Metadata Dependent Mondrian Processes

Yi Wang[‡†] YI.WANG@NICTA.COM.AU
Bin Li[†] BIN.LI@NICTA.COM.AU
Yang Wang[†] YANG.WANG@NICTA.COM.AU
Fang Chen[†‡] FANG.CHEN@NICTA.COM.AU
[†]Machine Learning Research Group, National ICT Australia, Eveleigh, NSW 2015, Australia
[‡]School of Computer Science & Engineering, University of New South Wales, Kensington, NSW 2033, Australia

Abstract

Stochastic partition processes in a product space play an important role in modeling relational data. Recent studies on the Mondrian process have introduced more flexibility into the block structure in relational models. A side-effect of such high flexibility is that, in data sparsity scenarios, the model is prone to overfit. In reality, relational entities are always associated with meta information, such as user profiles in a social network. In this paper, we propose a metadata dependent Mondrian process (MDMP) to incorporate meta information into the stochastic partition process in the product space and the entity allocation process on the resulting block structure. MDMP can not only encourage homogeneous relational interactions within blocks but also discourage meta-label diversity within blocks. Regularized by meta information, MDMP becomes more robust in data sparsity scenarios and easier to converge in posterior inference. We apply MDMP to link prediction and rating prediction and demonstrate that MDMP is more effective than the baseline models in prediction accuracy with a more parsimonious model structure.

1. Introduction

Relational data exist widely and many real-world applications boil down to relational data modeling, such as community detection, link prediction, and collaborative filtering. Although in different applications relational data may appear in different forms (e.g., binary links in a social network and discrete ratings in a recommender system), the essence of relational data modeling is similar – To represent the relational data as an adjacency matrix and cluster rows and columns simultaneously to uncover the block structure. Such "co-clustering" operation can be understood as permuting row/column entities on each dimension of the data matrix, with the objective to make the intensity of relational interactions consistent within blocks.

Block models (White et al., 1976) have been widely used in modeling, analyzing and predicting relational data. Stochastic block models were first proposed in (Holland et al., 1983) to establish a stochastic generalization of block models. By imposing a Chinese restaurant process on each dimension of the adjacency matrix, the infinite relational model (IRM) (Kemp et al., 2006) discards the restriction on the number of blocks. While IRM and its variants (Airoldi et al., 2009) have obtained the flexibility of the number of blocks, their block structures are restricted to regular grids (Muthukrishnan et al., 1999). The Mondrian process (MP) (Roy & Teh, 2009) was proposed to relax this restriction with a more flexible structure of blocks, which are generated recursively as a kd-tree.

While the Mondrian process is a powerful prior for complex relational modeling, its flexibility may lead to some side-effects compared to other regular block models: 1) It is prone to overfit in data sparsity scenarios; and 2) it is hard to converge based on a uniform prior assumption. In this paper, we aim to incorporate meta information of entities into MP to relieve these problems. The rationale behind is that the entities with similar meta information are more likely to have similar behaviors than those entities with different meta information. For example, people graduating from the same university are more likely to become friends on an online social network. Based on this observation, we propose a metadata dependent Mondrian process (MDMP), which seamlessly integrates the meta information into an MP-like hierarchical partition process.

MDMP can be viewed as a generalization of MP by integrating meta information into both the stochastic partition process in the product space and the entity allocation pro-

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

cess on the resulting block structure. MDMP adopts a similar hierarchical partition process as MP to generate blocks; while at each step of MDMP, a block is first reshaped according to the meta label distribution on each dimension for uniformly sampling the cutting position, such that the partition is more likely to occur on the dimension with relatively diverse meta labels. Due to the reshaping of blocks, partitioning a block with diverse meta labels becomes cheaper and MDMP has higher probability to accept the partition proposal on it. Thus, MDMP will produce a very different block structure compared to MP. For entity clustering on each dimension (i.e., allocating rows/columns onto the block structure), we rescale the cutting intervals on each dimension of the block structure such that a row/column is more likely to be allocated to an interval with the same meta label as the majority.

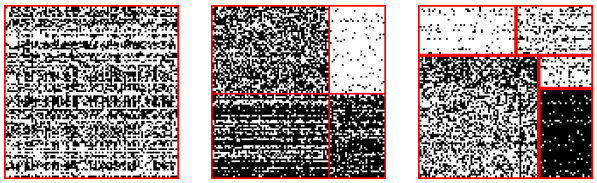
We empirically study the performance of MDMP and baseline models on three real-world data sets for link prediction and rating prediction. The experimental results demonstrate that, by incorporating meta information, MDMP is able to outperform the baselines in prediction accuracy with a more parsimonious model structure.

The remainder of the paper is organized as follows: Section 2 introduces the related work. The proposed MDMP relational model and its posterior inference method will be described in Section 3 and Section 4, respectively. The experimental results are reported in Section 5 and the paper is concluded in Section 6.

2. Related Work

The infinite relational model (IRM) (Kemp et al., 2006) is a Bayesian nonparametric model that does not require to know the number of partitions in advance. The Chinese restaurant process on each dimension of IRM enables an infinite partition on entities. IRM is only restricted to generate a regular grid partition (see Figure 1(b)), which is one of the three types of rectangular partitions (Muthukrishnan et al., 1999). IRM was extended by incorporating temporal dynamics to analyze time-varying relational data (Ishiguro et al., 2010). Then the mixed-membership stochastic blockmodel (MMSB) (Airoldi et al., 2009) was introduced to enable mixed memberships over the latent clusters. Another expressive feature-based block model, named nonparametric latent feature model (Miller et al., 2009), was proposed to model relational data through the combination of latent groups. In this paper, we only consider hard-membership block models.

