# Cumulative Restricted Boltzmann Machines for Ordinal Matrix Data Analysis

**Truyen Tran**[†‡]                                                    TRUYEN.TRAN@DEAKIN.EDU.AU
**Dinh Phung**[†]                                                      DINH.PHUNG@DEAKIN.EDU.AU
**Svetha Venkatesh**[†]                                          SVETHA.VENKATESH@DEAKIN.EDU.AU
[†]*Pattern Recognition and Data Analytics, Deakin University, Waurn Ponds, Vic 3216, Australia.*
[‡]*Department of Computing, Curtin University, Bentley, WA 6102, Australia*

**Editor:** Steven C.H. Hoi and Wray Buntine

## Abstract

Ordinal data is omnipresent in almost all multiuser-generated feedback - questionnaires, preferences etc. This paper investigates modelling of ordinal data with Gaussian restricted Boltzmann machines (RBMs). In particular, we present the model architecture, learning and inference procedures for both vector-variate and matrix-variate ordinal data. We show that our model is able to capture latent opinion profile of citizens around the world, and is competitive against state-of-art collaborative filtering techniques on large-scale public datasets. The model thus has the potential to extend application of RBMs to diverse domains such as recommendation systems, product reviews and expert assessments.

**Keywords:** Cumulative restricted Boltzmann machine, ordinal analysis, matrix data

## 1. Introduction

Restricted Boltzmann machines (RBMs) (Smolensky, 1986; Freund and Haussler, 1993; Le Roux and Bengio, 2008) have recently attracted significant interest due to their versatility in a variety of unsupervised and supervised learning tasks (Salakhutdinov et al., 2007; Larochelle and Bengio, 2008; Mohamed and Hinton, 2010), and in building deep architectures (Hinton and Salakhutdinov, 2006; Salakhutdinov and Hinton, 2009a). A RBM is a bipartite undirected model that captures the generative process in which a data vector is generated from a binary hidden vector. The bipartite architecture enables very fast data encoding and sampling-based inference; and together with recent advances in learning procedures, we can now process massive data with large models (Hinton, 2002; Tieleman, 2008; Benjamin Marlin and de Freitas., 2010).

This paper presents our contributions in developing RBM specifications as well as learning and inference procedures for multivariate ordinal data. This extends and consolidates the reach of RBMs to a wide range of user-generated domains - social responses, recommender systems, product/paper reviews, and expert assessments of health and ecosystems indicators. Ordinal variables are qualitative in nature – the absolute numerical assignments are not important but the relative order is. This renders numerical transforms and real-valued treatments inadequate. Current RBM-based treatments, on the other hand, ignore the ordinal nature and treat data as unordered categories (Salakhutdinov et al., 2007; Truyen et al., 2009). While convenient, this has several drawbacks: First, order information is not utilised, leading to more parameters than necessary - each category needs

parameters. Second, since categories are considered independently, it is less interpretable in terms of how ordinal levels are generated. Better modelling should account for the ordinal generation process.

Adapting the classic idea from (McCullagh, 1980), we assume that each ordinal variable is generated by an underlying latent utility, along with a threshold per ordinal level. As soon as the utility passes the threshold, its corresponding level is selected. As a result, this process would implicitly encode the order. Our main contribution here is a novel RBM architecture that accounts for multivariate, ordinal data. More specifically, we further assume that the latent utilities are Gaussian variables connected to a set of binary hidden factors (i.e., together they form a Gaussian RBM (Hinton and Salakhutdinov, 2006)). This offers many advantages over the standard approach that imposes a fully connected Gaussian random field over utilities (Kottas et al., 2005; Jeliazkov et al., 2008): First, utilities are seen as being generated from a set of binary factors, which in many cases represent the user's hidden profile. Second, utilities are decoupled given the hidden factors, making parallel sampling easier. And third, the posteriors of binary factors can be estimated from the ordinal observations, facilitating dimensionality reduction and visualisation. We term our model Cumulative RBM (CRBM)[1].

This new model behaves differently from standard Gaussian RBMs since utilities are never observed in full. Rather, when an ordinal level of an input variable is observed, it poses an *interval constraint* over the corresponding utility. The distribution over the utilities now becomes a *truncated* multivariate Gaussian. This also has another consequence during learning: While in standard RBMs we need to sample for the *free-phase* only (e.g., see (Hinton, 2002)), now we also need to sample for the *clamped-phase*. As a result, we introduce a double persistent contrastive divergence (PCD) learning procedure, as opposed to the single PCD in (Tieleman, 2008).

The second contribution is in advancing these ordinal RBMs from modelling i.i.d. vectors to modelling matrices of correlated entries. These ordinal matrices are popular in multiuser-generated assessments: Each user would typically judge a number of items producing a user-specific data vector where *intra-vector* entries are inherently correlated. Since user's choices are influenced by their peers, these *inter-vector* entries are no longer independent. The idea is borrowed from a recent work in (Truyen et al., 2009) which models both the user-specific and item-specific processes. More specifically, an ordinal entry is assumed to be jointly generated from user-specific latent factors and item-specific latent factors. This departs significantly from the standard RBM architecture: we no longer map from a visible vector to an hidden vector but rather map from a visible matrix to two hidden matrices.

In experiments, we demonstrate that our proposed CRBM is capable of capturing the latent profile of citizens around the world. Our model is also competitive against state-of-the-art collaborative filtering methods on large-scale public datasets.

We start with the RBM structure for ordinal vectors in Section 2, and end with the general structure for ordinal matrices in Section 3. Section 4 presents experiments validating our ordinal RBMs in modelling citizen's opinions worldwide and in collaborative filtering. Section 5 discusses related work, which is then followed by the conclusions.

---

1. The term 'cumulative' is to be consistent with the statistical literature when referring to the ordinal treatment in (McCullagh, 1980).
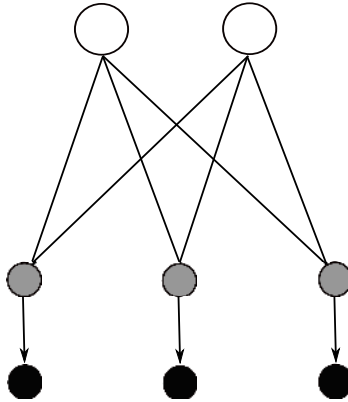
Figure 1: Model architecture of the Cumulative Restricted Boltzmann Machine (CRBM). Filled nodes represent observed ordinal variables, shaded nodes are Gaussian utilities, and empty nodes represent binary hidden factors.

