Model-Based Co-Clustering in Customer Targeting Utilizing Large-Scale Online Product Rating Networks

Qian Chen[1], Amal Agarwal[2], Duncan K.H. Fong[1], Wayne S. DeSarbo[1], and Lingzhou Xue[1]

[1] The Pennsylvania State University and [2] eBay Inc.

Abstract

Given the widely available online customer ratings on products, the individual-level rating prediction and clustering of customers and products are increasingly important for sellers to create targeting strategies for expanding the customer base and improving product ratings. However, the massive missing data problem is a significant challenge for modeling online product ratings. To address this issue, we propose a new co-clustering methodology based on a bipartite network modeling of large-scale ordinal product ratings. Our method extends existing co-clustering methods by incorporating covariates and ordinal ratings in the model-based co-clustering of a weighted bipartite network. We devise an efficient variational EM algorithm for model estimation. A simulation study demonstrates that our methodology is scalable for modeling large datasets and provides accurate estimation and clustering results. We further show that our model can successfully identify different groups of customers and products with meaningful interpretations and achieve promising predictive performance in a real application for customer targeting.

Keywords: co-clustering, bipartite network, online ratings, customer targeting, variational EM

1 Introduction

Online reviews can increase demand and drive revenue (e.g., Hu et al. 2008; Luca, 2016) and companies are paying close attention to online ratings of their products. For E-commerce companies and digital content providers which have collected a huge number of online product ratings, the data can be analyzed to produce useful information to benefit sellers on those platforms. For example, they can provide model-generated product ratings to help each online seller to identify potential customers in a database who will rate the seller's products highly. The strategy of targeting potential customers who will rate a given product higher than its current rating is better than targeting all customers in the database when the goal is to improve product rating. With such predicted product ratings, an online seller can focus on a smaller target population to maximize the impact of marketing efforts, have a better chance to expand customer base, and get a boost in product rating at the same time. Here we propose a methodology to cluster different groups of customers, based on which to generate model-based, individual-level rating predictions.

It is not an easy task to mine online ratings to identify, for any given product, the potential consumers who will rate the product highly. To develop a solution to the targeting problem, we start with the following assumption. If Customer A has the same or very similar opinion as Customer B on a product, and if B rates a different product highly, A is more likely to have B's opinion on the different product (and rates it highly) than that of a randomly chosen customer (e.g., Ricci et al. 2011; Schafer et al., 2007). Model-based clustering methods can then be used to identify customer groups with similar tastes. It is worth pointing out that, in the context of E-commerce, there are usually a large number of products (e.g., substitutable and complementary products) under study. When too many products are included in the study and the products are not liked or disliked by "similar" customers in a homogenous fashion, it will be difficult, if not impossible, to group customers with all products in one product group. As such, our model-based *co-clustering* becomes a useful tool because it simultaneously clusters customers and products into smaller and more homogeneous customer groups and product groups. Customers clustered together tend to have similar rating patterns towards various product groups, and products clustered together tend

to be rated by various customer groups in a similar manner. Having multiple product clusters allows us to study how customers' rating patterns vary by different product groups.

There are several challenges to overcome in applying a co-clustering technique to identify groups of customers with similar tastes. One major challenge is that the scale usage in ratings can vary substantially among the customers. Thus, when two customers provide the same rating on a product, they may imply very different levels of liking. For example, using a five-point scale for illustration purposes here, suppose both customers A and B give a 4-star rating on a product, but most of A's other ratings are 5-star, and most of B's other ratings range from 2-star to 4-star. In such a case, A's 4-star suggests some level of disliking, while B's 4-star suggests a high level of liking. In fact, many studies in survey research support the presence of heterogeneity in respondents' usage of scale, relatively independent of the content under evaluation (e.g., Baumgartner and Steenkamp 2001, 2006). Following similar notions in such literature, concerning online ratings, customers are also likely to have different types of scale usage relatively independent of the products under review: some of them tend to provide more friendly ratings, and some of them tend to be more critical in reviewing. Ignoring such difference in rating patterns can lead to misinterpretation of the ratings. Our proposed model-based co-clustering methodology that takes the potential heterogeneity in rating pattern into explicit consideration to identify different groups of customers with similar tastes. We simultaneously classify customers and products into separate customer clusters and product clusters so that members within a customer cluster are likely to have similar rating patterns toward product clusters, in terms of (1) their probabilities of rating the products – i.e., which products they rate and (2) their rating values – i.e., what ratings they assign. Conditional on cluster memberships, the proposed model provides model-based prediction of review propensities and product ratings, which are used to obtain individual-level rating predictions.

The proposed methodology also aims to address the following modeling and computational challenges for simultaneouly clustering customers and products based on their online ratings. First of all, online databases are usually fairly large, and such rating data generally contain a substantial

amount of missing values (Ying et al., 2006) as each customer typically rates a very small portion of available products. Although matrix factorization techniques have been widely used in recommendation systems since the success in the Netflix prize challenge (Koren et al., 2009), it is still a significant challenge for state-of-the-art matrix factorization methods to recover the massive missing online ratings. It is worth pointing out that the ground-breaking work such as Candès and Recht (2009), Candès and Tao (2010), and Keshavan et al. (2010) on the effectiveness and optimality results of matrix factorization quantified the minimum number of observed ratings as follows:

$$\text{number of observed ratings } \geq C_1(\text{number of customers}) \cdot r \cdot \log(\text{number of customers})$$

and

$$\text{number of observed ratings } \geq C_2(\text{number of products}) \cdot r \cdot \log(\text{the number of products})$$

where C_1 and C_2 are two positive constants, and r is the rank of the underying rating matrix to be recoved. However, the conditions are unlikely to hold for many online rating data sets. For example, in our empirical study with 13,600 customers, 2,657 products, and 24,419 product ratings, the right-hand side of both inequalities can be as large as one hundred and twenty thousand multiplied by the rank r, which can be at least twenty times the number of observed ratings even when $r = 4$. To deal with such few ratings, we need to carefully model the ordinal nature of rating data and incorporate more structural assumptions than the low-rank assumption in matrix factorization to extract useful information.

In this paper, we employ a different approach based on network modeling to overcome the massive missing data problem. We enrich data by considering an online product rating database as a bipartite network, thus taking advantage of not only the observed ratings but also the network structure (i.e., who rates what). As shown in Figure 1, a bipartite network contains two disjoint sets of nodes – one set of customers and one set of products, and an edge exists only between a customer node and a product node (Wasserman and Faust 1994; Hanneman and Riddle 2005). The edges of the bipartite network are weighted by the corresponding product ratings. We extend finite mixture clustering methods (e.g., DeSarbo and Cron 1988; Kamakura and Russell 1989; Wedel

and DeSarbo 1995) to the bipartite network to derive clusters for both customers and products, and assume a proportional odds model for the ordinal product ratings in our network setting.

[Insert Figure 1 Here]

Second, we allow the incorporation of covariates (e.g., customer and product attributes) in our network model, which produces a further computational challenge - simultaneously estimating regression type coefficients. This, together with a large dataset, call for a scalable clustering procedure for implementation. Here, to meet the challenge, we devise a variational expectation-maximization (VEM) algorithm and its stochastic version (S-VEM). We implemented both VEM and S-VEM with the templated C++ library RcppArmadillo, providing a balance between computational efficiency, estimation accuracy, and ease of use. Note that, the proposed method includes several commonly used network models as special cases, such as a bipartite version of the exponential-family random graph model (Vu et al. 2013) and a bipartite version of stochastic block model (Karrer and Newman 2011).

In a simulation study, we show that our model outperforms several benchmark models in terms of cluster recovery, parameter recovery, and computational efficiency. We apply the proposed model to an Amazon product rating dataset to demonstrate how our methodology can generate useful information for marketing managers and researchers. Results from our method can help an online seller to identify potential customers who are likely to rate her product and rate it highly. By targeting these potential customers, the seller may improve product rating and further expand her customer base with more focused marketing efforts.

The rest of the paper is organized as follows. We first review the relevant literature in Section 2. Then, we present our proposed methodology and theoretical properties in Section 3, and describe the model estimation and prediction technique in Section 4. In Section 5, we perform a simulation study to compare performance of the proposed method versus several network-based benchmark models. Section 6 provides a real data application to further examine the performance of the proposed method and discusses managerial implications. Finally, we present our conclusions and directions for future research in Section 7.

2 Review of Existing Co-Clustering Methodologies

Many data sets can be described as two-dimensional matrices, also called two-way data. One dimension represents a set of individuals (e.g., subjects, persons, cases), and the other dimension represents a set of variables or subjects (e.g., products, features). Co-clustering is defined as classifications on *each* of the two sets, (e.g., a set of products are classified into product groups and a set of customers are classified into customer groups). In particular, the model-based co-clustering refers to the class of co-clustering methods that design a model based on statistical assumptions and then estimate the model parameters based on the given data (Govaert and Nadif 2013). There are two main streams of model-based co-clustering methods. One is non-network approaches, and the other is network approaches or designated as graph models. The non-network models, also called the latent block models, typically require the two-way data matrix to be complete (without any missing data) for good model performance. The network models do not require the data matrix to be complete because they model both the existence of data entry in each cell and the data value if there is data entry in the cell.

In the context of E-commerce, traditional non-network methods utilize a two-way data matrix with customers as rows, products/questions as columns, and matrix entries as responses to perform simultaneous clustering of both rows and columns. More specifically, it categorizes rows and columns as corresponding homogeneous groups, which are assumed to have the same group-specific effects on the responses (e.g., Vichi 2001; DeSarbo et al. 2004; Rocci and Vichi 2008; Govaert and Nadif 2010; Pledger and Arnold 2014). More recent advancements in the non-network models are Jacques and Biernacki (2018) and Matechou et al. (2016) in which the authors extend the latent block models to ordinal data. However, this stream of models does not work well when the dataset is very large and contains a lot of missing values. Also, when partitioning the columns and the rows of the data matrix, they do not incorporate attributes of the columns and rows.

