International Journal of Management Science
Vol 12, No 2, November 2006

A Model-based Collaborative Filtering Through Regularized Discriminant Analysis Using Market Basket Data*

Jong-Seok Lee
Department of Industrial and Manufacturing Systems Engineering
Iowa State University, Ames, Iowa 50011, USA

Chi-Hyuck Jun**
Department of Industrial and Management Engineering, POSTECH
San 31 Hyoja-dong, Pohang 790-784, Seoul, Korea

Jaewook Lee
Department of Industrial and Management Engineering, POSTECH
San 31 Hyoja-dong, Pohang 790-784, Seoul, Korea

Sooyoung Kim
Department of Industrial and Management Engineering, POSTECH
San 31 Hyoja-dong, Pohang 790-784, Seoul, Korea

(Received Mar. 2005; Revised Jan. 2006/Mar. 2006; Accepted Apr. 2006)

ABSTRACT

Collaborative filtering, among other recommender systems, has been known as the most successful recommendation technique. However, it requires the user-item rating data, which may not be easily available. As an alternative, some collaborative filtering algorithms have been developed recently by utilizing the market basket data in the form of the binary user-item matrix. Viewing the recommendation scheme as a two-class classification problem, we proposed a new collaborative filtering scheme using a regularized discriminant analysis applied to the binary user-item data. The proposed discriminant model was built in terms of the major principal components and was used for predicting the probability of purchasing a particular item by an active user. The proposed scheme was illustrated with two modified real data sets and its performance was compared with the existing user-based approach in terms of the recommendation precision.

Keywords: Classification, Collaborative Filtering, Discriminant Analysis, Market Basket Data, Principal Component Analysis

---

* We would like to thank two anonymous referees for their helpful comments. This work was partially supported by KOSEF through the National Core Research Center for Systems Bio-Dynamics at POSTECH.

** Corresponding author, Email: chjun@postech.ac.kr

1. INTRODUCTION

The collaborative filtering (CF), first named by Goldberg *et al.* [5], is a technique that uses the known preferences of a group of users to predict the unknown preferences of a new user. Later, it was sometimes called as "social information filtering" [13] or "recommender system" [12]. These techniques are originally based on users' rating data for items. The basic assumption is that if users A and B rate a certain group of items similarly, then they are likely to rate other items similarly [6].

One of the problems in those CF schemes is that rating data may not be readily available. The usual customers seldom rate the products they used. It makes the CF suffer from the sparsity problem. To overcome such a problem, some researchers proposed a new CF scheme using the market basket data [2, 10, 11, 15]. The market basket data is just a list of past transaction records of products that were purchased by customers for a specified period. This scheme has the advantage that there is no need to gather rating data and so the probability of using possibly distorted data may be reduced. The market basket data can be transformed into a so-called binary user-item matrix having customers (users) and products (items) consisting of ones (purchases) and zeros (non-purchases). Deshpande and Karypis [3] developed a new item-based CF algorithm that uses item-to-item similarities from the user-item matrix to compute the relations between the different items. Huang *et al.* [8] viewed CF as a bipartite graph problem and proposed several new algorithms using the binary-type data. Lee *et al.* [9] proposed a recommendation scheme using binary logistic regression models applied to this type of binary user-item data.

Eventually, we would like to predict whether the zero value in an active user of interest for an item will stay at zero or change into one in the near future. This is exactly same as the two-class classification problem. So, any classification techniques may apply to develop a recommendation scheme when it is based on the market basket data. The discriminant analysis is a popularly used method in classification. Particularly, the regularized discriminant analysis proposed by Friedman [4] should be an alternative since it is quite flexible in choosing models [7]. Hence, we propose a model-based CF scheme which utilizes the regularized discriminant analysis combined with the principal components as a classification tool.

The organization of the paper is as follows: Section 2 describes the existing user-based CF scheme which is based on the binary user-item matrix. In Section 3, we proposed a new CF methodology based on the regularized discriminant analysis after performing a dimension reduction by principal component analysis.