## 2. Cumulative RBM for Vectorial Data

### 2.1. Model Definition

Denote by $\boldsymbol{v} = (v_1, v_2, ..., v_N)$ the set of ordinal observations. For ease of presentation we assume for the moment that observations are homogeneous, i.e., observations are drawn from the same discrete ordered category set $S = \{c_1 \prec c_2 \prec ..., \prec c_L\}$ where $\prec$ denotes the order in some sense. We further assume that each ordinal $v_i$ is solely generated from an underlying latent *utility* $u_i \in \mathbb{R}$ as follows (McCullagh, 1980)

$$P(v_i = c_l \mid u_i) = \begin{cases} \mathbb{I}\left[-\infty < u_i \le \theta_{i1}\right] & l = 1 \\ \mathbb{I}\left[\theta_{i(l-1)} < u_i \le \theta_{il}\right] & 1 < l \le L - 1 \\ \mathbb{I}\left[\theta_{i(L-1)} < u_i < \infty\right] & l = L \end{cases} \tag{1}$$

where $\theta_{i1} < \theta_{i2} < ... < \theta_{i(L-1)}$ are threshold parameters. In words, we choose an ordered category on the basis of the interval to which the underlying utility belongs.

The utilities are undirectionally connected with a set of hidden *binary factors* $\boldsymbol{h} = (h_1, h_2, ..., h_K) \in \{0, 1\}^K$ so that the two layers of $(\boldsymbol{u}, \boldsymbol{h})$ form a bipartite graph of Restricted Boltzmann Machines (RBMs) (Smolensky, 1986; Freund and Haussler, 1993; Hinton and Salakhutdinov, 2006). Binary factors can be considered as the hidden features that govern the generation of the observed ordinal data. Thus the generative story is: we start from the binary factors to generate utilities, which, in turn, generate ordinal observations. See, for example, Fig. 1 for a graphical representation of the model.

Let $\Psi(\boldsymbol{u}, \boldsymbol{h}) \ge 0$ be the model potential function, which can be factorised as a result of the bipartite structure as follows

$$\Psi(\boldsymbol{u}, \boldsymbol{h}) = \left[\prod_i \phi_i(u_i)\right] \left[\prod_{i,k} \psi_{ik}(u_i, h_k)\right] \left[\prod_k \phi_k(h_k)\right]$$

where $\phi_i, \psi_{ik}$ and $\phi_k$ are local potential functions. The model joint distribution is defined as

$$P(\boldsymbol{v}, \boldsymbol{u}, \boldsymbol{h}) \; = \frac{1}{Z} \Psi(\boldsymbol{u}, \boldsymbol{h}) \prod_i P(v_i \mid u_i) \tag{2}$$

where $Z = \int_{\boldsymbol{u}} \sum_{\boldsymbol{h}} \Psi(\boldsymbol{u}, \boldsymbol{h}) d\boldsymbol{u}$ is the normalising constant.

We assume the utility layer and the binary factor layer form a Gaussian RBM[2] (Hinton and Salakhutdinov, 2006). This translates into the local potential functions as follows

$$\phi_i(u_i) = \exp\left\{-\frac{u_i^2}{2\sigma_i^2} + \alpha_i u_i\right\}; \quad \psi_{ik}(u_i, h_k) = \exp\left\{w_{ik} u_i h_k\right\}; \quad \phi_k(h_k) = \exp\left\{\gamma_k h_k\right\} \tag{3}$$

where $\sigma_i$ is the standard deviation of the $i$-th utility, $\{\alpha_i, \gamma_k, w_{ik}\}$ are free parameters for $i = 1, 2, .., N$ and $k = 1, 2, .., K$.

The ordinal assumption in Eq. (1) introduces *hard constraints* that we do not see in standard Gaussian RBMs. Whenever an ordered category $v_i$ is observed, the corresponding utility is automatically *truncated*, i.e., $u_i \in \Omega(v_i)$, where $\Omega(v_i)$ is the new domain of $u_i$ defined by $v_i$ as in Eq. (1). In particular, the utility is truncated from above if the ordinal level is the lowest, from below if the level is the largest, and from both sides otherwise. For example, the conditional distribution of the latent utility $P(u_i \mid v_i, \boldsymbol{h})$ is a truncated Gaussian

$$P(u_i \mid v_i, \boldsymbol{h}) \quad \propto \quad \mathbb{I}\left[u_i \in \Omega(v_i)\right] \mathcal{N}\left(u_i; \mu_i(\boldsymbol{h}), \sigma_i\right) \tag{4}$$

where $\mathcal{N}(u_i; \mu_i(\boldsymbol{h}), \sigma_i)$ is the normal density distribution of mean $\mu_i(\boldsymbol{h})$ and standard deviation $\sigma_i$. The mean $\mu_i(\boldsymbol{h})$ is computed as

$$\mu_i(\boldsymbol{h}) = \sigma_i^2 \left(\alpha_i + \sum_{k=1}^{K} w_{ik} h_k\right) \tag{5}$$

As a generative model, we can estimate the probability that an ordinal level is being generated from hidden factors $\boldsymbol{h}$ as follows

$$P(v_i = c_l \mid \boldsymbol{h}) \quad = \quad \int_{u_i \in \Omega(c_l)} P(u_i | \boldsymbol{h}) = \begin{cases} \Phi\left(\theta_1^*\right) & l = 1 \\ \Phi\left(\theta_l^*\right) - \Phi\left(\theta_{(l-1)}^*\right) & 1 < l \le L-1 \\ 1 - \Phi(\theta_{L-1}^*) & l = L \end{cases} \tag{6}$$

where $\theta_l^* = \frac{\theta_l - \mu_i(\boldsymbol{h})}{\sigma_i}$, and $\Phi(\cdot)$ is the cumulative distribution function of the Gaussian. Given this property, we term our model by Cumulative Restricted Boltzmann Machine (CRBM).

Finally, the thresholds are parameterised so that the lowest threshold is fixed to a constant $\theta_{i1} = \tau_{i1}$ and the higher thresholds are spaced as $\theta_{il} = \theta_{i(l-1)} + e^{\tau_{il}}$ with free parameter $\tau_{il}$ for $l = 2, 3, .., L-1$.

---

2. This is for convenience only. In fact, we can replace Gaussian by any continuous distribution in the exponential family.

## 2.2. Factor Posteriors

Often we are interested in the posterior of factors $\{P(h_k \mid \boldsymbol{v})\}_{k=1}^K$ as it can be considered as a summary of the data $\boldsymbol{v}$. The nice thing is that it is now numerical and can be used for other tasks such as clustering, visualisation and prediction.

