Ordinal Non-negative Matrix Factorization for Recommendation

Olivier Gouvert, Thomas Oberlin, Cédric Févotte

Ordinal Non-negative Matrix Factorization for Recommendation

Olivier Gouvert [1] **Thomas Oberlin** [2] **Cédric Févotte** [1]

Abstract

We introduce a new non-negative matrix factorization (NMF) method for ordinal data, called OrdNMF. Ordinal data are categorical data which exhibit a natural ordering between the categories. In particular, they can be found in recommender systems, either with explicit data (such as ratings) or implicit data (such as quantized play counts). OrdNMF is a probabilistic latent factor model that generalizes Bernoulli-Poisson factorization (BePoF) and Poisson factorization (PF) applied to binarized data. Contrary to these methods, OrdNMF circumvents binarization and can exploit a more informative representation of the data. We design an efficient variational algorithm based on a suitable model augmentation and related to variational PF. In particular, our algorithm preserves the scalability of PF and can be applied to huge sparse datasets. We report recommendation experiments on explicit and implicit datasets, and show that OrdNMF outperforms BePoF and PF applied to binarized data.

1. Introduction

Collaborative filtering (CF) is a popular recommendation technique based only on the feedbacks of users on items. These feedbacks can be stored into a matrix $\mathbf{Y}$ of size $U \times I$, where U and I are the number of users and items respectively. Matrix factorization (MF) methods (Hu et al., 2008; Koren et al., 2009; Ma et al., 2011) aim to approximate the feedback matrix $\mathbf{Y}$ by a low-rank structure $\mathbf{W}\mathbf{H}^T$ where $\mathbf{W} \in \mathbb{R}_+^{U \times K}$ corresponds to user preferences and $\mathbf{H} \in \mathbb{R}_+^{I \times K}$ to item attributes.

Poisson factorization (PF) (Canny, 2004; Cemgil, 2009; Gopalan et al., 2015) is a non-negative matrix factorization (NMF) model (Lee & Seung, 1999; 2001; Févotte & Idier, 2011) which aims to predict future interactions between users and items in order to make recommendations. For this purpose, PF is often applied to a binarized version of the data, i.e., $\mathbf{Y} \in \{0, 1\}^{U \times I}$, containing only the information that a user is interacting with an item or not. A variant of PF, called Bernoulli-Poisson factorization (BePoF) (Acharya et al., 2015), has been proposed to explicitly model binary data. However, for both PF and BePoF, the binarization stage induces a loss of information, since the value associated to an interaction is removed. Although several attempts in the literature tried to directly model raw data, both for explicit (Hernandez-Lobato et al., 2014) and implicit data (Basbug & Engelhardt, 2016; Zhou, 2017; Gouvert et al., 2019), this remains a challenging problem.

In an attempt to keep as much information as possible, we propose in this paper to consider ordinal rather than binary data. Ordinal data (Stevens, 1946) are nominal/categorical data which exhibit a natural ordering (for example: cold $\prec$ warm $\prec$ hot). This type of data is encountered in recommender systems with explicit data such as ratings. It can also be created by quantizing implicit data such as play counts. Such a pre-processing remains softer than binarization and stays closer to the raw data, as soon as the number of classes is chosen big enough. In this paper, without loss of generality, we will work with ordinal data belonging to $\{0, \dots, V\}$. Note that for this type of data, the notion of distance between the different classes is not defined. For example, this implies that the mean is not adapted to these data, unlike the median.

There are two naive ways to process ordinal data. The first one consists in applying classification methods. The scale of the ordering relation which links the different categories is then ignored. The second way considers these data as real values in order to apply regression models. By doing this, it artificially creates a distance between the different categories. These two naive methods do not fully consider the specificity of ordinal data, since they remove or add information to the data. Threshold models (McCullagh, 1980; Verwaeren et al., 2012) are popular ordinal data processing methods that alleviate this issue. They assume that the data results from the quantization of continuous latent variables with respect to (w.r.t.) an increasing sequence of thresholds. The aim of these models is then to train a predictive model on the latent variables and to learn the sequence of thresh-

[1]IRIT, Université de Toulouse, CNRS, France [2]ISAE-SUPAERO, Université de Toulouse, France. Correspondence to: Olivier Gouvert <oliviergouvert@gmail.com>.

Proceedings of the 37^{th} International Conference on Machine Learning, Vienna, Austria, PMLR 119, 2020.

olds. Threshold models can thus be seen as an extension of naive regression models, where the distances between the different classes are learned through quantization thresholds. For a comprehensive and more detailed review on ordinal regression methods, we refer to (Gutierrez et al., 2015).

In this paper, we develop a new probabilistic NMF framework for ordinal data, called ordinal NMF (OrdNMF). OrdNMF is a threshold model where the latent variables have an NMF structure. In other words, this amounts to defining the approximation $\mathbf{Y} \approx G(\mathbf{W}\mathbf{H}^T)$, where $\mathbf{Y}$ is the ordinal data matrix, $\mathbf{W}$ and $\mathbf{H}$ are non-negative matrices and $G(\cdot)$ is a link function. OrdNMF allows us to work on more informative class of data than classical PF method by circumventing binarization. Contrary to ordinal MF (OrdMF) models (Chu & Ghahramani, 2005; Koren & Sill, 2011; Paquet et al., 2012; Hernandez-Lobato et al., 2014), OrdNMF imposes non-negativity constraints on both $\mathbf{W}$ and $\mathbf{H}$. This implies a more intuitive part-based representation of the data (Lee & Seung, 1999), and were shown to improve results in recommendation (Gopalan et al., 2015). OrdNMF can efficiently take advantage of the sparsity of $\mathbf{Y}$, scaling with the number of non-zero observations. Thereby, it can be applied to huge sparse datasets such as those commonly encountered in recommender systems. As opposed to learning-to-rank models, the aim of OrdNMF is to model ordinal data, via a generative probabilistic model, in order to predict the class of future interactions. Learning-to-rank models do not seek to predict a class but to rank items relatively to each other. For example, Bayesian personalized ranking (Rendle et al., 2009) is based on binary pairwise comparisons of the users' preferences and not on the raw matrix $\mathbf{Y}$. Although such models can also be used for recommendation, they are not generative.

The contributions of this paper are the following.

• We propose a new NMF model for ordinal data based on multiplicative noise. In particular, we study an instance of this model where the noise is assumed to be drawn from an inverse-gamma (IG) distribution. We show that this instance is an extension of BePoF (Acharya et al., 2015) and PF (Gopalan et al., 2015) applied to binarized data.

• We use a model augmentation trick to design an efficient variational algorithm, both for the update rules of the latent factors $\mathbf{W}$ and $\mathbf{H}$, and for those of the thresholds $\mathbf{b}$. In particular, this variational algorithm scales with the number of non-zero values in $\mathbf{Y}$.

