CS 181 Machine Learning
Practical 3 Report, Team *Deep Ellum*

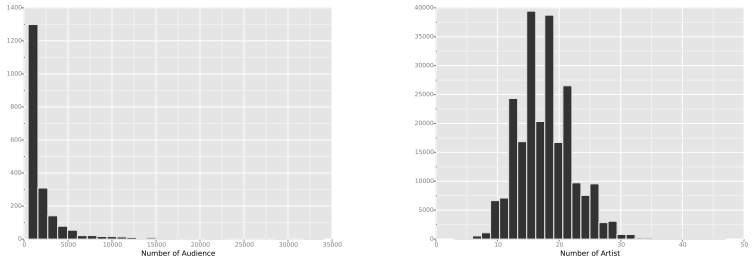
(Vivian) Wenwan Yang[1], Jing Wen[2], and (Jeremiah) Zhe Liu[2]

[1]Department of Computational Science and Engineering, SEAS
[2]Department of Biostatistics, Harvard School of Public Health

April 10, 2015

1 Exploratory Analysis

The training dataset features the gender, age, and country of origin of 233286 users, as well as their count of plays of tracks from 2000 artists. The current objective is to predict the unobserved count of plays for user-artists pairs.
The empirical distribution of number of artists listened (per user) and number of users listened (per artists) are shown in Figure 1. As shown, the data exhibits extremely sparse structure, with each user listening to only 5-50 out of artistis, and each artists were listened by at most 35000 out of 233286 users. Such sparsity in observation among the entrire user-artist count space will induce an user-artist count matrix with extremely sparse structure. Rendering traditional algorithms geared toward dense matrix with few missing data problematic.



(a) Number of Audience per Artist

(b) Number of Artist per Audience

Figure 1: Distribution of the number of audience per artist (**Left**) and the number of artist per audience (**Right**)

The marginal distribution of listening counts among all user-artists pairs were also considered and shown in Figure 2. As shown in the histogram (Left), the log-transformed listening counts displayed roughly symmetric distribution with long tails to the positive direction, which indicates severe right-skewness in the distribution of raw count, and also extreme-valued outliers in the positive direction.

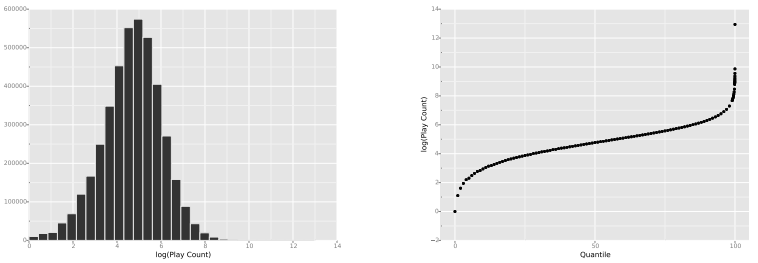


Figure 2: Histogram (Left) and Quantile plot (Right) of $\log(\text{Play Count})$

The distribution of outliers relative to the rest of the data is illustrated by the quantile plot of $\log(\text{Play Count})$ (Figure 2, right). As shown, while 99% of the data remained below 8 on log scale (or < 3000 on original scale), the log count increased dramatically during the $0.99 - 1.00$ quntile range from 8 up to 13 (or

from ~3000 to ~45000 on original scale), indicating a small amount ($< 1\%$) extreme listening behaviors. An empirical check indicates such counts come from either an avaracious user with high total listening count, or a focused user who listens to primarily one or two musician, or the combination of the above two cases. Such situation calls for the need of normalization during modelling stage.

2 Method

2.1 Rationale on Model Choice

In previous section we have identified following characteristics of the task at hand:

- Information Available:
    (a) Basic Demographic covariates from User
    (b) Large (233286×2000) User-Artist count matrix, with extremely sparse entries
- Data Distribution: Right-skewed distribution with extreme outliers
- Goal: Predict the unobserved play counts between users/artists observed in training set, with mean absolute error (MAE) as outcome metric.