Section 4 provides numerical experiments in order to test the performance of the proposed scheme. Finally, concluding remarks will be described in Section 5.

2. USER-BASED CF USING MARKET BASKET DATA

Suppose that rating scores by users are not available but only the market basket data can be obtained. The market basket data only tells us if a user has purchased a particular item or not. Suppose that there are n users and m items in the score database. The scores v_{ij} $(i=1,2,\ldots,n;\ j=1,2,\ldots,m)$ is therefore defined by

$$v_{ij} = \begin{cases} 1, & \text{when user } u_i \text{ purchased item } j \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Here, $V=(v_{ij})$ is called a binary user-item matrix.

In the user-based CF, the predicted scores by an active user for items of interest can be obtained from the same formula proposed by Breese *et al.* [1]. For the binary data, particularly, Mild and Reutterer [11] has proposed the use of the following modified version. That is, the predicted score by the active user u_a for the item j, P_{aj}, can be estimated as follows:

$$P_{aj} = \kappa_a \sum_{i=1}^{n} w(a,i) v_{ij} \tag{2}$$

where $w(a,i)$ is the weight or Pearson correlation coefficient reflecting the similarity between the active user u_a and the user u_i, and κ_a is a normalizing factor such that the absolute values of the weights sum to unity:

$$\kappa_a = \frac{1}{\sum_{i=1}^{n} | w(a,i) |} \tag{3}$$

From the user-item matrix we can summarize the data into the following proportions:

p_i = proportion of items that were purchased by user u_i, $i=1,\ldots,n$

p_{ij} = proportion of items that were purchased by both users u_i and u_j, $i \neq j$.

Then, the correlation coefficient between users u_a and u_i in (2) can be obtained by

$$w(a,i) = \frac{p_{ai} - p_a p_i}{\sqrt{p_a(1-p_a)}\sqrt{p_i(1-p_i)}} \quad (4)$$

3. PROPOSED CF BASED ON DISCRIMINANT ANALYSIS

As mentioned earlier, our goal is to predict the preference of an active user for the item *j* of interest that has not been purchased yet. Let v_j be an indicator variable for the item *j*, where '1' represents the purchase and '0' represents the non-purchase. Assume that the active user has already purchased at least one item and that the item *j* has been purchased by at least one user. Then, using item-to-item similarities we may predict whether the variable v_j belongs to 'class 1' (purchase state) or to 'class 0' (non-purchase state). This is a two-class classification problem where the user-item matrix *V* in (1) can be used as a training sample. More often, we will estimate the probability that v_j equals 1 and the classification rule can be derived on the basis of this value. So, our classification model for predicting the class of item *j* can be represented in a general form as follows:

$$v_j = g_j(v_1, \cdots, v_{j-1}, v_{j+1}, \cdots, v_m) + \varepsilon_j, \quad j = 1, \cdots, m. \quad (5)$$

where g_j is a suitable classification function, a discriminant function in our case, for the item *j* to be estimated and ε_j is a random error term. The model in (5) will be called the *j*-th model. In this paper we will consider the regularized discriminant analysis proposed by Friedman [4] for the classification.

Since the number of items under consideration in reality is very large and there may exist high correlation among items, some inefficiency may occur when estimating the model. Therefore, the dimension reduction through principal component analysis (PCA) is recommended before modeling [9].