Compared to non-network models, network models generally have better performance with large datasets and missing data. However, the existing clustering approaches with bipartite networks have strict assumptions and restrictions which limit their applicability in our setting. The

first type of existing approaches is projection-based (e.g., Newman 2001; Zhou et al. 2007). They perform the co-clustering in a two-step fashion by first projecting a bipartite network to one-mode networks and then perform some standard community detection algorithms. For example, in our case, these methods project a bipartite rating network into a network of customers in which customers who rate the same product are connected, or into a network of products where products rated by the same customers are connected. Unfortunately, after the one-mode projection, information of the eliminated set of nodes is lost, and it is difficult to establish links between customers and products. The second type of approaches employs a minimum description length based stochastic block model or mixture model to describe the structure and identifies the blocks (clusters) for a network (e.g., Larremore et al. 2014; Saldana et al. 2017; Zhou and Amini 2020; Razaee et al. 2019; Agarwal and Xue. 2020; Lee et al. 2020; Lee et al. 2022). For these types of network models, the bipartite version of the stochastic block model is the closest to our proposed model (Larremore et al. 2014; Keribin et al 2015; Zhou and Amini 2020; Razaee et al. 2019). For example, Zhou and Amini (2020) proposed a model for bipartite network clustering. Razaee et al. (2019) proposed a matched bipartite block model for mixed clustering that focuses on the latent one-to-one correspondence between clusters of the two sides. However, these models focus on binary data or categorical data. Differently, our proposed model, stemming from the bipartite version of the stochastic block model, focuses on ordinal data, which are typical of the Likert, Edwards, and semantic differential type ratings scales frequently encountered in online ratings. Modeling ordinal networks adds another computational challenge for that matter. In addition, we allow the incorporation of node attributes in the modeling. As a result, our model considers the effect of node covariates on network structure.

3 Methodology

3.1 The Proposed Model

We model an online rating dataset as a weighted bipartite network: when a customer rates a product, a weighted edge connects the customer node and the product node, and the weight represents the given ordinal product rating. The proposed model-based co-clustering methodology

provides a generative model for the weighted bipartite rating network and extends the methodology and applicability of stochastic block models. The proposed methodology can extract the common rating patterns like stochastic block models, and it can incorporate both customer and product covariates to form customer clusters and product clusters simultaneously. More specifically, the co-clusering is embedded in the following two aspects of the proposed bipartite rating network structure: for any given pair of customer cluster and product cluster, (i) the review propensity is homogeneous in terms of its responsiveness to the customer covariates and product attributes that are used to parameterize the review probabilities; and, (ii) the ordinal product ratings follow the same proportional odds model with cluster-specific parameters.

Our proposed method thus has three key components: (1) latent cluster memberships for customers and products, (2) rating network structure based on the latent cluster memberships, customer covariates, and product attributes, and (3) distributions of the ordinal product ratings given latent clusters and network edges. In what follows, we present the details for the components (1)-(3), respectively.

To begin with, we model the latent customer and product memberships in the rating network. Suppose that there are K clusters of customers and L clusters of products. (The values of K and L are determined by using an information criterion; See Web Appendix B). We assume that the cluster membership of customer i, denoted by $Z_i^u \epsilon \{1, \dots, K\}$, follows a multinomial distribution:

$$Z_i^u \sim_{iid} \text{Multinomial}(1; \pi_1, \dots, \pi_K),\ i \epsilon \{1, \dots, N\}, \quad (1)$$

where unknown parameters $\{\pi_1, \dots, \pi_K\}$ denote customer membership probabilities. Similarly, the cluster membership of product j, denoted by $Z_j^p \epsilon \{1, \dots, L\}$, follows a multinomial distribution:

$$Z_j^p \sim_{iid} \text{Multinomial}(1; \varphi_1, \dots, \varphi_L),\ j \epsilon \{1, \dots, M\}, \quad (2)$$

where $\{\varphi_1, \dots, \varphi_L\}$ denote product membership probabilities. Concomitant variables such as demographics and psychographics can be incorporated to profile the customer or product membership probabilities as in DeSarbo et al. (2017).

Next, we model the bipartite rating network structure (i.e., the existence of network edges) given the latent cluster memberships. For any customer i in cluster $k \epsilon \{1, \dots, K\}$ and product j in

cluster $l \epsilon \{1, \dots, L\}$, we model the existence of the network edge E_{ij} (that is, whether or not customer i rates product j) by an independent cluster-specific Bernoulli distribution parameterized by customer covariates and product attributes:

$$E_{ij}|Z_i^u = k, Z_j^p = l \sim_{ind} \text{Bernoulli}\left(P_{ij}(\boldsymbol{\theta}) = P_{ij}\left(\theta_{kl}^0, \boldsymbol{\theta}_k^u, \boldsymbol{\theta}_l^p\right)\right), \tag{3}$$

$$P_{ij}\left(\theta_{kl}^0, \boldsymbol{\theta}_k^u, \boldsymbol{\theta}_l^p\right) = \frac{\exp\left(\theta_{kl}^0 + (\boldsymbol{X}_i^u)'\boldsymbol{\theta}_k^u + \left(\boldsymbol{X}_j^p\right)'\boldsymbol{\theta}_l^p\right)}{1+\exp\left(\theta_{kl}^0 + (\boldsymbol{X}_i^u)'\boldsymbol{\theta}_k^u + \left(\boldsymbol{X}_j^p\right)'\boldsymbol{\theta}_l^p\right)}, \tag{4}$$

where $\boldsymbol{X}_i^u$ denotes covariates of customer i, $\boldsymbol{X}_j^p$ denotes attributes of product j, $\boldsymbol{\theta}_k^u$ and $\boldsymbol{\theta}_l^p$ denote cluster-specific parameters of interest, and θ_{kl}^0 denotes the cluster-specific intercept. Since $P_{ij}(\boldsymbol{\theta}) = P_{\boldsymbol{\theta}}\left(E_{ij} = 1|Z_i^u = k, Z_j^p = l\right)$ and $1 - P_{ij}(\boldsymbol{\theta}) = P_{\boldsymbol{\theta}}\left(E_{ij} = 0|Z_i^u = k, Z_j^p = l\right)$, we have

$$P_{\boldsymbol{\theta}}\left(E_{ij} = e_{ij}|Z_i^u = k, Z_j^p = l\right) = \frac{\exp\left(e_{ij}\theta_{kl}^0 + e_{ij}(\boldsymbol{X}_i^u)'\boldsymbol{\theta}_k^u + e_{ij}\left(\boldsymbol{X}_j^p\right)'\boldsymbol{\theta}_l^p\right)}{1 + \exp\left(\theta_{kl}^0 + (\boldsymbol{X}_i^u)'\boldsymbol{\theta}_k^u + \left(\boldsymbol{X}_j^p\right)'\boldsymbol{\theta}_l^p\right)} \tag{5}$$

for $e_{ij} \epsilon \{0,1\}$. Hence, $P_{\boldsymbol{\theta}}\left(E_{ij} = e_{ij}|Z_i^u = k, Z_j^p = l\right)$ uses the cluster-specific parameters to model the probability of the existence of a rating between customer i and product j. The proposed cluster-specific Bernoulli distribution provides a generative model for the bipartite rating network.

Now, it remains to model the distributions of product ratings given latent clusters and observed network edges. For any customer i in cluster k and product j in cluster l, we utilize the cluster-specific rating distribution based on the proportional odds model (McCullagh 1980) to model the ordinal product rating Y_{ij}. Suppose that ordinal ratings are based on an R-point Likert scale. To simplify the notation, we define the conditional probability of the rating Y_{ij} being $r = 1, \dots, R$, given the latent cluster memberships (i.e., $Z_i^u = k, Z_j^p = l$) and the existence of a review (i.e., $E_{ij} = 1$) as follows:

$$\omega_{ij,r}^{kl} = P(Y_{ij} = r|Z_i^u = k, Z_j^p = l, E_{ij} = 1\,). \tag{6}$$

Also, denote by $\Upsilon_{ij,r}^{kl}$, the cumulative probability that Y_{ij} is no greater than r:

$$\Upsilon_{ij,r}^{kl} = P\left(Y_{ij} \le r \middle| Z_i^u = k, Z_j^p = l, E_{ij} = 1\,\right) = \omega_{ij,1}^{kl} + \cdots + \omega_{ij,r}^{kl}. \tag{7}$$

Let $\boldsymbol{\delta} = (\delta_r^{kl})$ be the unknown proportional odds parameters. Given $Z_i^u = k$ and $Z_j^p = l$, we use the following proportional odds model to specify the cluster-specific rating distribution as:

$$\text{logit}\left(\Upsilon_{ij,r}^{kl}\right) = \log\left(\frac{\Upsilon_{ij,r}^{kl}}{1-\Upsilon_{ij,r}^{kl}}\right) = \delta_r^{kl}. \tag{8}$$

Thus $\omega_{ij,r}^{kl}$, $r = 1, \ldots, R$, can be expressed in terms of parameters $\boldsymbol{\delta}$ as follows:

$$\omega_{ij,r}^{kl} = \Upsilon_{ij,r}^{kl} - \Upsilon_{ij,r-1}^{kl} = \frac{\exp(\delta_r^{kl})}{1+\exp(\delta_r^{kl})} - \frac{\exp(\delta_{r-1}^{kl})}{1+\exp(\delta_{r-1}^{kl})},\ 2 \le r \le R-1$$

$$\omega_{ij,1}^{kl} = \frac{\exp(\delta_1^{kl})}{1+\exp(\delta_1^{kl})} \text{ and } \omega_{ij,R}^{kl} = 1 - \frac{\exp(\delta_{R-1}^{kl})}{1+\exp(\delta_{R-1}^{kl})}. \tag{9}$$

3.2 Parameter Identification

The proposed model includes a group of discrete latent random variables (i.e., cluster memberships of customers and products), which usually leads to the invariance of the likelihood function under relabeling of the cluster memberships. Thus, the identifiability of parameters is obtained up to a label switching on the cluster memberships (Stephens, 2000). Allman et al. (2011) and Matias and Miele (2017) studied the identifiability of the parameters in a broad class of random graph mixture models including the weighted random graphs. It is worth pointing out that the bipartite network model in this paper is a special case of weighted random graphs here. Thus, we can adapt their results to our context and obtain the conditions of parameter identification for our proposed model. Following Theorems 12-14 of Allman et al. (2011) and Proposition 1 of Matias and Miele (2017), we can study the identification of the membership probabilities, conditional probabilities of observing an edge, and proportional odds parameters when the network size is not too small. Furthermore, following Theorem 1 of Lee, Xue, and Hunter (2020), if consumer covariates and product attributes are linearly independent, all the parameters of the proposed model are identified up to label switching with probability one. After adapting the proof to our context, we have the following theorem about parameter identification, whose proof is presented in Web Appendix A.