Due to the lack of additional user/artist specific features, current task requires drawing inference on unobserved play counts using primarily observed counts, hence put the problem into the unsupervised setting.
However, most of the traditional, distance-based techniques (e.g. K-means/nearest neighbors, PCA/SVD) fail in our setting due to the sparsity-induced difficulty in defining distance, i.e. there does not exist a set of musician who are listened by most of the users. Yet imputation is intractable as it significantly increases the amount of data. In addition, the data may be considerably distorted due to inaccurate imputation.
Given such sitation, we choose to adopt a matrix-factorization approach to model directly only the observed play counts. Shown to be particularly effective in the famous Netflix competition [2], if denote the $(i,j)^{th}$ entry of the count matrix as $r_{i,j}$, such methods simulates the reconstructive view of traditional PCA approach by modelling the counts as:

$$r_{ij} = \mu_{ij} + \mathbf{q}_i^\top \mathbf{p}_j$$

where $\mathbf{p}$ and $\mathbf{q}$ are $f \times 1$ vectors which corresponds to the eigenvalue/vector in traditional PCA, and has the interpretation of "latent factors" which we will explain further in the following section. We see that since modelling is proceeded with an element-wise fashion, hence allowing us to consider only the observed entry in the count matrix.
In the next section, we present our implementation of this method. We will show how elements of above model is defined and interpreted in our current problem, and how to better model $(\mu_{ij}, \mathbf{q}_i, \mathbf{p}_j)$ by refining their structure and incorporate user-specific demographic informations. Furthermore, since such modification to PCA results in a difficult non-convex optimization problem [3], we also discuss practical issues in the estimation of $(\mu_{ij}, \mathbf{q}_i, \mathbf{p}_j)$, where a well-initiated stochastic gradient descent on normalized play counts seem to work reasonably well in our setting.

2.2 Model Definition and Interpretation

For each user u and artist i, we assume:

$$\begin{aligned} \text{count}_{ui} &\sim \text{Poisson}(r_{ui}) \\ r_{ui} &= \mu_{ui} + \mathbf{q}_i^\top \mathbf{p}'_u \end{aligned} \tag{1}$$

where r_{ui} measures the underlying "mean" preference by user u of artist i, where high values indicate stronger preference.

Modeling Systematic Preference

The parameter μ_{ui} measures the "baseline preference" that capture, relative to the population mean count (denoted as μ), the systematic tendencies for some artists to receive more numbers of plays than others (denoted as *artist-specific bias*, b_i), and for some users to play more often than others (denoted as *user-specific bias*, b_u). We also believe that the user-specific demographic information (Age, Gender, Country) would help explaining the user-specific tendency, which lead to below refinement of μ_{ij}:

$$\mu_{ui} = \mu + b_\mu + b_i + \beta_1 \text{Gender} + \beta_2 \text{Country} + \beta_1 \text{Age} \tag{2}$$

Modeling User Preference

The pair $(\mathbf{q}_i, \mathbf{p}'_u) \in \mathbb{R}^f \times \mathbb{R}^f$ indicates latent characteristics of artist i that is not explicitly measured ($\mathbf{q}_i$), and user u's preferences for these latent factors ($\mathbf{p}'_u$). For example, these latent factors might measure obvious dimensions such as genre and style; less well defined dimensions such as depth of the lyrics or structure of rhythm patterns; or completely uninterpretable dimensions. Hence the dot product, $\mathbf{q}_i^\top \mathbf{p}'_u$ represents the interaction between artist i and user u, where high values mean stronger overall interest of the user in the artist's characteristics. In addition, the user preference $\mathbf{p}'_u$ can be further augmented by recognizing the fact that the user u had expressed "implicit" preference to musician i just by listening to this musician. This idea is formalized in Koren (2008) [5] and leads to below refinement of $\mathbf{p}'_u$:

$$\mathbf{p}'_u = \mathbf{p}_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} \mathbf{y}_j \tag{3}$$

where $R(u)$ indicates the set of artists listened by user u, and $\mathbf{y}_j \in \mathbb{R}^f$ measures the aforementioned "implicit" preference.