PCA can be done by a spectral decomposition of the covariance matrix [14], from which eigenvalues and eigenvectors will be obtained. The reduced number of new variables or principal components can be determined by observing the magnitude of eigenvalues. Suppose that the number of principal components is chosen as p ($\le m$). Then, these principal components or new variables for the *j*-th model can be formed as follows:

$$\begin{aligned}
\xi_1^j &= w_{11}^j v_1 + \cdots + w_{1j-1}^j v_{j-1} + w_{1j+1}^j v_{j+1} + \cdots + w_{1m}^j v_m \\
\xi_2^j &= w_{21}^j v_1 + \cdots + w_{2j-1}^j v_{j-1} + w_{2j+1}^j v_{j+1} + \cdots + w_{2m}^j v_m \\
&\vdots \\
\xi_p^j &= w_{p1}^j v_1 + \cdots + w_{pj-1}^j v_{j-1} + w_{pj+1}^j v_{j+1} + \cdots + w_{pm}^j v_m
\end{aligned}, \tag{6}$$

where w_{kl}^j is the weight of l-th variable for the k-th principal component used in the j-th model. So, the number of predictor variables used for the j-th model reduces from $m-1$ to p, and the model for predicting the state of v_j will be in the following form instead of (5):

$$v_j = g_j(\xi_1^j, \xi_2^j, \cdots, \xi_p^j) + \varepsilon_j \tag{7}$$

The method of obtaining the principal components from the binary user-item matrix was described in Lee et al. [9].

The two-class discriminant analysis is based on the assumption that a vector x of p predictor variables for class k (k=0, 1) follows a multivariate normal distribution with density of

$$f_k(\mathbf{x}) = \frac{1}{(2\pi)^{p/2} \,|\, \boldsymbol{\Sigma}_k \,|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)'\boldsymbol{\Sigma}_k^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)\right\} \tag{8}$$

where $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ are mean vector and covariance matrix of the class k, respectively, which should be estimated from a training data to generate a classification rule or discriminant function. When $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_0$, the analysis is called a linear discriminant analysis (LDA) since the resultant discriminant function is linear. When $\boldsymbol{\Sigma}_1 \neq \boldsymbol{\Sigma}_0$, it is called a quadratic discriminant analysis (QDA) since the discriminant function is in a quadratic form. Friedman [4] proposed a fair regularization for the choice between LDA and QDA, which is called a regularized discriminant analysis (RDA). By introducing the regularization parameters, the sample covariance matrix of each class will be estimated.

Let n_k^j (k = 0, 1) be the number of observations (users) that belong to class k for the j-th model in the training data and let $n(= n_0^j + n_1^j)$ be the total number of users in the training data. Also, let $\xi_{i(k)}^j$ be the i-th principal component score in class k for the j-th model. Then, the mean vector in class k can be estimated by

$$\hat{\boldsymbol{\mu}}_k^j = \bar{\boldsymbol{\xi}}_{(k)}^j = \frac{1}{n_k^j}\sum_{i=1}^{n_k^j}\boldsymbol{\xi}_{i(k)}^j \tag{9}$$

and the sample covariance can be estimated by

$$\mathbf{S}_k^j = \frac{1}{n_k^j - 1}\sum_{i=1}^{n_k^j}(\xi_{i(k)}^j - \bar{\xi}_{(k)}^j)(\xi_{i(k)}^j - \bar{\xi}_{(k)}^j)' \tag{10}$$

However, RDA suggests using the following estimated covariance matrix having regularization parameters λ and γ ($0 \le \lambda, \gamma \le 1$):

$$\hat{\boldsymbol{\Sigma}}_k^j(\lambda,\gamma) = (1-\gamma)\hat{\boldsymbol{\Sigma}}_k^j(\lambda) + \frac{\gamma}{p}tr[\hat{\boldsymbol{\Sigma}}_k^j(\lambda)]\mathbf{I} \tag{11}$$

where

$$\hat{\boldsymbol{\Sigma}}_k^j(\lambda) = \frac{(1-\lambda)(n_k^j - 1)S_k^j + \lambda\sum_{k=0}^{1}(n_k^j - 1)S_k^j}{(1-\lambda)n_k^j + \lambda n}, \tag{12}$$

Here, regularization parameters can be estimated by a cross validation from the training data. Note that when $\lambda = \gamma = 0$ the result would be same as QDA while when $\lambda = 1$ and $\gamma = 0$ it leads to LDA.

