이학석사 학위논문

The Bayesian Mallows Rank Model for Individual Recommendation

베이즈 맬로우 순위 모형을 통한 개별 추천

2023 년 8 월

서울대학교 대학원
통계학과 통계학 (학과 및 전공)
장태영 (논문 작성자)

The Bayesian Mallows Rank Model for Individual Recommendation

지도교수 이재용

이 논문을 이학석사 학위논문으로 제출함

2023 년 4 월

서울대학교 대학원

통계학과 통계학 (학과 및 전공)

장태영 (논문 작성자)

장태영의 석사 학위논문을 인준함

2023 년 6 월

위 원 장 오희석 (인)

부위원장 이재용 (인)

위 원 원중호 (인)

The Bayesian Mallows Rank Model for Individual Recommendation

Taeyoung Chang

Abstract

The Bayesian Mallows Rank Model for Individual Recommendation

Taeyoung Chang
Department of Statistics
The Graduate School
Seoul National University

We introduce the Bayesian Mallows rank model suggested by Vitelli et al. (2018), which is utilized for analyzing ranking data. We observe how it works for the basic setting where complete ranking data are given and how it can be extended to more general setting. It leads to the part that explains how individual recommendation works well via this method. In a simulation to predict missing individual preference, our recommendation model shows better accuracy than random draw. We apply our approach to a movie rating data of individual users.

Keywords: Mallows rank model, Individual recommendation, Bayesian inference, Markov Chain Monte Carlo, Data augmentation

Student Number: 2021-21116

Contents

1 Introduction

In recent years, analyzing rank or preference data (for example, movies, music, restaurants) has been receiving much attention. Vitelli et al. [2018] developed the Bayesian framework for inference in the Mallows rank model which is among the most successful approaches to analyze rank data. The main advantage of the Bayesian approach is that probabilistic interpretation for the result of the analysis is available.

In Vitelli's paper, the authors wrote that typical tasks for rank or preference data are given as the following :

(i) Summarize multiple individual rankings to estimate the consensus ranking

(ii) Partition the assessors into clusters each sharing a consensus ranking of the items

(iii) Predict the ranks of unranked items at the individual level

The main interest of this paper lies in the third task. We suggest an individual recommendation system exploiting the Bayesian Mallows rank model.

In sections 2.1 and 2.2, we introduce the Bayesian Mallows model for rank data and how the MCMC algorithm for the posterior can be constructed. In section 2.3, we describe a mixture model to handle the heterogeneity of assessors. In section 2.4, we discuss the method to approximate the partition function in case of the exact partition function is not available. In section 3, we extend the Bayesian Mallows approach to partial rankings using data augmentation techniques and suggest an individual recommendation system utilizing the model. In section 4.1, a simulation for measuring the accuracy of our recommendation model for the case where the top-k ranking is performed. Section 4.2 is dedicated to the application of our recommendation model to a movie rating data of individual users.

2 Basic Model

2.1 Mallows Rank Model

For the elementary setting, we denote n as the number of items and N as the number of assessors. Define R_{ij} as the rank of the i-th item assessed by the j-th assessor. Then define an n-dimensional permutation vector

$$\mathbf{R}_j = (R_{1j}, R_{2j}, \cdots, R_{nj}) \in \mathcal{P}_n$$

where $\mathcal{P}_n$ is a set of all n-dimensional permutation vectors. Then we can say that $\mathbf{R}_j$ denotes the ranking vector of assessor j for all n items.

For measuring the distance between two ranking vectors, we shall use Footrule distance or Spearman's distance. We define it as

$$d(\cdot\ ,\ \cdot) : \mathcal{P}_n \times \mathcal{P}_n \longrightarrow [0, \infty)$$

where for the Footrule distance it is the same as ℓ_1 distance while for the Spearman's distance it is equivalent to the square of ℓ_2 distance in the Euclidean space.

We can observe that those distances are right-invariant distances so that for any ranking vector $\mathbf{r}$ and $\boldsymbol{\rho}$, it holds that $d(\mathbf{r}, \boldsymbol{\rho}) = d(\mathbf{r}\boldsymbol{\rho}^{-1}, \mathbf{1}_n)$ where $\mathbf{r} \mapsto \mathbf{r}\boldsymbol{\rho}^{-1}$ is a relabelling map by $\boldsymbol{\rho}$ and $\mathbf{1}_n$ is defined as $(1, 2, \cdots, n)$. Note that both the Footrule distance and Spearman's distance are right-invariant.

As the main model, we shall introduce the Mallows rank model. It is a class of non-uniform distribution for a ranking vector $\mathbf{r}$ on $\mathcal{P}_n$ given as below :

$$P(\mathbf{r}|\alpha, \boldsymbol{\rho}) = Z_n(\alpha, \boldsymbol{\rho})^{-1} \exp\big\{ -\frac{\alpha}{n} d(\mathbf{r}, \boldsymbol{\rho})\big\} \mathrm{I}(\mathbf{r} \in \mathcal{P}_n)$$

Here, we have two parameters $\boldsymbol{\rho}_{\in} \mathcal{P}_n$ and $\alpha > 0$. $\boldsymbol{\rho}$ is the latent consensus ranking while α represents the level of agreement between assessors so that as α gets larger, randomly generated ranking $\mathbf{r}$'s aggregate more to $\boldsymbol{\rho}$. In short, $\boldsymbol{\rho}$ is similar to the location parameter μ and α is similar to the precision parameter $\tau = 1/\sigma^2$ in normal distribution $\mathcal{N}(\mu, \sigma^2)$ Note that

$$Z_n(\alpha, \boldsymbol{\rho}) = \sum_{\mathbf{r} \in \mathcal{P}_n} \exp\big\{ -\frac{\alpha}{n} d(\mathbf{r}, \boldsymbol{\rho})\big\}$$

is called as the partition function.

Under the right-invariant distance assumption, the partition function $Z_n(\alpha, \boldsymbol{\rho})$ no longer depends on $\boldsymbol{\rho}$ so we can denote it as $Z_n(\alpha)$. The reason can be shown below :

$$\begin{aligned} Z_n(\alpha, \boldsymbol{\rho}) &= \sum_{\mathbf{r} \in \mathcal{P}_n} \exp\big\{ -\frac{\alpha}{n} d(\mathbf{r}, \boldsymbol{\rho})\big\} = \sum_{\mathbf{r} \in \mathcal{P}_n} \exp\big\{ -\frac{\alpha}{n} d(\mathbf{r}\boldsymbol{\rho}^{-1}, \mathbf{1}_n)\big\} \\ &= \sum_{\mathbf{r}' \in \mathcal{P}_n} \exp\big\{ -\frac{\alpha}{n} d(\mathbf{r}', \mathbf{1}_n)\big\} = \sum_{\mathbf{r} \in \mathcal{P}_n} \exp\big\{ -\frac{\alpha}{n} d(\mathbf{r}, \mathbf{1}_n)\big\} = Z_n(\alpha) \end{aligned}$$

Hence, from now on, we will denote the partition function as $Z_n(\alpha)$.