Like standard RBMs, the factor posteriors given the utilities are conditionally independent and assume the form of logistic units

$$P(h_k = 1 \mid \boldsymbol{u}) = \frac{1}{1 + \exp\left(-\gamma_k - \sum_i w_{ik} u_i\right)} \tag{7}$$

However, since the utilities are themselves hidden, the posteriors given only the ordinal observations are not independent:

$$P(h_k \mid \boldsymbol{v}) = \sum_{\boldsymbol{h}_{\neg k}} \int_{\boldsymbol{u} \in \boldsymbol{\Omega}(\boldsymbol{v})} P(\boldsymbol{h}, \boldsymbol{u} | \boldsymbol{v}) d\boldsymbol{u} \tag{8}$$

where $\boldsymbol{h}_{\neg k} = \boldsymbol{h} \backslash h_k$ and $\boldsymbol{\Omega}(\boldsymbol{v}) = \Omega(v_1) \times \Omega(v_2) \times ... \Omega(v_N)$ is the domain of the utility constrained by $\boldsymbol{v}$ (see Eq. (1)). Here we describe two approximation methods, namely Markov chain Monte Carlo (MCMC) and variational method (mean-field).

**MCMC.** We can exploit the bipartite structure of the RBM to run layer-wise Gibbs sampling: sample the truncated utilities in parallel using Eq. (4) and the binary factors using Eq. (7). Finally, the posteriors are estimated as $P(h_k \mid \boldsymbol{v}) \approx \frac{1}{n} \sum_{s=1}^n h_k^{(s)}$ for $n$ samples.

**Variational method.** We make the approximation

$$P(\boldsymbol{h}, \boldsymbol{u} \mid \boldsymbol{v}) \approx \prod_k Q_k(h_k) \prod_i Q_i(u_i)$$

Minimising the Kullback-Leibler divergence between $P(\boldsymbol{h}, \boldsymbol{u} \mid \boldsymbol{v})$ and its approximation leads the following recursive update

$$\begin{aligned} Q_k\left(h_k^{(t+1)} = 1\right) &\leftarrow \frac{1}{1 + \exp\left(-\gamma_k - \sum_i w_{ik} \langle u_i \rangle_{Q_i^{(t)}}\right)} \\ Q_i\left(u_i^{(t+1)}\right) &\leftarrow \frac{1}{\kappa_i^{(t)}} \mathbb{I}\left[u_i \in \Omega(v_i)\right] \mathcal{N}\left(u_i; \hat{\mu}_i(\boldsymbol{h}^{(t)}), \sigma_i\right) \end{aligned} \tag{9}$$

where $t$ is the update index of the recursion, $\langle u_i \rangle_{Q_i^{(t)}}$ is the mean of utility $u_i$ with respect to $Q_i\left(u_i^{(t)}\right)$, $\kappa_i^{(t)} = \int_{u_i \in \Omega(v_i)} \mathcal{N}\left(u_i; \mu_i(\boldsymbol{h}^{(t)}), \sigma_i\right)$ is the normalising constant, and $\hat{\mu}_i(\boldsymbol{h}^{(t)}) = \sigma_i^2\left(\alpha_i + \sum_{k=1}^K w_{ik} Q_k\left(h_k^{(t)} = 1\right)\right)$. Finally, we obtain $P(h_k \mid \boldsymbol{v}) \approx Q_k(h_k = 1)$.

## 2.3. Prediction

An important task is *prediction* of the ordinal level of an unseen variable given the other seen variables, where we need to estimate the following predictive distribution

$$P(v_j \mid \boldsymbol{v}) = \sum_{\boldsymbol{h}} \int_{u_j \in \Omega(v_j)} \int_{\boldsymbol{u} \in \boldsymbol{\Omega}(\boldsymbol{v})} P(\boldsymbol{h}, u_j, \boldsymbol{u} \mid \boldsymbol{v}) d\boldsymbol{u} du_j \tag{10}$$

Unfortunately, now $(h_1, h_2, ..., h_K)$ are coupled due to the integration over $\{u_j, \boldsymbol{u}\}$ making the evaluation intractable, and thus approximation is needed.

For simplicity, we assume that the seen data $\boldsymbol{v}$ is informative enough so that $P(\boldsymbol{h}|v_j, \boldsymbol{v}) \approx P(\boldsymbol{h}|\boldsymbol{v})$. Thus we can rewrite Eq. (10) as

$$P(v_j|\boldsymbol{v}) \approx \sum_{\boldsymbol{h}} P(\boldsymbol{h}|\boldsymbol{v})P(v_j|\boldsymbol{h})du_j$$

Now we make further approximations to deal with the exponential sum over $\boldsymbol{h}$.

**MCMC.** Given the sampling from $P(\boldsymbol{h}|\boldsymbol{v})$ described in Section 2.2, we obtain

$$P(v_j|\boldsymbol{v}) \approx \frac{1}{n} \sum_{s=1}^{n} P(v_j|\boldsymbol{h}^{(s)})du_j$$

where $n$ is the sample size, and $P(v_j|\boldsymbol{h}^{(s)})$ is computed using Eq. (6).

**Variational method.** The idea is similar to mean-field described in Section 2.2. In particular, we estimate $\hat{h}_k = P(h_k = 1|\boldsymbol{v})$ using either MCMC sampling or mean-field update. The predictive distribution is approximated as

$$P(v_j|\boldsymbol{v}) \approx \int_{u_i \in \Omega(v_j)} P(u_i \mid \hat{h}_1, \hat{h}_2, ..., \hat{h}_K)$$

where $P(u_i \mid \hat{h}_1, \hat{h}_2, ..., \hat{h}_K) = \mathcal{N}\left(u_i; \sigma_i^2 \left(\alpha_i + \sum_{k=1}^{K} w_{ik}\hat{h}_k\right), \sigma_i\right)$. The computation is identical to that of Eq. (6) if we replace $h_k$ (binary) by $\hat{h}_k$ (real-valued) .

### 2.4. Stochastic Gradient Learning with Persistent Markov Chains

Learning is based on maximising the data log-likelihood

$$\begin{aligned}
\mathcal{L} &= \log P(\boldsymbol{v}) = \log \sum_{\boldsymbol{h}} \int_{\boldsymbol{u}} P(\boldsymbol{v}, \boldsymbol{u}, \boldsymbol{h})d\boldsymbol{u} \\
&= \log Z(\boldsymbol{v}) - \log Z
\end{aligned}$$

where $P(\boldsymbol{v}, \boldsymbol{u}, \boldsymbol{h})$ is defined in Eq. (2) and $Z(\boldsymbol{v}) = \sum_{\boldsymbol{h}} \int_{\boldsymbol{u} \in \Omega(\boldsymbol{v})} \Psi(\boldsymbol{u}, \boldsymbol{h})d\boldsymbol{u}$. Note that $Z(\boldsymbol{v})$ includes $Z$ as a special case when the domain $\boldsymbol{\Omega}(\boldsymbol{v})$ is the whole real space $\mathbb{R}^N$.