Consequently, the joint density function in (12) will be estimated as follows for the j-th model:

$$f_k^j(\boldsymbol{\xi}) = \frac{1}{(2\pi)^{p/2}\left|\hat{\boldsymbol{\Sigma}}_k^j(\lambda,\gamma)\right|^{1/2}}\exp\left\{-\frac{1}{2}(\boldsymbol{\xi} - \hat{\boldsymbol{\mu}}_k^j)'\hat{\boldsymbol{\Sigma}}_k^j(\lambda,\gamma)^{-1}(\boldsymbol{\xi} - \hat{\boldsymbol{\mu}}_k^j)\right\} \tag{13}$$

Then, the probability of purchasing the j-th item for the active user will be predicted by

$$P^j = \frac{f_1^j(\xi)}{f_0^j(\xi) + f_1^j(\xi)} \tag{14}$$

where ξ is a vector of principal component scores for the active user, which should be obtained by purchase/non-purchase data of the active user for the rest of items. Note that (14) can be used for the binary classification (purchase/non-purchase) by introducing the cut-off value.

If there are many active users, the above procedure should be applied to each of the active users to estimate purchase probabilities. Similar procedure should be repeated in order to estimate purchase probabilities of other items under consideration. Then, we recommend the specified number of items for the active user in the order of estimated probabilities.

4. NUMERICAL EXPERIMENTS

4.1 Datasets used

In order to evaluate the proposed method, two data sets - the EachMovie data set and the MovieLenz data set, that are available on the web site, http://www.research.digital.com/SRC/eachmovie/ and http://www.cs.umn.edu/research/Group Lens/, respectively, were used. The EachMovie data set consists of 2,811,983 rating scores from 72,916 users on 1,628 movies and videos. The rating scores are on a numeric six-point scale with (0.0, 0.2, 0.4, 0.6, 0.8, 1.0). For this experiment 725 users and 257 movies were randomly selected. On the other hand, the MovieLenz data set consists of approximately 1 million ratings for 3,900 movies by 6,040 users and the rating scores are on a numeric five-point scale with (1, 2, 3, 4, 5). From the MovieLenz data set 1,397 users and 500 movies were randomly selected. Again, rating data are not required for our CF scheme but these data sets were chosen only due to the limitation on data collection.

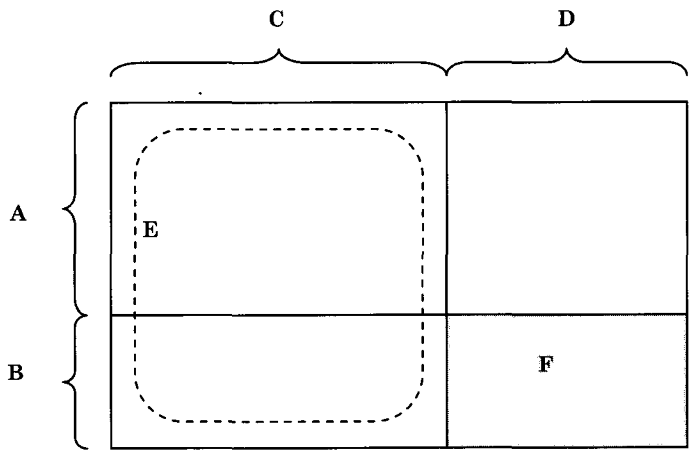
For our CF scheme of requiring binary user-item matrix we changed the values of rated cells including zero ratings into ones and the null values of non-rated cells into zeros. It should be noted that we are not arguing that we should always transform rating scores to binary data, but we would like to demonstrate that our model still works even when only binary data is available. Figure 1 describes how to transform the rating scores into the binary data. The same efforts were made in Deshpande and Karypis [3].

According to the procedure in Lee *et al.* [9], each of the above two data sets was divided into a training set (A; training users) and a test set (B; active users) as depicted in Figure 2. The data for the training set was used to build our discriminant models. We also divided the items for active users into a set of items that treated as predictor variables in our model (C) and a set consisting of items regarded as response variables (D). Note that these divisions were done randomly. The area by dotted line (section E) will be used to calculate similarities between training users and active users in the user-based approach by (2). The grey shaded area of the data (section F) will be first blinded and then used to measure the performance of the CF scheme.