Theorem 1. *Suppose that (i) the $K \times L$ parameter values $\left\{\theta_{kl}^0, \boldsymbol{\theta}_k^u, \boldsymbol{\theta}_l^p : k = 1, \ldots, K, l = 1, \ldots, L\right\}$ are distinct, (ii) the $K \times L$ parameter values $\left\{\delta_1^{kl}, \ldots, \delta_R^{kl} : k = 1, \ldots, K, l = 1, \ldots, L\right\}$ are distinct,*

(iii) the parameters of finite mixtures of proportional odds models are identifiable up to label switching, and (iv) consumer covariates and product attributes are linearly independent, *as long as the network size M or N is not too small, then the membership probabilities* $(\boldsymbol{\pi}, \boldsymbol{\varphi})$*, network parameters* $\boldsymbol{\theta}$*, and proportional odds parameters* $\boldsymbol{\delta}$ *are identified up to label switching with probability one, except for a subset of the parameter space whose Lebesgue measure is zero.*

4 Model Estimation and Prediction

In this section, we first present the estimation procedure for the proposed method, and then show how model-based predictions of review propensities and ratings are obtained.

4.1. The Likelihood Function

Given the proposed generative model in Section 3, we need to estimate the model parameters $\boldsymbol{\Theta} = (\boldsymbol{\theta}, \boldsymbol{\delta}, \boldsymbol{\pi}, \boldsymbol{\varphi})$ from observed reviews $\boldsymbol{E} = (E_{ij})$, ratings $\boldsymbol{Y} = (Y_{ij})$, customer covariates $\boldsymbol{X}^u = (\boldsymbol{X}_i^u)$ and product attributes $\boldsymbol{X}^p = (\boldsymbol{X}_j^p)$. To this end, we first write down the likelihood function for the proposed model:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\Theta}|\boldsymbol{E}, \boldsymbol{Y}, \boldsymbol{X}^u, \boldsymbol{X}^p) \\ = \sum_{k=1}^{K}\sum_{l=1}^{L}\prod_i\prod_j \Bigg\{ \pi_k \varphi_l \Bigg[P_{ij}\left(\theta_{kl}^0, \boldsymbol{\theta}_k^u, \boldsymbol{\theta}_l^p\right) \prod_{r=1}^{R} \left(\omega_{ij,r}^{kl}\right)^{\mathbf{1}_{Y_{ij}=r}} \Bigg]^{E_{ij}} \Big[1 \\ - P_{ij}\left(\theta_{kl}^0, \boldsymbol{\theta}_k^u, \boldsymbol{\theta}_l^p\right)\Big]^{1-E_{ij}} \Bigg\},\end{aligned} \tag{10}$$

where $\mathbf{1}_{Y_{ij}=r} = \begin{cases} 1, \text{if } Y_{ij} = r \\ 0, \text{if } Y_{ij} \neq r \end{cases}$ is the indicator function for the product ratings. After incorporating latent clusters, we obtain the complete-data likelihood for our proposed model as follows:

$$\begin{aligned}&\mathcal{L}_c(\boldsymbol{\Theta}|\boldsymbol{Z}, \boldsymbol{E}, \boldsymbol{Y}, \boldsymbol{X}^u, \boldsymbol{X}^p) = P(\boldsymbol{E}, \boldsymbol{Y}, \boldsymbol{Z}|\boldsymbol{\Theta}, \boldsymbol{X}^u, \boldsymbol{X}^p) = P(\boldsymbol{Z}|\boldsymbol{\Theta}) P(\boldsymbol{E}, \boldsymbol{Y}|\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{X}^u, \boldsymbol{X}^p) \\ &= \left(\prod_i \pi_{Z_i^u} \prod_j \varphi_{Z_j^p}\right) \Bigg\{ \prod_i\prod_j \Bigg[P_{ij}\left(\theta_{Z_i^u Z_j^p}^0, \boldsymbol{\theta}_{Z_i^u}^u, \boldsymbol{\theta}_{Z_j^p}^p\right) \prod_{r=1}^{R} \left(\omega_{ij,r}^{Z_i^u Z_j^p}\right)^{\mathbf{1}_{Y_{ij}=r}} \Bigg]^{E_{ij}} \Big[1 \\ &- P_{ij}\left(\theta_{Z_i^u Z_j^p}^0, \boldsymbol{\theta}_{Z_i^u}^u, \boldsymbol{\theta}_{Z_j^p}^p\right)\Big]^{1-E_{ij}} \Bigg\},\end{aligned} \tag{11}$$

where $\boldsymbol{Z} = (\boldsymbol{Z^u}, \boldsymbol{Z^p}) = \left((Z_1^u, \dots, Z_N^u), \left(Z_1^p, \dots, Z_M^p\right)\right)$, $\omega_{ij,r}^{Z_i^u Z_j^p}$ is a function of $\boldsymbol{\delta}$, $\mathbf{1}_{Y_{ij}=r} = 1$ if $Y_{ij} = r$ and zero otherwise. The traditional Expectation-Maximization (EM) algorithm is used extensively to solve the maximum likelihood estimation involving latent variables. However, the E-step here requires the computation of an intractable conditional expectation of the complete-data likelihood $\mathcal{L}_c(\boldsymbol{\Theta}|\boldsymbol{Z}, \boldsymbol{E}, \boldsymbol{Y}, \boldsymbol{X}^u, \boldsymbol{X}^p)$. To address this issue, we propose a variational EM algorithm and its stochastic version to estimate the model parameters $\boldsymbol{\Theta} = (\boldsymbol{\theta}, \boldsymbol{\delta}, \boldsymbol{\pi}, \boldsymbol{\varphi})$. In the following sub-sections, we first present the variational EM denoted by VEM (e.g., Neal and Hinton 1998; Jaakkola and Jordon 1996 & 1997) for our proposed model, and then examine the nature of the stochastic EM (SEM) (Celeux et al. 1995; Nielsen 2000) to introduce the stochastic version of VEM, denoted by S-VEM. The VEM uses the variational distribution to approximate the intractable conditional distribution in the traditional EM algorithm and enjoys the ascent property of evidence lower bound in the algorithm (Blei et al. 2017). The SEM estimates the conditional expectation by incorporating an additional simulation step after the E-step in the traditional EM algorithm and enjoys an appealing asymptotic result (Nielsen 2000). However, it is an open research question to provide a rigorous convergence guarantee for VEM and S-VEM when solving the general network-based clustering problem (Blei et al. 2017), which is important. Similar to numerous existing works (e.g., Zaheer et al 2016; Matias and Miele, 2017; Blei et al 2017), we find that VEM and S-VEM are well behaved, and both provide satisfactory performance in our simulation and empirical studies.

4.2. Variational EM (VEM)

The EM algorithm iterates between an expectation step (E-step) and a maximization step (M-step) until convergence:

- E-step: Compute the expected value of the complete-data log likelihood function:

$$\begin{aligned} Q\left(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)}\right) &= E_{\boldsymbol{Z}|\boldsymbol{E},\boldsymbol{Y};\boldsymbol{\Theta}^{(t)}}\left(\log \mathcal{L}_c(\boldsymbol{\Theta}|\boldsymbol{Z}, \boldsymbol{E}, \boldsymbol{Y}, \boldsymbol{X}^u, \boldsymbol{X}^p)\right) \\ &= \int \log P(\boldsymbol{E}, \boldsymbol{Y}, \boldsymbol{Z}|\boldsymbol{\Theta}, \boldsymbol{X}^u, \boldsymbol{X}^p)\, P\left(\boldsymbol{Z}\middle|\boldsymbol{E}, \boldsymbol{Y}; \boldsymbol{\Theta}^{(t)}\right) d\boldsymbol{Z} \end{aligned} \tag{12}$$

- M-step: Maximize the function obtained from E-step to obtain $\boldsymbol{\Theta}^{(t+1)}$:

$$\boldsymbol{\Theta}^{(t+1)} = \underset{\boldsymbol{\Theta}}{\operatorname{argmax}}\, Q\left(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)}\right). \tag{13}$$