Combine (1), (2), (3), we have reached below prediction rule:

$$\hat{r}_{ui} = E(\text{count}_{ui}) = \left[\mu + b_\mu + b_i + \beta_1 \text{Gender} + \boldsymbol{\beta}_2 \text{Country} + \beta_3 \text{Age}\right] + \mathbf{q}_i^\top \left(\mathbf{p}_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} \mathbf{y}_j\right)$$

In practise, however, we discarded the **Age** covariate due to high amount of missing/erroneous entries, and heuristically grouped **Country** into 8 continent-based categories[1] in order to reduce model dimension:

$$\hat{r}_{ui} = E(\text{count}_{ui}) = \left[\mu + b_\mu + b_i + \beta_1 \text{Gender} + \beta_2 \text{Continent}\right] + \mathbf{q}_i^\top \left(\mathbf{p}_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} \mathbf{y}_j\right) \tag{4}$$

[1] Continent = { Northern Europe, Europe, Africa, Middle East, South/Central Asia, East Asia, North America, South America }

2.3 Estimation

In order to estimate ($\mathbf{b} = \{\mathbf{b}_u, \mathbf{b}_i\}$, $\boldsymbol{\beta} = \{\beta_1, \beta_2\}$, $\{\mathbf{p}_u, \mathbf{q}_i\}$, $\mathbf{y}$), we chose to optimize on L_2 loss due to the availability of convience algorithm (stochastic gradient descent). A naive objective function for (4) would be:

$$\min_{\mathbf{b},\mathbf{q},\mathbf{p},\boldsymbol{\beta},\mathbf{y}} \sum_{(u,i)\in\mathcal{K}} (r_{ui} - \hat{r}_{ui})^2$$

However, notice that such approach will result in an unidentifiable model, since the solutions for $b_\mu + b_i$ and the inner product $\mathbf{q}_i^T \left(\mathbf{p}_u + |R(u)|^{-\frac{1}{2}} \sum_{j\in R(u)} \mathbf{y}_j\right)$ are not unique. However, since the goal is to predict unobserved play count, it is essential to reasonably model the latent variables ($\mathbf{b}, \{\mathbf{p}_u, \mathbf{q}_i\}$) within their plausible range, we hence place L_2 penalty on their norm:

$$\min_{\mathbf{b},\mathbf{q},\mathbf{p},\boldsymbol{\beta},\mathbf{y}} \sum_{(u,i)\in\mathcal{K}} (r_{ui} - \hat{r}_{ui})^2 + \lambda\left(b_i^2 + b_u^2 + \|q_i\|^2 + \|p_u\|^2 + |R(u)|^{-\frac{1}{2}} \sum_{j\in R(u)} \mathbf{y}_j\right)$$

A straightforward stochastic gradient descent hence follows. The optimization involves the algorithm which loops through all numbers of plays in our training data. Denoting $e_{ui} = r_{ui} - \hat{r}_{ui}$ the associated prediction error, and γ the step size in each gradient descent step, we move in the opposite direction of the gradient by modifying parameters as follows:

1. $\boldsymbol{\beta} \leftarrow \boldsymbol{\beta} + \gamma \cdot e_{ui} \cdot \mathbf{x}$ where $\mathbf{x}$ the 9×1 vector indicating user's gender and continent
2. $b_u \leftarrow b_u + \gamma \cdot (e_{ui} - \lambda \cdot b_u)$
3. $b_i \leftarrow b_i + \gamma \cdot (e_{ui} - \lambda \cdot b_i)$
4. $\mathbf{q}_i \leftarrow \mathbf{q}_i + \gamma \cdot (e_{ui} \cdot (\mathbf{p}_u + |R(u)|^{-\frac{1}{2}} \sum_{j\in R(u)} \mathbf{y}_j) - \lambda * \mathbf{q}_i)$
5. $\mathbf{p}_u \leftarrow \mathbf{p}_u + \gamma \cdot (e_{ui} \cdot \mathbf{q}_i - \lambda * \mathbf{p}_u)$
6. $\forall j \in R(u)$:

   $\mathbf{y}_j \leftarrow \mathbf{y}_j + \gamma \cdot (e_{ui} \cdot |R(u)|^{-\frac{1}{2}} \cdot \mathbf{q}_i - \lambda_6 \cdot \mathbf{y}_j)$