• We report the results of OrdNMF on recommendation tasks for two datasets (with explicit and implicit feedbacks). Moreover, posterior predictive checks (PPCs) demonstrate the excellent flexibility of OrdNMF and its ability to represent various kinds of datasets.

The rest of the paper is organized as follows. In Section 2, we present important related works on cumulative link models and on BePoF. In Section 3, we present our general OrdNMF model and detail a particular instance. In Section 4, we develop an efficient VI algorithm which scales with the number of non-zero values in the data. In Section 5, we test our algorithm on recommendation tasks for explicit and implicit datasets. Finally, in Section 6, we conclude and discuss the perspectives of this work.

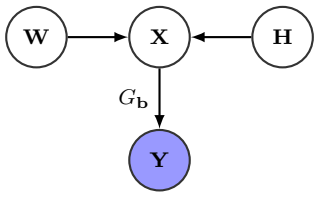

Figure 1. Graphical model of OrdMF. A latent variable $\mathbf{X}$ is introduced to make the link between the factorization term $\mathbf{W}\mathbf{H}^T$ and the ordinal data $\mathbf{Y}$.

2. Related Works

2.1. Cumulative Link Models (CLMs)

CLMs were one of the first threshold models proposed for ordinal regression (Agresti & Kateri, 2011). These models have been adapted to deal with the MF problem, leading to OrdMF models. They amount to finding the approximation $\mathbf{Y} \approx G(\mathbf{W}\mathbf{H}^T)$, where $\mathbf{Y} \in \{0, \dots, V\}^{U \times I}$ is an ordinal data matrix, $\mathbf{W} \in \mathbb{R}^{U \times K}$ and $\mathbf{H} \in \mathbb{R}^{I \times K}$ are latent factors, and $G(\cdot)$ is a parametrized link function described subsequently. OrdMF has been applied mainly to explicit data in order to predict users feedbacks (Chu & Ghahramani, 2005; Paquet et al., 2012).

The idea behind threshold models is to introduce a continuous latent variable $x_{ui} \in \mathbb{R}$ that is mapped to the ordinal data y_{ui}. This is done by considering an increasing sequence of thresholds $b_{-1} = -\infty < b_0 < \cdots < b_{V-1} < b_V = +\infty$, denoted by $\mathbf{b}$, which fully characterize the following quantization function, illustrated in Figure 2, by:

$$G_{\mathbf{b}} : \begin{array}{ccc} \mathbb{R} & \to & \{0, \dots, V\} \\ x & \mapsto & v \text{ such as } x \in [b_{v-1}, b_v). \end{array} \tag{1}$$

Therefore, ordinal data result from the quantization of the variable x_{ui} by the step function $G_{\mathbf{b}}$, i.e., $y_{ui} = G_{\mathbf{b}}(x_{ui})$. The latent variable x_{ui} corresponds to the variable $\lambda_{ui} = [\mathbf{W}\mathbf{H}^T]_{ui} \in \mathbb{R}$ perturbed by an additive noise ε_{ui}, whose cumulative density function (c.d.f.) is denoted by $F_\varepsilon : \mathbb{R} \to [0, 1]$. Thus, we obtain the following generative model, illustrated in Figure 1:

$$x_{ui} = \lambda_{ui} + \varepsilon_{ui}, \tag{2}$$

$$y_{ui} = G_{\mathbf{b}}(x_{ui}). \tag{3}$$

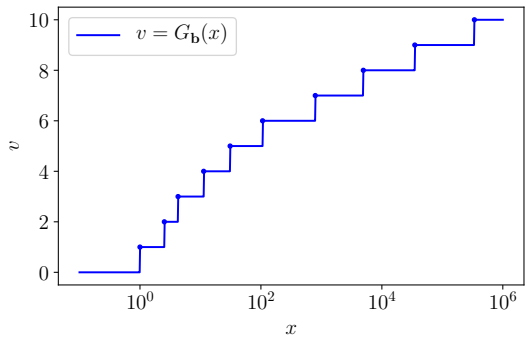

Figure 2. Example of a quantization function $x \mapsto G_{\mathbf{b}}(x)$.

The goal of MF models for ordinal data is therefore to jointly infer the latent variables $\mathbf{W}$ and $\mathbf{H}$ as well as the sequence of thresholds $\mathbf{b}$.

Cumulative distribution function. The c.d.f. associated with the random variable y_{ui} in Eqs. (2)-(3) can be calculated as follows:

$$\begin{aligned}\mathbb{P}[y_{ui} \leq v|\lambda_{ui}] &= \mathbb{P}[G_{\mathbf{b}}(x_{ui}) \leq v|\lambda_{ui}] && (4)\\ &= \mathbb{P}[\lambda_{ui} + \varepsilon_{ui} < b_v] && (5)\\ &= \mathbb{P}[\varepsilon_{ui} < b_v - \lambda_{ui}] && (6)\\ &= F_\varepsilon\left(b_v - \lambda_{ui}\right). && (7)\end{aligned}$$

It follows that the function $v \mapsto \mathbb{P}[y_{ui} \leq v|\lambda_{ui}]$ is increasing since the sequence of thresholds is itself increasing. Moreover, the probability mass function (p.m.f.) associated to the ordinal data can be written as:

$$\begin{aligned}\mathbb{P}[y_{ui} = v|\lambda_{ui}] &= \mathbb{P}[y_{ui} \leq v|\lambda_{ui}] - \mathbb{P}[y_{ui} \leq v-1|\lambda_{ui}]\\ &= F_\varepsilon\left(b_v - \lambda_{ui}\right) - F_\varepsilon\left(b_{v-1} - \lambda_{ui}\right). \quad (8)\end{aligned}$$

Some examples. If the c.d.f. is strictly increasing, we can rewrite Eq. (7) as:

$$F_\varepsilon^{-1}(\mathbb{P}[y_{ui} \leq v|\lambda_{ui}]) = b_v - \lambda_{ui}. \quad (9)$$

Hence the name of CLM, since the factorization model is related to the c.d.f. of the ordinal data through a link function $F_\varepsilon^{-1} : [0,1] \to \mathbb{R}$. Various choices of noise (equivalently, of link function F_ε^{-1}) have been considered in the literature. We present some of these choices in what follows.

• Logit function. The use of the logit function was first proposed in (Walker & Duncan, 1967). This model was popularized and renamed as "proportional odds model" by (McCullagh, 1980). The model can be rewritten as:

$$\operatorname{logit} \mathbb{P}[y_{ui} \leq v|\lambda_{ui}] = \log \frac{\mathbb{P}[y_{ui} \leq v|\lambda_{ui}]}{\mathbb{P}[y_{ui} > v|\lambda_{ui}]} = b_v - \lambda_{ui} \quad (10)$$

• Probit function. A common choice for the additive noise is $\varepsilon_{ui} \sim \mathcal{N}(0, \sigma^2)$ (Chu & Ghahramani, 2005; Paquet et al., 2012; Hernandez-Lobato et al., 2014). In that case the link function F_ε^{-1} is the probit function. Inference can be carried out with an EM algorithm based on the latent variable x_{ui}.