Since we exploit the Bayesian approach, we need to assume prior distributions for parameters $\boldsymbol{\rho}$ and α. For $\boldsymbol{\rho}$, we employ the uniform prior $\pi(\boldsymbol{\rho}) = \frac{1}{n!}\mathrm{I}(\boldsymbol{\rho} \in \mathcal{P}_n)$ and for the scale parameter, we use exponential prior $\pi(\alpha|\lambda) = \lambda e^{-\lambda\alpha}$ with a fixed value close to zero assigned for λ.

By the Bayes theorem, the posterior distribution for $\boldsymbol{\rho}$ and α given N observed ranks $\mathbf{R}_1, \cdots, \mathbf{R}_N$ satisfies the below :

$$P(\boldsymbol{\rho}, \alpha|\mathbf{R}_1, \cdots, \mathbf{R}_N) \propto \frac{\pi(\boldsymbol{\rho})\pi(\alpha)}{Z_n(\alpha)^N} \exp\big\{ -\frac{\alpha}{n} \sum_{j=1}^{N} d(\mathbf{R}_j, \boldsymbol{\rho})\big\}$$

2.2 Metropolis-Hastings Algorithm for The Posterior

Our purpose is to obtain samples from the posterior above. For this, we shall take advantage of the Metropolis-Hastings algorithm.

A general form of the Metropolis-Hastings algorithm is as follows: (Hoff [2009]) suppose the target probability distribution to approximate is $p(x)$ for a random variable X .

(a) Given a current value $x^{(s)}$ of X, generate x^* from a proposal distribution $J(x^*|x^{(s)})$

(b) Compute the acceptance ratio

$$r = \frac{p(x^*)}{p(x^{(s)})} \Big/ \frac{J(x^*|x^{(s)})}{J(x^{(s)}|x^*)}$$

(c) set $x^{(s+1)}$ to x^* with probability $\min(1, r)$

Back to the mallows rank model, to obtain samples from the posterior distribution, we shall alternate between two steps below :

(1) Given $\boldsymbol{\rho}$ and α , update $\boldsymbol{\rho}$ by proposing $\boldsymbol{\rho}'$

(2) Then, given α and $\boldsymbol{\rho}$, update α by proposing α'

For updating $\boldsymbol{\rho}$, we utilize Leap-and-Shift (L&S) proposal for the proposal distribution of the Metropolis-Hastings algorithm. It proceeds as the below :

(i) Draw a random number $u \sim Unif\{1, 2, \cdots, n\}$

(ii) Define $\mathcal{S} \subset \{1, 2, \cdots, n\}$ by $\mathcal{S} = \big[\max(1, \rho_u - L), \min(n, \rho_u + L)\big] - \{\rho_u\}$

(iii) Draw a random number $v \sim Unif(\mathcal{S})$

(iv) Let $\boldsymbol{\rho}^* \in \{1, 2, \cdots, n\}^n$ have elements $\begin{cases} \rho_i^* = \rho_i & i \in \{1, 2, \cdots, n\} - \{u\} \\ \rho_u^* = v \end{cases}$

(v) Let $\Delta = \rho_u^* - \rho_u$. Note that $\Delta \neq 0$

(vi) Define the proposed $\boldsymbol{\rho}' \in \mathcal{P}_n$ by below :

(a) If $\Delta > 0$ then

$$\begin{cases} \rho_u' = \rho_u^* \\ \rho_i' = \rho_i - 1 & \text{if } \rho_u < \rho_i \leq \rho_u^* \\ \rho_i' = \rho_i & \text{otherwise} \end{cases}$$

(b) If $\Delta < 0$ then

$$\begin{cases} \rho_u' = \rho_u^* \\ \rho_i' = \rho_i + 1 & \text{if } \rho_u > \rho_i \geq \rho_u^* \\ \rho_i' = \rho_i & \text{otherwise} \end{cases}$$

Step (i) - (iv) constitute the leap step while step (v) and (vi) constitute the shift step. Note that an integer $L \in \{1, 2, \cdots, \lfloor \frac{n-1}{2} \rfloor\}$ is fixed as a tuning parameter for the MCMC algorithm where 'L' stands for the leap size. For proposed $\boldsymbol{\rho}'$ and current $\boldsymbol{\rho}$, we should calculate transition probability $P_L(\boldsymbol{\rho}'|\boldsymbol{\rho})$ and $P_L(\boldsymbol{\rho}|\boldsymbol{\rho}')$ to derive the acceptance ratio of updating $\boldsymbol{\rho}$ in the Metropolis-Hastings algorithm.

As we calculate $P_L(\boldsymbol{\rho}'|\boldsymbol{\rho})$, we should consider two random draws ;
drawing $u \sim Unif\{1, 2, \cdots, n\}$ in step 1 and drawing $v \sim Unif(\mathcal{S})$ in step 3 where $\mathcal{S}$ depends on $\boldsymbol{\rho}_u$. Simply put, $P_L(\boldsymbol{\rho}'|\boldsymbol{\rho}) = \frac{1}{n} \cdot \frac{1}{|\mathcal{S}|}$ for typical cases. However, if $|\boldsymbol{\rho}_u' - \boldsymbol{\rho}_u| = 1$

then we should consider something more. When $|\boldsymbol{\rho}'_u - \boldsymbol{\rho}_u| > 1$ then u is the only possible index that proposes $\boldsymbol{\rho}'$ from $\boldsymbol{\rho}$. On the other hand, when $|\boldsymbol{\rho}'_u - \boldsymbol{\rho}_u| = 1$, there must be one index $\tilde{u}$ other than u s.t. $|\boldsymbol{\rho}'_{\tilde{u}} - \boldsymbol{\rho}_{\tilde{u}}| = 1$ so that $\tilde{u}$ becomes another index that can produce $\boldsymbol{\rho}'$ from $\boldsymbol{\rho}$. In this case, $P_L(\boldsymbol{\rho}'|\boldsymbol{\rho}) = \frac{1}{n} \cdot \frac{1}{|\mathcal{S}|} + \frac{1}{n} \cdot \frac{1}{|\tilde{\mathcal{S}}|}$ where $\mathcal{S}$ is produced from drawing u while $\tilde{\mathcal{S}}$ is produced from drawing $\tilde{u}$.

Using this logic, we can rewrite the equality about $P_L(\boldsymbol{\rho}'|\boldsymbol{\rho})$ as the following :

$$\begin{aligned} P_L(\boldsymbol{\rho}'|\boldsymbol{\rho}) &= \sum_{u=1}^{n} P_L(\boldsymbol{\rho}'|U = u, \boldsymbol{\rho}) P(U = u) \\ &= \frac{1}{n} \sum_{u=1}^{n} \mathrm{I}(\boldsymbol{\rho}', \boldsymbol{\rho}, u) \frac{1}{|\mathcal{S}^{(u)}|} \end{aligned}$$

where $\mathrm{I}(\boldsymbol{\rho}', \boldsymbol{\rho}, u)$ is an indicator for the possibility of the proposal from $\boldsymbol{\rho}$ to $\boldsymbol{\rho}'$ given u is drawed and $\mathcal{S}^{(u)}$ is the set $\mathcal{S}$ given u is drawn. If $\boldsymbol{\rho}'$ is proposed from $\boldsymbol{\rho}$ then typically $\mathrm{I}(\boldsymbol{\rho}', \boldsymbol{\rho}, u) = 1$ for only one u but if $|\boldsymbol{\rho}'_u - \boldsymbol{\rho}_u| = 1$ then $\mathrm{I}(\boldsymbol{\rho}', \boldsymbol{\rho}, \tilde{u}) = 1$ also holds for another $\tilde{u}$ different from u.