The Mondrian process (MP) (Roy & Teh, 2009) is the baseline model of MDMP (which can naturally degrade to MP if meta labels are uniformly distributed). MP is a stochastic partition process that recursively generates axis-aligned



(a) (b) (c)

Figure 1. Block models on a synthetic relational data set: (a) The input data; (b) a posterior regular block structure; and (c) a posterior hierarchical block structure.

cuts in a unit hypercube (see Figure 1(c)). In contrast to IRM, MP can partition the space in a hierarchical fashion (Muthukrishnan et al., 1999), known as kd-tree, and results in an irregular block structure. An MP in the product space $[0,1] \times [0,1]$ is started from a random cut on the perimeter and results in two sub-rectangles, in each of which a random cut is made in the same way and so forth. Before cutting on $[a_k^{\perp}, a_k^{\top}] \times [b_k^{\vdash}, b_k^{\dashv}]$ (block k), a cost is drawn from an exponential distribution $E_k \sim \text{Exponential}(a_k^{\top} - a_k^{\perp} + b_k^{\dashv} - b_k^{\vdash})$. If $\lambda - E_k < 0$ (λ is the budget), the recursive procedure halts; otherwise, a random cut is made $m_k \sim \text{MP}(\lambda, [a_k^{\perp}, a_k^{\top}], [b_k^{\vdash}, b_k^{\dashv}])$ and set $\lambda = \lambda - E_k$. Recently, rectangular tiling process (RTP) (Nakano et al., 2014) was proposed to produce arbitrary partitions (Muthukrishnan et al., 1999).

Under many circumstances, a more sophisticated model is required to capture dependence among entities (e.g., temporal dependence and spatial dependence). This constraint has been introduced into Bayesian nonparametric mixture models, such as dependent Dirichlet process (MacEachern, 2000) and distance-dependent Chinese restaurant process (Blei & Frazier, 2011), and Bayesian nonparametric latent feature models, such as dependent Indian buffet process (Williamson et al., 2010), dependent hierarchical beta process (Zhou et al., 2011), kernel beta process (Ren et al., 2011), and distance dependent Indian buffet process (Gershman et al., 2014). In the scope of Bayesian nonparametric relational models with regular block structures, the dependence based on the side information has been introduced in (Choi et al., 2011; Kim et al., 2012). In addition to considering constraints for grouping entities, MDMP directly uses meta information to rectify the generating process of hierarchical block structures.

3. Metadata Dependent Mondrian Process

The input relational data can be represented as an $N \times M$ matrix[1] $\mathbf{Y}$, where N and M are the numbers of entities in the two interacted sets, respectively, and $y_{n,m}$ denotes the

[1]MDMP can be straightforwardly extended to the cases of multi-arrays as the Mondrian process (Roy & Teh, 2009).

value of the interaction between entity n from one set and entity m from the other. Each entity is also associated with a meta label, $c_n^a \in \mathcal{C}_a$ for entity n and $c_m^b \in \mathcal{C}_b$ for entity m. In a social network, $\mathbf{Y}$ can be a symmetric binary matrix indicating links between users, which have meta information like locations as meta labels; in a recommender system, $\mathbf{Y}$ can be an asymmetric integer matrix indicating ratings of movies provided by users, which have meta information like occupations and genres as meta labels.

The quality of the block structure uncovered by an MP largely relies on the likelihood homogeneity of the relational data within blocks. In the cases that within-block interactions are very sparse, the MP is prone to overfit. To address this limitation, we incorporate meta information into the model, such that the block structure relies not only on the likelihood homogeneity but also the meta-label homogeneity. The goal of MDMP is to make use of metadata as side information to improve relational modeling by uncovering better block structures.

In the following, we introduce the proposed MDMP relational model, including one cutting strategy for generating partitions and one indexing strategy for allocating rows/columns to the resulting block structure. Both strategies are dependent on the meta information and will be exploited in Section 4 for inferring the block structure and row/column allocations.

3.1. Cutting Strategy

Given an axis-aligned block k in the product space $[0,1] \times [0,1]$, bounded in $[a_k^\perp, a_k^\top]$ on the vertical axis and $[b_k^\vdash, b_k^\dashv]$ on the horizontal axis, an MP makes a cut on either $[a_k^\perp, a_k^\top]$ or $[b_k^\vdash, b_k^\dashv]$ in proportional to its length (see Figure 2(a–b)). This cutting strategy is based on a reasonable assumption that the longer side is more likely to cover more heterogeneous entities. In the case of meta information being provided, an MDMP also reduces the meta label diversity on each side. Take community detection for example: Communities (blocks in a relational model) are likely to comprise users with similar occupations; in other words, it is more likely to detect reasonable communities by discouraging occupation diversity within blocks while modeling social interaction data.

To this end, an intuitive strategy is to increase (or decrease) the side-length of block k if high (or low) label diversity is observed on that side. We rescale the side-lengths of block k in the following way

$$\begin{aligned} &[a_k^\perp, a_k^\top] \times [b_k^\vdash, b_k^\dashv] \Rightarrow \\ &w_k^a[a_k^\perp, a_k^\top] \times w_k^b[b_k^\vdash, b_k^\dashv] = [\hat{a}_k^\perp, \hat{a}_k^\top] \times [\hat{b}_k^\vdash, \hat{b}_k^\dashv] \end{aligned} \tag{1}$$

where $w_k^a = \exp(\text{Entropy}(\mathbf{h}_k^a)) \in [1, +\infty)$, where $\mathbf{h}_k^a$ is the normalized histogram of meta labels (i.e., label proportions) on the vertical side of block k and $\text{Entropy}(\mathbf{h}_k^a)$ measures the diversity of meta label distribution in $\mathbf{h}_k^a$; and w_k^b is defined similarly for the horizontal side.