Recall that the model belongs to the exponential family in that we can rewrite the potential function as

$$\Psi(\boldsymbol{u}, \boldsymbol{h}) = \exp\left\{\sum_a W_a f_a(\boldsymbol{u}, \boldsymbol{h})\right\}$$

where $f_a(\boldsymbol{u}, \boldsymbol{h}) \in \{u_i, u_i h_k, h_k\}_{(i,k)=(1,1)}^{(N,K)}$ is a sufficient statistic, and $W_a \in \{\alpha_i, \gamma_k, w_{ik}\}_{(i,k)=(1,1)}^{(N,K)}$ is its associated parameter. Now the gradient of the log-likelihood has the standard form of difference of expected sufficient statistics (ESS)

$$\partial_{W_a}\mathcal{L} = \langle f_a \rangle_{P(\boldsymbol{u},\boldsymbol{h}|\boldsymbol{v})} - \langle f_a \rangle_{P(\boldsymbol{u},\boldsymbol{h})}$$

where $P(\boldsymbol{u}, \boldsymbol{h} \mid \boldsymbol{v})$ is a truncated Gaussian RBM and $P(\boldsymbol{u}, \boldsymbol{h})$ is the standard Gaussian RBM.

Put in common RBM-terms, there are two learning phases: the *clamped phase* in which we estimate the ESS w.r.t. the empirical distribution $P(\boldsymbol{u}, \boldsymbol{h} \mid \boldsymbol{v})$, and the *free phase* in which we compute the ESS w.r.t. model distribution $P(\boldsymbol{u}, \boldsymbol{h})$.

### 2.4.1. Persistent Markov Chains

The literature offers efficient stochastic gradient procedures to learn parameters, in which the method of (Younes, 1989) and its variants – the Contrastive Divergence of (Hinton, 2002) and its persistent version of (Tieleman, 2008) – are highly effective in large-scale settings. The strategy is to update parameters after short Markov chains. Typically only the free phase requires the MCMC approximation. In our setting, on the other hand, both the clamped phase and the free phase require approximation.

Since it is possible to integrate over utilities when the binary factors are known, it is tempting so sample only the binary factors in the Rao-Blackwellisation fashion. However, here we take the advantage of the bipartite structure of the underlying RBM: the layer-wise sampling is efficient and much simpler. Once the hidden factor samples are obtained, we integrate over utilities for better numerical stability. The ESSes are the averaged over all factor samples.

For the clamped phase, we maintain one Markov chain per data instance. For memory efficiency, only the binary factor samples are stored between update steps. For the free phase, there are two strategies:

- *Contrastive chains*: one short chain is needed per data instance, but initialised from the clamped chain. That is, we discard those chains after each update.

- *Persistent chains*: free-phase chains are maintained during the course of learning, independent of the clamp-phase chains. If every data instance has the same dimensions (which they do not, in the case of missing data), we need to maintain a moderate number of chains (e.g., $20 - 100$). Otherwise, we need one chain per data instance.

At each step, we collect a small number of samples and estimate the approximate distributions $\tilde{P}(\boldsymbol{u}, \boldsymbol{h} \mid \boldsymbol{v})$ and $\tilde{P}(\boldsymbol{u}, \boldsymbol{h})$. The parameters are updated according to the stochastic gradient ascent rule

$$W_s \leftarrow W_s + \nu \left( \langle f_a \rangle_{\tilde{P}(\boldsymbol{u}, \boldsymbol{h} \mid \boldsymbol{v})} - \langle f_a \rangle_{\tilde{P}(\boldsymbol{u}, \boldsymbol{h})} \right)$$

where $\nu \in (0, 1)$ is the learning rate.

### 2.4.2. Learning Thresholds

Thresholds appear only in the computation of $Z(\boldsymbol{v})$ as they define the utility domain $\boldsymbol{\Omega}(\boldsymbol{v})$. Let $\bar{\boldsymbol{\Omega}}(\boldsymbol{v})^+$ be the upper boundary of $\boldsymbol{\Omega}(\boldsymbol{v})$, and $\bar{\boldsymbol{\Omega}}(\boldsymbol{v})^-$ the lower boundary. The gradient of the log-likelihood w.r.t. boundaries reads

$$
\begin{aligned}
\partial_{\bar{\boldsymbol{\Omega}}(\boldsymbol{v})^+} \mathcal{L} &= \frac{1}{Z(\boldsymbol{v})} \sum_{\boldsymbol{h}} \partial_{\bar{\boldsymbol{\Omega}}(\boldsymbol{v})^+} \int_{\boldsymbol{u} \in \boldsymbol{\Omega}(\boldsymbol{v})} \Psi(\boldsymbol{h}, \boldsymbol{u}) d\boldsymbol{u} = \sum_{\boldsymbol{h}} P(\boldsymbol{u} = \bar{\boldsymbol{\Omega}}(\boldsymbol{v})^+, \boldsymbol{h} \mid \boldsymbol{v}) \\
\partial_{\bar{\boldsymbol{\Omega}}(\boldsymbol{v})^-} \mathcal{L} &= -\sum_{\boldsymbol{h}} P(\boldsymbol{u} = \bar{\boldsymbol{\Omega}}(\boldsymbol{v})^-, \boldsymbol{h} \mid \boldsymbol{v})
\end{aligned}
$$

Recall from Section 2.1 that the boundaries $\bar{\Omega}(v_i = l)^-$ and $\bar{\Omega}(v_i = l)^+$ are the lower-threshold $\theta_{i(l-1)}$ and the upper-threshold $\theta_{il}$, respectively, where $\theta_{il} = \theta_{i(l-1)} + e^{\tau_{il}} = \tau_{i1} + \sum_{m=2}^{l} e^{\tau_{im}}$. Using the chain rule, we would derive the derivatives w.r.t. to $\{\tau_{im}\}_{m=2}^{L-1}$.

### 2.5. Handling Heterogeneous Data

We now consider the case where ordinal variables do not share the same ordinal scales, that is, we have a separate ordered set $S_i = \{c_{i1} \prec c_{i2} \prec ..., \prec c_{iL_i}\}$ for each variable $i$. This requires only slight change from the homogeneous case, e.g., by learning separate set of thresholds for each variable.

### 3. CRBM for Matrix Data

Often the data has the matrix form, i.e., a list of column vectors and we often assume columns as independent. However, this assumption is too strong in many applications. For example, in collaborative filtering where each user plays the role of a column, and each item the role of a row, a user's choice can be influenced by other users' choices (e.g., due to the popularity of a particular item), then columns are correlated. Second, it is also natural to switch the roles of the users and items and this clearly destroys the i.i.d assumption over the columns.