Unfortunately, $P\left(\boldsymbol{Z}\middle|\boldsymbol{E},\boldsymbol{Y};\boldsymbol{\Theta}^{(t)}\right)$ in the E-step is intractable and $Q\left(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)}\right)$ does not have a closed form, which leads to the numeric challenge of computing and updating $\boldsymbol{\Theta}^{(t+1)}$ in the M-step (Beal 2003; Daudinet al. 2008; Blei et al. 2017). To overcome this problem, we use the variational EM approach (e.g., Neal and Hinton 1998; Jaakkola and Jordon 1996 & 1997; Jordan et al. 1999; Beal 2003; Wainwright and Jordan 2008; Blei et al. 2017; Ansari et al. 2018) in which we approximate the intractable distribution $P\left(\boldsymbol{Z}\middle|\boldsymbol{E},\boldsymbol{Y};\boldsymbol{\Theta}^{(t)}\right)$ by a simple mean-field distribution $q(\boldsymbol{Z};\boldsymbol{u},\boldsymbol{v})$, called the *variational distribution*, indexed by mean-field parameters $(\boldsymbol{u},\boldsymbol{v})$. Here we set:

$$q(\boldsymbol{Z};\boldsymbol{u},\boldsymbol{v}) = q(\boldsymbol{Z^u};\boldsymbol{u})q(\boldsymbol{Z^p};\boldsymbol{v}) = \prod_{i=1}^{N} q(Z_i^u;\boldsymbol{u}_i)\prod_{j=1}^{M} q\left(Z_j^p;\boldsymbol{v}_j\right), \tag{14}$$

$$q(Z_i^u;\boldsymbol{u}_i) \sim \text{Multinomial}(1;\boldsymbol{u}_i), \tag{15}$$

$$q\left(Z_j^p;\boldsymbol{v}_j\right) \sim \text{Multinomial}(1;\boldsymbol{v}_j), \tag{16}$$

where $\boldsymbol{u}_i = (u_{i,1},\dots,u_{i,K})$ and $\boldsymbol{v}_j = (v_{j,1},\dots,v_{j,L})$ are variational parameters. We search over the space of variational distributions to find a member that is closest to $P\left(\boldsymbol{Z}\middle|\boldsymbol{E},\boldsymbol{Y};\boldsymbol{\Theta}^{(t)}\right)$ based on the Kullback-Leibler (KL) divergence (Kullback and Leibler 1951). The KL divergence measures the closeness between $q(\boldsymbol{Z};\boldsymbol{u},\boldsymbol{v})$ and $P\left(\boldsymbol{Z}\middle|\boldsymbol{E},\boldsymbol{Y};\boldsymbol{\Theta}^{(t)}\right)$, so our goal is to minimize the KL divergence:

$$\text{KL}\left(q(\boldsymbol{Z};\boldsymbol{u},\boldsymbol{v})||P\left(\boldsymbol{Z}\middle|\boldsymbol{E},\boldsymbol{Y};\boldsymbol{\Theta}^{(t)}\right)\right) = E_q\left(\log\frac{q(\boldsymbol{Z};\boldsymbol{u},\boldsymbol{v})}{P\left(\boldsymbol{Z}\middle|\boldsymbol{E},\boldsymbol{Y};\boldsymbol{\Theta}^{(t)}\right)}\right), \tag{17}$$

where, $E_q(\cdot)$ refers to taking expectations with respect to the variational distribution. The KL divergence is not directly computable, but it can be further derived as shown in the following expression (see Web Appendix B for details):

$$\begin{aligned}\text{KL}\left(q(\boldsymbol{Z};\boldsymbol{u},\boldsymbol{v})||P\left(\boldsymbol{Z}\middle|\boldsymbol{E},\boldsymbol{Y};\boldsymbol{\Theta}^{(t)}\right)\right) &= E_q\left(\log q(\boldsymbol{Z};\boldsymbol{u},\boldsymbol{v})\right) - E_q\left(\log P\left(\boldsymbol{Z}\middle|\boldsymbol{E},\boldsymbol{Y};\boldsymbol{\Theta}^{(t)}\right)\right)\\ &= \log\mathcal{L}\left(\boldsymbol{\Theta}^{(t)}\middle|\boldsymbol{E},\boldsymbol{Y}\right) - E_q\left(\log\mathcal{L}_c\left(\boldsymbol{\Theta}^{(t)}\middle|\boldsymbol{Z},\boldsymbol{E},\boldsymbol{Y}\right)\right) + E_q\left(\log q(\boldsymbol{Z};\boldsymbol{u},\boldsymbol{v})\right).\end{aligned} \tag{18}$$

We define the following term as the evidence lower bound (ELBO), which is a lower bound on the logarithm of the likelihood function:

$$\begin{aligned}&\text{ELBO}\left(\boldsymbol{\Theta}^{(t)},\boldsymbol{u},\boldsymbol{v}\,|\boldsymbol{E},\boldsymbol{Y},\boldsymbol{X}^u,\boldsymbol{X}^p\right)\\ &= E_q\left(\log\mathcal{L}_c\left(\boldsymbol{\Theta}^{(t)}\middle|\boldsymbol{Z},\boldsymbol{E},\boldsymbol{Y},\boldsymbol{X}^u,\boldsymbol{X}^p\right)\right) - E_q\left(\log q(\boldsymbol{Z};\boldsymbol{u},\boldsymbol{v})\right).\end{aligned} \tag{19}$$

Thus, minimizing the KL divergence is equivalent to maximizing ELBO with respect to $(\boldsymbol{u},\boldsymbol{v})$:

$$\begin{aligned} &\mathrm{KL}\left(q(\boldsymbol{Z};\boldsymbol{u},\boldsymbol{v})||P\left(\boldsymbol{Z}\middle|\boldsymbol{E},\boldsymbol{Y};\boldsymbol{\Theta}^{(t)}\right)\right) \\ &= \log\mathcal{L}\left(\boldsymbol{\Theta}^{(t)}\middle|\boldsymbol{E},\boldsymbol{Y},\boldsymbol{X}^u,\boldsymbol{X}^p\right) - \mathrm{ELBO}\left(\boldsymbol{\Theta}^{(t)},\boldsymbol{u},\boldsymbol{v}\,|\boldsymbol{E},\boldsymbol{Y},\boldsymbol{X}^u,\boldsymbol{X}^p\right). \end{aligned} \tag{20}$$

This evidence lower bound can be computed in a closed form (see Web Appendix B) and we maximize ELBO with respect to $(\boldsymbol{u},\boldsymbol{v})$ in the Variational EM algorithm. In the following variational M-step, maximization with respect to $\boldsymbol{\Theta}$ may be accomplished, which is presented in Web Appendix B**.** We summarize the algorithm details as Algorithm 1.

1. Initialize $\boldsymbol{\Theta}^{(0)}, \boldsymbol{u}^{(0)}, \boldsymbol{v}^{(0)}$
2. **Repeat**
3. Variational E-step: update variational parameters

$$\left(\boldsymbol{u}^{(t+1)},\boldsymbol{v}^{(t+1)}\right) = \underset{(\boldsymbol{u},\boldsymbol{v})}{\operatorname{argmax}}\,\mathrm{ELBO}\left(\boldsymbol{\Theta}^{(t)},\boldsymbol{u},\boldsymbol{v}\,|\boldsymbol{E},\boldsymbol{Y},\boldsymbol{X}^u,\boldsymbol{X}^p\right)$$

4. Variational M-step: update $\boldsymbol{\Theta}$

$$\boldsymbol{\Theta}^{(t+1)} = \underset{\boldsymbol{\Theta}}{\operatorname{argmax}}\,\mathrm{ELBO}\left(\boldsymbol{\Theta},\boldsymbol{u}^{(t+1)},\boldsymbol{v}^{(t+1)}\,|\boldsymbol{E},\boldsymbol{Y},\boldsymbol{X}^u,\boldsymbol{X}^p\right)$$

5. **Until convergence**

Algorithm 1 The Proposed Variational EM (VEM)

After obtaining the parameter estimates, one may assign customer *i* and product *j* to the derived customer and product clusters according to the corresponding estimated probabilities:

$$\widehat{Z_i^u} = \underset{k=1,\ldots,K}{\operatorname{argmax}}\, q(Z_i^u;\widehat{\boldsymbol{u}_\iota}), \tag{21}$$

$$\widehat{Z_j^p} = \underset{l=1,\ldots,L}{\operatorname{argmax}}\, q\left(Z_j^p;\widehat{\boldsymbol{v}_j}\right). \tag{22}$$

Remark 1. When calculating the expectation terms $E_q[\cdot]$ of ELBO in equation (19), for each customer i, $i\epsilon\{1,\ldots,N\}$, we have to consider each possible value of $k \in \{1,\ldots,K\}$, and for each product j, $j\epsilon\{1,\ldots,M\}$, we consider each possible value of $l \in \{1,\ldots,L\}$. This step is separable over all pairs of nodes. For each pair, the expectation operator puts different weights $(\boldsymbol{u},\boldsymbol{v})$ based on different possibilities of the pair belonging to different clusters. This step $(\mathrm{O(K \times L)})$ can be very computationally expensive when *K* and *L* are large. Following the notion of stochastic EM (SEM) (Celeux et al. 1995; Nielsen 2000), we propose a stochastic version of VEM by introducing an additional S-step after the variational E-step in VEM, thus overcome the computational

bottleneck by replacing $E_q[\cdot]$ with an empirical estimate. We included the technical details in the Web Appendix C.

Remark 2. In the literature, to the best of our knowledge, it is still an open question to establish the asymptotic properties of variational approximation in model-based clustering of network data. Westling and McCormick (2019) pointed out the connection between variational approximation in a class of mixture models based on the *i.i.d.* observations and *M*-estimation and then studied the theoretical properties for variational estimators in this class of mixture models by using the results of *M*-estimation (van der Vaart, 2000). Specifically, Proposition 1 and Theorem 1 of Westling and McCormick (2019) require the *i.i.d.* assumption to study their defined profiled objective function of variational approximation through the theory of *M*-estimation such as Theorem 5.14 of van der Vaart (2000). Thus, the theory of Westling and McCormick (2019) excludes the dyadic data in network models. It will be an important research question to fill this gap.

