Probabilistic Canonical Tensor Decomposition for Predicting User Preference

Maja Rita Rudolph
Department of Electrical Engineering
Columbia University
New York, NY 10027
mrr2163@columbia.edu

San Gultekin
Department of Electrical Engineering
Columbia University
New York, NY 10027
sg3108@columbia.edu

John Paisley
Department of Electrical Engineering
Columbia University
New York, NY 10027
jpaisley@columbia.edu

Shih-Fu Chang
Department of Electrical Engineering
Columbia University
New York, NY 10027
sfchang@columbia.edu

Abstract

We propose a model to infer a user's preference ranking over musicians from a sparse set of pairwise preferences of the form "user k prefers artist i over artist j". The goal is to approximate the data with a low-rank factor model using canonical tensor decomposition. A user-specific pairwise preference is modeled as the sign of a 3-way tensor inner product of latent factor vectors associated with the user and the two musicians being compared. The latent factors are learned using mean-field variational inference and can be used to predict all missing preference pairs. We validate our approach on a real data set of 80M pairwise preferences aggregated from the interaction of 200K users with an online radio.

1 Introduction

A good recommendation system is crucial for many web applications. The goal of this work is to learn a user's preference between any two artists based on data collected from user listening patterns to an online radio. A common paradigm for recommendation systems is collaborative filtering which explores the data for patterns in user behavior and makes recommendations to a user based on behaviors of other similar users. Latent factor models are often used to implicitly capture those patterns using a low-rank approximation of the observation data. Tot this end, matrix factorization has been successfully applied to the Netflix challenge for the prediction of movie ratings [1]. The idea is that each user and each item is associated with a latent factor vector. The rating by user k for item j is then modeled as a function of the inner product of the latent factor vector associated with user k and the latent factor vector associated with item j. A probabilistic treatment of matrix factorization has since improved the predictive capabilities of this formulation [2, 3].

Our goal in this work is to understand user listening behavior to last.fm, an online radio. The information we have is of the form "user k listened to artist j a total of c_j^k times". One approach is to directly model c_j^k. However, results can be skewed by outliers in listening patterns and in many cases the information desired is less granular, i.e., it is simply desired to know if a user likes an artist, or which artist a user prefers.

In this paper we focus on a model for this user preference scenario. To this end, we map the click count data to pairwise preferences [4–7] and obtain an incomplete 3-dimensional binary array $\{z_{ij}^{(k)}\}$

where $z_{ij}^{(k)} = 1$ if user k prefers artist i over artist j. The goal is to approximate this multi-way array with a low-rank factor model using canonical tensor decomposition [8]. Each entry of the array is modeled as the 3-way tensor inner product of 3 latent factor vectors. Each user k is associated with a latent factor vector $u_k \in \mathbb{R}^d$ and each artist i is associated with two latent factor vectors $V_i \in \mathbb{R}^d$ and $v_i \in \mathbb{R}^d$. The dimensionality d governs the rank of the approximation. Introducing two different sets of latent factor vectors for each item allows us to compare two items and to keep track of the order in which two items appear in the statement about a users' preference.

$$\text{user } k \text{ score of } i \text{ vs } j = \langle u_k, V_i, v_j \rangle \approx -\langle u_k, V_j, v_i \rangle = \text{user } k \text{ score of } j \text{ vs } i$$

where $\langle \cdot, \cdot, \cdot \rangle$ denotes the 3-way tensor inner product.

The generative process for the model is described in the next section. Variational inference, presented in Section 3, enables us to train our model on a large dataset of approximately 80M lines of the form "user k prefers item i over item j" aggregated from the online radio last.fm. The data set and experimental results on held out data are described in Section 4 which is followed by a discussion of our work including our planned future developments.

