}{J}\right) \quad (2)$$

where J is the number of movie factors, $X_{um} = v$, $z^u_{um} = k$ and $\beta = \sum_v \beta^v$. The conditional distribution is the same for the user indicator variables with the role of user and movie reversed. The Gibbs sampler thus cycles through the indicator variables $z^m_{um}, z^u_{um} \forall u, m$. Ratings are conditionally independent given $\mathbf{\Phi}$ so we can marginalize out unobserved ratings from the model. The Gibbs sampler therefore only scans over the observed entries in the matrix X.

Prediction

In a typical collaborative filtering scenario the product-consumer rating matrix is very sparse. In the movie user example, for each user the data will contain ratings for only a small fraction of the movies. One task is to estimate the ratings for movies the user has not seen. To make predictions, once converged we collect samples from the Markov chain. Given a single sample from the chain for $\mathbf{z}$ we start by calculating a mean estimate for $\Phi_{jk}, \pi^m_m, \pi^u_u$

$$\Phi_{jk}[v] = \frac{N^v_{jk} + \beta^v}{N_{jk} + \beta} \qquad \pi^m_m[j] = \frac{N_{mj} + \alpha^m/J}{\sum_j N_{mj} + \alpha^m}$$
$$\pi^u_u[k] = \frac{N_{uk} + \alpha^u/K}{\sum_k N_{uk} + \alpha^u} \quad (3)$$

Then we calculate the expected value of X_{um}

$$E(X_{mu}) = \sum_{j,k}\left(\sum_v v\ \Phi_{jk}[v])\right)\pi^m_m[j]\pi^u_u[k].$$

To see the connection with matrix factorization define $\mathbf{\Phi}_{jk} = \sum_v v\Phi_{jk}[v]$ as the core matrix, $U = \boldsymbol{\pi}^u$ and

$V = \boldsymbol{\pi}^m$. Then $E(X_{mu})$ is given by the matrix multiplication

$$X \sim U\boldsymbol{\Phi}V^t$$

It may also be useful to estimate how confident we should be in the predicted value X_{um}. The distribution of X_{um} is multinomial (4),

$$P(X_{mu} = v|\Phi_{jk}, \pi^m_m, \pi^u_u) = \sum_{j,k} \Phi_{jk}[v]\pi^m_m[j]\pi^u_u[k] \quad (4)$$

so it's variance can easily be calculated.

In the previous calculations for the predicted value of X_{mu}, we used just a single sample. We would like to take many samples from the chain and use them to find estimates of $\Phi_{jk}, \pi^m_m, \pi^u_u$ marginalized over $\mathbf{z}$. However, there may be multiple equivalent modes of the distribution where the assignment variables $\mathbf{z}$ have different values which would cause the average of equations (3) calculated over samples from different modes to be incorrect. Instead we can marginalize over $\mathbf{z}$ implicitly by averaging over predictions from many Gibbs iterations. Call $\boldsymbol{\Phi}_s$, $\boldsymbol{\pi}^m_s$ and $\boldsymbol{\pi}^u_s$ the mean estimates of $\boldsymbol{\Phi}$, $\boldsymbol{\pi}^m$, $\boldsymbol{\pi}^u$ based on a single sample $\mathbf{z}_s$ We initialize $X = 0$. After scan number "s" through the data-matrix we update our average as follows,

$$\bar{X} \rightarrow \frac{s-1}{s}\bar{X} + \frac{1}{s}\boldsymbol{\Phi}_s \prod_m \boldsymbol{\pi}^m_s$$

where we suppressed summations to avoid clutter. We find this online averaging results in a significant improvement in prediction accuracy.

Bi-LDA Experiments

Netflix

In these experiments we evaluate Bi-LDA on the Netflix movie ratings dataset. The Netflix dataset consists of user-movie ratings provided by the Netflix corporation as part of a data-mining contest. The input data is movie-user rating pairs where the rating can take integer values 1 to 5 ($X_{movie,user} \in 1, 2, 3, 4, 5$). We use the Netflix training dataset of approximately 100 million ratings for our training data and 10% of the Netflix probe dataset for our held out test data. We use Root Mean Squared Error (RMSE) to measure the prediction performance on the test data. Using 50 user groups and 500 movie groups we ran the Gibbs sampler for 1600 epochs (an epoch is resampling all the variables once). The parameters $\alpha^m, \alpha^u, \beta$ were set to 1. The resulting RMSE on the test data set was .933.

Methods focused on the best RMSE for the Netflix dataset have achieved a better RMSE on the probe set, such as .9089 using an ensemble of methods (Takacs *et al.* 2007). However, Bi-LDA offers the advantage of a distribution for the predictions. In figure 1(top) we show a histogram of the average RMSE values binned and ordered by variance. When we compare this with the percentage of predictions falling in those variance bins, we may conclude that we can classify 90% of the data with an RMSE of .9 and 40% with an RMSE of .75. Ordering the prediction by variance would for example, enable a company to target the most interested customers.