In this way, a uniform sampling of cutting position on $[\hat{a}_k^\perp, \hat{a}_k^\top] \times [\hat{b}_k^\vdash, \hat{b}_k^\dashv]$ will be in terms of both the true side-length, corresponding to the number of entities, and the meta label diversity on that side, corresponding to the heterogeneity of entities (see Figure 2(c–d)).

After sampling a cutting position γ from a uniform distribution on the perimeter of the reshaped block

$$\gamma \sim \text{Uniform}\big(0, \hat{a}_k^\top - \hat{a}_k^\perp + \hat{b}_k^\dashv - \hat{b}_k^\vdash\big) \tag{2}$$

the physical cutting position m_k should be mapped back to its original coordinate system (see Figure 2(d–e))

$$m_k^a = a_k^\perp + \frac{\gamma}{w_k^a}, \text{ if } \gamma < \hat{a}_k^\top - \hat{a}_k^\perp \tag{3}$$

$$m_k^b = b_k^\vdash + \frac{\gamma - (\hat{a}_k^\top - \hat{a}_k^\perp)}{w_k^b}, \text{ otherwise} \tag{4}$$

where m_k^a (or m_k^b) denotes the cutting position on the vertical (or horizontal) axis of block k.

It is worth noting that cost sampling is also influenced by the reshaping: $E_k \sim \text{Exponential}(\hat{a}_k^\top - \hat{a}_k^\perp + \hat{b}_k^\dashv - \hat{b}_k^\vdash)$. If the meta label diversity has become reasonably low in block k, the cost will increase and the recursive partition process in this branch is likely to halt earlier.

3.2. Indexing Strategy

Given the current block structure (e.g., Figure 3(a)) obtained in the cutting step, the indexing step aims to allocate rows/columns of $\mathbf{Y}$ to the vertical/horizontal axis of the partition space $[0,1] \times [0,1]$ by making each block have homogeneous relational data (e.g., Figure 1(c)). Since $\mathbf{Y}$ is separately exchangeable given the partition structure and the meta information, the allocation of rows/columns is equivalent to the permutation of rows/columns.

Suppose the current blocks are $\{[a_k^\perp, a_k^\top] \times [b_k^\vdash, b_k^\dashv]\}_{k=1}^K$ (K is the number of blocks[2]). We can obtain all the vertical/horizontal cutting positions projected onto the axes. Let $[r_{1:E}^\perp, r_{1:E}^\top]$ and $[s_{1:F}^\vdash, s_{1:F}^\dashv]$ be the intervals on the vertical and horizontal axes, respectively. An MP assumes a uniform prior distribution for indexing rows/columns on these intervals: $\xi_n \sim \text{Uniform}(\bigcup_{i=1}^E [r_i^\perp, r_i^\top])$ and $\eta_m \sim \text{Uniform}(\bigcup_{j=1}^F [s_j^\vdash, s_j^\dashv])$, where ξ_n and η_m are the indexing variables of the nth row and mth column in $\mathbf{Y}$.

To make within-block meta label distribution more homogeneous, we can increase the probability of sampling ξ_n (or

[2]The blocks correspond to the leaves of the underlying kd-tree, which is produced by hierarchically partitioning the space.

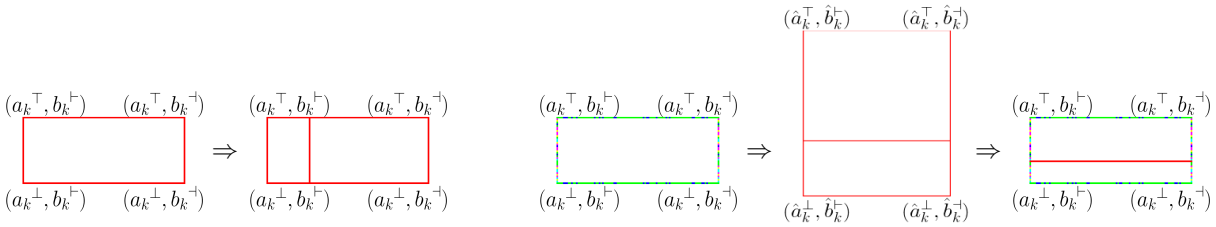


Figure 2. Cutting strategy: (a–b) MP only considers side-length (number of entities) and performs a uniform sampling on the perimeter. (c–e) MDMP considers both side-length (number of entities) and meta label diversity (heterogeneity of entities). The original block (c) with meta label information (different colors denote different labels) is first reshaped to (d); after a uniform sampling on the perimeter of the reshaped block, the cutting position should be mapped back to the original block.

η_m) from the intervals with higher proportion of the corresponding meta label c_n^a (or c_m^b). We also use rescaling to this end: An MDMP samples indexing variables, ξ_n and η_m, on the rescaled intervals conditioned on the index assignments of the other rows $\xi_{\neg n}$ and columns $\eta_{\neg m}$. The rescaled intervals are calculated as

$$[r_i^{\perp}, r_i^{\top}] \quad \Rightarrow \quad v_i^{c_n^a}[r_i^{\perp}, r_i^{\top}] = [\hat{r}_{i,c_n^a}^{\perp}, \hat{r}_{i,c_n^a}^{\top}] \tag{5}$$

$$[s_j^{\vdash}, s_j^{\dashv}] \quad \Rightarrow \quad v_j^{c_m^b}[s_j^{\vdash}, s_j^{\dashv}] = [\hat{s}_{j,c_m^b}^{\vdash}, \hat{s}_{j,c_m^b}^{\dashv}] \tag{6}$$

where $v_i^{c_n^a}$ and $v_j^{c_m^b}$ are rescaling weighs implicitly defined in Eqs. 7 and 8; c_n^a and c_m^b denote meta labels. For example, in a rating matrix, $c_n^a \in \{\text{student}, \text{engineer}, \text{professor}\}$ can be occupation of users while $c_m^b \in \{\text{classic}, \text{folk}, \text{jazz}\}$ can be genre of music.