	Item 1	Item 2	…	Item m-1	Item m
u_1	0.2		…	0.8	
u_2		0.0	…		1.0
⋮	⋮	⋮	⋮	⋮	⋮
u_n		0.2	…		0.4

	Item 1	Item 2	…	Item m-1	Item m
u_1	1	0	…	1	0
u_2	0	1	…	0	1
⋮	⋮	⋮	⋮	⋮	⋮
u_n	0	1	…	0	1

Figure 1. Transformation into binary data



A : Training users
B : Active users
C : Items referred to as independent variables
D : Items referred to as dependent variables
E : Data for calculating the similarities between A and B in user-based approach

Figure 2. Division of the experimental data set

Table 1 summarizes the two data sets, EachMovie and MovieLenz, by sections shown in Figure 2. Note that the proportion of ones for active users is higher than that for training users in the EachMovie data set, whereas the proportions of ones for active users and training users are similar for the MovieLenz data set.

Table 1. Summary of each section of data

Dataset	Section	No. of Users	No. of Items	No. of Ones	Proportion of Ones (%)
EachMovie	A×C	604	207	33,972	27.17
	A×D	604	50	6,519	21.59
	B×C	121	207	9,234	36.87
	B×D	121	50	2,046	33.82
MovieLenz	A×C	1,068	411	133,110	30.32
	A×D	1,068	89	25,921	27.27
	B×C	329	411	43,273	32.00
	B×D	329	89	8,349	28.51

4.2 Performance Comparisons

Our performance measure is the precision or hit rate, which is generally used in information retrieval research and calculated by

$$\text{Precision} = \frac{\text{hit number}}{\text{Top-N}} \tag{15}$$

where 'Top-N' is the specified number of items to be recommended by a CF scheme and 'hit number' is the actual number of items that were recommended and appeared in the section F having values of one. For example, if an active user actually purchased 4 items among 5 items that a CF scheme recommended, then the precision equals 0.8. We will consider various values of Top-N ranging from 1 to 10. When multiple active users are considered, the average of the precision for each active user will be taken, which will be referred to as the average precision.

As mentioned in Section 3, we reduced the dimension of predictor variables by using PCA. The eigenvalues corresponding to the first 50 principal components for each data set are listed in Table A1. Figure A1 shows all eigenvalues as a scree plot, from which the first 10 principal components (bold figures in Table A1) for the EachMovie data set and the first 18 principal components (bold figures in Table A1) for the MovieLenz data set were chosen, respectively, as predictor variables for our proposed scheme.

When applying RDA, we determined the regularization parameters (λ, γ) by two-fold cross validation using the training data, which minimizes the sum of squared prediction errors between the actual values in the section A×D and the predicted probabilities by (14). Note that the model corresponding to each item should have its own regularization parameters. The parameters for 50 items in EachMovie data set and 89 items in MovieLenz data set were summarized in Table A2.

Table 2 shows the average precisions calculated by (15) of the proposed scheme over all active users considered for EachMovie data set as well as for MovieLenz data set according to various values of Top-N. It can be seen that the precision of the proposed CF scheme is quite high enough to be used in practice. For example, when recommending the top 5 movies were recommended by RDA for an active user in the EachMovie data set, 85.5 percent of items were actually purchased on the average. When compared with the proportion of 1's in Table 1, the level of the average precision shown in Table 2 seems to be high.

In this table the average precision of the existing user-based approach was also included. We also compared with the average precision of the approach by Lee et al. (2005) using the binary logistic regression (BLR). As we can see, the precision of the proposed approach is absolutely higher than the precision of the user-based approach for all values of Top-N. The BLR seems to be slightly superior but RDA is quite comparable to this. Obviously, the performance of a CF scheme may depend on the characteristics of data sets, so it cannot be known until it is actually implemented. Hence, a variety of methods should be tested before taking the best one.