Hence, the acceptance probability for updating $\boldsymbol{\rho}$ is equal to $\min(1, r)$ where r is given as below :

$$\begin{aligned} r &= \frac{P(\boldsymbol{\rho}', \alpha|\mathbf{R})}{P(\boldsymbol{\rho}, \alpha|\mathbf{R})} \cdot \frac{P_L(\boldsymbol{\rho}|\boldsymbol{\rho}')}{P_L(\boldsymbol{\rho}'|\boldsymbol{\rho})} \\ &= \frac{P_L(\boldsymbol{\rho}|\boldsymbol{\rho}')}{P_L(\boldsymbol{\rho}'|\boldsymbol{\rho})} \cdot \frac{\pi(\boldsymbol{\rho}')}{\pi(\boldsymbol{\rho})} \exp\Big\{ -\frac{\alpha}{n} \sum_{j=1}^{N} \big[d(\mathbf{R}_j, \boldsymbol{\rho}') - d(\mathbf{R}_j, \boldsymbol{\rho}) \big] \Big\} \end{aligned}$$

where $\mathbf{R}$ represents all observed rankings $(\mathbf{R}_1, \cdots, \mathbf{R}_N)$

Next, we need to update α. Here, log-normal distribution is used as a proposal distribution. Sample a proposal α' from a log-normal distribution $\log\mathcal{N}(\log(\alpha), \sigma_\alpha^2)$ with tuning parameter σ_α^2 fixed. Then the transition probability is calculated as

$$P_{\sigma_\alpha^2}(\alpha'|\alpha) = \frac{1}{\sqrt{2\pi\sigma_\alpha^2}} \exp\big(-\frac{1}{2\sigma_\alpha^2} (\log\alpha' - \log\alpha)^2\big) \frac{1}{\alpha'}$$

Accordingly, we have the ratio $P_{\sigma_\alpha^2}(\alpha'|\alpha) \,/\, P_{\sigma_\alpha^2}(\alpha|\alpha') = \alpha \,/\, \alpha'$. Thus acceptance probability for updating α is equal to $\min(1, r)$ where r is given as below :

$$\begin{aligned} r &= \frac{P(\boldsymbol{\rho}, \alpha'|\mathbf{R})}{P(\boldsymbol{\rho}, \alpha|\mathbf{R})} \Big/ \frac{P_{\sigma_\alpha^2}(\alpha'|\alpha)}{P_{\sigma_\alpha^2}(\alpha|\alpha')} \\ &= \frac{\alpha'}{\alpha} \frac{\pi(\alpha')}{\pi(\alpha)} \frac{Z_n(\alpha)^N}{Z_n(\alpha')^N} \exp\Big\{ -\frac{\alpha' - \alpha}{n} \sum_{j=1}^{N} d(\mathbf{R}_j, \boldsymbol{\rho}) \Big\} \end{aligned}$$

Note that additional parameter α_{jump} can be exploited to update α only every α_{jump} update of $\boldsymbol{\rho}$.

2.3 A Mixture Mallows Model

So far we have assumed that there exists a unique consensus ranking shared by all assessors. However, the possibility of dividing assessors into more homogeneous subsets, each sharing a consensus ranking of the items, brings the model closer to reality.

To handle this idea, we can consider a mixture Mallows model. It can be described as the following :

- $z_j \in \{1, \cdots, C\}$ assigns assessor j to one of C clusters for each assessor $j = 1, \cdots, N$. In other words, $z_1, \cdots, z_N$ are the cluster labels.
- The ranking vectors $\mathbf{R}$'s within each cluster $c \in \{1, \cdots, C\}$ are described by a Mallows model with parameters α_c and $\boldsymbol{\rho}_c$. Here, $\boldsymbol{\rho}_c$ can be viewed as the cluster consensus.

Then the likelihood for the observed rankings $\mathbf{R}_1, \cdots, \mathbf{R}_N$ is derived as

$$P(\mathbf{R}_1, \cdots, \mathbf{R}_N | \{\alpha_c, \boldsymbol{\rho}_c\}_{c=1,\cdots,C}, z_1, \cdots, z_N) = \prod_{j=1}^{N} \frac{1}{Z_n(\alpha_{z_j})} \exp\{-\frac{\alpha_{z_j}}{n} d(\mathbf{R}_j, \boldsymbol{\rho}_{z_j})\}$$

Given the number of clusters C , the parameters and corresponding prior distributions for the mixture model is given in the following :

- $\boldsymbol{\rho}_1, \cdots, \boldsymbol{\rho}_C \sim \pi_{\boldsymbol{\rho}}$ independently where $\pi_{\boldsymbol{\rho}}$ is a uniform prior on $\mathcal{P}_n$ as in the homogeneous setting.
- $\alpha_1, \cdots, \alpha_C \sim \pi_\alpha$ independently where π_α is an exponential prior with fixed λ as in the homogeneous setting.
- $\tau_1, \cdots, \tau_C$ are the probabilities that an assessor belongs to c-th cluster. The standard symmetric Dirichlet prior $(\tau_1, \cdots, \tau_C) \sim \mathcal{D}(\psi, \cdots, \psi)$ is assumed with a fixed value of ψ.
- $z_1, \cdots, z_N$ are the cluster labels as mentioned above. $P(z_j = c \,|\, \boldsymbol{\tau}) = \tau_c$ for each $j = 1, \cdots, N$ and $c = 1, \cdots, C$ where $\boldsymbol{\tau} = (\tau_1, \cdots, \tau_C)$

To obtain MCMC samples from the posterior, the algorithm will alternate between two steps below :

- Sample $\boldsymbol{\rho}_1, \cdots, \boldsymbol{\rho}_C$ and $\alpha_1, \cdots, \alpha_C$ by a Metropolis-Hastings algorithm.
- Sample $\tau_1, \cdots, \tau_C$ and $z_1, \cdots, z_N$ by a Gibbs sampler.