We rescale the intervals as follows: Calculate the normalized portion of meta labels over vertical cuts and horizontal cuts; then use this proportion to weight $[r_{1:E}^{\perp}, r_{1:E}^{\top}]$ and $[s_{1:F}^{\vdash}, s_{1:F}^{\dashv}]$

$$(\hat{r}_{i,c_n^a}^{\top} - \hat{r}_{i,c_n^a}^{\perp}) = \frac{(r_i^{\top} - r_i^{\perp})\mathcal{N}_{i,c_n^a}}{\sum_{i'=1}^{E}(r_{i'}^{\top} - r_{i'}^{\perp})\mathcal{N}_{i',c_n^a}} \tag{7}$$

$$(\hat{s}_{j,c_m^b}^{\dashv} - \hat{s}_{j,c_m^b}^{\vdash}) = \frac{(s_j^{\dashv} - s_j^{\vdash})\mathcal{N}_{j,c_m^b}}{\sum_{j'=1}^{F}(s_{j'}^{\dashv} - s_{j'}^{\vdash})\mathcal{N}_{j',c_m^b}} \tag{8}$$

where $\mathcal{N}_{i,c_n^a}$ denotes the number of rows with meta label c_n^a allocated to the ith interval on the vertical axis; $\mathcal{N}_{j,c_m^b}$ is similarly defined for the columns. In implementation, a small number can be added to $\mathcal{N}_{i,c_n^a}$ and $\mathcal{N}_{j,c_m^b}$ for regularization. This rescaling method is illustrated in Figure 3. We will use the rescaled lengths of intervals for sampling indexing variables in Eqs. 16 and 17.

3.3. Graphical Model

The generative process of the MDMP relational model is as follows (the corresponding graphical model is shown in Figure 4):

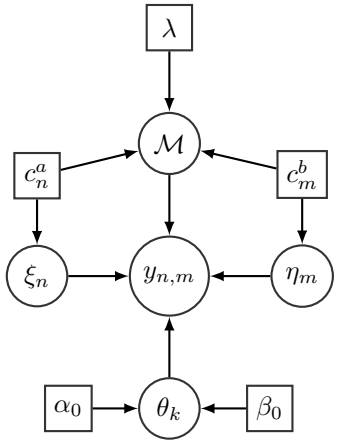


Figure 4. The graphical representation of the MDMP relational model (with the beta-Bernoulli model in each block).

- $\theta_k \sim \text{Beta}(\alpha_0, \beta_0)$ (link data) or $\theta_k \sim \text{Dirichlet}(\alpha_0)$ (rating data), $k = 1, \ldots, K$;
- $[a_k^{\perp}, a_k^{\top}, b_k^{\vdash}, b_k^{\dashv}] \sim \text{MDMP}(\lambda, c_{1:N}^a, c_{1:M}^b)$, $k = 1, \ldots, K$;
- $\xi_n \sim \text{Uniform}(\bigcup_{i=1}^{E}[\hat{r}_{i,c_n^a}^{\perp}, \hat{r}_{i,c_n^a}^{\top}])$, $n = 1, \ldots, N$;
- $\eta_m \sim \text{Uniform}(\bigcup_{j=1}^{F}[\hat{s}_{j,c_m^b}^{\vdash}, \hat{s}_{j,c_m^b}^{\dashv}])$, $m = 1, \ldots, M$;
- $y_{n,m} \sim \text{Bernoulli}(\theta_{\hbar(\xi_n,\eta_m)})$ (link data) or $y_{n,m} \sim \text{Discrete}(\theta_{\hbar(\xi_n,\eta_m)})$ (rating data), $n = 1, \ldots, N$, $m = 1, \ldots, M$.

where $\mathcal{M} = \{[a_k^{\perp}, a_k^{\top}, b_k^{\vdash}, b_k^{\dashv}]\}_{k=1}^{K}$ denotes the block structures (a kd-tree on $[0,1] \times [0,1]$) and $\hbar(\xi_n, \eta_m)$ denotes a mapping from a row-column index pair to a block index in $\mathcal{M}$. Note that we neglect some intermediate steps, such as $\{[\hat{r}_{i,c_n^a}^{\perp}, \hat{r}_{i,c_n^a}^{\top}]\}_{i=1}^{E}$ and $\{[\hat{s}_{j,c_m^b}^{\vdash}, \hat{s}_{j,c_m^b}^{\dashv}]\}_{j=1}^{F}$ are calculated based on $\{[a_k^{\perp}, a_k^{\top}, b_k^{\vdash}, b_k^{\dashv}]\}_{k=1}^{K}$.

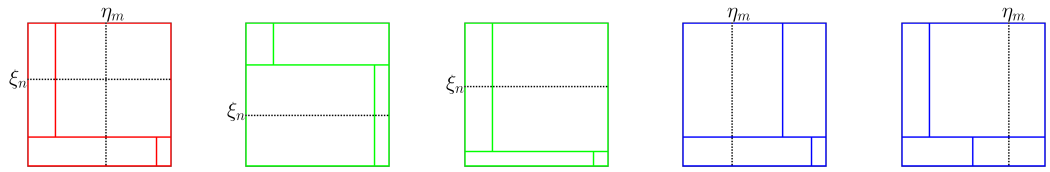


Figure 3. Indexing strategy: (a) The block structure; (b–c) rescaled intervals on the vertical axis for two different meta labels; (d–e) rescaled intervals on the horizontal axis for two different meta labels. After rescaling, indices ξ_n and η_m are more likely to be assigned to those vertical/horizontal intervals with higher proportion of the same meta label.