• Other choices like log-log or cauchit functions have also been considered (Agresti & Kateri, 2011). The survey (Ananth & Kleinbaum, 1997) recaps some of these choices.

2.2. Bernoulli-Poisson Factorization (BePoF)

In this section, we present BePoF (Acharya et al., 2015) which is a variant of PF for binary data (not directly related to the CLMs introduced above). It employs a model augmentation trick for inference that we will use in our own algorithm presented in Section 4.

The Poisson distribution can easily be "augmented" to fit binary data $y_{ui} \in \{0, 1\}$. Indeed, it suffices to introduce a thresholding operation that binarizes the data. The corresponding generative hierarchical model is therefore given by:

$$\begin{aligned}n_{ui} &\sim \text{Poisson}([\mathbf{W}\mathbf{H}^T]_{ui}), && (11)\\ y_{ui} &= \mathbb{1}[n_{ui} > 0], && (12)\end{aligned}$$

where $n_{ui} \in \mathbb{N}$ is a latent variable and $\mathbb{1}$ is the indicator function. We denote by $\mathbf{N} \in \mathbb{N}^{U \times I}$ the matrix such that $[\mathbf{N}]_{ui} = n_{ui}$. This variable can easily be marginalized by noting that $\mathbb{P}[y_{ui} = 0] = \text{Poisson}(0|[\mathbf{W}\mathbf{H}^T]_{ui}) = e^{-[\mathbf{W}\mathbf{H}^T]_{ui}}$. We obtain:

$$y_{ui} \sim \text{Bern}(1 - e^{-[\mathbf{W}\mathbf{H}^T]_{ui}}) \quad (13)$$

where Bern refers to the Bernoulli distribution. The conditional distribution of the latent variable n_{ui} is given by:

$$n_{ui}|y_{ui} \sim \begin{cases}\delta_0, & \text{if } y_{ui} = 0,\\ \text{ZTP}([\mathbf{W}\mathbf{H}^T]_{ui}), & \text{if } y_{ui} = 1.\end{cases} \quad (14)$$

where ZTP refers to the zero-truncated Poisson distribution and δ_0 to the Dirac distribution located in 0. The latent variable $\mathbf{N}$ can be useful to design Gibbs or variational inference (VI) algorithms for binary PF (Acharya et al., 2015) and we will employ a similar trick in Section 4.

Remark. The generative model presented in Eq. (13) is in the form $y_{ui} \sim \text{Bern}(G([\mathbf{W}\mathbf{H}^T]_{ui}))$ where $G : \mathbb{R}$ (or $\mathbb{R}_+$) $\to [0,1]$. When $[\mathbf{W}\mathbf{H}^T]_{ui} \in \mathbb{R}$, the function G can be the inverse of the probit (Consonni & Marin, 2007) or of the logit function for example. They are special cases of the model presented in Section 2.1 with $V = 1$. Mean-parametrized Bernoulli MF models have also been considered (Lumbreras et al., 2018). They correspond to $G = \text{Id}$ and require additional constraints on the latent factors $\mathbf{W}$ and $\mathbf{H}$ in order to satisfy $[\mathbf{W}\mathbf{H}^T]_{ui} \in [0,1]$.

3. Ordinal NMF (OrdNMF)

In this section, we introduce OrdNMF which is a NMF model pecially designed for ordinal data. A difference with Section 2.1 is that we impose that both matrices $\mathbf{W}$ and $\mathbf{H}$ are non-negative. Thus, we now have $[\mathbf{W}\mathbf{H}^T]_{ui} \in \mathbb{R}_+$ instead of $[\mathbf{W}\mathbf{H}^T]_{ui} \in \mathbb{R}$. We denote $\lambda_{uik} = w_{uk}h_{ik}$ so that $\lambda_{ui} = \sum_k \lambda_{uik} = [\mathbf{W}\mathbf{H}^T]_{ui}$.

3.1. Quantization of the Non-negative Numbers

Our model works on the same principle as OrdMF (see Section 2.1) and seeks to quantize the non-negative real line $\mathbb{R}_+$. For this, we introduce the increasing sequence of thresholds $\mathbf{b}$ given by $b_{-1} = 0 < b_0 < \cdots < b_{V-1} < b_V = +\infty$ (the thresholds are here non-negative). Moreover, we define the quantization function $G_{\mathbf{b}} : \mathbb{R}_+ \to \{0, \dots, V\}$ like in Eq. (1) but with support $\mathbb{R}_+$.

As compared to Section 2.1, we now assume a non-negative multiplicative noise on x_{ui}. This ensures the non-negativity of x_{ui} and it seems well suited for modeling over-dispersion, a common feature of recommendation data. Let ε_{ui} be a non-negative random variable with c.d.f. F_ε, we thus propose the following generative model:

$$x_{ui} = \lambda_{ui} \cdot \varepsilon_{ui}, \tag{15}$$
$$y_{ui} = G_{\mathbf{b}}(x_{ui}). \tag{16}$$

Like before, our goal is to jointly infer the latent variables $\mathbf{W}$ and $\mathbf{H}$ as well as the sequence of the thresholds $\mathbf{b}$. In our model, the c.d.f. associated to the ordinal random variable y_{ui} becomes:

$$\begin{aligned}
\mathbb{P}[y_{ui} \le v|\lambda_{ui}] &= \mathbb{P}[G_{\mathbf{b}}(x_{ui}) \le v|\lambda_{ui}] && (17)\\
&= \mathbb{P}[\lambda_{ui} \cdot \varepsilon_{ui} < b_v] && (18)\\
&= \mathbb{P}\left[\varepsilon_{ui} < \frac{b_v}{\lambda_{ui}}\right] && (19)\\
&= F_\varepsilon\left(\frac{b_v}{\lambda_{ui}}\right). && (20)
\end{aligned}$$

Therefore, we can deduce that the p.m.f. is given by:

$$\begin{aligned}
&\mathbb{P}[y_{ui} = v|\lambda_{ui}]\\
&= \mathbb{P}[y_{ui} \le v|\lambda_{ui}] - \mathbb{P}[y_{ui} \le v-1|\lambda_{ui}] && (21)\\
&= F_\varepsilon\left(\frac{b_v}{\lambda_{ui}}\right) - F_\varepsilon\left(\frac{b_{v-1}}{\lambda_{ui}}\right). && (22)
\end{aligned}$$

Various functions F_ε can be used which determine the exact nature of the multiplicative noise. Figure 3 displays the function $\lambda \mapsto F_\varepsilon(\lambda^{-1})$ for the examples considered next.