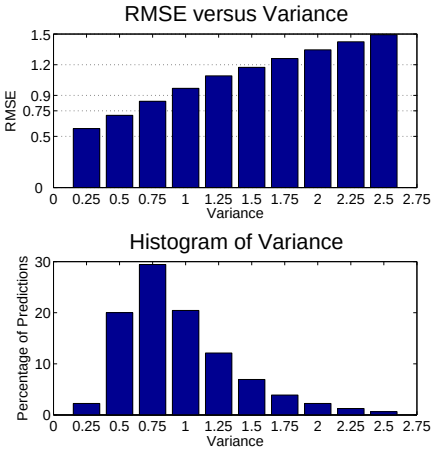

Figure 1: Top: The RMSE of predictions binned by their predicted variance. The first bar is predictions with variance less then .25, the second bar is the RMSE of predictions with variance between .25 and .5, and so on by increments of .25. Bottom: The percentage of predictions that fall within each bin.

Multi-LDA.

In this section we extend the model to more then two modalities. For example, we may want to consider the release date of the movie to learn if there was a shift in the movie groups users preferred over the years. Although it is not difficult to extend Bi-LDA to Multi-LDA, we introduced Bi-LDA first because it is the most common case and the description of Multi-LDA involves bookkeeping that clutters the description of the model.

In order to make the transition to Multi-LDA we need to replace the two modalities user-movie with a list of modalities 1..M. Thus instead of π^{user} and π^{movie} we have $\pi^m, m \in 1...M$. To index the observations associated with the combination of modalities we replace m, u with $i_1..i_M$.

In Bi-LDA movie and user groups were indexed by j and k, in Multi-LDA j and k are replaced by $j_1..j_M$. For instance $z^1_{i_1..i_M} = j_1$ tells us that for modality 1 and data item $i_1..i_M$ group j_1 is assigned. Also,$\Phi_{j_1..j_M}[v]$ is the probability of value v for the cluster identified by the factors $j_1..j_M$. Thus the equation for the Bi-LDA model 1 becomes the following for the Multi-LDA model

$$P(X, \{\mathbf{z}^m\}, \boldsymbol{\Phi}, \{\boldsymbol{\pi}^m\}) =$$
$$P(X|\{\mathbf{z}^m\}, \boldsymbol{\Phi})P(\boldsymbol{\Phi}|\boldsymbol{\beta}) \prod_m P(\mathbf{z}^m|\boldsymbol{\pi}^m)\ P(\boldsymbol{\pi}^m|\boldsymbol{\alpha}^m)$$

The conditional distribution for sampling $\mathbf{z}$ becomes

$$P(z^m_{i_1..i_M} = j_m|\mathbf{z}\backslash z^m_{i_1..i_M}, X) \propto$$
$$\left(\frac{N^{v,\neg(i_1..i_M)}_{j_1..j_M} + \beta^v}{N^{\neg(i_1..i_M)}_{j_1..j_M} + \beta}\right)\left(N^{\neg(i_1..i_M)}_{j_1..j_M} + \frac{\alpha^m}{J}\right)$$

Where $X_{i_1..i_M} = v$ and $z^{m'}_{i_1..i_M} = j_{m'}\ \forall m' \backslash m$.

Bi-HDP

One consideration when applying the Multi-LDA model to a new data set, is how to choose the number of groups for each modality. Nonparametric Bayesian models offer an elegant solution, providing a prior over possible partitions of each modality into groups.

Using the movie ratings example again, we note that if we hold the assignment variables for the movie modality constant, inference in the user-branch is the same as inference in an LDA model. This observation suggests an easy procedure to take the infinite limit: replace each LDA branch with the nonparametric version of LDA, the Hierarchical Dirichlet Process (HDP) (Teh *et al.* 2006).

HDP introduces a $root$ pool of groups. Each movie draws a distribution over groups, π_m^{movie} using the $root$ distribution as a prior. Starting with the finite version of this extended model with J movie groups and K user groups we replace the distribution for π^m, π^u in the Bi-LDA model with the following: $\tau^m \sim \text{Dir}(\gamma/J, ..., \gamma/J)$, $\pi_m^m \sim \text{Dir}(\alpha^m \tau^m)$, $\tau^u \sim \text{Dir}(\gamma/K, ..., \gamma/K)$, $\pi_u^u \sim \text{Dir}(\alpha^u \tau^u)$. The rest of the model remains the same as in Bi-LDA.

We use the results from (Teh *et al.* 2006) to take the limit as $J, K \to \infty$ and get the nonparametric version of Bi-LDA, Bi-HDP.