2.4 Numerical Challenges & Further Modification

2.4.1 Normalization

As noted in Section 1, the distribution of play counts features extreme outliers (maximum play counts: 419157), which consequently leads to unstable performance of the gradient descent steps. Specifically, an unusually large play counts will led to large e_{ui}, which may led to an usually large updating step thus leading to overshooting issues. In practice, since the e_{ui} is shared by all parameters, the issue of overshooting will quickly propagate as the algorithm loop over all observed counts, and led to `Inf` estimates by the end of iteration 1. We hence decide to optimize on the probability of each user listen to a specific musician, and use the total play count of this user as a weight vector in optimization, i.e. for user u and $R(u)$ the set of musician listened by user u, we have:

$$w_u^0 = \sum_{i\in R(u)} r_{ui}$$
$$p_{ui} = \frac{r_{ui}}{\sum_{i\in R(u)} r_{ui}}$$

Note since only the relative ratio between w_{u0}'s matters in the optimization procedure, we may further normalize $\mathbf{w}^0 = \{w_1^0, \ldots, w_{233286}^0\}$ by its standard error, i.e. $w_u = w_u^0 / \sigma_{\mathbf{w}^0}$.
Thus our prediction rule and objective function becomes:

$$\hat{r}_{ui} = \sigma_{\mathbf{w}^0} * w_u * \hat{p}_{ui} = \sigma_{\mathbf{w}^0} * w_u * \left\{ \left[\mu + b_\mu + b_i + \beta_1 \text{Gender} + \beta_2 \text{Continent} \right] + \mathbf{q}_i^T \left(\mathbf{p}_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} \mathbf{y}_j \right) \right\}$$

$$\min_{\mathbf{b},\mathbf{q},\mathbf{p},\boldsymbol{\beta},\mathbf{y}} \sum_{(u,i) \in \mathcal{K}} w_u * (p_{ui} - \hat{p}_{ui})^2 + \lambda \left(b_i^2 + b_u^2 + \|\mathbf{q}_i\|^2 + \|\mathbf{p}_u\|^2 + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} \mathbf{y}_j \right)$$

Given the non-convex nature [3] of our current problem, it is important to initialize our algorithm the closest possible to the theoretical global minimum. In our implementation, we choose to initialize on user-specific median. Operationally, if denote the median play count for user u as m_u, and global median as m. For all observed (u, i, j) combination, we initialize $(\mu, \mathbf{b}, \boldsymbol{\beta}, \{\mathbf{q}, \mathbf{p}\}, \mathbf{y})$ as:

$$\begin{aligned} \mu &= m, \quad b_u = m_u - m \\ b_i &= \beta_1 = \beta_2 = 0 \\ \mathbf{q}_i &= \mathbf{p}_u = \mathbf{y}_j = \mathbf{0} \end{aligned}$$

Thus if properly tuned, in the worst case, the algorithm is expected to converge at a local minimum that is at as bad as user-specific median.

2.4.2 Parameter Selection

In our current implementation, two parameters (λ, f) appeared to have non-trivial influence on model fit. Specifically, λ determines the upper bound of the square norm of our latent factors $(\mathbf{b}, \boldsymbol{\beta}, \{\mathbf{q}, \mathbf{p}\}, \mathbf{y})$. While a small λ will lead to ill-identified latent factors with arbitrarily large value as discussed in Section 2.3, large λ will bound all latent factors toward 0 and reduce our model estimates to a global median.
f indicates the dimension of latent factors. While a growing number of factor dimensions enables the model to better express complex user- artist interactions, a overly high dimension may lead to the issue of overfitting, hence reduced prediction accuracy. According to recommendation from Koren and Bell [1](Chapter 3), for Netflix data, best predictin performance of the implicit preference augmented model is achieved at $\lambda = 0.02, f = 200$. Since the user rating in Netflix is in the scale of $0 - 5$ and our listening probability p_{ui} ranges between $[0, 1]$, we slightly downscale λ accordingly. As a result, we decide to search for the optimal combination in the joint parameter space $(\lambda, f) \in L \times F$, where $L = [0.004, 0.020]$ and $F = \{10, 20, 50, 100, 150, 200\}$.
If define MAE as $MAE = \sum_{(u,i)} |r_{ui} - \hat{r}_{ui}| / N_{users}$, our results is shown in Table 1 and Figure 3. As shown, it appears the optimal MAE is achieved at $(\lambda = 0.004, f = 100)$, with MAE= 2202.33. outperforming the user median estimates ($MAE \approx 2307.06$).