4.3. Prediction

After estimating parameters and cluster memberships from the proposed VEM or S-VEM, we can use the estimates to obtain model-based predictions of review propensities and ratings between any customer $i\epsilon\{1,\dots,N\}$ and any product $j\epsilon\{1,\dots,M\}$. Let $\widehat{\boldsymbol{\theta}}$ be the estimate of network parameters $\boldsymbol{\theta}$ and $\widehat{\boldsymbol{\delta}}$ be the estimate of proportional odds parameters $\boldsymbol{\delta}$. Given the model parameter estimates, conditional on the latent cluster memberships, we predict the review propensity of customer i for product j as

$$\begin{aligned}&\sum_{k,l} P_{\widehat{\boldsymbol{\theta}}}\left(E_{ij}=1\middle|Z_i^u=k\,,Z_j^p=l\right)q(Z_i^u=k;\widehat{\boldsymbol{u}}_\iota)q\left(Z_j^p=l;\widehat{\boldsymbol{v}}_j\right)\\&=\sum_{k,l}\frac{\exp\left(\widehat{\theta_{kl}^0}+(\boldsymbol{x}_i^u)'\widehat{\boldsymbol{\theta}_k^u}+\left(\boldsymbol{x}_j^p\right)'\widehat{\boldsymbol{\theta}_l^p}\right)}{1+\exp\left(\widehat{\theta_{kl}^0}+(\boldsymbol{x}_i^u)'\widehat{\boldsymbol{\theta}_k^u}+\left(\boldsymbol{x}_j^p\right)'\widehat{\boldsymbol{\theta}_l^p}\right)}q(Z_i^u=k;\widehat{\boldsymbol{u}}_\iota)q\left(Z_j^p=l;\widehat{\boldsymbol{v}}_j\right).\end{aligned}\tag{23}$$

We obtain the conditional distribution of ordinal rating Y_{ij} while products being rated as

$$P_{\widehat{\boldsymbol{\delta}}}\left(Y_{ij}=r\middle|Z_i^u=k\,,Z_j^p=l,E_{ij}=1\right)=\widehat{\omega_{ij,r}^{kl}},\ r=1,\dots,R,\tag{24}$$

or more specifically, for $2\le r\le R-1$:

$$P_{\hat{\boldsymbol{\delta}}}\left(Y_{ij}=r\middle|Z_i^u=k\,,Z_j^p=l,E_{ij}=1\right)=\frac{\exp\left(\widehat{\delta_r^{kl}}\right)}{1+\exp\left(\widehat{\delta_r^{kl}}\right)}-\frac{\exp\left(\widehat{\delta_{r-1}^{kl}}\right)}{1+\exp\left(\widehat{\delta_{r-1}^{kl}}\right)}, \tag{25}$$

And, for $r=1$ or $r=R$:

$$P_{\hat{\boldsymbol{\delta}}}\left(Y_{ij}=1\middle|Z_i^u=k\,,Z_j^p=l,E_{ij}=1\right)=\frac{\exp\left(\widehat{\delta_1^{kl}}\right)}{1+\exp\left(\widehat{\delta_1^{kl}}\right)}, \tag{26}$$

$$P_{\hat{\boldsymbol{\delta}}}\left(Y_{ij}=R\middle|Z_i^u=k\,,Z_j^p=l,E_{ij}=1\right)=1-\frac{\exp\left(\widehat{\delta_{R-1}^{kl}}\right)}{1+\exp\left(\widehat{\delta_{R-1}^{kl}}\right)}. \tag{27}$$

Therefore, given parameter estimates, we predict the rating of customer i on product j via:

$$\widehat{Y_{ij}}=\sum_{r,k,l}r\times P_{\hat{\boldsymbol{\delta}}}\left(Y_{ij}=r\middle|Z_i^u=k\,,Z_j^p=l,E_{ij}=1\right)q(Z_i^u=k;\widehat{\boldsymbol{u}_i})q\left(Z_j^p=l;\widehat{\boldsymbol{v}_j}\right). \tag{28}$$

5 A Simulation Study

A full factorial experiment design was used to compare the performance of the proposed model versus several network-based benchmark models. The results in the simulation study are used to demonstrate the effectiveness and efficiency of the proposed algorithm and examine the numerical performance of our proposed method when the data follow the generative model introduced in the paper. Consistent with past research involving factorial experiment design for newly proposed segmentation methods (e.g., Kim et al. 2012; DeSarbo et al. 2017), we experimentally manipulated three factors (see Table 1) and generated $3\times2\times2=12$ settings based on the size of the network (X1: small network vs. medium network vs. large network), the number of clusters (X2: more vs. less), and the existence of covariates (X3: yes vs. no). These factors and their levels were specified to reflect various conditions representing a variety of potential marketing applications. We attempted to create a variety of empirical settings which would realistically test the comparative performance of these methods. With each generated setting, we have run 5 replications. Please refer to Web Appendix D for details.

[Insert Table 1 Here]

We consider the proposed method with S-VEM versus the following benchmark models in the numerical comparison:

(1) The bipartite version of stochastic block model with binary data (Karrer and Newman 2011), which cannot incorporate covariates.

(2) The bipartite version of stochastic block model with binary data, which incorporates covariates.

(3) The bipartite version of stochastic block model with ordinal data, which cannot incorporate covariates.

Moreover, we consider two versions of the proposed method, one with VEM and the other without covariates, as two additional benchmark methods:

(4) The proposed network model with covariates but implemented with VEM.

(5) The proposed network model without covariates.

In the literature, the model-based non-network co-clustering method for ordinal data proposed by Jacques and Biernacki (2018) can be another potential benchmark model. However, it is not designed to deal with large data sets containing a large number of missing data points, and thus takes longer than the job time limit of the university server we run our codes on when applied to our simulated datasets. Therefore, we exclude this method in the numerical comparison and focus on comparing the results between our proposed model and the various benchmark models described in (1) to (5) above.

We consider the following performance measures in our simulation study: (1) *Cluster membership recovery*, measured by the percentage of correctly predicted cluster memberships; and, (2) *Parameter recovery*, measured by the root mean squared error (RMSE) between the actual and recovered coefficients. Table 2 presents the summary of performance measures of the proposed model versus the five benchmark models under study across all settings. Based on the overall performance, our proposed model outperforms all benchmark models in terms of customer cluster membership recovery and has a comparable recovery rate for product memberships with benchmark models 1, 3, and 5. In addition, our proposed model outperforms the benchmarks in terms of parameter recovery. We summarize the performance measures by factor level (i.e., network size, the number of customer clusters) in Web Appendix D.

[Insert Table 2 Here]

To demonstrate the advantage of the stochastic version of variational EM (S-VEM) over the non-stochastic version of varational EM (VEM) when implementing the proposed model, we compare the *computational efficiency* of the proposed model and benchmark model 4. More specifically, we measure the total running time, the number of iterations required for convergence and the average time of each iteration. We present the medians instead of means of these computational performance measures in Table 3, to avoid the outliers due to the instability issue of the university server.

[Insert Table 3 Here]

Consistent with our expectation, as shown in Table 3, the S-VEM does not shorten running time with small and medium networks, but it significantly reduces the running time for large networks. In the large network setting, the S-VEM algorithm can search the parmater space more quickly, therefore it has shorter time per iteration in general together with a larger number of iterations.

6 The Empirical Application

6.1 The Amazon Product Rating Dataset

To demonstrate the capability of our proposed methodology, we apply our model to a publicly available Amazon product rating data set for the pre-defined 'Clothing, Shoes and Jewelry' category (He and McAuley 2016; McAuley et al. 2015). This pre-defined category dataset contains many products, which consist of both substitutable and complementary items. Thus, there are likely multiple ratings from each customer as well as some overlaps of products rated by different customers, allowing us to evaluate customers' review patterns. Here we use the 1-5-core dataset in which each customer provides at least one rating, and each product receives at least five ratings. We focus on ratings posted in the year 2014, which is the most recent year of data available to us.

[Insert Table 4 Here]

Table 4 summarizes the descriptive statistics of our dataset (Web Appendix E lists the correlation matrices of the variables of interests), which contains 13,600 customers, 2,657 products,

and 24,419 product ratings. This means, out of 13,600×2,657=36,135,200 customer-product pairs, only 0.06% have ratings. The massive missing rating issue posts a significant challenge for matrix factorization techniques for recommendation systems. The number of observed product ratings is significantly smaller than the stated threshold (that is 13,600×4×log(13,600)≈224,865) to achieve an effective reconstruction of the rating matrix (Candès and Recht, 2009).

On average, a product receives 9.19 ratings (median=7, SD=7.95), and a customer provides 4.68 ratings (median=4, SD=3.04). These statistics confirm the fact that customers rate only a small handful of products out of a huge number of available products which presents a massive missing data challenge for the analysis ahead. Looking at the average rating received per product, the mean is 4.25 (median=4.33, SD=0.52). The average rating provided per customer has a mean of 4.22 (median=4.38, SD=0.75). The product ratings are heavily skewed towards the high end of the scale.

6.2 Implementation and Selection of Covariates

We applied the proposed model to the Amazon rating dataset described above. More specifically, our proposed methodology models the online rating dataset as a weighted bipartite network and performs two-way clustering: a derived customer (product) cluster is a group of customer nodes with similar connectivity pattern to product (customer) cluster, and their connectivity patterns are affected similarly by the customer and product attributes. In this context, our methodology thus considers the ratings given by each customer, the ratings received by each product, as well as who rates what. Also, to account for the effects of other variables on who rates what, we employ product price as the product attribute ($\boldsymbol{X}_j^p$), and customer spending level (average purchase price) as customer covariates ($\boldsymbol{X}_i^u$) in the *model of rating network structure* to parameterize the model component on whether a customer rates a product (i.e., the network edges). Product price and customer spending level are included because price/spending level may affect customers' expectation, and in turn affect the after-purchasing satisfaction. Limited by the data

availability, we do not have demographic variables. But our proposed model is able to incorporate demographics and other variables when they are available.

6.3 Co-Clustering Results

With our dataset, we choose four customer clusters and three product clusters, given their satisfactory predictive performance in the network cross-validation (Chen et al., 2018; Li et al., 2020). Tables F-1, F-2, and F-3 in Web Appendix F list the network model parameter estimates. After obtaining the parameter estimates, we derive the latent memberships by assigning each customer and product to various clusters separately according to the corresponding (largest) estimated probabilities. Table 5 summarizes the descriptive statistics of customers by cluster. The clustering results suggest that customers have highly heterogeneous tastes and review patterns, as shown by their rating distributions, number of ratings, and review propensities. For the customers under study, the mixing proportions (i.e., the relative sizes of the derived clusters) are 18% for Customer Cluster 1, 25% for Customer Cluster 2, 6% for Customer Cluster 3, and 50% for Customer Cluster 4. We perform both parametric and nonparametric statistical tests to compare the differences across clusters (See Web Appendix G) and draw inferences about the population. Based on Table 5, we find:

- Customer Cluster 1 (labeled as "critical reviewers") consists of customers whose ratings are polarized. They tend to have strong opinions on the products and use the 1-, 2- and 3-star much more often than the other customer clusters: 25% of their ratings are 1-star, 20% are 2-star, 26% are 3-star, while 24% are 5-star and 4% are 4-star. On average, they have the lowest mean rating per customer (mean=3.32, median=3.50), and a relatively low review propensity (mean=0.064%, median=0.054%).
- Customer Cluster 2 (labeled as *"fair reviewers"*) consists of customers who tend to use the middle of scale by assigning 4-star (57%) and 3-star (20%), along with a small number of 1-star (0.5%), 2-star (5%) and 5-star (16%). On average, they have a relatively high review propensity (mean=0.071%, median=0.053%).