The former is updated in a similar way as in section 2.2. The acceptance probability is slightly changed according to the cluster index c. The latter is divided into two stages. The first one is a Gibbs step for $\boldsymbol{\tau} = (\tau_1, \cdots, \tau_C)$. Since Dirichlet prior is conjugate to the multinomial distribution, $\boldsymbol{\tau} \sim \mathcal{D}(\psi, \cdots, \psi)$ and $(n_1, \cdots, n_C) \,|\, \boldsymbol{\tau} \sim \text{Multinomial}(N, \boldsymbol{\tau})$ (here n_c is defined as $n_c = \sum_{j=1}^{N} \mathrm{I}(z_j = c)$) lead to $\boldsymbol{\tau} \,|\, (n_1, \cdots, n_C) \sim \mathcal{D}(\psi + n_1, \cdots, \psi + n_C)$. Hence we sample $\boldsymbol{\tau}$ from $\mathcal{D}(n_1 + \psi, \cdots, n_C + \psi)$ in the first stage. The second one is a Gibbs step for $(z_1, \cdots, z_N)$. For sampling z_j, we will use the information of current $\boldsymbol{\tau}, \boldsymbol{\rho}, \boldsymbol{\alpha}$ and $\mathbf{R}_j$ where $\boldsymbol{\rho} = (\boldsymbol{\rho}_1, \cdots, \boldsymbol{\rho}_C)$ and $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_C)$

$$
\begin{aligned}
P(z_j = c \,|\, \boldsymbol{\tau}, \boldsymbol{\rho}, \boldsymbol{\alpha}, \mathbf{R}_j) &\propto P(z_j = c \,|\, \boldsymbol{\tau}) P(\mathbf{R}_j \,|\, \boldsymbol{\rho}, \boldsymbol{\alpha}, z_j = c) \quad \because \text{prior} \times \text{likelihood} \\
&= P(z_j = c \,|\, \boldsymbol{\tau}) P(\mathbf{R}_j \,|\, \boldsymbol{\rho}_c, \alpha_c) \\
&= \tau_c Z_n(\alpha_c)^{-1} \exp\Big\{ -\frac{\alpha_c}{n} d(\mathbf{R}_j, \boldsymbol{\rho}_c) \Big\}
\end{aligned}
$$

Using this, we can sample z_j from $P(z_j = c \,|\, \boldsymbol{\tau}, \boldsymbol{\rho}, \boldsymbol{\alpha}, \mathbf{R}_j)$

One thing important for fitting a mixture Mallows model is how to determine the number of clusters C. The number of clusters C is often unknown and the selection of C can be based on different criteria. We take advantage of the within-cluster sum of distances given as the following :

$$
\sum_{c=1}^{C} \sum_{j\,:\,z_j = c} d(\tilde{\mathbf{R}}_j, \boldsymbol{\rho}_c)
$$

We use the posterior MCMC outputs of the within-cluster sum of distances of the observed ranks from the corresponding cluster consensus. After separate analyses were performed for $C = 1, 2, \cdots \mathcal{C}$ for some $\mathcal{C}$, we expect to observe an 'elbow' in the within-cluster distance posterior distribution as a function of C. It leads to identifying the optimum number of clusters.

2.4 Approximation of The Partition Function via Off-line Importance Sampling

Notice that we need to know the value of the partition function $Z_n(\alpha)$ to calculate the acceptance probability in the MCMC algorithm. Recall that with the right-invariant distance assumption, we can represent the partition function $Z_n(\alpha)$ as

$$
Z_n(\alpha) = \sum_{\mathbf{r} \in \mathcal{P}_n} \exp\big\{ -\frac{\alpha}{n} d(\mathbf{r}, \mathbf{1}_n) \big\}
$$

To calculate $Z_n(\alpha)$ for some fixed value α, we can recognize that directly summing over all $\mathbf{r}$ in $\mathcal{P}_n$ is very inefficient since $\mathcal{P}_n$ has $n!$ number of permutation vectors. Note that $d(\mathbf{r}, \mathbf{1}_n)$ takes only the finite number of discrete values $\mathcal{D} = \{d_1, \cdots, d_z\}$ where z depends on n and distance d. We can express the partition function as

$$
Z_n(\alpha) = \sum_{d_i \in \mathcal{D}} |L_i| \exp\big\{ -\frac{\alpha}{n} d_i \big\}
$$

where $L_i = \{\mathbf{r} \in \mathcal{P}_n : d(\mathbf{r}, \mathbf{1}_n) = d_i\}$. To compute $Z_n(\alpha)$, we only need to know $|L_i|$ for all values of d_i in $\mathcal{D}$.

- When d is footrule distance
  - $\mathcal{D}$ includes all even numbers from 0 to $\lfloor n^2/2 \rfloor$

– $|L_i|$ corresponds to the sequence A061869 available for $n \leq 50$ on the OEIS(Online Encyclopedia of Integer Sequences)

- When d is Spearman's distance
  – $\mathcal{D}$ includes all even numbers from 0 to $2\binom{n}{3}$
  – $|L_i|$ corresponds to the A175929 available for $n \leq 14$ on the OEIS

Although the exact value of the partition function $Z_n(\alpha)$ is available for not large n (when $n \leq 50$ or $n \leq 14$ for the case of the footrule distance or the Spearman's distance respectively), we have trouble when n gets larger. To handle it, we need an approximation of the partition function. Here, we take advantage of importance sampling for the approximation.

A general form of importance sampling can be given as the following : (Owen [2009-2013, 2018]) suppose our goal is to estimate $\mu = E_p[f(X)]$ i.e. the expected value of $f(X)$ under $X \sim p$. For a probability density q other than p, we can yield

$$\mu = E_p[f(X)] = \int f(x)p(x)\,dx = \int \frac{f(x)p(x)}{q(x)}q(x)\,dx = E_q\Big[\frac{f(X)p(X)}{q(X)}\Big]$$

i.e. μ equals the expected value of $\frac{f(X)p(X)}{q(X)}$ under $X \sim q$. Then, the importance sampling estimate of μ is

$$\hat{\mu}_q = \frac{1}{K}\sum_{k=1}^{K}\frac{f(X_i)p(X_i)}{q(X_i)} \quad \text{where } X_i \sim q$$

The basic idea of importance sampling is to sample the states from a different distribution when we want to lower the variance of estimation of μ or sampling from original density p is difficult.

Back to the situation of estimating the partition function, given K number of ranking vectors $\mathbf{R}^1, \cdots, \mathbf{R}^K$ sampled from an IS auxiliary distribution $q(\mathbf{R})$, the unbiased IS estimate of $Z_n(\alpha)$ can be written as

$$\hat{Z}_n(\alpha) = \frac{1}{K}\sum_{k=1}^{K}\exp\big\{-\frac{\alpha}{n}d(\mathbf{R}^k, \mathbf{1}_n)\big\}\frac{1}{q(\mathbf{R}^k)}$$