λ / f	10	20	50	100	150	200
0.004	4500.41	2499.81	2298.69	2202.33	2798.67	2951.46
0.008	3750.61	2950.56	2400.06	2350.54	2400.62	2850.97
0.012	3001.39	2599.22	2970.77	2450.32	2501.98	2450.26
0.016	2250.06	2499.30	2700.20	3001.74	2699.39	2549.48
0.020	2398.99	2147.78	2598.63	2900.39	3100.31	2800.54

Table 1: Model MAE under $(\lambda, f) \in L \times F$

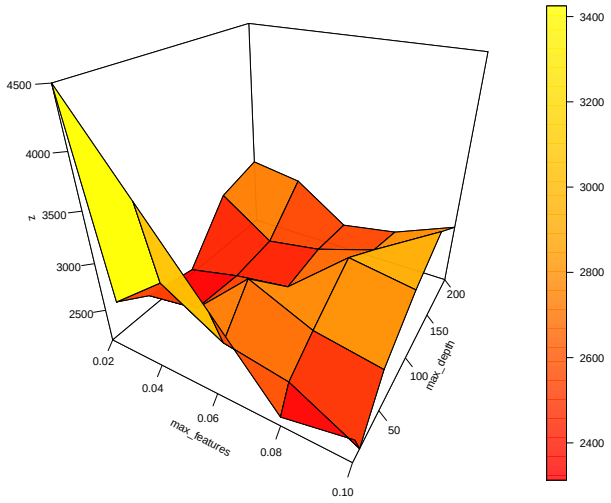


Figure 3: MAE with respect to λ and f

3 Discussion & Possible Directions

Given our current implementation, authors believe that we have achieved maximum usage of information available in training data. However, due to time and resource contraint, we were not able to attempt all planned implementations. Specifically, we find below ideas potentially promising and decribe them as below:

3.1 Optimize on Absolute Error through Linear Programming

If minimize $\text{MAE} = \sum_{(u,i)} |r_{ui} - \hat{r}_{ui}| + \lambda\left(b_i^2 + b_u^2 + \|\mathbf{q}_i\|^2 + \|\mathbf{p}_u\|^2 + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} \mathbf{y}_j\right)$, we may convert them into below linear programing problem:

$$\begin{aligned} \text{minimize} \quad & \sum_{(u,i)} t_{ui} + \lambda\left(b_i^2 + b_u^2 + \|\mathbf{q}_i\|^2 + \|\mathbf{p}_u\|^2 + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} \mathbf{y}_j\right) \\ \text{subject to} \quad & r_{ui} - \hat{r}_{ui} \leqslant t_{ui} \\ & r_{ui} - \hat{r}_{ui} \geqslant -t_{ui} \end{aligned}$$

If set $\lambda = 0$ (i.e. no penalization), such problem can be solved by simplex methods such as Barrodale-Roberts algorithm. However, it is yet not clear if the penalized can solved in a similar way, or if such implementation exists in Python.

3.2 Alternative Objective Function through Hierarchical Model

Due to the nature of play counts, we may pursue a more statistics-oriented approach through an Hierarchical structure (denote $\theta = \{\mathbf{b}, \boldsymbol{\beta}, \{\mathbf{p}_u, \mathbf{q}_i\}, \mathbf{y}\}$) as:

$$\begin{aligned} \text{count}_u &\sim \text{Pois}(r_{ui} | i \in R(u)) \qquad \text{where Pois is a Poisson Process} \\ r_{ui} &= E(\text{count}_{ui} | \mathbf{x}, \theta) = \left[\mu + b_\mu + b_i + \beta_1 \text{Gender} + \beta_2 \text{Continent}\right] + \mathbf{q}_i^T \left(\mathbf{p}_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} \mathbf{y}_j\right) \end{aligned}$$

We may assume conditional exchangeability among $count_{ui}$ observations conditional on the user/artist specific parameters. If given enough thoughts, we should be able to develope a likelihood-based loss based on above construction, with possibility of incoporating Dirichlet Process ideas (since the probability for a user to listen to a specific musician can be considered as an 200d multinomial distribution) and it would be interesting to see if such methods offers better performance. (However, another Bayesian-based idea is formalized in below section)

3.3 Probabilistic Matrix Factorization (PMF)

The deterministic factoring algorithms described above may not be successful to model the number of plays, as the existing algorithms have trouble making accurate predictions for users with very few plays. Alternatively, probabilistic algorithms that scale linearly with the number of observations have been proved to perform well on very sparse and imbalanced datasets, such as our Streaming Music dataset.