- Customer Cluster 3 (labeled as *"friendly reviewers"*) consists of customers whose ratings are very friendly. These customers tend to use 5- and 4-star: 59% of their ratings are 5-star, and 34% are 4-star. On average, they have the largest number of ratings provided per customer (mean=6.27, median=6), and a relatively low review propensity (mean=0.064%, median =0.054%).
- Customer Cluster 4 (labeled as *"super nice reviewers"*) consists of customers who are most likely to give a 5-star (97% of ratings). On average, they have the highest average rating per customer (mean = 4.65, median=4.86), the highest spending power (mean = $22.17, median=$16.26), the smallest number of ratings provided per customer (mean=4.46, median=4), and a relatively high review propensity (mean=0.072%, median=0.055%).

[Insert Table 5 Here]

To facilitate customer clustering, our model also identifies 3 product clusters of distinct characteristics as summarized in Table 6. The mixing proportions are 9% for Product Cluster 1, 59% for Product Cluster 2, and 32% for Product Cluster 3. Based on Table 6, we find:

- Product Cluster 1 (labeled as "*most rated products*") consists of products which are most rated on average. The average number of ratings per product is three to four times more than the other two clusters. On average, they have a medium level of average rating per product (mean=4.22, median=4.27). A lot of products in this cluster have very good selling performance: 16% of the products in this cluster are listed as Top 100 products based on Amazon's best seller rank, and 18% are listed between Top 101 to Top 500. Both percentages are much higher than those of the other two clusters. They have particularly high average review propensity (mean=0.219%, median=0.214%).
- Product Cluster 2 (labeled as "*lower and less rated products*") consists of products which tend to be less rated together with lower ratings. On average, these products have the lowest average rating per product (mean=4.01, median=4.13), the smallest number of ratings per product (mean=6.70, median=6), and the lowest review propensity (mean=0.052%, median =0.053%).
- Product Cluster 3 (labeled as "*highest rated products*") consists of products which are highest rated on average. They have the highest average ratings per product (mean=4.70,

median=4.71), and the highest price (mean=22.00, median=14.99) on average. Products from well-known brands tend to show up in this cluster. On average, products in this cluster have a relatively low review propensity (mean=0.063%, median=0.062%).

[Insert Table 6 Here]

To gain a higher-level understanding of the identified customer groups' review pattern with the product groups, we present the histogram of observed ratings for each product-customer cluster pair (see Figure 2). As shown in Figure 2, the rating distributions are much more homogenous within each customer cluster than within each product cluster. In other words, customers within each customer cluster (of similar tastes) appear to be quite homogeneous: the majority of customers in clusters 2, 3 and 4 have similar tastes, respectively, no matter what products are considered, but the rating patterns for Customer Cluster 1 vary with the product clusters being considered.

We also present the estimated review propensities at the cluster level. Figure H in Web Appendix H shows the predicted review propensities as a network: a node represents a customer cluster or a product cluster; the area of a square represents cluster size; and the link width represents the corresponding review propensity. As indicated in Figure H, customers' review propensities vary substantially toward different product clusters. Product Cluster 1, although its size is small, has a significantly larger propensity to be reviewed by all customer clusters. We consider this group of products as "*buzz*" products. Their review propensities are on average 2 to 6 times larger than that of other product clusters. This is aligned with the fact that Product Cluster 1 contains many "*most rated*" products. Table H in Web Appendix H summarizes the descriptive statistics of the review propensities for each customer-product cluster pair.

6.4 Predictive Performance

For predictive validation, we randomly pick 5% of customers in the dataset and withhold one rating from each of them. Using this hold-out data, we compared the predictive performance of the proposed methods and its sub-model which does not use any covariates to parameterize the rating network structure. In addition, we compared the predictive performance of the proposed

methods with two state-of-the-art approaches, including the spectral algorithm OptSpace proposed by Raghunandan et al. (2010) and one of the famous matrix factorization approaches introduced by Koren et al (2009). It is worth pointing out that Raghunandan et al. (2010) achieved the optimal sample complexity in the matrix completion problem and Koren et al (2009) presented the matrix factorization models in the Netflix Prize competition. The OptSpace algorithm of Raghunandan et al. (2010) was implemented in R package ROptSpace (https://cran.r-project.org/web/packages/ROptSpace/index.html). The implementation of matrix factorization using the stochastic gradient descent optimization popularized by Simon Funk that won the third place in the Netflix Prize competition was implemented as the FunkSVD function in R package recommenderlab (https://cran.r-project.org/web/packages/recommenderlab/index.html).

As shown in Table 7, our proposed method has the smallest RMSE, and the differences between the proposed methods and its sub-model as well as the two state-of-the-art approaches (FunkSVD and OptSpace) are statistically significant, since the differences are at least 8.5 standard errors away from zero. To further test the proposed model's predictive performance, we compared its 25% hit rate of top reviews versus that of the its sub-model as well as the two state-of-the-art approaches: with the hold-out data, we selected the first quantiles of the reviews based on the withhold ratings and the predicted ratings separately; then we calculated what percentage of the 'true' top 25% reviews were overlapped by the top 25% based on the predicted ratings. Also, our proposed model has much higher hit rates than the others. This indicates that our model has the potential to help sellers identify who are more likely to rate their products highly.

Our method empowers E-commerce platforms to help sellers who are interested in boosting their product ratings with better customer targeting strategies. For example, the seller of a product could use results from the proposed methodology to identify "high value" potential customers who will rate the product higher than its current rating. Then the online seller can send reminders to "*high value*" customers to review their recent purchases and offer special discounts to invite potential "*high value*" customers to participate in an "early reviewer" program. By targeting the potential "*high value*" customers, an online seller may expand her customer base, achieve a higher

product rating, and maximize marketing efforts without wasting resources to target "*low value*" customers who are not interested in the product or likely to give poor ratings.

[Insert Table 7 Here]

7 Conclusion

In this paper, we propose a new model-based co-clustering methodology that utilizes large-scale online product rating networks to simultaneously cluster customers and products. The proposed model identifies customers with similar tastes and rating patterns. More specifically, our method classifies customers and products into separate customer clusters and product clusters, identifies memberships of the derived clusters, estimates the cluster-level model parameters, and provides probability estimates linking individual members from a customer cluster to those from a product cluster. Results from the co-clustering are used to generate individual-level rating predictions. Different from previous research, we extend finite mixture clustering methods to bipartite network modeling, focus on ordinal data, and devise efficient variational inference methods for computation so that our method better accommodates large-scale rating data in E-commerce. In addition, we incorporate not only the rating information but also customer and product attributes to derive the clusters.

To conclude, the proposed methodology provides marketing researchers and managers with a powerful tool for analyzing online rating data and help online sellers to design effective customer targeting strategies. Future research could consider incorporating concomitant variables directly in the model to automatically profile identified clusters. One might also consider incorporating dynamics to capture the evolvement of clusters over time.

Acknowledgements

The authors would like to thank the editor, associate editor, and referees for their insightful comments and constructive suggestions that improved the quality and presentation of this paper. Lingzhou Xue was supported in part by the National Science Foundation grants (DMS-2210775, CCF-2007823, DMS-1953189).

Disclosure Statement

The authors report there are no competing interests to declare.

References

Allman, E., C. Matias, and J. Rhodes (2011). Parameters identifiability in a class of random graph mixture models. *Journal of Statistical Planning and Inference*. 141, 1719–1736.

Agarwal, A., & Xue, L. (2020). Model-based clustering of nonparametric weighted networks with application to water pollution analysis. *Technometrics*, *62*(2), 161-172.

Ansari, A., Li, Y., & Zhang, J. Z. (2018). Probabilistic Topic Model for Hybrid Recommender Systems: A Stochastic Variational Bayesian Approach. *Marketing Science*, 37 (6), 987–1008.

Baumgartner, H., & Steenkamp, J. B. E. (2001). Response styles in marketing research: A cross-national investigation. *Journal of Marketing Research*, *38*(2), 143-156.

Baumgartner, H., & Steenkamp, J. B. E. (2006). An extended paradigm for measurement analysis of marketing constructs applicable to panel data. *Journal of Marketing Research*, *43*(3), 431-442.

Beal, M. J. (2003). *Variational algorithms for approximate Bayesian inference.* (Doctoral dissertation, University College London).

Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association*, *112*(518), 859-877.

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, *30*(1-7), 107-117.

Candès, E. J., & Recht, B. (2009). Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, *9*(6), 717–772.

Candès, E. J., & Tao, T. (2010). The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5), 2053-2080.

Celeux, G., Chauveau, D., & Diebolt, J. (1995). *On stochastic versions of the EM algorithm*, Research Report RR-2514, INRIA.

Chen, K., & Lei, J. (2018). Network cross-validation for determining the number of communities in network data. *Journal of the American Statistical Association*, 113(521), 241-251.

Daudin, J. J., Picard, F., & Robin, S. (2008). A mixture model for random graphs. *Statistics and Computing*, *18*(2), 173-183.

Dellarocas, C., Gao, G., & Narayan, R. (2010). Are consumers more likely to contribute online reviews for hit or niche products? *Journal of Management Information Systems*, *27*(2), 127-158.