Table 2. Average precisions in percentage for two data sets

EachMovie				MovieLenz			
Top-N	RDA	BLR	User-based	Top-N	RDA	BLR	User-based
1	91.736	92.562	89.256	1	76.900	80.547	74.468
2	89.669	92.149	86.777	2	76.292	77.508	70.669
3	89.256	91.736	86.226	3	74.468	76.393	69.098
4	86.983	90.702	84.711	4	73.252	74.848	67.857
5	85.455	89.256	81.157	5	72.340	73.435	67.234
6	83.058	85.537	78.788	6	71.682	72.239	65.856
7	80.992	83.235	76.860	7	70.821	71.298	64.655
8	78.409	80.165	73.760	8	69.529	70.137	63.526
9	75.023	77.502	71.717	9	68.794	69.537	62.682
10	72.975	75.207	70.413	10	68.207	68.936	61.216

5. CONCLUSIONS

We proposed a new model-based collaborative filtering (CF) scheme by utilizing the market basket data. Regularized discriminant analysis was employed as a classification method after dimension reduction by principal component analysis. Through the numerical experiments using two modified real data sets, we found that the proposed scheme outperformed the user-based approach. Even though the method by Lee et al. [9] slightly was slightly better than the proposed one for two data sets used in these experiments, the regularized discriminant analysis may be a good alternative for a CF scheme. The result also supports the use of the market basket data in a recommender system instead of the rating data.

There may be a practical issue regarding the computational time when the CF scheme is used for the on-line recommendation. If the model training were performed by off-line operations, then this problem can be lightened since the computational time of recommendation for an active user may be short enough. A further study should be needed to develop a more elaborate model by utilizing classification methods. Also, a new CF scheme is expected to be developed in a future, which utilizes the user-to-user similarities as well as the item-to-item similarities.

REFERENCES

[1] Breese, J. S., D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," *Microsoft Research Technical Report*, MSR-TR-98-12, 1998.

[2] Demiriz, A., "Enhancing product recommender systems on sparse binary data," *Data Mining and Knowledge Discovery* 9, 2 (2004), 147-170.

[3] Deshpande, M. and G. Karypis, "Item-based top-N recommendation algorithms," *ACM Transactions on Information Systems* 22, 1 (2004), 143-177.

[4] Friedman, J. H., "Regularized discriminant analysis," *Journal of the American Statistical Association* 84, 405 (1989), 165-175.

[5] Goldberg, D., D. Nichols, B. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM* 35, 12 (1992), 61-70.

[6] Goldberg, K., T. Roeder, D. Gupta, and C. Perkins, "Eigentaste: a constant

time collaborative filtering algorithm," *Information Retrieval* 4, 2 (2001), 133-151.

[7] Hastie, T., R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, New York, 2001.

[8] Huang, Z., H. Chen, and D. Zeng, "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering," *ACM Transactions on Information Systems* 22, 1 (2004), 116-142.

[9] Lee, J.-S., C.-H. Jun, J. Lee, and S. Kim, "Classification-based collaborative filtering using market basket data," *Expert Systems with Application* 29, 3 (2005), 700-704.

[10] Mild, A. and T. Reutterer, "Collaborative filtering methods for binary market basket data analysis," *Lecture Notes in Computer Science* 2252 (2001), 302-313.

[11] Mild, A. and T. Reutterer, "An improved collaborative filtering approach for predicting cross-category purchase based on binary market basket data," *Journal of Retailing and Consumer Services* 10, 3 (2003), 123-133.

[12] Resnick, P., N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: an open architecture for collaborative filtering of netnews," *Proceedings of the ACM 1994 Conference on Computer Supported Cooperative*, (1994), 175-186.

[13] Shardanand, U. and P. Maes, "Social information filtering: algorithms for automating word of mouth," *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, (1995), 210-217.