This IS estimate of $Z_n(\alpha)$ is derived as the following

$$\begin{aligned} Z_n(\alpha) &= \sum_{\mathbf{R}\in\mathcal{P}_n}\exp\{-\frac{\alpha}{n}d(\mathbf{R}, \mathbf{1}_n)\} = \sum_{\mathbf{R}\in\mathcal{P}_n}\frac{1}{P(\mathbf{R})}\exp\{-\frac{\alpha}{n}d(\mathbf{R}, \mathbf{1}_n)\}P(\mathbf{R}) \\ &= E_{R\sim P(R)}\Big[\frac{1}{P(\mathbf{R})}\exp\{-\frac{\alpha}{n}d(\mathbf{R}, \mathbf{1}_n)\}\Big] = E_{R\sim P(R)}\big[f(\mathbf{R})\big] \end{aligned}$$

where $f(\mathbf{R}) = \frac{1}{P(\mathbf{R})}\exp\{-\frac{\alpha}{n}d(\mathbf{R}, \mathbf{1}_n)\}$ and $P(\mathbf{R})$ is an abbreviation of the Mallows model density $P(\mathbf{R}|\alpha, \mathbf{1}_n)$. Using the general form of importance sampling estimate above, we get

$$\hat{Z}_n(\alpha) = \frac{1}{K}\sum_{k=1}^{K}\frac{f(\mathbf{R}^k)P(\mathbf{R}^k)}{q(\mathbf{R}^k)} = \frac{1}{K}\sum_{k=1}^{K}\exp\big\{-\frac{\alpha}{n}d(\mathbf{R}^k, \mathbf{1}_n)\big\}\frac{1}{q(\mathbf{R}^k)} \quad \text{where } \mathbf{R}^k \sim q(\mathbf{R})$$

While we cannot sample $\mathbf{R}$ from $P(\mathbf{R}|\alpha, \mathbf{1}_n)$ because the exact value of $Z_n(\alpha)$ is not available, it must be computationally feasible to sample $\mathbf{R}$ from $q(\mathbf{R})$. The more $q(\mathbf{R})$ resembles the Mallows likelihood $P(\mathbf{R}^k|\alpha, \mathbf{1}_n)$, the smaller is the variance of $\hat{Z}_n(\alpha)$. We shall use the following pseudo-likelihood approximation for $q(\mathbf{R})$. It is proceeded as below.

(i) Sample a permutation $(i_1, \cdots, i_n) \in \mathcal{P}_n$ which gives the order of the pseudo-likelihood factorization.

(ii) Factorization is given as

$$q(\mathbf{R}) = P(\mathbf{R}|\mathbf{1}_n) = P(R_{i_n}|\mathbf{1}_n)P(R_{i_{n-1}}|R_{i_n}, \mathbf{1}_n) \cdots P(R_{i_2}|R_{i_3}, \cdots, R_{i_n}, \mathbf{1}_n)P(R_{i_1}|R_{i_2}, \cdots, R_{i_n}, \mathbf{1}_n)$$

(iii) The conditional distributions are given by

$$P(R_{i_n}|\mathbf{1}_n) = \frac{\exp\{-\frac{\alpha}{n}d(R_{i_n}, i_n)\} \cdot \mathrm{I}(R_{i_n} \in \{1, \cdots, n\})}{\sum_{r_n \in \{1,\cdots,n\}} \exp\{-\frac{\alpha}{n}d(r_n, i_n)\}}$$

$$P(R_{i_{n-1}}|R_{i_n}, \mathbf{1}_n) = \frac{\exp\{-\frac{\alpha}{n}d(R_{i_{n-1}}, i_{n-1})\} \cdot \mathrm{I}(R_{i_{n-1}} \in \{1, \cdots, n\} - \{R_{i_n}\})}{\sum_{r_{n-1} \in \{1,\cdots,n\}-\{R_{i_n}\}} \exp\{-\frac{\alpha}{n}d(r_{n-1}, i_{n-1})\}}$$

$$\vdots$$

$$P(R_{i_2}|R_{i_3}, \cdots, R_{i_n}, \mathbf{1}_n) = \frac{\exp\{-\frac{\alpha}{n}d(R_{i_2}, i_2)\} \cdot \mathrm{I}(R_{i_2} \in \{1, \cdots, n\} - \{R_{i_n}, \cdots, R_{i_3}\})}{\sum_{r_2 \in \{1,\cdots,n\}-\{R_{i_n},\cdots,R_{i_3}\}} \exp\{-\frac{\alpha}{n}d(r_2, i_2)\}}$$

$$P(R_{i_1}|R_{i_2}, \cdots, R_{i_n}, \mathbf{1}_n) = \mathrm{I}(R_{i_1} \in \{1, \cdots, n\} - \{R_{i_n}, \cdots, R_{i_3}, R_{i_2}\})$$

Note that each factor is a simple univariate distribution.

For the given value of α, we sample K number of rankings from $q(\mathbf{R})$ so that we get $\mathbf{R}^1, \cdots, \mathbf{R}^K$ which are exploited for calculating $\hat{Z}_n(\alpha)$. Over a discrete grid of 100 equally spaced α values between small numbers as 0.01 and 10 (this is the range of α that turned out to be relevant in many applications), we shall produce a smooth partition function simply using a polynomial of degree 10. What we have is 100 data points of $(\alpha, \hat{Z}_n(\alpha))$'s for these grid points. A smooth partition function is produced by fitting multiple linear regression for the model

$$\log \hat{Z}_n(\alpha) = \beta_0 + \beta_1\alpha + \beta_2\alpha^2 + \cdots + \beta_{10}\alpha^{10}$$

so that for the partition function, only thing we should store before initializing the MCMC algorithm is those estimated beta parameter values.

3 Partial Rankings and Individual Recommendations

Often, only a subset of the items is ranked by each assessor. These situations can be handled conveniently in the Bayesian framework by applying data augmentation techniques. Furthermore, with these MCMC samples of augmented individual rankings, we can suggest a method of individual recommendation.

In this section, we shall see two cases of partial rankings (top-k ranks & pairwise comparison) with corresponding MCMC algorithms to obtain posterior samples and individual recommendation method.

3.1 Ranking of The Top Ranked Items

First, we shall consider the case of the top-k ranks. Assume that among n items $\{A_1, \cdots, A_n\}$, each assessor j has ranked the subset of items $\mathcal{A}_j \subset \{A_1, \cdots, A_n\}$ giving them top ranks from 1 to $n_j = |\mathcal{A}_j|$. We had a complete ranking $\mathbf{R}_j \in \mathcal{P}_n$ before, but now we will denote $\mathbf{R}_j$ as a partial ranking. Also, we will write $\tilde{\mathbf{R}}_j \in \mathcal{P}_n$ as an augmented ranking vector where the unknown part follows uniform prior on the set of permutations of $(n_j + 1, \cdots, n)$ for each $j = 1, \cdots, N$. For simplicity, we assume a homogeneous setting here.

Our goal is to sample from the posterior distribution

$$P(\alpha, \boldsymbol{\rho} \,|\, \mathbf{R}_1, \cdots, \mathbf{R}_N) = \sum_{\tilde{\mathbf{R}}_1 \in \mathcal{S}_1} \cdots \sum_{\tilde{\mathbf{R}}_N \in \mathcal{S}_N} P(\alpha, \boldsymbol{\rho}, \tilde{\mathbf{R}}_1, \cdots, \tilde{\mathbf{R}}_N \,|\, \mathbf{R}_1, \cdots, \mathbf{R}_N)$$

where $\mathcal{S}_j$ is a set of all possible augmented ranking vectors given original partial ranks with the allowable 'fill-ins' of the missing ranks for each $j = 1, \cdots, N$. For this, our MCMC algorithm will alternate between two steps below :

- Sample the augmented ranks given the current values of α and $\boldsymbol{\rho}$
- Sample α and $\boldsymbol{\rho}$ given the current values of the augmented ranks.