4. Inference

The joint probability of relational data $\mathbf{Y}$, model parameters $\{\mathcal{M}, \theta_{1:K}\}$, and indexing variables $\{\xi_{1:N}, \eta_{1:M}\}$ is

$$\begin{aligned} & p(\mathbf{Y}, \mathcal{M}, \theta_{1:K}, \xi_{1:N}, \eta_{1:M} | \lambda, \alpha_0, \beta_0, c^a_{1:N}, c^b_{1:M}) \\ = & \prod_{n=1}^{N} \prod_{m=1}^{M} p(y_{n,m} | \mathcal{M}, \theta_{1:K}, \xi_n, \eta_m) \\ & \times \prod_{n=1}^{N} p(\xi_n | \mathcal{M}, c^a_{1:N}) \prod_{m=1}^{M} p(\eta_m | \mathcal{M}, c^b_{1:M}) \\ & \times p(\mathcal{M} | \lambda, c^a_{1:N}, c^b_{1:M}) \times \prod_{k=1}^{K} p(\theta_k | \alpha_0, \beta_0) \end{aligned} \tag{9}$$

where $\theta_{1:K}$ can be marginalized out (if we use beta-Bernoulli, Dirichlet-multinomial, or other conjugate distributions). Thus, we need to estimate $\mathcal{M}$ and $\{\xi_{1:N}, \eta_{1:M}\}$.

The inference framework for MDMP is outlined in Algorithm 1. We adopt two nested loops of MCMC sampling for approximate inference: The outer loop is to infer the block structure $\mathcal{M}$ by proposing adding or removing a cut. Since this part of inference involves dimensionality change of the parameter space, we adopt the reversible-jump MCMC (Green, 1995) algorithm (see Section 4.1). The inner loop is to infer row/column indexing variables $\xi_{1:N}$ and $\eta_{1:M}$, given the current $\mathcal{M}$. This part of inference can be simply solved by using the collapsed Gibbs sampling algorithm (see Section 4.2).

4.1. Sampling Partitions $\mathcal{M}$

The reversible-jump MCMC (RJMCMC) (Green, 1995) is aimed to sample posterior distributions in which the dimensionality of parameter space varies between iterations of a Markov chain. RJMCMC has been used in MP-based co-clustering ensembles in (Wang et al., 2011). We also adopt this technique for sampling the block structure $\mathcal{M}$ in an MDMP. Each step of the RJMCMC algorithm proposes to add or remove a cut in $\mathcal{M}$.

Suppose from iteration t to $t+1$ (the outer loop in Algorithm 1), the RJMCMC algorithm proposes to add a cut (the change of the block structure is denoted by $\mathcal{M}_t \to \mathcal{M}_{t+1}$), the acceptance ratio of this proposal is

Algorithm 1 Approximate inference for MDMP

Input: $\mathbf{Y}$ and $\{\lambda, \alpha_0, \beta_0, c^a_{1:N}, c^b_{1:M}\}$
Output: $\mathcal{M}$ and $\{\xi_{1:N}, \eta_{1:M}\}$
repeat
  Initialize $\mathcal{M}$ as $[0,1] \times [0,1]$;
  Propose a partition (add/remove a cut) in $\mathcal{M}$;
  Accept/Reject the proposal (RJMCMC);
  if Accepted **then**
    Sample $\xi_{1:N}$ and $\eta_{1:M}$ from $\mathcal{M}$ (Gibbs sampling);
  end if
until Exceed the number of iterations

$$\begin{aligned} & \alpha_{\mathcal{M}_t \to \mathcal{M}_{t+1}} \\ = & \min \left\{ 1, \frac{p(\mathcal{M}_{t+1} | \mathbf{Y}, \xi_{1:N}, \eta_{1:M}, \lambda, \alpha_0, \beta_0)}{p(\mathcal{M}_t | \mathbf{Y}, \xi_{1:N}, \eta_{1:M}, \lambda, \alpha_0, \beta_0)} \right. \\ & \left. \times \frac{q(\mathcal{M}_{t+1} \to \mathcal{M}_t)}{q(\mathcal{M}_t \to \mathcal{M}_{t+1})} \times |\mathcal{J}_{\mathcal{M}_t \to \mathcal{M}_{t+1}}| \right\} \end{aligned} \tag{10}$$

where the first term $\frac{p(\mathcal{M}_{t+1}|-)}{p(\mathcal{M}_t|-)}$ is the ratio of the posterior probabilities of the two block structures given the data; the second term $\frac{q(\mathcal{M}_{t+1} \to \mathcal{M}_t)}{q(\mathcal{M}_t \to \mathcal{M}_{t+1})}$ is the ratio of the proposal probabilities; and the last term $|\mathcal{J}_{\mathcal{M}_t \to \mathcal{M}_{t+1}}|$ is the determinant of the Jacobian inter-model transition matrix.

Let $\mathcal{M}_{t+1} = \{\mathcal{M}_t, m_k, E_k\}$ (as defined before, m_k denotes a cutting and E_k denotes the associated cost) and $\{u_1, u_2\}$ (generated using the sample proposal distributions as $\{m_k, E_k\}$) be the corresponding auxiliary variables for $\{m_k, E_k\}$, there is a bijection between $\{\mathcal{M}_t, u_1, u_2\}$ and $\mathcal{M}_{t+1}$ characterised by an identity inter-model transition matrix; thus we have $|\mathcal{J}_{\mathcal{M}_t \to \mathcal{M}_{t+1}}| = 1$. For simplicity, we can also assume that the state transition proposal distribution is symmetric.