Thus, it is more precise to assume that an observation is *jointly* generated by both the row-wise and column-wise processes (Truyen et al., 2009). In particular, let $d$ be the index of the data instance, each observation $v_{di}$ is generated from an utility $u_{di}$. Each data instance (column) $d$ is represented by a vector of binary hidden factors $\boldsymbol{h}_d \in \{0,1\}^K$ and each item (row) $i$ is represented by a vector of binary hidden factors $\boldsymbol{g}_i \in \{0,1\}^S$. Since our data matrix is usually incomplete, let us denote by $W \in \{0,1\}^{D \times N}$ the incidence matrix where $W_{di} = 1$ if the cell $(d, i)$ is observed, and $W_{di} = 0$ otherwise. There is a single model for the whole incomplete data matrix. Every observed entry $(d, i)$ is connected with two sets of hidden factors $\boldsymbol{h}_d$ and $\boldsymbol{g}_i$. Consequently, there are $DK + NS$ binary factor units in the entire model.

Let $\boldsymbol{H} = \left( \{u_{di}\}_{W_{di}=1}, \{\boldsymbol{h}_d\}_{d=1}^{D}, \{\boldsymbol{g}_i\}_{i=1}^{N} \right)$ denote all latent variables and $\boldsymbol{V} = \{v_{di}\}_{W_{di}=1}$ all visible ordinal variables. The matrix-variate model distribution has the usual form

$$P(\boldsymbol{V}, \boldsymbol{H}) = \frac{1}{Z^*} \Psi^*(\boldsymbol{H}) \prod_{d,i|W_{di}=1} P(v_{di} \mid u_{di})$$

where $Z^*$ is the normalising constant and $\Psi^*(\boldsymbol{H})$ is the product of all local potentials. More specifically,

$$\Psi^*(\boldsymbol{H}) = \prod_{d,i|W_{di}=1} \left( \phi_{di}(u_{di}) \prod_k \psi_{ik}(u_{di}, h_{dk}) \prod_s \varphi_{is}(u_{di}, g_{is}) \right) \left[ \prod_{d,k} \phi_k(h_{dk}) \right] \left[ \prod_{i,s} \phi_s(g_{is}) \right]$$

where $\psi_{ik}(u_{di}, h_k), \phi_k(h_{dk})$ are the same as those defined in Eq. (3), respectively, and

$$\phi_{di}(u_{di}) = \exp \left\{ -\frac{u_{di}^2}{2\sigma_{di}^2} + (\alpha_i + \beta_d)u_i \right\}; \quad \varphi_{ds}(u_{di}, g_s) = \exp \{\omega_{ds} u_{di} g_s\}; \quad \phi_s(g_{is}) = \exp \{\xi_s g_{is}\}$$

The ordinal model $P(v_{di} \mid u_{di})$ is similar to that defined in Eq. (1) except for the thresholds, which are now functions of both the data instance and the item, that is $\theta_{di1} = \tau_{i1} + \kappa_{d1}$ and $\theta_{dil} = \theta_{di(l-1)} + e^{\tau_{il} + \kappa_{dl}}$ for $l = 2, 3, .., L-1$.

### 3.1. Model Properties

It is easy to see that conditioned on the utilities, the posteriors of the binary factors are still factorisable. Likewise, given the factors, the utilities are univariate Gaussian

$$
\begin{aligned}
P(u_{di} \mid \boldsymbol{h}_d, \boldsymbol{g}_i) &= \mathcal{N}\left(\mu_{di}^*(\boldsymbol{h}_d, \boldsymbol{g}_i), \sigma_{di}^2\right) \\
P(u_{di} \mid \boldsymbol{h}_d, \boldsymbol{g}_i, v_{di}) &\propto \mathbb{I}\left[u_{di} \in \Omega(v_{di})\right] P(u_{di} \mid \boldsymbol{h}_d, \boldsymbol{g}_i)
\end{aligned}
$$

where $\Omega(v_{di})$ is the domain defined by the thresholds at the level $l = v_{di}$, and the mean structure is

$$
\mu_{di}^*(\boldsymbol{h}_d, \boldsymbol{g}_i) = \sigma_{di}^2 \left( \alpha_i + \beta_d + \sum_{k=1}^{K} w_{ik} h_{dk} + \sum_{s=1}^{S} \omega_{ds} g_{is} \right) \tag{11}
$$

Previous inference tricks can be re-used by noting that for each column (i.e., data instance), we still enjoy the Gaussian RBM when conditioned on other columns. The same holds for rows (i.e., items).

### 3.2. Stochastic Learning with Structured Mean-Fields

Although it is possible to explore the space of the whole model using Gibbs sampling and use the short MCMC chains as before, here we resort to structured mean-field methods to exploit the modularity in the model structure. The general idea is to alternate between the column-wise and the row-wise conditional processes:

- In the *column-wise* process, we estimate item-specific factor posteriors $\{\hat{\boldsymbol{g}}_i\}_{i=1}^N$, where $\hat{g}_{is} \leftarrow P\left(g_{is} = 1 \mid (v_{di})_{di \mid W_{id}=1}\right)$ and use them *as if* the item-specific factors $(\boldsymbol{g}_i)_{i=1}^N$ are given. For example, the mean structure in Eq. (11) now has the following form

$$
\mu_{di}^*(\boldsymbol{h}_d, \hat{\boldsymbol{g}}_i) = \sigma_{di}^2 \left( \alpha_i + \beta_d + \sum_{k=1}^{K} w_{ik} h_{dk} + \sum_{s=1}^{S} \omega_{ds} \hat{g}_{is} \right)
$$

  which is essentially the mean structure in Eq. (5) when $\beta_d + \sum_{s=1}^{S} \omega_{ds} \hat{g}_{is}$ is absorbed into $\alpha_i$. Conditioned on the estimated posteriors, the data likelihood is now factorisable $\prod_d P\left(\boldsymbol{v}_{d\bullet} \mid \{\hat{\boldsymbol{g}}_i\}_{i=1}^N\right)$, where $\boldsymbol{v}_{d\bullet}$ denotes the observations of the $d$-th data instance.

- Similarly, in the *row-wise* process we estimate data-specific posteriors $\left\{\hat{\boldsymbol{h}}_d\right\}_{d=1}^D$, where $\hat{h}_{dk} = P\left(h_{dk} = 1 \mid (v_{di})_{W_{id}=1}\right)$ and use them *as if* the data-specific factors $(\boldsymbol{h}_d)_{d=1}^D$ are given. The data likelihood has the form $\prod_i P\left(\boldsymbol{v}_{\bullet i} \mid \left\{\hat{\boldsymbol{h}}_d\right\}_{d=1}^D\right)$, where $\boldsymbol{v}_{\bullet i}$ denotes the observations of the $i$-th item.