The latter is done similarly as in the previous section where in this case $\mathbf{R}_1, \cdots, \mathbf{R}_N$ are replaced by augmented ranks $\tilde{\mathbf{R}}_1, \cdots, \tilde{\mathbf{R}}_N$. For the former, given the current $\tilde{\mathbf{R}}_j$, we propose $\tilde{\mathbf{R}}'_j$ in $\mathcal{S}_j$ from a uniform proposal distribution. Then with the current values of α and $\boldsymbol{\rho}$, the proposed $\tilde{\mathbf{R}}_j$ is accepted with probability $\min(1, r)$ where r is equal to

$$\begin{aligned} r &= \frac{P(\tilde{\mathbf{R}}_1, \cdots, \tilde{\mathbf{R}}'_j, \cdots, \tilde{\mathbf{R}}_N \,|\, \alpha, \boldsymbol{\rho})}{P(\tilde{\mathbf{R}}_1, \cdots, \tilde{\mathbf{R}}_j, \cdots, \tilde{\mathbf{R}}_N \,|\, \alpha, \boldsymbol{\rho})} \\ &= \exp[-\frac{\alpha}{n}\{d(\tilde{\mathbf{R}}'_j, \boldsymbol{\rho}) - d(\tilde{\mathbf{R}}_j, \boldsymbol{\rho})\}] \end{aligned}$$

Note that we can generalize this algorithm to generic partial ranking, where items partially ranked by each assessor are not necessarily the top-ranked items. Also, if we add clustering in the model, then α and $\boldsymbol{\rho}$ in the acceptance ratio are replaced by α_{z_j} and $\boldsymbol{\rho}_{z_j}$ respectively where z_j is the current value of the cluster label for the j-th assessor.

For individual recommendations, we will take advantage of MCMC outputs of augmented ranks $\tilde{\mathbf{R}}_1, \cdots, \tilde{\mathbf{R}}_N$. In the case of top-k ranks, j-th assessor has already picked his or her top-k items among n items. Suppose the j-th assessor has a latent complete ranking for all n items in his mind whether consciously or unconsciously. We can denote it as $\tilde{\mathbf{R}}_{j,true}$. We can assume that when the items are movies or TV programs, it is meaningless to recommend the items that the assessor has already ranked as top-k since probably he or she has already seen those items. Then it is natural that we want to recommend top-ranked items in $\tilde{\mathbf{R}}_{j,true}$ with items in $\mathcal{A}_j$ removed. (Recall that $\mathcal{A}_j$ denotes the set of items already ranked as top-k by the j-th assessor.) For the case of the top-k rank, we can say those items are ranked as $k+1, k+2$, or $k+3$ in $\tilde{\mathbf{R}}_{j,true}$.

We shall find the estimate for $\tilde{\mathbf{R}}_{j,true}$ exploiting the MCMC outputs of augmented rank $\tilde{\mathbf{R}}_j$. Then we will recommend top-ranked items in the estimated ranking with items in $\mathcal{A}_j$ removed. We suggest a cumulative probability(CP) ranking estimate as a single-point estimate for $\tilde{\mathbf{R}}_{j,true}$. Given the MCMC outputs of augmented rank, the CP ranking estimate is derived as the following :

(1) First, select the item which has the maximum a posteriori marginal probability of being ranked first.

(2) Second, select the item which has the maximum a posteriori marginal probability of being ranked first or second among the remaining ones.

(3) Third, select the item which has the maximum a posteriori marginal probability of being ranked first, second, or third among the remaining ones.

(4) Keep following this sequential scheme.

3.2 Pairwise Comparison

Now, we shall consider the case of the pairwise comparison. We often have a situation where assessors compare pairs of items rather than determine ranks of all or a subset of items. This case can be also handled by the Bayesian framework with data augmentation techniques.

Before dealing with how the MCMC algorithm is implemented, we first introduce some notations.

- $A_r \prec A_s$: A_s is preferred to A_r, so that A_s has a higher rank than A_r
- $\mathcal{B}_j$: pairwise orderings or preferences stated by assessor j
- $\mathcal{A}_j$: the set of items constrained by assessor j
- $\text{tc}(\mathcal{B}_j)$: the transitive closure of $\mathcal{B}_j$, containing all pairwise orderings of the elements in $\mathcal{A}_j$ induced by $\mathcal{B}_j$.

Some examples of transitive closure are given as the below :

$$\mathcal{B}_j = \{A_1 \prec A_2, A_2 \prec A_5\} \;\Rightarrow\; \text{tc}(\mathcal{B}_j) = \{A_1 \prec A_2, A_2 \prec A_5, A_1 \prec A_5\}$$
$$\mathcal{B}_k = \{A_1 \prec A_2, A_2 \prec A_5, A_4 \prec A_5\} \;\Rightarrow\; \text{tc}(\mathcal{B}_k) = \{A_1 \prec A_2, A_2 \prec A_5, A_1 \prec A_5, A_4 \prec A_5\}$$

The main idea of the MCMC algorithm remains the same as the one for the case in section 3.1. In the algorithm, we should propose augmented ranking which obeys the partial ordering constraints $\text{tc}(\mathcal{B}_j)$ for each assessor j. If we had decided to imitate the algorithm in section 3.1 the same, then we would propose augmented rank $\tilde{\mathbf{R}}'_j$ in $\mathcal{S}_j$ from a uniform proposal distribution where $\mathcal{S}_j$ is a set of all possible augmented ranking vectors compatible with $\text{tc}(\mathcal{B}_j)$. But instead of doing this, we shall utilize a 'modified' leap-and-shift proposal distribution as a proposal distribution for updating augmented ranks. For the modified version of the L&S proposal, only the leap step is changed from the original one described in section 2.2. It proceeds as the following.

Given a full augmented rank vector $\tilde{\mathbf{R}}_j$ compatible with $\text{tc}(\mathcal{B}_j)$, we shall propose $\tilde{\mathbf{R}}'_j$.

(i) Draw a random number $u \sim Unif\{1, 2, \cdots, n\}$

(ii) If $A_u \notin \mathcal{A}_j$ then complete the leap step by drawing $\tilde{R}^*_{uj} \sim Unif\{1, 2, \cdots, n\}$

(iii) If $A_u \in \mathcal{A}_j$ then complete the leap step by drawing $\tilde{R}^*_{uj} \sim Unif\{l_j + 1, \cdots, r_j - 1\}$ where l_j and r_j are defined by

- $l_j = \max\{\tilde{R}_{kj} : A_k \in \mathcal{A}_j, k \neq u, (A_k \succ A_u) \in \text{tc}(\mathcal{B}_j)\}$ with a convention that $l_j = 0$ if the set is empty
- $r_j = \min\{\tilde{R}_{kj} : A_k \in \mathcal{A}_j, k \neq u, (A_k \prec A_u) \in \text{tc}(\mathcal{B}_j)\}$ with a convention that $r_j = n + 1$ if the set is empty
- Briefly, l_j is the current rank of the item whose rank is closest to A_u among all assessed items preferred to A_u, and r_j is the current rank of the item whose rank is closest to A_u among all assessed items less preferred than A_u.