The ratio of the posterior probabilities $\frac{p(\mathcal{M}_{t+1}|-)}{p(\mathcal{M}_t|-)}$ can be rewritten as a production of a prior ratio and a likelihood

ratio $\frac{p(\mathcal{M}_{t+1}|\lambda)}{p(\mathcal{M}_t|\lambda)} \times \frac{\mathcal{L}(\mathcal{M}_{t+1})}{\mathcal{L}(\mathcal{M}_t)}$. The prior ratio is

$$\frac{p(\mathcal{M}_{t+1}|\lambda)}{p(\mathcal{M}_t|\lambda)} = \frac{p(m_k)p(E_k)\varrho_{k'}\varrho_{k''}}{\varrho_k} \tag{11}$$

where $p(m_k)p(E_k) = \exp(-\phi_k E_k)$ denotes the probability of sampling a cut m_k in block k ($\phi_k = \hat{a}_k^{\top} - \hat{a}_k^{\perp} + \hat{b}_k^{\dashv} - \hat{b}_k^{\vdash}$); $\varrho_k = \exp(-\phi_k\lambda_k)$ denotes the probability of terminating at block k (same definitions for $\varrho_{k'}$ and $\varrho_{k''}$ in block k' and k'', respectively).

If we adopt the compound beta-Bernoulli distribution, the likelihoods given $\mathcal{M}_{t+1}$ and $\mathcal{M}_t$ are

$$\begin{aligned}\mathcal{L}(\mathcal{M}_t) &= \prod_{k_t=1}^{K_t}\prod_{y_{n,m}\in\mathcal{M}_{k_t}} p(y_{n,m}|\theta_{k_t}) \\ &= \prod_{k_t=1}^{K_t}\theta_{k_t}^{\mathcal{N}_{k_t,+}}(1-\theta_{k_t})^{\mathcal{N}_{k_t,-}} \qquad (12)\\ \mathcal{L}(\mathcal{M}_{t+1}) &= \prod_{k_{t+1}=1}^{K_{t+1}}\prod_{y_{n,m}\in\mathcal{M}_{k_{t+1}}} p(y_{n,m}|\theta_{k_{t+1}}) \\ &= \prod_{k_{t+1}=1}^{K_{t+1}}\theta_{k_{t+1}}^{\mathcal{N}_{k_{t+1},+}}(1-\theta_{k_{t+1}})^{\mathcal{N}_{k_{t+1},-}} \qquad (13)\end{aligned}$$

where K_t denotes the number of blocks in $\mathcal{M}_t$ and $\mathcal{N}_{k_t,+/-}$ denotes the number of observed positive/negtive entries in block k_t.

Similarly, if we adopt the compound Dirichlet-multinomial distribution, the likelihoods given $\mathcal{M}_{t+1}$ and $\mathcal{M}_t$ are

$$\begin{aligned}\mathcal{L}(\mathcal{M}_t) &= \prod_{k_t=1}^{K_t}\prod_{l=1}^{L}\theta_{k_t,l}^{\mathcal{N}_{k_t,l}} \qquad (14)\\ \mathcal{L}(\mathcal{M}_{t+1}) &= \prod_{k_{t+1}=1}^{K_{t+1}}\prod_{l=1}^{L}\theta_{k_{t+1},l}^{\mathcal{N}_{k_{t+1},l}} \qquad (15)\end{aligned}$$

where $\theta_{k_{t,1:L}}$ denotes the parameter of the Dirichlet distribution in block k_t of $\mathcal{M}_t$; L denotes the number of meta label categories; and $\mathcal{N}_{k_t,l}$ denotes the number of observed entries with meta label l in block k_t.

The RJMCMC algorithm for removing a cut from $\mathcal{M}_t$ is as similar as adding a cut. We assume that the probability of proposing to add or remove a cut is equal.

4.2. Sampling Indices $\xi_{1:N}$ and $\eta_{1:M}$

Given a block structure $\mathcal{M}$, we can use Gibbs sampling (Geman & Geman, 1984) to approximate the posterior distribution of the indexing variables $\xi_{1:N}$ and $\eta_{1:M}$.

Based on the joint probability Eq. 9, the conditional posterior of ξ_n (i.e., probability of allocating nth row to the ith vertical interval) with beta-Bernoulli likelihood gives

$$\begin{aligned}p(\xi_n \in [r_i^{\perp}, r_i^{\top}]|\mathbf{Y}, \mathcal{M}, \xi_{1:N}^{\neg n}, \eta_{1:M}, c_{1:N}^a) \propto \\ (\hat{r}_{i,c_n^a}^{\top} - \hat{r}_{i,c_n^a}^{\perp}) \prod_{k\in\mathcal{S}_i}\binom{\mathcal{N}_{n,+}^{i,k}+\mathcal{N}_{n,-}^{i,k}}{\mathcal{N}_{n,+}^{i,k}} \\ \frac{\mathcal{B}(\mathcal{N}_{n,+}^{i,k}+\alpha_k, \mathcal{N}_{n,-}^{i,k}+\beta_k)}{\mathcal{B}(\alpha_k,\beta_k)}\end{aligned} \tag{16}$$

where $(\hat{r}_{1:E,c_n^a}^{\top} - \hat{r}_{1:E,c_n^a}^{\perp})$ are the rescaled vertical intervals according to Eq. 7, $\mathcal{S}_i$ denotes the set of blocks which have interactions with the ith vertical interval, and $\mathcal{N}_{n,+/-}^{i,k}$ denotes the number of positive/negtive entries in the nth row if it is assigned to the kth block which is traversed by the ith vertical interval.