• Gamma noise: $\varepsilon_{ui} \sim \text{Gamma}(\alpha, 1)$.[1] The c.d.f. is given by $F_\varepsilon(x) = \frac{\gamma(\alpha,x)}{\Gamma(\alpha)}$ where $\gamma(\alpha, x) = \int_0^x t^{\alpha-1}e^{-t}\text{d}t$ is the lower incomplete gamma function. If $\alpha = 1$, we recover an exponential noise $\varepsilon_{ui} \sim \text{Exp}(1)$ whose c.d.f. is $F_\varepsilon(x) = 1 - e^{-x}$.

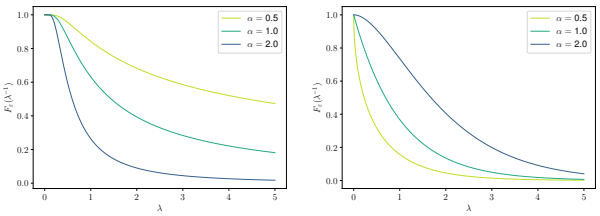

Figure 3. Functions $\lambda \mapsto F_\varepsilon(\lambda^{-1})$ for gamma (left) and inverse-gamma (right) noises.

• Inverse-gamma (IG) noise: $\varepsilon_{ui} \sim \text{IG}(\alpha, 1)$.[1] The c.d.f. is given by $F_\varepsilon(x) = \frac{\Gamma(\alpha,x^{-1})}{\Gamma(\alpha)}$ where $\Gamma(\alpha, x) = \int_x^\infty t^{\alpha-1}e^{-t}\text{d}t$ is the upper incomplete gamma function. If $\alpha = 1$, we obtain the c.d.f. $F_\varepsilon(x) = e^{-1/x}$.

• Any increasing function $F_\varepsilon : \mathbb{R}_+ \to [0, 1]$ defines a non-negative random variable which can be used in OrdNMF.

3.2. OrdNMF with IG Noise (IG-OrdNMF)

In the rest of the paper, we focus on the special case where ε_{ui} is a multiplicative IG noise with shape parameter $\alpha = 1$, i.e., $\varepsilon_{ui} \sim \text{IG}(1, 1)$[2]. We use the acronym IG-OrdNMF for this particular instance of OrdNMF.

For convenience we write $\theta_v = b_v^{-1}$. The sequence $\boldsymbol{\theta}$ corresponds to the inverse of the thresholds and is therefore decreasing, i.e., $\theta_{-1} = +\infty > \theta_0 > \cdots > \theta_{V-1} > \theta_V = 0$. Moreover, we denote by $\boldsymbol{\Delta}$ the positive sequence of decrements defined by $\Delta_v = \theta_{v-1} - \theta_v$ for $v \in \{1, \dots, V\}$. We have $\theta_v = \sum_{l=v+1}^V \Delta_l$ and, in particular, $\theta_{V-1} = \Delta_V$.

Interpretation. In IG-OrdNMF model, the c.d.f. associated with an ordinal data y_{ui} is given by:

$$\mathbb{P}[y_{ui} \le v|\lambda_{ui}] = e^{-\lambda_{ui}\theta_v}, \tag{23}$$
$$\text{or } \mathbb{P}[y_{ui} > v|\lambda_{ui}] = 1 - e^{-\lambda_{ui}\theta_v}, \tag{24}$$

with $v \in \{0, \dots, V\}$. Therefore, BePoF (see Section 2.2) is a particular case of IG-OrdNMF with $V = 1$ and $\theta_0 = 1$.

This formulation allows for a new interpretation of IG-OrdNMF. As a matter of fact, the event $\{y_{ui} > v\}$ is a binary random variable which follows a Bernoulli distribution: $\{y_{ui} > v\} \sim \text{Bern}(1-e^{-\lambda_{ui}\theta_v})$. Then, we can see IG-OrdNMF as the aggregation of V dependent BePoF models for different thresholds of binarization $v \in \{0, \dots, V-1\}$.

[1] The rate parameter β is fixed to 1 because of a scale invariance with λ_{ui}.

[2] The expectation of a IG variable is not defined for $\alpha \le 1$, however the model is still well-defined.

Probability mass function. The p.m.f. of an observation is given by:

$$\mathbb{P}[y_{ui} = v|\lambda_{ui}] = \begin{cases} e^{-\lambda_{ui}\theta_0}, & \text{for } v = 0, \\ e^{-\lambda_{ui}\theta_v} - e^{-\lambda_{ui}\theta_{v-1}}, & \text{for } 1 \leq v < V, \\ 1 - e^{-\lambda_{ui}\theta_{V-1}}, & \text{for } v = V. \end{cases} \tag{25}$$

Then, the log-likelihood of λ_{ui} can be written as:

$$\log \mathbb{P}[y_{ui} = v|\lambda_{ui}] = \begin{cases} -\lambda_{ui}\theta_0, \text{ if } v = 0, \\ -\lambda_{ui}\theta_v + \log(1 - e^{-\lambda_{ui}\Delta_v}), \text{ else.} \end{cases} \tag{26}$$

This expression brings up a linear term in λ_{ui} and a non-linear term of the form $x \mapsto \log(1 - e^{-x})$, similar to the function used in Section 2.2.

Moreover, the expectation of the observations is well-defined and given by :

$$\mathbb{E}(y_{ui}|\lambda_{ui}) = V - \sum_{v=0}^{V-1} e^{-\lambda_{ui}\theta_v}. \tag{27}$$

Note that, in the context of ordinal data processing, the expectation is not a good statistic since it implicitly implies a notion of distance between classes. However, this quantity will be useful to build lists of recommendations. Indeed, the function $\lambda_{ui} \mapsto \mathbb{E}(y_{ui}|\lambda_{ui})$ is increasing. Thus, the higher the $\lambda_{ui} = [\mathbf{W}\mathbf{H}^T]_{ui}$, the higher (in expectation) the level of interaction between the user and the item.

4. Bayesian Inference

We impose a gamma prior on the entries of both matrices $\mathbf{W}$ and $\mathbf{H}$, i.e., $w_{uk} \sim \text{Gamma}(\alpha^W, \beta_u^W)$ and $h_{ik} \sim \text{Gamma}(\alpha^H, \beta_i^H)$. Gamma prior is known to induce sparsity which is a desirable property in NMF methods.