Denote the number of plays of user u for artist i by R_{ui}. Suppose we have M artists and N users. Let $U \in R^{D*N}$ and $V \in R^{D*M}$ be latent user and artist feature matrices, with column vectors U_u and V_i representing user-specific and artist-specific latent feature vectors respectively. We can define the conditional distribution over the observed numbers of plays as

$$p(R|U,V,\sigma^2) = \prod_{u=1}^{N}\prod_{i=1}^{M}\left[\mathcal{N}(R_{ui}|U_u^T V_i, \sigma^2)\right]^{I_{ui}}$$

where $\mathcal{N}(x|\mu,\sigma^2)$ represents the probability density function of the Gaussian distribution with mean μ and variance σ^2, and I_{ui} is the indicator function that is equal to 1 if user u listened to the track from artist i and equal to 0 otherwise. We also place zero-mean spherical Gaussian priors on user and artist feature vectors:

$$p(U|\sigma_U^2) = \prod_{i=1}^{N}\mathcal{N}(U_u|0,\sigma_U^2 I), \qquad p(V|\sigma_V^2) = \prod_{j=1}^{M}\mathcal{N}(V_i|0,\sigma_V^2 I)$$

The log of the posterior distribution over the use and the artist feature is:

$$\begin{aligned} ln p(U,V|R,\sigma^2,\sigma_V^2,\sigma_U^2) = &-\frac{1}{2\sigma^2}\sum_{u=1}^{N}\sum_{i=1}^{M} I_{ui}(R_{ui}-U_u^T V_i)^2 - \frac{1}{2\sigma_U^2}\sum_{u=1}^{N} U_u^T U_u - \frac{1}{2\sigma_V^2}\sum_{i=1}^{M} V_i^T V_i \\ &-\frac{1}{2}\Big(\Big(\sum_{u=1}^{N}\sum_{i=1}^{M} I_{ui}\Big) ln\sigma^2 + ND ln\sigma_U^2 + MD ln\sigma_V^2\Big) + C \end{aligned}$$

Maximizing the above log-posterior over artist and user features is equivalent to minimizing the below sum-of-squared-errors objective function:

$$E = \frac{1}{2}\sum_{u=1}^{N}\sum_{i=1}^{M} I_{ij}(R_{ij}-U_i^T V_j)^2 + \frac{\lambda_U}{2}\sum_{i=1}^{N}||U_u||_{Fro}^2 + \frac{\lambda_V}{2}\sum_{i=1}^{M}||V_j||_{Fro}^2$$

where $\lambda_U = \sigma^2/\sigma_U^2$, $\lambda_V = \sigma^2/\sigma_V^2$, and $\|\cdot\|_{Fro}^2$ represents the Frobenius norm. This model is a probabilistic extension of the SVD model, since the objective function reduces to the SVD objective in the limit of prior variances going to infinity.

Additionally, the PMF offers other desirable extensions such as automatic complexity control through spherical prior on user and artist feature vectors. Unfortunately, current implementation is available only in Matlab.

Reference

1. Ricci F, Rokach L, Shapira B et al. (2010) **Recommender Systems Handbook**. *Springer*.

2. Koren Y, Bell R, Volinsky C. (2009) **Matrix factorization techniques for recommender systems**. *IEEE Computer* Aug 2009, 42-49.

3. Srebro N, Jaakkola T.(2003) **Weighted low-rank approximations**. *Proceedings of the Twentieth International Conference* 720727.

4. R Salakhutdinov, A Mnih. (2008) **Probabilistic Matrix Factorization**. *Advances in Neural Information Processing Systems* Vol. 20

5. Koren, Y. (2008) **Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Model**, *Proc. 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.