The conditional posterior of ξ_n with Dirichlet-multinomial likelihood gives

$$\begin{aligned}p(\xi_n \in [r_i^{\perp}, r_i^{\top}]|\mathbf{Y}, \mathcal{M}, \xi_{1:N}^{\neg n}, \eta_{1:M}, c_{1:N}^a) \propto \\ (\hat{r}_{i,c_n^a}^{\top} - \hat{r}_{i,c_n^a}^{\perp}) \prod_{k\in\mathcal{S}_i}\frac{\prod_{l=1}^{L}\Gamma(\mathcal{N}_{n,l}^{i,k}+\alpha_{k,l})}{\Gamma(\sum_{l=1}^{L}\mathcal{N}_{n,l}^{i,k}+\alpha_k)}\frac{\Gamma(\alpha_k)}{\prod_{l=1}^{L}\Gamma(\alpha_{k,l})}\end{aligned} \tag{17}$$

where $\mathcal{N}_{n,l}^{i,k}$ denotes the number of entries with meta label l in the nth row if it is assigned to the kth block which is traversed by the ith vertical interval.

The conditional posterior of η_m can be derived in the same way as ξ_n. The conditional posterior of θ_k is simple. For beta-Bernoulli likelihood, we have

$$\theta_k \propto \frac{\alpha_0 + \mathcal{N}_{k,+}}{\alpha_0+\beta_0+\mathcal{N}_{k,+}+\mathcal{N}_{k,-}} \tag{18}$$

while for Dirichlet-multinomial likelihood, we have

$$\theta_{k,l} \propto \mathcal{N}_{k,l} + \alpha_0 \tag{19}$$

5. Experiment

We empirically test the proposed MDMP relational model on three real-world data sets with various meta information. We compare MDMP to IRM (Kemp et al., 2006) (block model with Bernoulli distribution in each block) for link prediction, BiLDA (Porteous et al., 2008) (block model with discrete distribution in each block) for rating prediction, and MP (Roy & Teh, 2009) for both.

We adopt the following performance measures: 1) Log-likelihood (LL) for measuring the fitness of block modeling; 2) Bayesian information criterion (BIC) for measuring the fitness of block modeling penalized by free parameters; 3) Area under curve (AUC) for measuring the link prediction performance; and 4) Root mean square error (RMSE) for measuring the rating prediction performance.

In our experiments, each data set is partitioned into 5 splits, and each time 4 splits are used for training and the rest one is used for testing. All the reported results are average values over five runs. For MP and MDMP, we perform 500 iterations of RJMCMC sampling.

5.1. Link Prediction: Lazega's Lawyer

The first data set adopted for link prediction is the Lazega's lawyer data[3] (Lazega, 2003). In this data set, there are three different relationships among 71 lawyers in a law firm, which are "Advisory", "Friendship" and "Workmate". For each lawyer in the network, seven different types of side information is also provided, including gender, law school they graduated from, office, practice, status, years with the firm, and age. The first five types of this side information is incorporated into MDMP for evaluation.

The block modeling results are visualized in Figure 5 and the performance comparison results are reported in Table 1. For a fair comparison, we select proper hyper-parameters to make the total number of blocks in IRM, MP and MDMP be approximately at the same level. From Figure 5, we can see that MDMP can uncover clearer block structures than IRM and MP. From Table 1, we can see that MDMP not only most fits the data (in LL) but also has the most parsimonious model structure (in BIC). It is worth noting that, among five types of meta information, "gender" is most helpful for predicting friendships, "law school" is most helpful for predicting advisory, and "practice" is most helpful for predicting workmates.

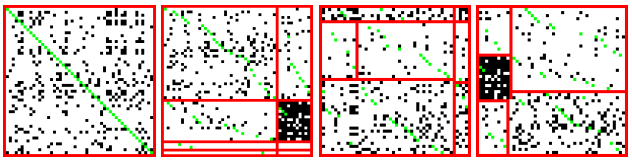
5.2. Link Prediction: Douban

Douban is an SNS provider which allows users to share and review movies, books, and music. The user connections in Douban form an asymmetric network, on which each user has a profile with demographical information. We adopt "City" information of each user as the meta information for MDMP. In this experiment, we adopt a preprocessed data set[4] (Ma et al., 2011), which comprises 21593 users. We randomly select 50 users for evaluation. The density of links in the resulting network is around 14.3%.

The block modeling results are visualized in Figure 6 and the performance comparison results are reported in Table 2. We can see that MDMP has a more parsimonious block structure than IRM and a clearer block structure than MP. MDMP performs best; while MP performs even worse than IRM and seems to have not yet converged given the same number of sampling iterations.

[3] https://www.stats.ox.ac.uk/~snijders/siena/Lazega_lawyers_data.htm

[4] https://www.cse.cuhk.edu.hk/irwin.king.new/pub/data/douban



(a) Input (b) IRM (c) MP (d) MDMP

Figure 6. Block structure visualization on the Douban dataset.

Dataset: Douban	IRM	MP	MDMP
LL (Blockmodeling)	-782.8	-935.6	**-757.5**
BIC (Blockmodeling)	1690.7	1925.8	**1509.5**
AUC (Prediction)	0.7420	0.6987	**0.7638**

Table 2. Performance comparison on the Douban dataset.