At each step, we then improve the conditional data likelihood using the gradient technique described in Section 2.4, e.g., by running through the whole data once.
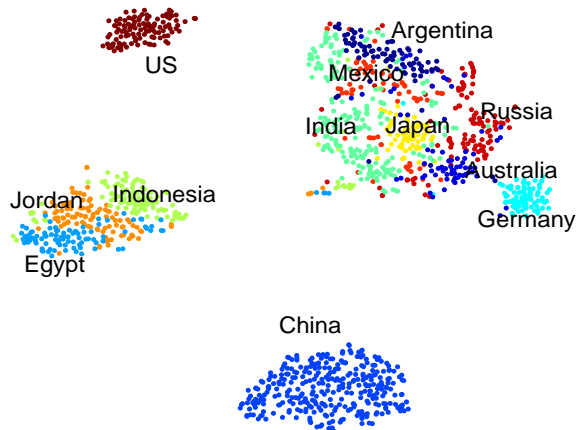
Figure 2: Visualisation of world's opinions in 2008 by projecting latent posteriors $\hat{\boldsymbol{h}} = \left(P(h_1^1|\boldsymbol{v}), P(h_2^1|\boldsymbol{v}, ..., P(h_K^1|\boldsymbol{v})\right)$ on 2D using t-SNE (van der Maaten and Hinton, 2008), where $h_k^1$ is a shorthand for $h_k = 1$. Best viewed in colours. .

### 3.2.1. ONLINE ESTIMATION OF POSTERIORS

The structured mean-fields technique requires the estimation of the factor posteriors. To reduce computation, we propose to treat the trajectory of the factor posteriors during learning as a *stochastic process*. This suggests a simple smoothing method, e.g., at step $t$:

$$\hat{\boldsymbol{h}}_d^{(t)} \leftarrow \eta \hat{\boldsymbol{h}}_d^{(t-1)} + (1 - \eta) P\left(h_{dk} = 1 \mid \boldsymbol{u}_d^{(t)}\right)$$

where $\eta \in (0, 1)$ is the *smoothing factor*, and $\boldsymbol{u}_d^{(t)}$ is a utility sample in the *clamped phase*. This effectively imposes an exponential decay to previous samples. The estimation of $\eta$ would be of interest in its own right, but we would empirically set $\eta \in (0.5, 0.9)$ and do not pursue the issue further.

### 4. Experiments

In this section, we demonstrate how CRBM can be useful in real-world data analysis tasks. To monitor learning progress, we estimate the data pseudo-likelihood $P(v_i \mid \boldsymbol{v}_{\neg i})$. For simplicity, we treat $v_i$ as if it is not in $\boldsymbol{v}$ and replace $\boldsymbol{v}_i$ by $\boldsymbol{v}$. This enables us to use the same predictive methods in Section 2.3. See Fig. 3(a) for an example of the learning curves. To sample from the truncated Gaussian, we employ methods described in (Robert, 1995), which is more efficient than standard rejection sampling techniques. Mapping parameters $\{w_{ik}\}$ are initialised randomly, bias paramters are from zeros, and thresholds $\{\theta_{il}\}$ are spaced evenly at the begining.

### 4.1. Global Attitude Analysis: Latent Profile Discovery

In this experiments we validate the capacity to discover meaningful latent profiles from people's opinions about their life and the social/political conditions in their country and

around the world. We use the public world-wide survey by PewResearch Centre[3] in 2008 which interviewed 24, 717 people from 24 countries. After re-processing, we keep 165 ordinal responses per respondent. Example questions are: "(Q1) [..] how would you describe your day today—has it been a typical day, a particularly good day, or a particularly bad day?", "(Q5) [...] over the next 12 months do you expect the economic situation in our country to improve a lot, improve a little, remain the same, worsen a little or worsen a lot?".

The data is heterogeneous since question types are different (see Section 2.5). For this we use a vector-based CRBM with $K = 50$ hidden units. After model fitting, we obtain a posterior vector $\hat{\boldsymbol{h}} = \left(P(h_1^1|\boldsymbol{v}), P(h_2^1|\boldsymbol{v}), ..., P(h_K^1|\boldsymbol{v})\right)$, which is then used as the representation of the respondent's latent profile. For visualisation, we project this vector onto the 2D plane using a locality-preserving dimensionality reduction method known as *t-SNE*[4] (van der Maaten and Hinton, 2008). The opinions of citizens of 12 countries are depicted in Fig. 2. This clearly reveals how cultures (e.g., Islamic and Chinese) and nations (e.g., the US, China, Latin America) see the world.

### 4.2. Collaborative Filtering: Matrix Completion

We verify our models on three public rating datasets: MovieLens[5] – containing 1 million ratings by 6 thousand users on nearly 4 thousand movies; Dating[6] – consisting of 17 million ratings by 135 thousand users on nearly 169 thousand profiles; and Netflix[7] – 100 millions ratings by 480 thousand users on nearly 18 thousand movies. The Dating ratings are on the 10-point scale and the other two are on the 5-star scale. We then transform the Dating ratings to the 5-point scale for uniformity. For each data we remove those users with less than 30 ratings, 5 of which are used for tuning and stopping criterion, 10 for testing and the rest for training. For MovieLens and Netflix, we ensure that rating timestamps are ordered from training, to validation to testing. For the Dating dataset, the selection is at random.

For comparison, we implement state-of-the-art methods in the field, including: Matrix Factorisation (MF) with Gaussian assumption (Salakhutdinov and Mnih, 2008), MF with cumulative ordinal assumption (Koren and Sill, 2011) (without item-item neighbourhood), and RBM with multinomial assumption (Salakhutdinov et al., 2007).For prediction in the CRBM, we employ the variational method (Section 10). The training and testing protocols are the same for all methods: Training stops where there is no improvement on the likelihood of the validation data. Two popular performance metrics are reported on the test data: the *root-mean square error* (RMSE), the *mean absolute error* (MAE). Prediction for ordinal MF and RBMs is a numerical mean in the case of RMSE: $v_j^{RMSE} = \sum_{l=1}^{L} P(v_j = l|\boldsymbol{v})l$, and an MAP estimation in the case of MAE: $v_j^{MAE} = \arg\max_l P(v_j = l|\boldsymbol{v})$.