4.1. Augmented Model

As described in Section 3.2 , the log-likelihood for ordinal data such that $v \in \{1, \dots, V\}$ brings up a non-linear term $\log(1 - e^{-x})$ which is not conjugate with the gamma distribution, making the inference complicated. To solve this issue, we use the trick presented in Section 2.2 by augmenting our model with the latent variable:

$$n_{ui}|y_{ui}, \lambda_{ui} \sim \begin{cases} \delta_0, & \text{if } y_{ui} = 0, \\ \text{ZTP}(\lambda_{ui}\Delta_{y_{ui}}), & \text{if } y_{ui} > 0. \end{cases} \tag{28}$$

Moreover, as commonly done in the PF setting (Cemgil, 2009; Gopalan et al., 2015), we augment our model with the latent variable $\mathbf{c}_{ui}|n_{ui}, \lambda_{ui} \sim \text{Mult}(n_{ui}, \boldsymbol{\phi}_{ui})$, where Mult is the multinomial distribution and $\boldsymbol{\phi}_{ui}$ is a probability vector with entries $\frac{\lambda_{uik}}{\lambda_{ui}}$. Therefore, for ordinal data $y_{ui} \in \{1, \dots, V\}$, we obtain the following joint log-likelihood:

$$\begin{aligned} &\log p(y_{ui}, n_{ui}, \mathbf{c}_{ui}|\lambda_{ui}) = -\lambda_{ui}\theta_{y_{ui}-1} \\ &+ n_{ui} \log \Delta_{y_{ui}} + \sum_k \left(c_{uik} \log \lambda_{uik} - \log c_{uik}!\right), \\ &\text{s.t. } n_{ui} \in \mathbb{N}^* \text{ and } n_{ui} = \sum_k c_{uik}. \end{aligned} \tag{29}$$

Table 1. Variational distributions for IG-OrdNMF.

Var.	Distribution
$\mathbf{C}$	$q(\mathbf{c}_{ui}\|n_{ui}) = \text{Mult}\left(\mathbf{c}_{ui};\ n_{ui},\ \tilde{\boldsymbol{\phi}}_{ui}\right)$
$\mathbf{N}$	$q(n_{ui}) = \begin{cases} \delta_0, & \text{if } y_{ui} = 0 \\ \text{ZTP}(n_{ui};\ \Lambda_{ui}\Delta_{y_{ui}}), & \text{if } y_{ui} > 0 \end{cases}$
$\mathbf{W}$	$q(w_{uk}) = \text{Gamma}\left(w_{uk};\ \tilde{\alpha}_{uk}^W,\ \tilde{\beta}_{uk}^W\right)$
$\mathbf{H}$	$q(h_{ik}) = \text{Gamma}\left(h_{ik};\ \tilde{\alpha}_{ik}^H,\ \tilde{\beta}_{ik}^H\right)$

Joint log-likelihood of IG-OrdNMF. We denote by $\mathbf{Z} = \{\mathbf{N}, \mathbf{C}, \mathbf{W}, \mathbf{H}\}$ the set of latent variables of the augmented model. Moreover, we define T_v such that:

$$T_v = \begin{cases} \theta_0, & \text{if } v = 0, \\ \theta_{v-1}, & \text{if } v > 0. \end{cases} \tag{30}$$

The joint log-likelihood of IG-OrdNMF is therefore given by:

$$\begin{aligned} &\log p(\mathbf{Y}, \mathbf{N}, \mathbf{C}|\mathbf{W}, \mathbf{H}) = \sum_{ui} \Big[n_{ui} \log \Delta_{y_{ui}} \\ &+ \sum_k \left(c_{uik} \log \lambda_{uik} - \log c_{uik}!\right) - \lambda_{ui} T_{y_{ui}} \Big]. \end{aligned} \tag{31}$$

It is important to note that $n_{ui} = 0$ and $\mathbf{c}_{ui} = \mathbf{0}_K$ when $y_{ui} = 0$. Consequently, the variables $\mathbf{N}$ and $\mathbf{C}$ are partially observed and the inference take advantage of the sparsity of the observed matrix $\mathbf{Y}$.

4.2. Variational Inference

The posterior distribution $p(\mathbf{Z}|\mathbf{Y})$ is intractable. We use VI to approximate this distribution by a simpler variational distribution q. Here, we assume that q belongs to the mean-field family and can be written in the following factorized form:

$$q(\mathbf{Z}) = \prod_{ui} q(n_{ui}, \mathbf{c}_{ui}) \prod_{uk} q(w_{uk}) \prod_{ik} q(h_{ik}). \tag{32}$$

Note that the variables $\mathbf{N}$ and $\mathbf{C}$ remains coupled. We use a coordinate-ascent VI (CAVI) algorithm to optimize the

parameters of q. The variational distributions are described in Table 1. The associated update rules are summarized in Algorithm 1.

Approximation and link with PF. Algorithm 1 can be simplified by assuming that $q(n_{ui}) = \delta_1$ if $y_{ui} > 0$. This amounts to replacing the non-linear term $\log(1 - e^{-x})$ by $\log x$ in Eq. (26). However, this approximation will produce similar results only if x is very small, since $\log(1 - e^{-x}) = \log x + o(x)$. In practice, this can only be verified a posteriori by observing that $\mathbb{E}_q(n_{ui}) \approx 1$.

As mentioned above, BePoF is a special case of IG-OrdNMF for $V = 1$ and $\theta_0 = 1$. Thus, we can notice that PF algorithm applied to binary data is an approximation of BePoF algorithm for $q(n_{ui}) = \delta_1$ if $y_{ui} = 1$.

4.3. Thresholds Estimation

A key element of threshold models is the learning of thresholds (corresponding here to $\boldsymbol{\theta}$ parameters). For this, we use a VBEM algorithm. It aims to maximize the term $\mathbb{E}_q(\log p(\mathbf{Y}, \mathbf{Z}; \boldsymbol{\theta}))$, w.r.t. the variables $\boldsymbol{\theta}$, which is given by:

$$\mathbb{E}_q(\log p(\mathbf{Y}, \mathbf{Z}; \boldsymbol{\theta})) = \sum_{ui} \left[\mathbb{E}_q(n_{ui}) \log \Delta_{y_{ui}} - \mathbb{E}_q(\lambda_{ui}) T_{y_{ui}} \right] + cst, \quad \text{s.t. } \theta_0 > \theta_1 > \cdots > \theta_{V-1} > \theta_V = 0. \tag{33}$$

Note that both terms T_v (defined in Eq. (30)) and $\Delta_v = \theta_{v-1} - \theta_v > 0$ depend on the sequence $\boldsymbol{\theta}$.

Decrements optimization. We choose to work on the decrement sequence $\boldsymbol{\Delta}$ rather than on the threshold sequence $\boldsymbol{\theta}$. Indeed, by doing so, the decreasing constraint of $\boldsymbol{\theta}$ becomes a non-negativity constraint of $\boldsymbol{\Delta}$. Moreover, we obtain only terms in x and $\log x$ in the function to be maximized. Thus, the problem can be solved analytically.