2 Generative Process for Canonical Tensor Decomposition

Consider a 3-way tensor $Z \in \{-1, +1\}^{N \times M \times M}$ that encodes the pairwise preferences of N users between M items as follows,

$$z_{ij}^{(k)} = \begin{cases} 1 & \text{if user k prefers item i over item j} \\ -1 & \text{if user k does not prefer item i over item j} \\ \emptyset & \text{no observed preference.} \end{cases}$$

We construct these values from the last.fm data by calculating the total number of times a user listens to artist i and j and setting the preference to be the one with the greater number.

We index the observations by $\Omega = \{(i, j, k) | z_{ij}^{(k)} \neq \emptyset\}$. In our approach, a rank d canonical decomposition of the tensor Z models each entry of as the 3-way tensor inner product of latent factor vectors u_k, V_i, $v_j \in \mathbb{R}^d$ plus Gaussian noise of variance $\sigma^2 > 0$

$$z_{ij}^{(k)} \sim N(< u_k, V_i, v_j >, \sigma^2). \tag{1}$$

Since $z_{ij}^{(k)} \notin \mathbb{R}$, we are making an approximation here similar that used by [2] in probabilistic matrix factorization. The priors on the factor vectors are multivariate Gaussian with diagonal precision matrix $\lambda^{-1} I \in \mathbb{S}_{++}^{d \times d}$, $\lambda > 0$:

$$\begin{aligned} u_k &\sim N(0, \lambda^{-1} I), \\ V_i &\sim N(0, \lambda^{-1} I), \\ v_j &\sim N(0, \lambda^{-1} I). \end{aligned}$$

In the next section we describe an inference scheme which allows us to learn these latent factor vectors from the data.

3 Variational Inference

To make predictions for unseen data we want to learn the latent factor vectors which maximize the posterior $p(\theta | Z)$, where $\theta = \{u_{1:N}, V_{1:M}, v_{1:M}\}$. We use the mean-field assumption and approximate the posterior with a fully factored variational distribution [9, 10],

$$q(\theta) = \big[\prod_{k=1}^{N} q(u_k)\big]\big[\prod_{i=1}^{M} q(V_i)\big]\big[\prod_{i=1}^{M} q(v_i)\big], \tag{2}$$

which requires that we define the functional forms of the q distributions.

The general variational objective we seek to maximize with respect to the parameters of a particular q is

$$\mathcal{L} = \mathbb{E}_q[\log p(Z, u, V, v)] - \mathbb{E}q[\log q(\theta)]. \tag{3}$$

which can be shown to minimize the Kullback-Leibler divergence between $q(\theta)$ and the true posterior $p(\theta|Z)$. Using the standard procedure, each $q(\theta_i)$ can be computed using the formula

$$q(\theta_i) \propto \exp\{\mathbb{E}_{q(\theta_{-i})}[\log p(Z, u, V, v)]\}. \tag{4}$$

In our case, the log joint likelihood of the model is

$$\begin{aligned}\log p(Z, u, V, v) = &\sum_{(i,j,k)\in\Omega} \log p(z_{ij}^{(k)}|u_k, V_i, v_j) \\ &+ \sum_{k=1}^{N} \log p(u_k) + \sum_i \log p(V_i) + \sum_i \log p(v_i).\end{aligned} \tag{5}$$

Since our model is fully conditionally conjugate, finding q with this joint likelihood gives distributions in the same family as the prior,

$$q(u_k) \sim N(\mu_{u_k}, \Sigma_{u_k}) \quad q(V_i) \sim N(\mu_{V_i}, \Sigma_{V_i}) \quad q(v_i) \sim N(\mu_{v_i}, \Sigma_{v_i}).$$