Fig. 3(a) depicts the learning curve of the vector-based and matrix-based CRBMs, and Fig. 3(b) shows their predictive performance on test datasets. Clearly, the effect of matrix treatment is significant. Tables 1,2,3 report the performances of all methods on the three datasets. The (matrix) CRBM are often comparable with the best rivals on the RMSE scores and are competitive against all others on the MAE.
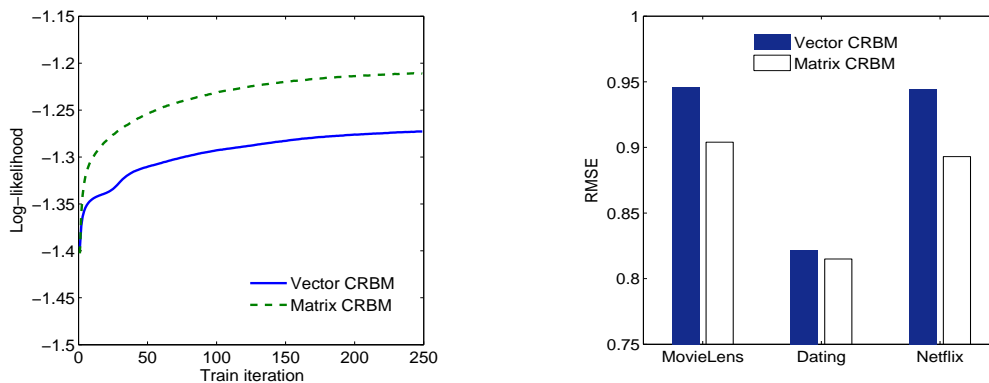
---

3. http://pewresearch.org/

4. Note that the t-SNE does not do clustering, it tries only to map from the input to the 2D so that local properties of the data in preserved.

5. http://www.grouplens.org/node/12

6. http://www.occamslab.com/petricek/data/

7. http://netflixprize.com/

(a) Monitoring pseudo-likelihood in training

(b) RMSE on test data

Figure 3: Vector versus matrix CRBMs, where $K = 50$.

|  | $K = 50$ | | $K = 100$ | | $K = 200$ | |
|---|---|---|---|---|---|---|
|  | *RMSE* | *MAE* | *RMSE* | *MAE* | *RMSE* | *MAE* |
| Gaussian Matrix Fac. | 0.914 | 0.720 | 0.911 | 0.719 | 0.908 | 0.716 |
| Ordinal Matrix Fac. | **0.904** | 0.682 | **0.902** | 0.682 | **0.902** | 0.680 |
| Multinomial RBM | 0.928 | 0.711 | 0.926 | 0.707 | 0.928 | 0.708 |
| Matrix Cumul. RBM | **0.904** | **0.666** | 0.904 | **0.662** | 0.906 | **0.664** |

Table 1: Results on MovieLens (the smaller the better).

|  | $K = 50$ | | $K = 100$ | | $K = 200$ | |
|---|---|---|---|---|---|---|
|  | *RMSE* | *MAE* | *RMSE* | *MAE* | *RMSE* | *MAE* |
| Gaussian Matrix Fac. | 0.852 | 0.596 | 0.848 | 0.592 | 0.840 | 0.586 |
| Ordinal Matrix Fac. | 0.857 | 0.511 | 0.854 | 0.507 | 0.849 | 0.502 |
| Multinomial RBM | **0.815** | 0.483 | **0.794** | 0.470 | 0.787 | 0.463 |
| Matrix Cumul. RBM | **0.815** | **0.475** | 0.799 | **0.461** | 0.794 | **0.458** |

Table 2: Results on Dating (the smaller the better).

|  | $K = 50$ | | $K = 100$ | |
|---|---|---|---|---|
|  | *RMSE* | *MAE* | *RMSE* | *MAE* |
| Gaussian Matrix Fac. | **0.890** | 0.689 | 0.888 | 0.688 |
| Ordinal Matrix Fac. | 0.904 | 0.658 | 0.902 | 0.657 |
| Multinomial RBM | 0.894 | 0.659 | **0.887** | 0.650 |
| Matrix Cumul. RBM | 0.893 | **0.641** | 0.892 | **0.640** |

Table 3: Results on Netflix (the smaller the better).

## 5. Related Work

This work partly belongs to the thread of research that extends RBMs for a variety of data types, including *categories* (Salakhutdinov et al., 2007)*, counts* (Gehler et al., 2006; Salakhutdinov and Hinton, 2009c,b), *bounded* variables (Le et al., 2011) and a mixture of these types (Tran et al., 2011). Gaussian RBMs have been only used for continuous variables (Hinton, 2002; Mohamed and Hinton, 2010) – thus our use for ordinal variables is novel. There has also been recent work extending Gaussian RBMs to better model highly correlated input variables (Ranzato and Hinton, 2010; Courville et al., 2011). For ordinal data, to the best of our knowledge, the first RBM-based work is (Truyen et al., 2009), which also contains a treatment of matrix-wise data. However, their work indeed models multinomial data with knowledge of orders rather than modelling the ordinal nature directly. The result is that it is over-parameterised but less efficient and does not offer any underlying generative mechanism for ordinal data.

Ordinal data has been well investigated in statistical sciences, especially quantitative social studies, often under the name of *ordinal regression*, which refers to single ordinal output given a set of input covariates. The most popular method is by (McCullagh, 1980) which examines the level-wise cumulative distributions. Another well-known treatment is the sequential approach, also known as continuation ratio (Mare, 1980), in which the ordinal generation process is considered stepwise, starting from the lowest level until the best level is chosen. For reviews of recent development, we refer to (Liu and Agresti, 2005). In machine learning, this has attracted a moderate attention in the past decade (Herbrich et al., 1999; Chu and Ghahramani, 2006; Chu and Keerthi, 2007; Cardoso and da Costa, 2007), adding machine learning flavours (e.g., large-margins) to existing statistical methods.

Multivariate ordinal variables have also been studied for several decades (Anderson and Pemberton, 1985). The most common theme is the assumption of the latent multivariate normal distribution that generates the ordinal observations, often referred to as *multivariate probit* models (Chib and Greenberg, 1998; Grilli and Rampichini, 2003; Kottas et al., 2005; Podani, 2005; Jeliazkov et al., 2008; Chagneau et al., 2010). The main problem with this setting is that it is only feasible for problems with small dimensions. Our treatment using RBMs offer a solution for large-scale settings by transferring the low-order interactions among the Gaussian variables onto higher-order interactions through the hidden binary layer. Not only this offers much faster inference, it also enables automatic discovery of latent aspects in the data.

For matrix data, the most well-known method is perhaps matrix factorisation (Lee and Seung, 1999; Rennie and Srebro, 2005; Salakhutdinov and Mnih, 2008). However, this method assumes that the data is normally distributed, which does not meet the ordinal characteristics well. Recent research has attempted to address this issue (Paquet et al., 2011; Koren and Sill, 2011; Tran et al., 2012). In particular, (Paquet et al., 2011; Koren and Sill, 2011) adapt cumulative models of (McCullagh, 1980), and (Tran et al., 2012) tailors the sequential models of (Mare, 1980) for task.

## 6. Conclusion

We have presented CRBM, a novel probabilistic model to handle vector-variate and matrix-variate ordinal data. The model is based on Gaussian restricted Boltzmann machines and we present the model architecture, learning and inference procedures. We show that the

model is useful in profiling opinions of people across cultures and nations. The model is also competitive against state-of-art methods in collaborative filtering using large-scale public datasets. Thus our work enriches the RBMs, and extends their use on multivariate ordinal data in diverse applications.

