Probabilistic Models for Data Combination in Recommender Systems

Sinead Williamson
Engineering Dept.
University of Cambridge
Cambridge, CB2 1PZ, UK

Zoubin Ghahramani
Engineering Dept.
University of Cambridge
Cambridge, CB2 1PZ, UK

1 Introduction

In a typical collaborative filtering problem, the dataset is an incomplete matrix of ratings $\mathbf{R}$ given by a set $\mathcal{U}$ of users to a set $\mathcal{I}$ of items, and the task is to predict what ratings the users would give to the items they have not yet rated. A common approach to this problem is to use matrix factorization techniques to find a lower dimensional representation, $\mathbf{R} \approx \mathbf{U}\mathbf{M}^T$ of this matrix [3, 6, 5, 7].

Collaborative filtering systems suffer from two significant, and related, problems: sparsity and extension to new items. Due to the fact that most users rate a very small subset of the universe of items, the user-item rating matrix is generally very sparse, particularly in the case of new users or obscure items, leading to poor predictions in such cases. Furthermore, an item cannot be recommended until it has been rated, which is a significant disadvantage since most commercial recommender systems will hope to promote new items.

In many real-life situations, our knowledge of the users and the items may not be restricted to the aforementioned ratings matrix. In a movie recommender system, we may have synopses of the films and details of the cast list. A system recommending electronic goods may have technical specifications, manuals or manufacturer's descriptions. A music recommendation system is likely to contain reviews, and may allow users to 'tag' songs or artists with words or phrases. We may also have information about the users, through blogs, social networks and demographic information. In all the cases described above, the additional information can be described in matrix form.

We propose a method for jointly learning multiple related matrices, and show that, by sharing information between the two matrices, such an approach allows us to improve predictive performances for items where one of the matrices contains very sparse, or no, information. While the above justification has focused on recommender systems, the approach described is applicable to any two datasets that relate to a common set of items and can be represented in matrix form. Examples of such problems could include image data where each image is associated with a set of words (for example captioned or tagged images); sets of scientific papers that can be represented either using a bag-of-words representation or in terms of their citation links to and from other papers; corpora of documents that exist in two languages.

2 Probabilistic data combination

Our method assumes the presence of two sources of information about a common set of items, for example a (sparse) user-item ratings matrix $\mathbf{R}$, and a (full) item-content matrix $\mathbf{S}$. Our method is motivated by the assumption that the latent features that determine whether a given user will like a given item, and the features that determine the "content" of that item, can be mapped into a space where they are likely to be similar. We constrain our factorizations to use a common matrix to model the features of each item, as shown in figure 1. Here, the (partially observed) ratings matrix $\mathbf{R} \approx \mathbf{C}\mathbf{B}^T$, and the (fully observed) content matrix $\mathbf{S} \approx \mathbf{C}\mathbf{B}^T$, with the latent-feature matrix $\mathbf{B}$ contributing to both matrices.

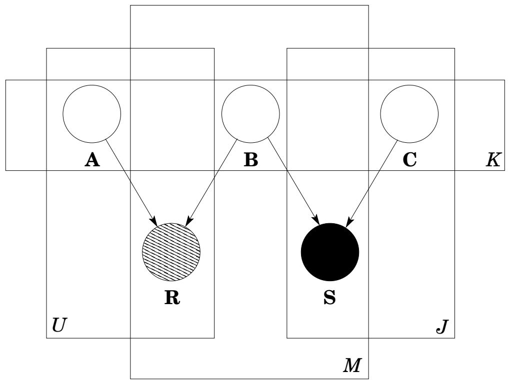

Figure 1: Graphical model detailing the relationship between the ratings matrix $\mathbf{R}$ and the content matrix $\mathbf{S}$

Depending on the form of the data being modeled, any Generalized Linear Model can be used to describe the relationships between the latent matrices and the observed matrices. In the results presented below, we used a linear Gaussian model, and inference was carried out using a Variational Bayesian approach.

3 Results

Here we present the performance of our model on two datasets, one containing ordinal ratings data, and one containing continuous inferred ratings data. In each case, we compared our model with a variational implementation of probabilistic Singular Value Decomposition described in the joint variational matrix factorization model described in [4]. Where ordinal data was available, we also compared with the JRank algorithm of [1] (using the identity kernel for the user kernel), and with a weighted combination of the single variational matrix factorization collaborative filtering model and a series of naive Bayes classifiers trained on the content data for each user. Since these two methods require discrete classes, they were not used on the continuous dataset.

3.1 MovieLens ratings and genre data

The MovieLens dataset[1] contain 100,000 ratings given by 943 users to 1682 movies. The ratings are integer values between one and five. In addition to the ratings data, this dataset also contains binary labels classifying each movie into one or more of 19 genres. We use the sparse rating matrix as the $\mathbf{R}$ matrix in figure 1, and the full, binary genre matrix as the $\mathbf{S}$ matrix.