[14] Sharma, S., *Applied multivariate techniques*, New York: John Wiley & Sons, (1995), 58-87.

[15] Weiss, S. M. and N. Indurkhya, "Lightweight collaborative filtering method for binary-encoded data," *Lecture Notes in Computer Science* 2168 (2001), 484-491.

APPENDIX

Table A1. Eigenvalues of the first 50 principal components

EachMovie									
1	**8.930**	11	0.356	21	0.261	31	0.216	41	0.186
2	**1.754**	12	0.344	22	0.259	32	0.214	42	0.185
3	**1.348**	13	0.325	23	0.250	33	0.210	43	0.184
4	**0.976**	14	0.307	24	0.245	34	0.207	44	0.180
5	**0.872**	15	0.299	25	0.241	35	0.204	45	0.180
6	**0.761**	16	0.289	26	0.236	36	0.200	46	0.177
7	**0.533**	17	0.285	27	0.234	37	0.196	47	0.176
8	**0.453**	18	0.278	28	0.232	38	0.195	48	0.173
9	**0.406**	19	0.273	29	0.229	39	0.192	49	0.171
10	**0.394**	20	0.272	30	0.225	40	0.189	50	0.170

MovieLenz									
1	**6.436**	**11**	**0.723**	21	0.423	31	0.339	41	0.291
2	**5.568**	**12**	**0.609**	22	0.404	32	0.334	42	0.290
3	**3.404**	**13**	**0.603**	23	0.387	33	0.331	43	0.289
4	**2.852**	**14**	**0.552**	24	0.379	34	0.323	44	0.284
5	**1.825**	**15**	**0.538**	25	0.371	35	0.319	45	0.283
6	**1.647**	**16**	**0.519**	26	0.361	36	0.314	46	0.277
7	**1.305**	**17**	**0.505**	27	0.356	37	0.310	47	0.275
8	**1.146**	**18**	**0.452**	28	0.351	38	0.305	48	0.272
9	**1.023**	19	0.436	29	0.347	39	0.300	49	0.272
10	**0.830**	20	0.430	30	0.341	40	0.298	50	0.270

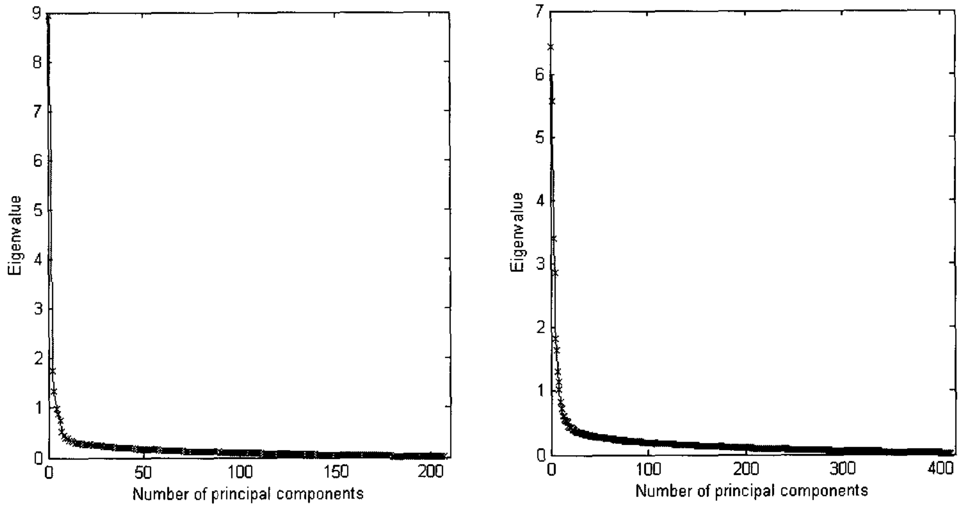
Note: 1. The eigenvalues corresponding to first 50 principal components are listed in the order of their magnitude.