For inference we use the direct assignment method of Gibbs sampling for a HDP distribution. Unlike in Bi-LDA and Multi-LDA where all variables other then $\mathbf{z}, X$ were marginalized over, we keep τ^u, τ^m.

The equations for the conditional probability of $\mathbf{z}$ (the user and movie group assignments for a rating) are the following:

$$P(z_{um}^m = j | \mathbf{z} \backslash z_{um}^m, \boldsymbol{\tau}) \propto (\alpha_m \tau_j^m + N_{mj}^{\neg(um)}) \times \left(\frac{N_{jk}^{v,\neg(um)} + \beta^v}{N_{jk}^{\neg(um)} + \beta} \right)$$

$$P(z_{um}^u = k | \mathbf{z} \backslash z_{um}^u, \boldsymbol{\tau}) \propto (\alpha_u \tau_k^u + N_{uk}^{\neg(um)}) \times \left(\frac{N_{jk}^{v,\neg(um)} + \beta^v}{N_{jk}^{\neg(um)} + \beta} \right)$$

The key difference between these equations and the finite case (2), is that τ^m and τ^u are distributions over J+1 and K+1 possible groups. If there currently exist K user groups, then $\alpha_u \tau_{(K+1)}^u$ is proportional to the accumulated probability of the infinite pool of 'empty' clusters. The same holds true for the movies. At every sampling step there is the possibility of choosing a new group from the countably infinite pool of empty groups. In this way the Gibbs sampler samples over the number of groups or if we make the connection to Matrix factorization again, the rank of the matrix decomposition. We must also sample τ^m, τ^u the details of which are omitted for space.

Although with the transition to Bi-HDP we have eliminated the need to choose the number of user and movie groups, we still have the parameters $\alpha^m, \gamma^m, \alpha^u, \gamma^u$ to choose. Fortunately we can also sample these parameters using the auxiliary variable trick explained in (Teh *et al.* 2006). Again we omit the detail for space. The Gibbs sampler thus alternates sampling the assignment variables $\{z^m, z^u\}$ with $\alpha^m, \alpha^u, \gamma^m, \gamma^u, \tau^m, \tau^u$. This is guaranteed to converge to the equilibrium distribution of the Bi-HDP model.

Finally, We can again extend Bi-HDP to Multi-HDP in the same way that we extended Bi-LDA to Multi-LDA.

Discussion

We have introduced a novel model for nonparametric Bayesian tensor factorization. The model has several advantages. First, it provides a full distribution over predictions for missing data. In collaborative filtering experiments we show we can use the variance for a prediction's distribution to reliably order the predictions by there accuracy. Second, the model can infer structure in multiple modalities concurrently. Third, the model infers the rank of the matrix/tensor decomposition and only requires a few hyper-parameters.

Throughout our description of the model, we use a multinomial distribution, Multinomial($\phi_{j_1..j_m}$), for a data value. However, the model works with other distributions with a conjugate prior, such as normal, with little modification.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 0447903 and No. 0535278 and by ONR under Grant No. 00014-06-1-073

References

Airoldi, E. M.; Blei, D. M.; Fienberg, S. E.; and Xing, E. P. 2008. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*. in press.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research* 3:993–1022.

Griffiths, T., and Steyvers, M. 2002. A probabilistic approach to semantic representation. In *Proceedings of the 24th Annual Conference of the Cognitive Science Society*.

Hofmann, T. 2004. Latent semantic models for collaborative filtering. *ACM Trans. Inf. Syst.* 22(1):89–115.

Kemp, C.; Tenenbaum, J. B.; Griffiths, T. L.; Yamada, T.; and Ueda, N. 2006. Learning systems of concepts with an infinite relational model. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence (AAAI '06)*.

Mansinghka, V.; Kemp, C.; Tenenbaum, J. B.; and Griffiths, T. L. 2006. Structured priors for structure learning. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence (UAI 2006)*.

Marlin, B. 2003. Modeling user rating profiles for collaborative filtering. In *Neural Information Processing Systems (NIPS-03)*.

Meeds, E.; Ghahramani, Z.; Neal, R. M.; and Roweis, S. T. 2007. Modeling dyadic data with binary latent factors. In Schölkopf, B.; Platt, J.; and Hoffman, T., eds., *Advances in Neural Information Processing Systems 19*. Cambridge, MA: MIT Press. 977–984.

Takacs, G.; Pilaszy, I.; Nemeth, B.; and Tikk, D. 2007. On the gravity recommendation system. In *Proceedings of KDD Cup and Workshop 2007*.

Teh, Y. W.; Jordan, M. I.; Beal, M. J.; and Blei, D. M. 2006. Hierarchical Dirichlet processes. *Journal of the American Statistical Association* 101(476):1566–1581.