We next give the analytic updates for these parameters of q. Let $\odot$ be the elementwise multiplication of two vectors and define

$$\begin{aligned}
\Sigma_{u_k} &= (\tfrac{1}{\sigma^2}\textstyle\sum_{i,j} S_{ij}^{(u)} + \lambda \mathrm{I})^{-1}, & \mu_{u_k} &= \Sigma_{u_k} \tfrac{1}{\sigma^2} \textstyle\sum_{i,j} z_{ij}^{(k)} (\mu_{V_i} \odot \mu_{v_j}), \\
\Sigma_{V_i} &= (\tfrac{1}{\sigma^2}\textstyle\sum_{j,k} S_{jk}^{(V)} + \lambda \mathrm{I})^{-1}, & \mu_{V_i} &= \Sigma_{V_i} \tfrac{1}{\sigma^2} \textstyle\sum_{j,k} z_{ij}^{(k)} (\mu_{u_k} \odot \mu_{v_j}), \\
\Sigma_{v_j} &= (\tfrac{1}{\sigma^2}\textstyle\sum_{i,k} S_{ik}^{(v)} + \lambda \mathrm{I})^{-1}, & \mu_{v_j} &= \Sigma_{v_j} \tfrac{1}{\sigma^2} \textstyle\sum_{i,k} z_{ij}^{(k)} (\mu_{V_i} \odot \mu_{u_k}),
\end{aligned} \tag{6}$$

where

$$\begin{aligned}
S_{ij}^{(u)} =& (\mu_{V_i} \odot \mu_{v_j})(\mu_{V_i} \odot \mu_{v_j})^T + \operatorname{diag}(\operatorname{diag}(\Sigma_{V_i}) \odot \operatorname{diag}(\Sigma_{v_j}) + \mu_{V_i} \odot \mu_{V_i} \odot \operatorname{diag}(\Sigma_{v_j}) + \mu_{v_j} \odot \mu_{v_j} \odot \operatorname{diag}(\Sigma_{V_i})) \\
S_{jk}^{(V)} =& (\mu_{u_k} \odot \mu_{v_j})(\mu_{u_k} \odot \mu_{v_j})^T + \operatorname{diag}(\Sigma_{u_k} \odot \Sigma_{v_j} + \mu_{u_k}\mu_{u_k}^T \odot \Sigma_{v_j} + \mu_{v_j}\mu_{v_j}^T \odot \Sigma_{u_k}) \\
S_{ik}^{(v)} =& (\mu_{V_i} \odot \mu_{u_k})(\mu_{V_i} \odot \mu_{u_k})^T + \operatorname{diag}(\Sigma_{V_i} \odot \Sigma_{u_k} + \mu_{V_i}\mu_{V_i}^T \odot \Sigma_{u_k} + \mu_{u_k}\mu_{u_k}^T \odot \Sigma_{V_i}).
\end{aligned}$$

Then a coordinate ascent inference algorithm can be run by iterating between Equation (6) using the most recent values of the other parameters.

4 Experimental Results

We experiment with data collected in 2008 from the online radio last.fm [11]. The data is aggregated by artist and each line has the form "user k listened to artist j a total of c_j^k times". We selected the 796 most popular artists, namely all artists to which at least 1% of users listened, and we selected the users who listened to between $20 - 50$ musicians from this set of artists. This gave 201,147 users, for which we constructed the pairwise preference data set yielding 80 million lines of the form "user k prefers artist i over artist j".

For testing, for each user we randomly selected one artist and the user specific preference pairs involving this artist were held out and used as test data. We set the rank $d = 10$, precision parameter $\lambda = 0.1$ and noise variance $\sigma^2 = 1$ and randomly initialized the latent factor vectors according to