We can rewrite the term T_v w.r.t. the sequence $\boldsymbol{\Delta}$ by noting that: $T_v = \sum_{l=1}^{V} \mathbb{1}[v \leq l] \Delta_l$, $\forall v \in \{0, \ldots, V\}$. Therefore, the optimization problem presented in Eq. (33) amounts to maximizing the following function:

$$\mathbb{E}_q(\log p(\mathbf{Y}, \mathbf{Z}; \boldsymbol{\Delta})) = \sum_{ui} \sum_{l=1}^{V} \Big[\mathbb{1}[y_{ui} = l] \mathbb{E}_q(n_{ui}) \log \Delta_l - \mathbb{1}[y_{ui} \leq l] \mathbb{E}_q(\lambda_{ui}) \Delta_l \Big] + cst, \text{ s.t. } \boldsymbol{\Delta} \geq 0. \tag{34}$$

Algorithm 1 CAVI for IG-OrdNMF.

Data: Matrix $\mathbf{Y}$
Result: Variational distribution q and thresholds $\boldsymbol{\theta}$

1 Initialization of variational parameters and thresholds $\boldsymbol{\theta}$;
repeat
2 **foreach** *couple* (u, i) *such as* $y_{ui} > 0$ **do**
3 $\Lambda_{uik} = \exp\left(\mathbb{E}_q(\log w_{uk}) + \mathbb{E}_q(\log h_{ik})\right)$;
  $\Lambda_{ui} = \sum_k \Lambda_{uik}$;
  $\mathbb{E}_q(n_{ui}) = \frac{\Lambda_{ui} \Delta_{y_{ui}}}{1 - e^{-\Lambda_{ui} \Delta_{y_{ui}}}}$;
  $\mathbb{E}_q(c_{uik}) = \mathbb{E}_q(n_{ui}) \frac{\Lambda_{uik}}{\Lambda_{ui}}$;
4 **end**
5 **foreach** *user* $u \in \{1, \ldots, U\}$ **do**
6 $\tilde{\alpha}^W_{uk} = \alpha^W + \sum_i \mathbb{E}_q(c_{uik})$;
  $\tilde{\beta}^W_{uk} = \beta^W_u + \sum_i T_{y_{ui}} \mathbb{E}_q(h_{ik})$;
7 **end**
8 **foreach** *item* $i \in \{1, \ldots, I\}$ **do**
9 $\tilde{\alpha}^H_{ik} = \alpha^H + \sum_u \mathbb{E}_q(c_{uik})$;
  $\tilde{\beta}^H_{ik} = \beta^H_i + \sum_u T_{y_{ui}} \mathbb{E}_q(w_{uk})$;
10 **end**
11 Update of thresholds: Eq. (35) and Eq. (36);
  Update of rate parameters β^W_u and β^H_i;
  Calculate ELBO$(q, \boldsymbol{\theta})$;
12 **until** *ELBO converge*;

Thus, we obtain the following update rules:

$$\Delta_l = \frac{\sum_{ui} \mathbb{1}[y_{ui} = l] \mathbb{E}_q(n_{ui})}{\sum_{ui} \mathbb{1}[y_{ui} \leq l] \mathbb{E}_q(\lambda_{ui})}, \forall l \in \{1, \ldots, V\}, \tag{35}$$

$$\theta_v = \sum_{l=v+1}^{V} \Delta_l, \forall v \in \{0, \ldots, V-1\}. \tag{36}$$

Algorithm 1 scales with the number of non-zero values in the observation matrix $\mathbf{Y}$. The complexity of OrdNMF is of the same order of magnitude as BePoF and PF. The only difference with these algorithms in terms of computational complexity is the update of the thresholds (Line 11 of Alg. 1).

4.4. Posterior Predictive Expectation.

The posterior predictive expectation $\mathbb{E}(\mathbf{Y}^*|\mathbf{Y})$ corresponds to the expectation of the distribution of new observations $\mathbf{Y}^*$ given previously observed data $\mathbf{Y}$. This quantity allows us to create the list of recommendations for each user. We can approximate it by using the variational distribution q:

$$\mathbb{E}(\mathbf{Y}^*|\mathbf{Y}) \approx \int_{\mathbf{W},\mathbf{H}} \mathbb{E}(\mathbf{Y}^*|\mathbf{W}, \mathbf{H}) q(\mathbf{W}) q(\mathbf{H}) \mathrm{d}\mathbf{W} \mathrm{d}\mathbf{H}. \tag{37}$$

Unfortunately, this expression is not tractable. But for recommendation we are only interested in ordering items w.r.t.

Table 2. Recommendation performance of OrdNMF using the MovieLens dataset. Bold: best NDCG score. R: raw data. B: binarized data.

			NDCG @100 with threshold s				
Model	Data	K	$s=1$	$s=4$	$s=6$	$s=8$	$s=10$
OrdNMF	R	150	**0.444**	**0.444**	**0.439**	**0.414**	0.353
BePoF	B (≥ 1)	50	0.433	0.430	0.421	0.383	0.310
PF	B (≥ 1)	100	0.431	0.428	0.418	0.380	0.306
BePoF	B (≥ 8)	50	0.389	0.393	0.399	0.408	**0.369**
PF	B (≥ 8)	150	0.386	0.389	0.395	0.403	0.365

Table 3. Recommendation performance of OrdNMF using the Taste Profile dataset. Bold: best NDCG or log-likelihood score. Q: quantized data. B: binarized data. R: raw data.

			NDCG @100 with threshold s						
Model	Data	K	$s=1$	$s=3$	$s=6$	$s=11$	$s=21$	$s=51$	log-lik
OrdNMF	Q	250	**0.213**	**0.174**	0.153	0.135	0.123	0.117	$\mathbf{-2.8 \cdot 10^5}$
dcPF	R	150	0.209	0.173	**0.154**	**0.137**	**0.128**	**0.121**	$-3.0 \cdot 10^5$
BePoF	B (≥ 1)	250	0.210	0.170	0.149	0.131	0.120	0.115	N/A
PF	B (≥ 1)	250	0.206	0.167	0.146	0.129	0.118	0.115	N/A

this quantity. The function $\lambda_{ui} \mapsto \mathbb{E}(y^*_{ui}|\lambda_{ui})$ being increasing, we can use instead of Eq. (37) the simpler score $s_{ui} = [\mathbb{E}_q(\mathbf{W})\mathbb{E}_q(\mathbf{H})^T]_{ui}$.

5. Experimental Results

5.1. Experimental Set Up

Datasets. We report experimental results for two datasets described below.

• MovieLens (Harper & Konstan, 2015). This dataset contains the ratings of users on movies on a scale from 1 to 10. These explicit feedbacks correspond to ordinal data. We consider that the class 0 corresponds to the absence of a rating for a couple user-movie. The histogram of the ordinal data is represented in blue on Figure 4. We pre-process a subset of the data as in (Liang et al., 2016), keeping only users and movies that have more than 20 interactions. We obtain $U = 20$k users and $I = 12$k movies.