## References

J.A. Anderson and JD Pemberton. The grouped continuous model for multivariate ordered categorical variables and covariate adjustment. *Biometrics*, pages 875–885, 1985.

Bo Chen Benjamin Marlin, Kevin Swersky and Nando de Freitas. Inductive Principles for Restricted Boltzmann Machine Learning. In *Proceedings of the 13rd International Conference on Artificial Intelligence and Statistics*, Chia Laguna Resort, Sardinia, Italy, May 2010.

J.S. Cardoso and J.F.P. da Costa. Learning to classify ordinal data: the data replication method. *Journal of Machine Learning Research*, 8(1393-1429):6, 2007.

P. Chagneau, F. Mortier, N. Picard, and J.N. Bacro. A hierarchical bayesian model for spatial prediction of multivariate non-gaussian random fields. *Biometrics*, 2010.

S. Chib and E. Greenberg. Analysis of multivariate probit models. *Biometrika*, 85(2): 347–361, 1998.

W. Chu and Z. Ghahramani. Gaussian processes for ordinal regression. *Journal of Machine Learning Research*, 6(1):1019, 2006.

W. Chu and S.S. Keerthi. Support vector ordinal regression. *Neural computation*, 19(3): 792–815, 2007.

A. Courville, J. Bergstra, and Y. Bengio. A spike and slab restricted Boltzmann machine. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 15. Fort Lauderdale, USA, 2011.

Y. Freund and D. Haussler. Unsupervised learning of distributions on binary vectors using two layer networks. *Advances in Neural Information Processing Systems*, pages 912–919, 1993.

P.V. Gehler, A.D. Holub, and M. Welling. The rate adapting Poisson model for information retrieval and object recognition. In *Proceedings of the 23rd international conference on Machine learning*, pages 337–344. ACM New York, NY, USA, 2006.

Leonardo Grilli and Carla Rampichini. Alternative specifications of multivariate multilevel probit ordinal response models. *Journal of Educational and Behavioral Statistics*, 28(1): 31–44, 2003. ISSN 10769986. URL http://www.jstor.org/stable/3701277.

R. Herbrich, T. Graepel, and K. Obermayer. Large margin rank boundaries for ordinal regression. *Advances in Neural Information Processing Systems*, pages 115–132, 1999. ISSN 1049-5258.

G.E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14:1771–1800, 2002.

G.E. Hinton and R.R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

I. Jeliazkov, J. Graves, and M. Kutzbach. Fitting and comparison of models for multivariate ordinal outcomes. *Advances in Econometrics*, 23:115–156, 2008.

Y. Koren and J. Sill. OrdRec: an ordinal model for predicting personalized item rating distributions. In *Proceedings of the fifth ACM conference on Recommender systems*, pages 117–124. ACM, 2011.

A. Kottas, P. Müller, and F. Quintana. Nonparametric Bayesian modeling for multivariate ordinal data. *Journal of Computational and Graphical Statistics*, 14(3):610–625, 2005.

H. Larochelle and Y. Bengio. Classification using discriminative restricted Boltzmann machines. In *Proceedings of the 25th international conference on Machine learning*, pages 536–543. ACM, 2008.

Q.V. Le, W.Y. Zou, S.Y. Yeung, and A.Y. Ng. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. *CVPR*, 2011.

N. Le Roux and Y. Bengio. Representational power of restricted Boltzmann machines and deep belief networks. *Neural Computation*, 20(6):1631–1649, 2008.

D.D. Lee and H.S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

I. Liu and A. Agresti. The analysis of ordered categorical data: an overview and a survey of recent developments. *TEST*, 14(1):1–73, 2005.

R.D. Mare. Social background and school continuation decisions. *Journal of the American Statistical Association*, pages 295–305, 1980.

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

A.R. Mohamed and G. Hinton. Phone recognition using restricted Boltzmann machines. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pages 4354–4357. IEEE, 2010.

U. Paquet, B. Thomson, and O. Winther. A hierarchical model for ordinal matrix factorization. *Statistics and Computing*, pages 1–13, 2011.

J. Podani. Multivariate exploratory analysis of ordinal data in ecology: pitfalls, problems and solutions. *Journal of Vegetation Science*, 16(5):497–510, 2005.

M.A. Ranzato and G.E. Hinton. Modeling pixel means and covariances using factorized third-order Boltzmann machines. In *CVPR*, pages 2551–2558. IEEE, 2010.

J.D.M. Rennie and N. Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, pages 713–719, Bonn, Germany, 2005.

C.P. Robert. Simulation of truncated normal variables. *Statistics and computing*, 5(2): 121–125, 1995.

R. Salakhutdinov and G. Hinton. Deep Boltzmann Machines. In *Proceedings of The Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS'09*, volume 5, pages 448–455, 2009a.

R. Salakhutdinov and G. Hinton. Replicated softmax: an undirected topic model. *Advances in Neural Information Processing Systems*, 22, 2009b.

R. Salakhutdinov and G. Hinton. Semantic hashing. *International Journal of Approximate Reasoning*, 50(7):969–978, 2009c.

R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. *Advances in neural information processing systems*, 20:1257–1264, 2008.

R. Salakhutdinov, A. Mnih, and G. Hinton. Restricted Boltzmann machines for collaborative filtering. In *Proceedings of the 24th International Conference on Machine Learning (ICML)*, pages 791–798, 2007.

P. Smolensky. Information processing in dynamical systems: Foundations of harmony theory. *Parallel distributed processing: Explorations in the microstructure of cognition*, 1: 194–281, 1986.

T. Tieleman. Training restricted Boltzmann machines using approximations to the likelihood gradient. In *Proceedings of the 25th international conference on Machine learning*, pages 1064–1071. ACM, 2008.

T. Tran, D.Q. Phung, and S. Venkatesh. Mixed-variate restricted Boltzmann machines. In *Proc. of 3rd Asian Conference on Machine Learning (ACML)*, Taoyuan, Taiwan, 2011.

T. Tran, D.Q. Phung, and S. Venkatesh. Sequential decision approach to ordinal preferences in recommender systems. In *Proc. of the 26th AAAI Conference*, Toronto, Ontario, Canada, 2012.

T.T. Truyen, D.Q. Phung, and S. Venkatesh. Ordinal Boltzmann machines for collaborative filtering. In *Twenty-Fifth Conference on Uncertainty in Artificial Intelligence (UAI)*, Montreal, Canada, June 2009.

L. van der Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.

L. Younes. Parametric inference for imperfectly observed Gibbsian fields. *Probability Theory and Related Fields*, 82(4):625–645, 1989.