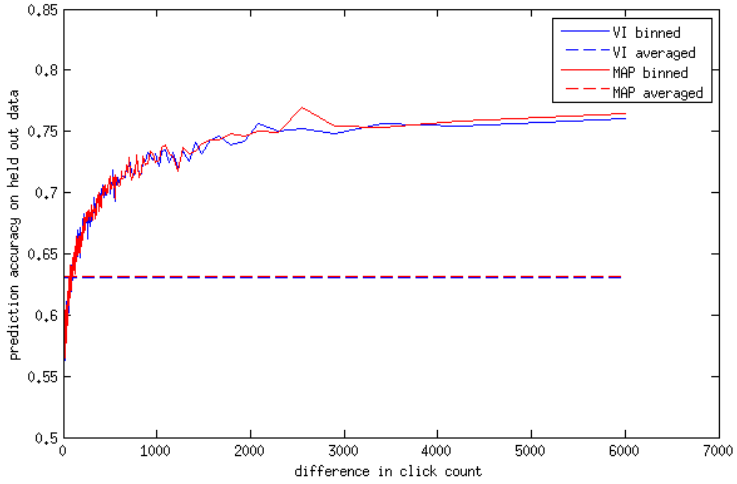

Figure 1: Prediction accuracy across 200K users and 800 artists from last.fm.

their priors. The test results after 50 iterations of our variational inference updates can be found in Figure 1. The average prediction accuracy on held out data is 62% depicted as a straight dashed line.

To better understand the results we also sorted the test data by click difference ($|c_i^k - c_j^k|$) and showed prediction accuracy based on this number. This is to give a sense of how our prediction accuracy varies by preference strength. That is, for pairs (k, i, j) that have a large difference in listening number, we would expect to have a higher accuracy, while the more borderline cases should be harder to predict. As expected the model performs better in the strong preference regime, i.e., when the click difference between two artists is large. This can also be seen in Table 1 which provides some qualitative results.

Table 1 shows the prediction results for 2 users on the randomly held out bands *guano apes* and *snow patrol* compared to artists the model was trained on. Those training artists are sorted from most listened to (measured in c_j^k) to least listened to. According to the underlying ground truth the users prefer the artists in the first column over the artists in the second column. Checks in the third column indicate whether our model predicts the pairwise preference correctly. Note that the click difference is largest at the beginning and the end of the list and smallest when the held out band switches from the right column to the left. Consistent with Figure 1, the prediction errors for the examples shown here are more frequent in the region where the click difference is small.

We compare our results with MAP estimates. Rather than updating the entire distribution the MAP algorithm only updates the locations of the latent factor vectors. This results in shorter running times but a model that doesn't capture uncertainty. For comparison Figure 1 also contains the test results after running MAP for 150 iterations.

5 Discussion and Future Work

We have presented a probabilistic model for the prediction of pairwise user preferences. The binary preferences are modeled as the 3-way tensor inner product of the latent factor vectors associated with the user and the two items being compared. We validate our model on a large data set collected from the online radio last.fm.

Both the advantage and the drawback of our model comes from introducing $N + 2M$ latent factor vectors rather than $N + M$ such as in other comparable approaches for pairwise preference prediction. More parameters to train means additional computational cost (time and space) but also results in more flexibility and easier fitting to the structure inherent in the data.

We are currently considering the following future developments:

user 130 pairwise preference predicted correctly?		
my dying bride	guano apes	✓
system of a down	guano apes	✓
john williams	guano apes	✓
manowar	guano apes	✓
behemoth	guano apes	✓
pantera	guano apes	✓
cannibal corpse	guano apes	✓
apocalyptica	guano apes	✓
t.i.	guano apes	✗
metallica	guano apes	✗
guano apes	ac/dc	✓
guano apes	lamb of god	✓
guano apes	howard shore	✓
guano apes	lil wayne	✓
guano apes	aerosmith	✓
guano apes	akon	✓
guano apes	black eyed peas	✓
guano apes	tiamat	✓
guano apes	jimmy eat world	✓
guano apes	hans zimmer	✓
guano apes	atb	✓
guano apes	red hot chili peppers	✓
guano apes	eminem	✓