We carried out two experiments on this dataset. First, we removed approximately one third of the ratings (at random), and calculated the average RMSE on the held-out data for the four models described above. This experiment was designed to test the average predictive performance of the algorithms. The results are shown in table 1

Secondly, we designed an experiment to test the performance of the algorithm on new movies with no, or few, reviews. We selected 200 movies at random from the subset of movies with over 200 ratings. For each movie, we selected between 0 and 10 of the ratings to appear in the training set, and used our model to predict the remaining ratings. The results are shown in table 2. The total size of the training set is the same as that of the training set

[1]www.grouplens.org

Table 1: MovieLens: RMSE on randomly held-out movies

Method	RMSE
Joint variational matrix factorization	0.9725
Single variational matrix factorization	0.9928
JRank	1.2108
Naive Bayes classifier + single variational matrix factorization	1.2734

in the first example, allowing comparisons to be made between the performances on 'new' movies and randomly selected movies.

Table 2: MovieLens: RMSE on movies with all bar [0,10] ratings held out

Method	RMSE
Joint variational matrix factorization	1.1048
Single variational matrix factorization	1.2135
JRank	2.8722
Naive Bayes classifier + single variational matrix factorization	1.4013

In each case we used a rank 20 decomposition for both the joint and the single matrix factorization models. This value was chosen to optimize performance in the single variational matrix factorization model, to avoid bias towards the joint matrix factorization method, which will generally favour a larger value.

3.2 Inferred preference data from Last.FM

Last.FM is an online radio that records data of the listening habits of its subscribers, and plays them songs based on its own collaborative filtering algorithms. The Last.FM site also contains information on recording artists, including biographies, discographies, and user-generated 'tags'. While Last.FM does not obtain ratings from its users, the number of times a user has listened to a given artist can be used as an indicator of their opinion of a band.

We tested our algorithm, and the single-matrix variational matrix factorization model, on a dataset consisting of 1100 users and 1079 artists, with a total of 9440 user-artist pairs. This dataset was obtained by selecting 1500 artists at random from the May 2005 Last.FM data dump[2], and then selecting 1100 users at random from the subset of users who had listened to one or more of these artists. The raw pseudo-ratings data for a given user-artist pair consisted of the number of times a that user had listened to that artist. This raw data ranged between 0-10,000. To prevent a few high user-artist values from dominating the dataset, we trained and tested the algorithms on the log of the raw play counts.

The content data consisted of the tags assigned to an artist by listeners, taken from the artist's Last.FM profile. The 1000 most commonly used tags across the dataset were used. Not all artists had associated content. The RMSE errors calculated over 1000 hold-out points is given in table 3.

Table 3: LastFM: RMSE on 1000 hold-out points

Method	RMSE
Joint variational matrix factorization	1.5741
Single variational matrix factorization	1.8782

[2]http://www-etud.iro.umontreal.ca/∼bergstrj/audioscrobbler_data.html

4 Discussion and conclusions

We have proposed a Bayesian model for jointly factorizing two separate matrices relating to a common set of items, and have demonstrated its performance in a recommender system framework. We have shown that our method demonstrates good performance when compared with both factorization of the ratings matrix alone, and two methods that make predictions based on both the ratings and the content data. Our method performs particularly well in cases where the ratings data is sparse, for example a movie that has just been released. By comparison, while the JRank and weighted naive Bayes/variational matrix factorization methods performed reasonably well in the case of randomly removed data (see table 1), they performed poorly in the case of 'new' movies (see table 2).

The ability to perform well in low-ratings-data regimes, and in the case of new items, is often lacking in collaborative filtering-based recommender systems. This is unfortunate, because new items are particularly important in recommendation scenarios. A useful goal in recommender systems is to offer novel items, with which the user is unfamiliar, and in particular serendipitous items, which the user would have been unlikely to find or sample were it not for the recommendation of the system [2]. A user is less likely to already have seen a new movie, or read a new book. Also, a retailer will often be most interested in promoting newly released items, or items that do not already have a large audience.

In the examples above, we have demonstrated the model on matrices containing rating and item-content data. It is, however, applicable to a wide range of matrix factorization problems, as was described in Section 1. The model could easily be extended to jointly factorize more than two related matrices - for example if we wished to include demographic information about the users as well as content information about the items. include further information about the users, for example demographic information or social network data. While we have focused on the application of such a model to recommender systems, it has much wider general applicability - for example, jointly modeling web page content and link structure; jointly modeling image properties and associated text; jointly modeling the text and the citation structure of scientific papers.

References

[1] Justin Basilico and Thomas Hofmann, *Unifying collaborative and content-based filtering*, ICML '04: Proceedings of the twenty-first international conference on Machine learning (New York, NY, USA), ACM, 2004, p. 9.

[2] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl, *Evaluating collaborative filtering recommender systems*, ACM Trans. Inf. Syst. **22** (2004), no. 1, 5–53.

[3] Thomas Hofmann, *Probabalistic latent semantic indexing*, Research and Development in Information Retrieval, 1999, pp. 50–57.

[4] Y. J. Lim and Y. W. Teh, *Variational Bayesian approach to movie rating prediction*, Proceedings of KDD Cup and Workshop, 2007.

[5] Edward Meeds, Zoubin Ghahramani, Radford M. Neal, and Sam T. Roweis, *Modeling dyadic data with binary latent factors*, Neural Information Processing Systems (NIPS), 2006.

[6] Ruslan Salakhutdinov and Andriy Mnih, *Probabilistic matrix factorization*, Advances in Neural Information Processing Systems, vol. 20, 2008.

[7] Nathan Srebro, Jason D. M. Rennie, and Tommi S. Jaakkola, *Maximum-margin matrix factorization*, Advances in Neural Information Processing Systems 17, MIT Press, 2005.