• Taste Profile (Bertin-Mahieux et al., 2011). This dataset, provided by the Echo Nest, contains the play counts of users on a catalog of songs. As mentioned in the introduction, we choose to quantize these counts on a predefined scale in order to obtain ordinal data. We arbitrarily select the following quantization thresholds: $[1, 2, 5, 10, 20, 50, 100, 200, 500]$. For example, the class labeled 6 corresponds to a listening counts between 21 and 50. As for MovieLens, the class 0 corresponds to users who have not listen to a song. The histogram of the ordinal data are displayed in blue on Figure 4. We pre-process a subset of the data as before and obtain $U = 16$k users and $I = 12$k songs.

Be careful not to confuse the predefined quantization used to obtain ordinal data, with the quantization of the latent variable in OrdNMF model which is estimated during inference. Although we expect OrdNMF to recover a relevant scaling between the categories, there is no reason to get the same quantization function that was used for pre-processing.

Evaluation. Each dataset is split into a train set $\mathbf{Y}^{\text{train}}$ and a test set $\mathbf{Y}^{\text{test}}$: the train set contains 80% of the non-zero values of the original dataset $\mathbf{Y}$, the other values are set to the class 0; the test set contains the remaining 20%. All the compared methods are trained on the train set and then evaluated on the test set.

First, we evaluate the recommendations with a ranking metric. For each user, we propose a list of $m = 100$ items (movies or songs) ordered w.r.t. the prediction score presented in Section 4.4: $s_{ui} = [\mathbb{E}_q(\mathbf{W})\mathbb{E}_q(\mathbf{H}^T)]_{ui}$. The quality of these lists is then measured through the NDCG metric (Järvelin & Kekäläinen, 2002). The NDCG rewards relevant items placed at the top of the list more strongly than those placed at the end. We use the relevance definition proposed in (Gouvert et al., 2019):

$$\text{rel}(u, i) = \mathbb{1}[y^{\text{test}}_{ui} \geq s]. \tag{38}$$

In other words, an item is considered as relevant if it belongs at least to the class s in the test set. The NDCG metric is between 0 and 1, the higher the better.

Moreover, for the Taste Profile dataset, we calculate the log-likelihood of the non-zero entries on the test set, as it is

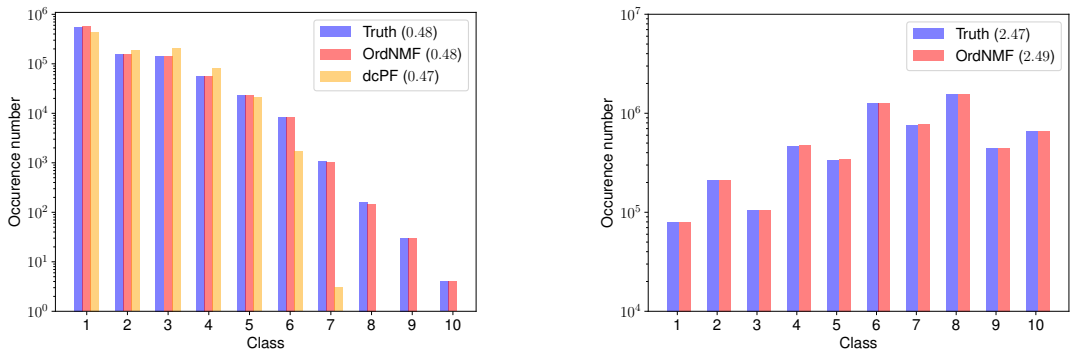

Figure 4. PPC of the distribution of the classes in the Taste Profile dataset (left) and MovieLens dataset (right). The blue bars (Truth) represents the histogram of the classes in the train set. The colored bars represent the simulated histograms obtained from the different inferred OrdNMF or dcPF models. The percentages of non-zero values are written in parentheses.

done in (Basbug & Engelhardt, 2016):

$$\mathcal{L}_{NZ} = \sum_{(u,i)\in \text{Testset}} \log p(y_{ui}^{\text{test}} | y_{ui}^{\text{test}} > 0, \hat{\mathbf{W}}, \hat{\mathbf{H}}), \qquad (39)$$

where y_{ui}^{test} is the quantized data, $\hat{w}_{uk} = \mathbb{E}[w_{uk}]$ and $\hat{h}_{ik} = \mathbb{E}[h_{ik}]$ are the estimated latent factors.

Compared methods. We compare OrdNMF with three other models: PF, BePoF and discrete compound PF (dcPF) (Basbug & Engelhardt, 2016) with a logarithmic element distribution as implemented in (Gouvert et al., 2019). Each model is applied either to raw data (R), quantized data (Q) or binarized data (B). For the MovieLens dataset, two different binarizations are tested: one with a threshold at 1 (≥ 1) and one with a threshold at 8 (≥ 8). For the Taste Profile dataset, dcPF is applied to the count data (R) whereas OrdNMF is applied to the quantized data (Q).

For all models, we select the shape hyperparameters $\alpha^W = \alpha^H = 0.3$ among $\{0.1, 0.3, 1\}$ (Gopalan et al., 2015). The number of latent factors is chosen among $K \in \{25, 50, 100, 150, 200, 350\}$ for the best NDCG score with threshold $s = 8$ for the MovieLens dataset, and $s = 1$ for the Taste Profile dataset. All the algorithms are run 5 times with random initializations and are stopped when the relative increment of the expected lower bound (ELBO) falls under $\tau = 10^{-5}$. The computer used for these experiments was a MacBook Pro with an Intel Core i5 processor (2,9 GHz) and 16 Go RAM. All the Python codes are available on `https://github.com/Oligou/OrdNMF`.

5.2. Prediction Results

Table 2 displays the results for the MovieLens dataset. First, we can compare BePoF with its approximation, i.e., PF applied to binarized data. BePoF is slightly better than PF for both binarizations, and requires less latent factors. Then, we observe that the choice of the binarization has a big impact on the NDCG scores. BePoF with data thresholded at 1 (≥ 1) perform well on small NDCG threshold s but has poor performance after. On the contray, with data thresholded at 8 (≥ 8), BePoF achieves best performances for NDCG $s = 10$ but poor performances for small s. OrdNMF does not exhibit such differences between NDCG scores and benefits from the additional information brought by the ordinal classes. Nevertheless, we can note a small decrease of the performance with $s = 10$ which is the hardest class to predict.

Table 3 displays the same kind of results for the Taste Profile dataset. Again, OrdNMF outperforms BePoF and PF which exploit less data information. OrdNMF is competitive with dcPF which gives the best results for the highest thresholds s. However, OrdNMF presents a higher log-likelihood score than dcPF. Thus, OrdNMF seems better suited to predict the feedback class of a user than dcPF. This observation is confirmed by the posterior predictive checks (PPC) presented below.