For individual recommendation, we will again exploit MCMC outputs of augmented ranks $\tilde{\mathbf{R}}_1, \cdots, \tilde{\mathbf{R}}_N$. Assume the same situation as in section 3.1. so that j-th assessor has a latent complete ranking $\tilde{\mathbf{R}}_{j,true}$ and when those items are movies or TV programs he or she has already seen the constrained items $\mathcal{A}_j$. Then we want to recommend top-ranked items in $\tilde{\mathbf{R}}_{j,true}$ with items in $\mathcal{A}_j$ removed as before. We will use the CP ranking estimate as a single point estimate for $\tilde{\mathbf{R}}_{j,true}$ using MCMC outputs for augmented ranking $\tilde{\mathbf{R}}_j$ and recommend top-ranked items in the estimated ranking with items in $\mathcal{A}_j$ removed.

4 Simulation and Application to Real Dataset

4.1 Simulation

We test the accuracy of our recommendation model through the simulation. We generate $N = 1000$ number of ranking vectors for $n = 25$ items under a mixture Mallows rank model with $C = 4$. Cluster consensus $\boldsymbol{\rho}_1, \cdots, \boldsymbol{\rho}_C$ and cluster scale parameters $\alpha_1, \cdots, \alpha_C$ are generated from the uniform distribution on $\mathcal{P}_n$ and exponential distribution with $\lambda = 1$ respectively. The footrule distance is exploited for metric d.

Note that to sample complete rankings $\mathbf{R}_1, \cdots, \mathbf{R}_N$ from Mallows$(\boldsymbol{\rho}, \alpha)$, we run a basic Metropolis-Hastings algorithm in section 2.2 with fixed $\boldsymbol{\rho}$ and α so that we update $\mathbf{R}$ with the L&S proposal. Acceptance probability in section 2.2 is changed as the below :

$$
\begin{aligned}
r &= \frac{P(\boldsymbol{\rho}, \alpha|\mathbf{R}')}{P(\boldsymbol{\rho}, \alpha|\mathbf{R})} \cdot \frac{P_L(\mathbf{R}|\mathbf{R}')}{P_L(\mathbf{R}'|\mathbf{R})} \\
&= \frac{P_L(\mathbf{R}|\mathbf{R}')}{P_L(\mathbf{R}'|\mathbf{R})} \cdot \exp\Big\{ -\frac{\alpha}{n} \sum_{j=1}^{N} \big[d(\mathbf{R}'_j, \boldsymbol{\rho}) - d(\mathbf{R}_j, \boldsymbol{\rho}) \big] \Big\}
\end{aligned}
$$

From the MCMC samples for $\mathbf{R}$'s, we store obtained rankings at regular intervals (in simulation, we collect one sample every 1000 iterations after 1000 burn-in iterations). By using this method, we gain complete rankings for $N = 1000$ assessors and $n = 25$ items.

We assume the case of top-10 rankings so that we fit our model using only top-10 ranked items for each assessor. The optimal number of clusters is identified by exploiting within-cluster distance posterior distribution for separate analyses with $C = 1, 2, \cdots, 10$. We fit our model using hyperparameters $L = 5, \sigma_\alpha = 0.1, \lambda = 0.1, \psi_1, \cdots, \psi_C = 10$ and $\alpha_{jump} = 10$. We run the MCMC for 10^5 iterations with a burn-in of 10^4 iterations. After fitting, we have MCMC outputs for augmented ranking $\tilde{\mathbf{R}}_1, \cdots, \tilde{\mathbf{R}}_N$. As mentioned in section 3.1, we derive the CP ranking estimate for each assessor. Since top-10 items are assumed to be already observed, we can recommend five items ranked as $11, 12, \cdots, 15$ in the CP ranking estimate. Note that in this simulation we know the true complete ranking $\tilde{\mathbf{R}}_{j,true}$ for each assessor j. Thus we can measure the number of correctly recommended items that are recommended by our model and lie between rank 11 and rank 15 in true complete ranking. We store such number for $N = 1000$ individuals and iterate the same work 10 times by setting a random seed from 1 to 10 to measure the accuracy of our recommendation model. The result is shown in the tables below.

Iteration	$A = 5$	$A \geq 4$	$A \geq 3$	$A \geq 2$	$A \geq 1$
1	0.000	0.022	0.182	0.597	0.929
2	0.001	0.017	0.188	0.620	0.923
3	0.001	0.048	0.296	0.687	0.953
4	0.002	0.022	0.213	0.602	0.917
5	0.005	0.095	0.396	0.766	0.964
6	0.002	0.030	0.193	0.611	0.937
7	0.001	0.042	0.255	0.674	0.936
8	0.008	0.076	0.352	0.728	0.953
9	0.006	0.099	0.364	0.701	0.946
10	0.000	0.048	0.252	0.666	0.954

Table 1: Relative frequency of the number of correctly recommended items for each iteration. Here A stands for the number of correctly recommended items. Random seed 1 is used for the iteration 1, random seed 2 is used for the iteration 2, and so on.

Method	$A = 5$	$A \geq 4$	$A \geq 3$	$A \geq 2$	$A \geq 1$
Our model (mean)	0.0026	0.0499	0.2691	0.6652	0.9412
Our model (sd)	0.0028	0.0302	0.0791	0.0573	0.0153
Random draw	0.0003	0.0170	0.1668	0.5665	0.9161

Table 2: The first two rows of the table shows the mean and standard deviation of the relative frequency for 10 iterations using our recommendation model. The last row of the table shows the exact probability when we randomly recommend 5 items out of 15 items.

As we can see in Table 2, our recommendation model has better accuracy than random draw on average. For each number of correctly recommended items (5, larger than or equal to 4, larger than or equal to 3, larger than or equal to 2, and larger than or equal to 1), the mean relative frequency of our model is always higher than the exact probability for the random draw. Note that we can calculate this exact probability by hand using a combination (for example, the exact probability for $A = 5$ is calculated as $\binom{5}{5}\binom{10}{0} / \binom{15}{5}$). Every iteration has a higher relative frequency for all of those events except $(A = 5)$ as we can see in Table 1.

4.2 Application to Movie Rating Data

Now, we shall move on to the application to the real dataset. Here, we use the data of 9000 movies and 100,000 ratings from 700 users, where those ratings have been obtained from the Official GroupLens website. This dataset can be downloaded on Kaggle. Since $n = 9000$ may not be computationally feasible with the Bayesian Mallows rank model, we focus on $n = 151$ movies which are assessed by more than 100 users. Users give rates scaling from 0 to 5 where a half point is allowed.

In order to take advantage of our recommendation model, we should transform this data into partial ranking data. We can interpret "Movie A is given a higher rate than Movie B by assessor 1" as "Movie A is preferred to Movie B by assessor 1" so that we can transform our data into a pairwise comparison case. Every possible pair is compared for this conversion. Note that a pair of movies given the same rate is discarded since we cannot judge the preference of the user between those two movies.