2. The bold figures are eigenvalues of the principal components that are used in the discriminant model.

Table A2. Regularization parameters

EachMovie														
Model	λ	γ	Model	λ	γ	Model	λ	γ	Model	λ	γ	Model	λ	γ
1	0.0	0.0	11	1.0	0.0	21	1.0	0.0	31	0.5	0.1	41	1.0	0.1
2	1.0	0.1	12	0.9	0.0	22	0.8	0.0	32	0.6	0.0	42	1.0	0.0
3	1.0	0.0	13	1.0	0.0	23	1.0	0.0	33	1.0	0.0	43	1.0	0.0
4	1.0	0.0	14	1.0	0.0	24	0.0	0.0	34	1.0	0.0	44	1.0	0.1
5	1.0	0.0	15	1.0	0.0	25	0.5	0.2	35	0.0	0.4	45	1.0	0.0
6	0.9	0.0	16	1.0	0.0	26	1.0	0.0	36	1.0	0.0	46	1.0	0.2
7	0.3	0.0	17	1.0	0.1	27	0.6	0.0	37	0.6	0.0	47	1.0	0.0
8	1.0	0.0	18	1.0	0.0	28	0.5	0.2	38	1.0	0.0	48	0.0	0.0
9	1.0	0.0	19	1.0	0.0	29	1.0	0.7	39	1.0	0.0	49	0.0	0.0
10	1.0	0.0	20	1.0	0.0	30	1.0	0.8	40	0.8	0.3	50	1.0	0.1

MovieLenz														
Model	λ	γ	Model	λ	γ	Model	λ	γ	Model	λ	γ	Model	λ	γ
1	0.6	0.0	19	1.0	0.0	37	0.8	0.0	55	1.0	0.0	73	0.8	0.7
2	0.8	0.2	20	0.4	0.2	38	0.4	0.2	56	1.0	0.2	74	0.4	0.3
3	1.0	0.0	21	0.8	0.0	39	1.0	0.0	57	0.9	0.0	75	0.3	0.3
4	0.9	0.0	22	0.9	0.0	40	0.8	0.0	58	0.7	0.3	76	1.0	0.1
5	1.0	0.1	23	0.7	0.0	41	0.8	0.1	59	0.8	0.0	77	1.0	0.0
6	1.0	0.0	24	0.8	0.2	42	1.0	0.0	60	0.7	0.0	78	0.3	0.5
7	0.6	0.0	25	0.9	0.0	43	0.5	0.2	61	0.5	0.2	79	1.0	0.0
8	0.7	0.1	26	0.9	0.0	44	0.8	0.0	62	0.8	0.1	80	0.2	0.3
9	0.5	0.0	27	1.0	0.0	45	0.7	0.0	63	1.0	0.1	81	0.7	0.2
10	1.0	0.0	28	0.5	0.0	46	0.9	0.0	64	1.0	0.2	82	0.9	0.0
11	0.9	0.0	29	0.6	0.0	47	0.0	0.4	65	1.0	0.0	83	0.7	0.0
12	0.9	0.0	30	0.8	0.1	48	0.6	0.1	66	0.4	0.3	84	0.8	0.0
13	0.7	0.5	31	0.5	0.2	49	1.0	0.0	67	0.0	0.4	85	0.8	0.1
14	0.4	0.2	32	1.0	0.0	50	0.5	0.4	68	1.0	0.0	86	1.0	0.2
15	0.8	0.1	33	0.8	0.0	51	1.0	0.0	69	0.8	0.0	87	0.9	0.0
16	0.0	0.3	34	0.9	0.0	52	0.6	0.0	70	0.9	0.0	88	1.0	0.0
17	0.9	0.4	35	0.0	0.4	53	1.0	0.2	71	1.0	0.1	89	0.8	0.0
18	0.8	0.0	36	1.0	0.2	54	0.8	0.0	72	1.0	0.0			



EachMovie MovieLenz

Figure A1. Scree plot of eigenvalues