DeSarbo, W. S., Chen, Q., & Blank, A. S. (2017). A parametric constrained segmentation methodology for application in sport marketing. *Customer Needs and Solutions*, *4*(4), 37-55.

DeSarbo, W. S., & Cron, W. L. (1988). A maximum likelihood methodology for clusterwise linear regression. *Journal of Classification*, *5*(2), 249-282.

DeSarbo, W. S., Fong, D. K., Liechty, J., & Saxton, M. K. (2004). A hierarchical Bayesian procedure for two-mode cluster analysis. *Psychometrika*, *69*(4), 547-572.

Efron, B., & Tibshirani, R. (1986). Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical Science*, 54-75.

Efron, B., & Tibshirani, R. (1997). Improvements on cross-validation: the 632+ bootstrap method. *Journal of the American Statistical Association*, *92*(438), 548-560.

Efron , G., & Nadif, M. (2003). Clustering with block mixture models. *Pattern Recognition*, *36*, 463–473.

Govaert, G., & Nadif, M. (2010). Latent block model for contingency table. *Communications in Statistics—Theory and Methods*, *39*, 416–425.

Govaert, G., & Nadif, M. (2013). *Co-clustering: models, algorithms and applications*. John Wiley & Sons.

Hanneman, R. A., & Riddle, M. (2005). Introduction to social network methods. Riverside, CA: University of California, Riverside (published in digital form at http://faculty.ucr.edu/~hanneman/)

He, R., & McAuley, J. (2016). Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web* (pp. 507-517). International World Wide Web Conferences Steering Committee.

Hu, N., Liu, L., & Zhang, J. J. (2008). Do online reviews affect product sales? The role of reviewer characteristics and temporal effects. *Information Technology and management*, *9*(3), 201-214.

Jaakkola, T. S., & Jordan, M. I. (1996). Computing upper and lower bounds on likelihoods in intractable networks. In *Proceedings of the Twelfth International Conference on Uncertainty in Artificial Intelligence* (pp. 340-348). Morgan Kaufmann Publishers Inc.

Jaakkola, T., & Jordan, M. I. (1997). Recursive algorithms for approximating probabilities in graphical models. In *Advances in Neural Information Processing Systems* (pp. 487-493).

Jacques, J., & Biernacki, C. (2018). Model-based co-clustering for ordinal data. *Computational Statistics & Data Analysis*, *123*, 101-115.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., & Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, *37*(2), 183-233.

Kamakura, W. A., & Russell, G. (1989). A probabilistic choice model for market segmentation and elasticity structure. *Journal of Marketing Research*, *26*, 379-390.

Karrer, B., & Newman, M. E. (2011). Stochastic block models and community structure in networks. *Physical Review E*, *83*(1), 016107.

Keribin, C., Brault, V., Celeux, G., & Govaert, G. (2015). Estimation and selection for the latent block model on categorical data. *Statistics and Computing*, 25(6), 1201-1216.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, *46*(5), 604-632.

Kim, S., Fong, D. K., & DeSarbo, W. S. (2012). Model-based segmentation featuring simultaneous segment-level variable selection. *Journal of Marketing Research*, *49*(5), 725-736.

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *IEEE Computer*, *42*(8), 30-37.

Kumar, S., Spezzano, F., Subrahmanian, V. S., & Faloutsos, C. (2016). Edge weight prediction in weighted signed networks. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on* (pp. 221-230). IEEE.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *The annals of mathematical statistics*, *22*(1), 79-86.

Larremore, D. B., Clauset, A., & Jacobs, A. Z. (2014). Efficiently inferring community structure in bipartite networks. *Physical Review E*, *90*(1), 012805.

Lee, K. H., Xue, L., & Hunter, D. R. (2020). Model-based clustering of time-evolving networks through temporal exponential-family random graph models. *Journal of Multivariate Analysis*, 175, 104540.

Lee, K. H., Agarwal, A., Zhang, A. Y., & Xue, L. (2022). Model-based clustering of semiparametric temporal exponential-family random graph models. *Stat*, 11(1), e459.

Li, T., Levina, E., & Zhu, J. (2020). Network cross-validation by edge sampling. *Biometrika*, 107(2), 257-276.

Luca, M. (2016). Reviews, reputation, and revenue: The case of Yelp. com. *Com (March 15, 2016). Harvard Business School NOM Unit Working Paper*, (12-016).

Matechou, E., Liu, I., Fernández, D., Farias, M., & Gjelsvik, B. (2016). Biclustering models for two-mode ordinal data. *Psychometrika*, *81*(3), 611-624.

Matias, C., & Miele, V. (2017). Statistical clustering of temporal networks through a dynamic stochastic block model. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *79*(4), 1119-1141.

McAuley, J., Pandey, R., & Leskovec, J. (2015, August). Inferring networks of substitutable and complementary products. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785-794). ACM.

McCullagh, P. (1980). Regression models for ordinal data. *Journal of the Royal Statistical Society. Series B*, 109-142.

Neal, R. M., & Hinton, G. E. (1998). A view of the EM algorithm that justifies incremental, sparse, and other variants. In *Learning in Graphical Models* (pp. 355-368). Springer, Dordrecht.

Nielsen, S. F. (2000). The stochastic EM algorithm: estimation and asymptotic results. *Bernoulli*, *6*(3), 457-489.

Pledger, S., & Arnold, R. (2014). Multivariate methods using mixtures: Correspondence analysis, scaling and pattern-detection. *Computational Statistics & Data Analysis*, *71*, 241-261.

Keshavan, R. H., Montanari, A., & Oh, S. (2010). Matrix completion from a few entries. *IEEE Transactions on Information Theory*, 56(6), 2980-2998.

Razaee, Z. S., Amini, A. A., & Li, J. J. (2019). Matched Bipartite Block Model with Covariates. *Journal of Machine Learning Research*, *20*(34), 1-44.

Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook. In *Recommender systems handbook* (pp. 1-35). Springer, Boston, MA.

Rocci, R.,&Vichi, M. (2008). Two-mode multi-partitioning. *Computational Statistics and Data Analysis*, *52*, 1984–2003.

Saldana, D. Franco, Yi Yu, and Yang Feng. How many communities are there? *Journal of Computational and Graphical Statistics* 26.1 (2017): 171-181.

Schafer, J. B., Frankowski, D., Herlocker, J., & Sen, S. (2007). Collaborative filtering recommender systems. In *The adaptive web* (pp. 291-324). Springer, Berlin, Heidelberg.

Stephens, M. (2000). Dealing with label switching in mixture models. *Journal of the Royal Statistical Society: Series B*, 62(4), 795-809.

Vichi, M. (2001). Double k-means clustering for simultaneous classification of objects and variables. In S. Borra, R. Rocci, M. Vichi, & M. Schader (Eds.), *Advances in Classification and Data Analysis*. (pp. 43–52). Berlin: Springer.

Vu, D. Q., Hunter, D. R., & Schweinberger, M. (2013). Model-based clustering of large networks. *The Annals of Applied Statistics*, *7*(2), 1010.

Van der Vaart, A. W. (2000). *Asymptotic Statistics*. Cambridge University Press.

Wasserman, S., & Faust, K. (1994). *Social Network Analysis: Methods and Applications* (Vol. 8). Cambridge University Press.

Wainwright, M. J., & Jordan, M. I. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, *1*(1–2), 1-305.

Wedel, M., & DeSarbo, W. S. (1995). A mixture likelihood approach for generalized linear models. *Journal of Classification*, *12*(1), 21-55.

Westling, T., & McCormick, T. H. (2019). Beyond prediction: A framework for inference with variational approximations in mixture models. *Journal of Computational and Graphical Statistics*, 28(4), 778-789.

Ying, Y., Feinberg, F., & Wedel, M. (2006). Leveraging missing ratings to improve online recommendation systems. *Journal of Marketing Research*, *43*(3), 355-365.

Zaheer, M., Wick, M. Tristan, J.-B., Smola, A. and Steele, G. (2016). Exponential stochastic cellular automata for massively parallel inference. In *Artificial Intelligence and Statistics*, pp. 966-975.

Zhou, Z., & Amini, A. A. (2020). Optimal bipartite network clustering. *Journal of Machine Learning Research*, *21*(40), 1-68.

Zhou, T., Ren, J., Medo, M., & Zhang, Y. C. (2007). Bipartite network projection and personal recommendation. *Physical Review E*, *76*(4), 046115.

Table 1 Monte Carlo Experimental Design Factors

Factors	Levels	Codes
Size of the Network (X1)	Small network: N=250, M=50	1
	Medium network: N=3000, M=300	2
	Large network: N=10000, M=3000	3
Number of Clusters (X2)	More: K=4, L=3	1
	Less: K=3, L=3	2
Existence of Covariates (X3)	With Covariates	1
	Without Covariates	2

Table 2 Overall Performance of Proposed Model and Benchmark Models for the Monte Carlo Study (Mean)

	Proposed Model	Benchmark 1	Benchmark 2	Benchmark 3	Benchmark 4	Benchmark 5
Hit Rate: Customers	0.90 (0.004)	0.77 (0.008)	0.77 (0.007)	0.81 (0.005)	0.87 (0.006)	0.86 (0.004)
Hit Rate: Products	0.96 (0.004)	0.96 (0.002)	0.81 (0.008)	0.97 (0.003)	0.92 (0.005)	0.98 (0.002)
RMSE ($\boldsymbol{\theta}^0$)	0.09 (0.004)	0.54 (0.017)	0.13 (0.004)	0.50 (0.017)	0.14 (0.004)	0.48 (0.018)
RMSE ($\boldsymbol{\theta}^u$)	0.08 (0.005)	NA	0.07 (0.004)	NA	0.10 (0.008)	NA
RMSE ($\boldsymbol{\theta}^p$)	0.02 (0.001)	NA	0.04 (0.002)	NA	0.02 (0.001)	NA
RMSE ($\boldsymbol{\omega}$)	0.19 (0.008)	NA	NA	0.30 (0.008)	0.27 (0.008)	0.18 (0.008)

Note: Numbers in parentheses are standard errors.