We fit our recommendation model for this preference data with hyperparameter $L = 5, \sigma_\alpha = 0.1, \lambda = 0.1$, and $\alpha_{jump} = 10$. We run the MCMC for 10^5 iterations with a burn-in of 10^4 iterations. The number of clusters C is chosen as 1 after inspecting the plot of within-cluster distances. Note that since $n > 50$, the exact value of the partition function is not available under the footrule distance. Therefore the partition function is approximated by importance sampling described in section 2.4. After running MCMC, we have a CP ranking estimate and from this ranking, we will recommend top-ranked movies. Of course for the recommendation, we shall exclude the movies that are already assessed by the user.

In the following, for three users with user IDs 10, 100, and 250, we will display the ratings by each user and the corresponding CP ranking estimate with the resulting individual recommendation by our model.

user ID	movie title	rating
10	The Usual Suspects	5
	The Shawshank Redemption	4
	Ace Ventura : Pet Detective	3
	Batman	3
	Die Hard	3
	Reservoir Dogs	3
	The Empire Strikes Back	4
	The Princess Bride	4
	Raiders of the Lost Ark	4
	Aliens	4
	Return of the Jedi	4
	The Terminator	4
	Indiana Jones and the Last Crusade	4
	Good Will Hunting	4
	There's Something About Mary	5
	The Matrix	5

Table 3: Rating by user 10

user ID	movie title	ranking
10	There's Something About Mary	1
	The Usual Suspects	2
	The Matrix	3
	American Beauty	4
	Pulp Fiction	5
	Schindler's List	6
	Taxi Driver	7
	Fight Club	8
	The Godfather	9
	Monty Python and the Holy Grail	10
	The Godfather : Part II	11
	The Dark Knight	12
	Amélie	13
	The Lord of the Rings : The Fellowship of the Ring	14
	Leon : The Professional	15
	The Lord of the Rings : The Return of the King	16
	GoodFellas	17
	Memento	18
	Ferris Bueller's Day Off	19
	Dr. Strangelove	20

Table 4: CP estimate ranking for user 10

user ID	movie title	order
10	American Beauty	1st
	Pulp Fiction	2nd
	Schindler's List	3rd
	Taxi Driver	4th
	Fight Club	5th

Table 5: Recommended movies for user 10

user ID	movie title	rating
100	Toy Story	4
	Heat	3
	Leaving Las Vegas	4
	Twelve Monkeys	5
	Fargo	4
	Mission : Impossible	3
	The Rock	3
	Twister	3
	Independence Day	3
	Willy Wonka & the Chocolate Factory	5

Table 6: Rating by user 100

user ID	movie title	ranking
100	Twelve Monkeys	1
	Willy Wonka & the Chocolate Factory	2
	The Godfather	3
	A Clockwork Orange	4
	Taxi Driver	5
	Pulp Fiction	6
	Casablanca	7
	The Empire Strikes Back	8
	Amélie	9
	The Shawshank Redemption	10
	Eternal Sunshine of the Spotless Mind	11
	Raiders of the Lost Ark	12
	Memento	13
	Schindler's List	14
	The Matrix	15
	The Lord of the Rings : The Fellowship of the Ring	16
	American Beauty	17
	The Lord of the Rings : The Two Towers	18
	Fight Club	19
	The Godfather : Part II	20

Table 7: CP estimate ranking for user 100

user ID	movie title	order
100	The Godfather	1st
	A Clockwork Orange	2nd
	Taxi Driver	3rd
	Pulp Fiction	4th
	Casablanca	5th

Table 8: Recommended movies for user 100

user ID	movie title	rating
250	Taxi Driver	4
	Die Hard : With a Vengeance	4.5
	The Shawshank Redemption	5
	Forrest Gump	5
	Schindler's List	5
	The Silence of the Lambs	5
	The Godfather	5
	One Flew Over the Cuckoo's Nest	5
	The Godfather : Part II	5
	Back to the Future	4.5
	Austin Powers : International Man of Mystery	4
	The Truman Show	4.5
	Good Will Hunting	4.5
	Armageddon	4
	Saving Private Ryan	4.5
	American History X	4
	The Matrix	5
	Ghostbusters	4
	American Beauty	4.5
	Total Recall	4
	Fight Club	4.5
	The Green Mile	5
	Gladiator	4.5
	Memento	4.5
	Shrek	5
	Monsters, Inc.	4.5
	Harry Potter and the Philosopher's Stone	4
	Amélie	5
	A Beautiful Mind	5
	Pirates of the Caribbean : The Curse of the black Pearl	4
	The Dark Knight	5
	Inception	5

Table 9: Rating by user 250

user ID	movie title	ranking
250	The Shawshank Redemption	1
	Schindler's List	2
	The Godfather	2
	The Dark Knight	3
	The Matrix	4
	The Godfather : Part II	5
	Amélie	6
	Fargo	7
	Inception	8
	The Silence of the Lambs	9
	The Lord of the Rings : The Return of the King	10
	One Flew Over the Cuckoo's Nest	11
	The Usual Suspects	12
	The Green Mile	13
	Casablanca	14
	Shrek	15
	Forrest Gump	16
	A Beautiful Mind	17
	GoodFellas	18
	Pulp fiction	19
	The Lord of the Rings : The Fellowship of the Ring	20

Table 10: CP estimate ranking for user 250

user ID	movie title	order
250	Fargo	1st
	The Lord of the Rings : The Return of the King	2nd
	The Usual Suspects	3rd
	Casablanca	4th
	GoodFellas	5th

Table 11: Recommended movies for user 250

References

Valeria Vitelli, Øystein Sørensen, Marta Crispino, Arnoldo Frigessi, and Elja Arjas. Probabilistic preference learning with the mallows rank model. *Journal of Machine Learning Research*, 18(158):1–49, 2018.

Peter D. Hoff. *A First Course in Bayesian Statistical Methods.* Springer, 2009.

Art Owen. Importance sampling, 2009-2013, 2018. URL https://artowen.su.domains/mc/Ch-var-is.pdf.

국문 초록

본 논문은 Vitelli의 2018년 논문에서 다루어진 베이즈 맬로우 순위 모형에 대해서 소개한다. 해당 모형은 순위 데이터를 분석하는 데 사용된다. 우선 해당 모형이 가장 기본이 되는 완전 순위 데이터가 주어진 상황에서 어떻게 작동하는 지를 살펴보고, 보다 더 일반적인 상황으로 어떻게 확장되는 지 다루었다. 그런 다음, 해당 방법론으로 개별 추천이 어떻게 이루어질 수 있는 지 설명하였다. 개별 사용자 선호도 중 결측치가 있는 데이터를 임의로 생성하여 시뮬레이션 실험을 한 결과, 완전 임의 선택을 통해 추천할 아이템을 고르는 방법에 비해 해당 모형을 사용한 추천 방식이 더 나은 정확도를 보였음을 확인하였다. 마지막으로, 영화 평점 사이트에서 개별 유저의 평점 부여 데이터를 활용하여 각 사용자에게 새로운 영화를 추천하는 데에 해당 모형을 적용해보았다.

주요어 : 맬로우 순위 모형, 개별 추천, 베이즈 추론, 마코프체인-몬테카를로, 데이터 증강

학번 : 2021-21116