5.3. Posterior Predictive Check (PPC)

A PPC consists of generating new data based on the posterior predictive distribution $p(\mathbf{Y}^*, \mathbf{W}, \mathbf{H}|\mathbf{Y}) \approx p(\mathbf{Y}^*|\mathbf{W}, \mathbf{H})q(\mathbf{W})q(\mathbf{H})$, and then compare the structure of the original data $\mathbf{Y}$ with the artificial data $\mathbf{Y}^*$. Here, we focus on the distribution of the ordinal categories. Figure 4 presents the results of these PPCs for the Taste Profile dataset. The blue bars correspond to the empirical histogram of the data ($\mathbf{Y}^{\text{train}}$), the red and orange bars correspond to the histograms of the simulated data obtained with OrdNMF and dcPF respectively. While dcPF fails to model the very large values present in the data (from the class 7, which corresponds to values greater than 50 plays), OrdNMF seems to precisely describe all the ordinal categories. This is also the case on the MovieLens dataset too. Even if the empirical histogram is here less regular, OrdNMF can adapt itself to

all type of data through the inferred thresholds $\mathbf{b}$.

6. Conclusion

We developed a new probabilistic NMF framework to process ordinal data. In particular, we presented IG-OrdNMF which is an extension of BePoF and conducted experiments on two different datasets. We show the ability of OrdNMF to process different kinds of ordinal data both explicit and implicit. This work opens up several exciting perspectives. As we described in Section 3.1, OrdNMF can be used for different choices of multiplicative noise. It would be interesting to develop OrdNMF for the exponential noise in a similar way than IG-OrdNMF. Finally, when applied to implicit data, it would be of particular interest to learn the pre-processing during the factorization, in order to automatically tune the level of pre-processing adapted to a given dataset. This is yet left for future investigations.

Acknowledgements

Supported by the European Research Council (ERC FACTORY-CoG-6681839) and the ANR-3IA (ANITI).

References

Acharya, A., Ghosh, J., and Zhou, M. Nonparametric Bayesian factor analysis for dynamic count matrices. In *Proc. International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

Agresti, A. and Kateri, M. *Categorical data analysis*. Springer, 2011.

Ananth, C. V. and Kleinbaum, D. G. Regression models for ordinal responses: A review of methods and applications. *International journal of epidemiology*, pp. 1323–1333, 1997.

Basbug, M. E. and Engelhardt, B. E. Hierarchical compound Poisson factorization. In *Proc. International Conference on Machine Learning (ICML)*, 2016.

Bertin-Mahieux, T., Ellis, D. P., Whitman, B., and Lamere, P. The million song dataset. In *Proc. International Society for Music Information Retrieval (ISMIR)*, pp. 10, 2011.

Canny, J. GaP: A factor model for discrete data. In *Proc. ACM International on Research and Development in Information Retrieval (SIGIR)*, pp. 122–129, 2004.

Cemgil, A. T. Bayesian inference for nonnegative matrix factorisation models. *Computational Intelligence and Neuroscience*, 2009.

Chu, W. and Ghahramani, Z. Gaussian processes for ordinal regression. *The Journal of Machine Learning Research*, pp. 1019–1041, 2005.

Consonni, G. and Marin, J.-M. Mean-field variational approximate Bayesian inference for latent variable models. *Computational Statistics & Data Analysis*, pp. 790–798, 2007.

Févotte, C. and Idier, J. Algorithms for nonnegative matrix factorization with the beta-divergence. *Neural computation*, pp. 2421–2456, 2011.

Gopalan, P., Hofman, J. M., and Blei, D. M. Scalable recommendation with hierarchical Poisson factorization. In *Proc. Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 326–335, 2015.

Gouvert, O., Oberlin, T., and Févotte, C. Recommendation from raw data with adaptive compound Poisson factorization. In *Proc. Conference on Uncertainty in Artificial Intelligence (UAI)*, 2019.

Gutierrez, P. A., Perez-Ortiz, M., Sanchez-Monedero, J., Fernandez-Navarro, F., and Hervas-Martinez, C. Ordinal regression methods: Survey and experimental study. *IEEE Transactions on Knowledge and Data Engineering*, pp. 127–146, 2015.

Harper, F. M. and Konstan, J. A. The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems (TIIS)*, pp. 1–19, 2015.

Hernandez-Lobato, J. M., Houlsby, N., and Ghahramani, Z. Probabilistic matrix factorization with non-random missing data. In *Proc. International Conference on Machine Learning (ICML)*, pp. 1512–1520, 2014.

Hu, Y., Koren, Y., and Volinsky, C. Collaborative filtering for implicit feedback datasets. In *Proc. IEEE International Conference on Data Mining (ICDM)*, pp. 263–272, 2008.

Järvelin, K. and Kekäläinen, J. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)*, pp. 422–446, 2002.

Koren, Y. and Sill, J. OrdRec: An ordinal model for predicting personalized item rating distributions. In *Proc. ACM Conference on Recommender Systems (RecSys)*, pp. 117–124, 2011.

Koren, Y., Bell, R., and Volinsky, C. Matrix factorization techniques for recommender systems. *Computer*, pp. 30–37, 2009.

Lee, D. D. and Seung, H. S. Learning the parts of objects by non-negative matrix factorization. *Nature*, pp. 788–791, 1999.

Lee, D. D. and Seung, H. S. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 556–562, 2001.

Liang, D., Charlin, L., McInerney, J., and Blei, D. M. Modeling user exposure in recommendation. In *Proc. International Conference on World Wide Web (WWW)*, pp. 951–961, 2016.

Lumbreras, A., Filstroff, L., and Févotte, C. Bayesian mean-parameterized nonnegative binary matrix factorization. *arXiv preprint arXiv:1812.06866*, 2018.

Ma, H., Liu, C., King, I., and Lyu, M. R. Probabilistic factor models for web site recommendation. In *Proc. ACM International on Research and Development in Information Retrieval (SIGIR)*, pp. 265–274, 2011.

McCullagh, P. Regression models for ordinal data. *Journal of the Royal Statistical Society: Series B (Methodological)*, (2):109–127, 1980.

Paquet, U., Thomson, B., and Winther, O. A hierarchical model for ordinal matrix factorization. *Statistics and Computing*, pp. 945–957, 2012.

Rendle, S., Freudenthaler, C., Gantner, Z., and Schmidt-Thieme, L. Bpr: Bayesian personalized ranking from implicit feedback. In *Proc. Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 452–461, 2009.

Stevens, S. S. On the theory of scales of measurement. 1946.

Verwaeren, J., Waegeman, W., and De Baets, B. Learning partial ordinal class memberships with kernel-based proportional odds models. *Computational Statistics & Data Analysis*, pp. 928–942, 2012.

Walker, S. H. and Duncan, D. B. Estimation of the probability of an event as a function of several independent variables. *Biometrika*, pp. 167–179, 1967.

Zhou, M. Nonparametric Bayesian negative binomial factor analysis. *Bayesian Analysis*, 2017.