Table 3 Computational Efficiency Measures of Proposed Model and Benchmark Models for the Monte Carlo Study (Median)

		Network Size		
		Small Network	**Medium Network**	**Large Network**
Total Time (Unit: hour)	Proposed Model	0.07	1.42	26.20
	Benchmark 4	0.02	0.94	42.61
The Number of Iterations	Proposed Model	768.50	1528.50	507.50
	Benchmark 4	95.50	476.00	442.00
Time per Iteration (Unit: second)	Proposed Model	0.36	3.08	163.22
	Benchmark 4	0.68	7.09	232.52

Table 4 Descriptive Statistics of the Dataset

		Mean	Median	Min	Max	SD
Customer (N=13600)	Number of Products Rated Per Customer	4.68	4.00	1	48	3.04
	Average Rating Provided Per Customer	4.22	4.38	1	5	0.75
	Customer's Average Spending Level	21.33	15.96	0.01	342	21.41
Product (M=2657)	Number of Ratings Received Per Product	9.19	7	5	172	7.95
	Average Rating Received Per Product	4.25	4.33	1.8	5.0	0.52
	Product Price	20.17	12.99	0.01	295	24.26
	Best Sellers Rank: Top 100	5%				
	Best Sellers Rank: Top 101-500	11%				

Table 5 Descriptive Statistics of Customers by Cluster

† in the format Mean|Median|Min|Max

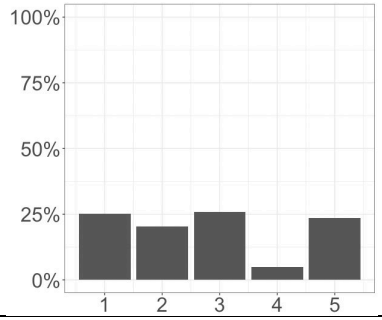
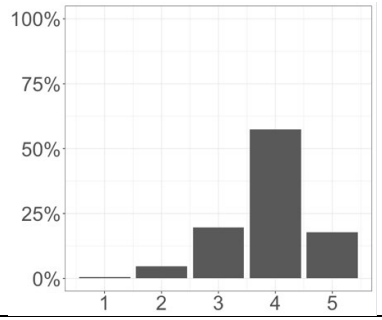
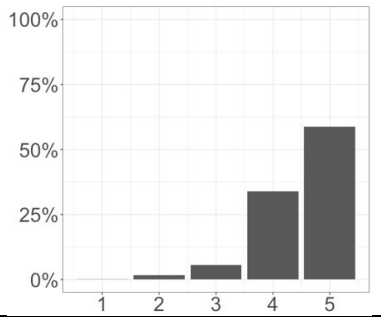
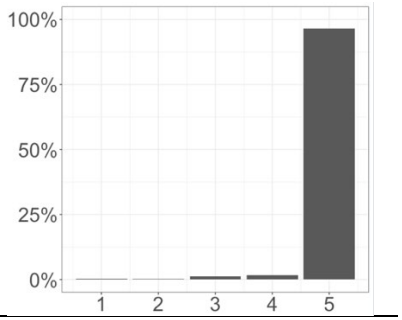
Customer Cluster	1: Critical Reviewer	2: Fair Reviewer	3: Friendly Reviewer	4: Super Nice
Cluster Size	2511	3456	841	6792
Mixing Proportion	18%	25%	6%	50%
Rating Distribution				
Avg Rating Per Customer†	3.32|3.50|1.00|4.80	3.98|4.00|1.60|4.92	4.40|4.50|2.83|4.93	4.65|4.86|1.80|5.00
No. of Ratings Per Customer†	4.59|4|1|46	4.77|4|1|35	6.27|6|2|44	4.46|4|1|48
Spending Power†	20.03|14.95|0.01|294.99	20.91|15.99|0.01|194.99	20.22|15.89|1.00|147.08	22.17|16.26|0.01|342.00
Review Propensity†	0.064%|0.054%|0.011%|0.220%	0.071%|0.053%|0.008%|0.355%	0.064%|0.054%|0.020%|0.194%	0.072%|0.055%|0.010%| 0.255%

Table 6 Descriptive Statistics of Products by Cluster

† in the format Mean|Median|Min|Max

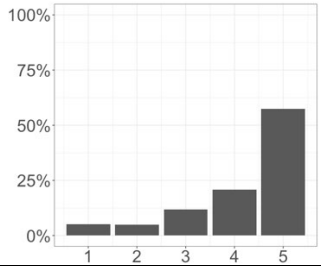
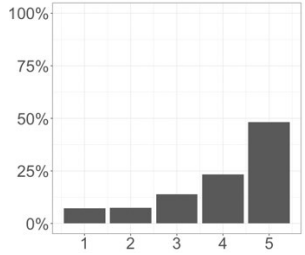
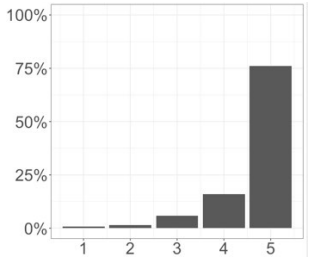
Product Cluster	1: Most Rated	2: Less and Lower Rated	3: Highest Rated
Cluster Size	229	1575	853
Mixing Proportion	9%	59%	32%
Rating Distribution			
Avg Rating Per Product†	4.22|4.27|2.45|4.89	4.01|4.13|1.80|4.80	4.70|4.71|4.12|5.00
No. of Ratings Per Product†	27.79|22|13|172	6.70|6|5|17	8.79|8|5|17
Price†	16.52|9.99|0.01|149.00	19.70|12.99|0.01|281.25	22.00|14.99|0.01|295.00
Review Propensity†	0.219%|0.214%|0.023%|0.355%	0.052%|0.053%|0.008%|0.061%	0.063%|0.062%|0.009%|0.081%
% as Top 100 Best Sellers	16%	2%	5%
% as Top 100-500 Best Sellers	18%	8%	13%

Table 7 Predictive Performance of Ratings

	Proposed	Sub-model	FunkSVD	ROptSpace
RMSE	1.111 (0.004)	1.145 (0.004)	1.526 (0.005)	4.359 (0.004)
Hit Rate of Top 25% Raters	0.320 (0.003)	0.283 (0.003)	0.299 (0.003)	0.229 (0.003)

Note: Numbers in parentheses are standard errors. Numbers are based on 500 random subsets of size 100.

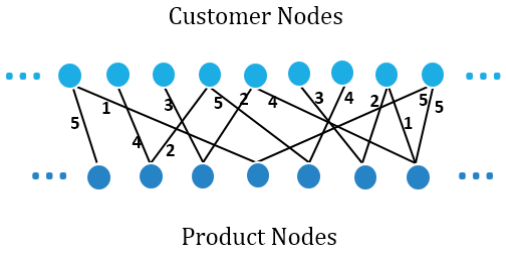


Figure 1 A Bipartite Network Representation of a Rating Database

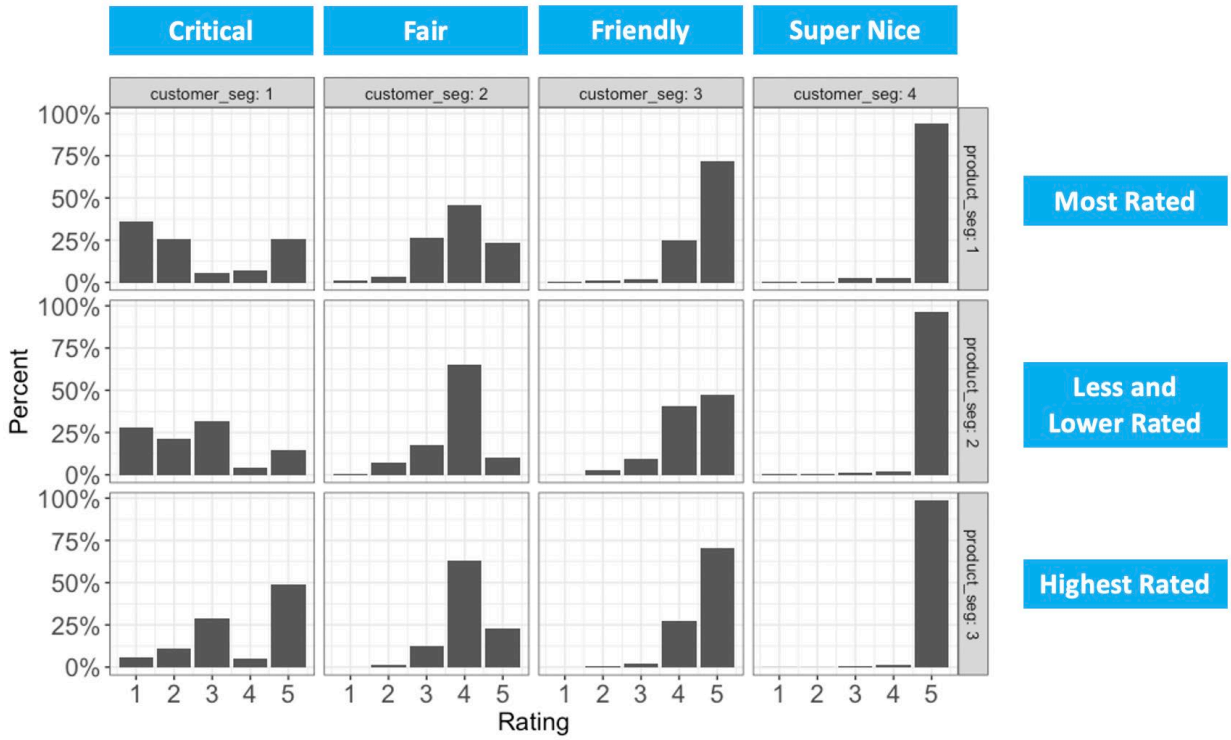


Figure 2 The Distributions of Observed Ratings for Customer-Product Cluster Pairs