user 160 pairwise preference predicted correctly?		
avenged sevenfold	snow patrol	✓
nightwish	snow patrol	✓
metallica	snow patrol	✗
30 seconds to mars	snow patrol	✓
anberlin	snow patrol	✓
ensiferum	snow patrol	✗
morcheeba	snow patrol	✓
system of a down	snow patrol	✗
staind	snow patrol	✗
the killers	snow patrol	✓
p.o.d.	snow patrol	✓
paramore	snow patrol	✓
rammstein	snow patrol	✓
kaiser chiefs	snow patrol	✓
in flames	snow patrol	✗
3 doors down	snow patrol	✗
kon	snow patrol	✗
snow patrol	hans zimmer	✓
snow patrol	robbie williams	✓
snow patrol	travis	✓
snow patrol	muse	✓
snow patrol	fear factory	✓
snow patrol	keane	✓
snow patrol	coldplay	✓
snow patrol	switchfoot	✓
snow patrol	band of horses	✓
snow patrol	disturbed	✓
snow patrol	epica	✓
snow patrol	my dying bride	✓
snow patrol	lady gaga	✓

Table 1: Prediction results for 2 users on the randomly held out bands *guano apes* and *snow patrol* compared to artists the model was trained on. Those training artists are sorted from most listened to to least listened to. According to the underlying ground truth the users prefer the artist in the first column over the artist in the second column. Checks indicate whether our model predicts the pairwise preference correctly. Note that the click difference is largest at the beginning and the end of the list and smallest when the held out band switches from the right column to the left. As alrady quantified in Figure 1 prediction occur more frequently in regions where the click difference is small.

- In the future, we hope to speed up our approach by using stochastic variational inference [12] (SVI). Since there is no clear way to partition our parameters into global and local variables, some additional thought will be required for applying stochastic inference to this problem.
- Modeling $z_{ij}^{(k)} \in \{-1, +1\}$ as a Gaussian random variable is clearly a rough approximation. We plan to consider probit models that use a latent Gaussian random variable of the same form as Equation (1), but then have a binning function that maps this latent Gaussian variable to ± 1.
- We will also extend the framework to the case where there is no clear user preference, possibly because of an equality in number of clicks, by having a third partition in our latent probit model for "no clear preference". This will give an added level of flexibility to the model in the cases where certain elements in Z cannot be formed based on an inequality.
- In addition, we hope to extend our model and inference scheme to a larger number of dimensions, for example a time dimension could give a 4-way tensor decomposition and allow for the modeling of temporal dynamics [13]. We also want to explore other schemes to incorporate temporal dynamics.

References

[1] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[2] Ruslan Salakhutdinov and Andriy Mnih. Probabilistic matrix factorization. In *NIPS*, volume 1, pages 2–1, 2007.

[3] Ruslan Salakhutdinov and Andriy Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th international conference on Machine learning*, pages 880–887. ACM, 2008.

[4] Nathan N Liu, Luheng He, and Min Zhao. Social temporal collaborative ranking for context aware movie recommendation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(1):15, 2013.

[5] Ulrich Paquet, Blaise Thomson, and Ole Winther. A hierarchical model for ordinal matrix factorization. *Statistics and Computing*, 22(4):945–957, 2012.

[6] Tim Salimans, Ulrich Paquet, and Thore Graepel. Collaborative learning of preference rankings. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 261–264. ACM, 2012.

[7] Mohammad Emtiyaz Khan, Young Jun Ko, and Matthias Seeger. Scalable collaborative bayesian preference learning. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics*, number EPFL-CONF-196605, 2014.

[8] Henk AL Kiers. Towards a standardized notation and terminology in multiway analysis. *Journal of chemometrics*, 14(3):105–122, 2000.

[9] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[10] Martin J Wainwright and Michael I Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1-2):1–305, 2008.

[11] O. Celma. *Music Recommendation and Discovery in the Long Tail*. Springer, 2010.

[12] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

[13] Liang Xiong, Xi Chen, Tzu-Kuo Huang, Jeff G Schneider, and Jaime G Carbonell. Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *SDM*, volume 10, pages 211–222. SIAM, 2010.