5.3. Rating Prediction: MovieLens

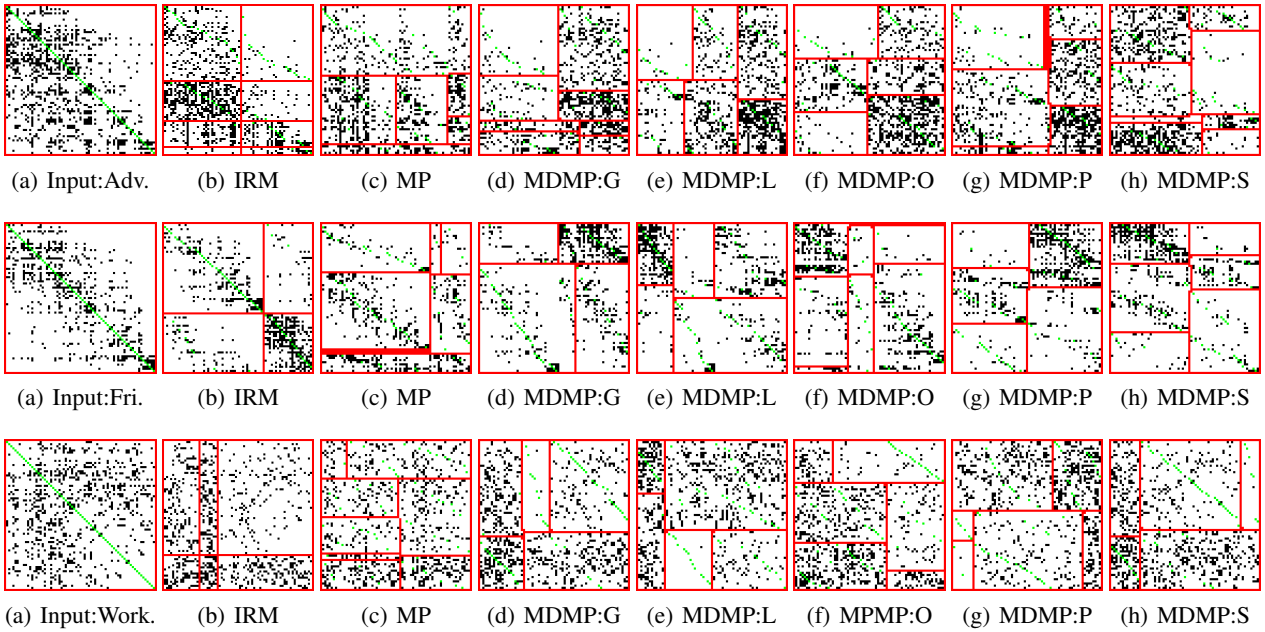
We adopt the MovieLens data set[5] for rating prediction. It comprises 6040 users and 3883 movies. Each user is associated with three types of meta information: gender, age and career. We don't consider movie meta information for simplicity. The three types of user meta information are incorporated into MDMP. We randomly select 70 users and 70 items from the entire data set and keep the sparsity of the rating matrix being 80% for evaluation.

The block modeling results are visualized in Figure 7 and the performance comparison results are reported in Table 3. From Figure 7, we can see that the ratings within blocks are more homogeneous in MDMP than in BiLDA and MP, especially in the case of incorporating gender information. From Table 3, we can see that MDMP with gender gives the best block modeling result; while MDMP with career gives the best rating prediction result.

6. Conclusion

In this paper, we propose a metadata dependent Mondrian process (MDMP) that incorporates meta information of entities into the partition process. MDMP can not only encourage homogeneous relational data within blocks but also discourage meta-label diversity within blocks. By incorporating meta information, MDMP becomes more robust in data sparsity scenarios and converges faster in posterior inference. The empirical tests on three real-world data sets demonstrate that, regularized by meta information, MDMP can uncover clearer block structures than IRM and MP with a more parsimonious model structure and higher prediction accuracy. In our future work, we will 1) investigate how to make better use of meta information and 2) exploit MDMP for generating related rating matrices (Li et al., 2009).

[5] https://movielens.org/



(a) Input:Adv. (b) IRM (c) MP (d) MDMP:G (e) MDMP:L (f) MDMP:O (g) MDMP:P (h) MDMP:S

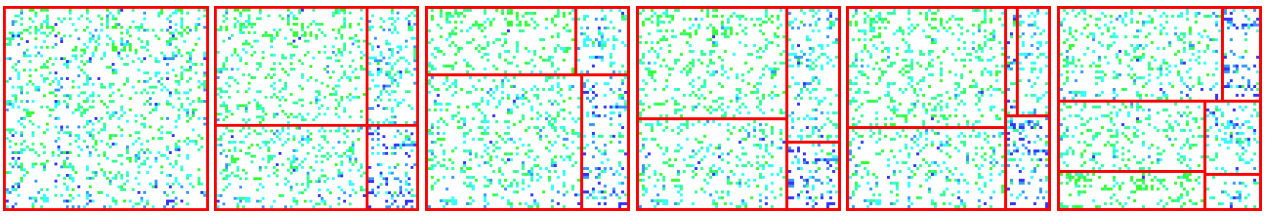
(a) Input:Fri. (b) IRM (c) MP (d) MDMP:G (e) MDMP:L (f) MDMP:O (g) MDMP:P (h) MDMP:S

(a) Input:Work. (b) IRM (c) MP (d) MDMP:G (e) MDMP:L (f) MPMP:O (g) MDMP:P (h) MDMP:S

Figure 5. Block structure visualization on the Lazega's Lawyer dataset: (a) Input data with three types of links (Advisory, Friend, Workmate); (b) IRM; (c) MP; (d–h) MDMP with 5 different meta information (Gender, Lawschool, Office, Practice, Status).

Dataset: Lazega Lawyer's	IRM	MP	MDMP:G	MDMP:L	MDMP:O	MDMP:P	MDMP:S
LL (Blockmodeling): Advisory	-1961.3	-2206.8	-1996.0	**-1953.7**	-1970.7	-1994.2	-1974.9
LL (Blockmodeling): Friend	-1546.6	-1653.6	**-1507.9**	-1521.8	-1546.0	-1552.3	-1534.0
LL (Blockmodeling): Work	-1960.7	-2048.5	-1955.2	-1958.2	-1958.1	**-1954.3**	-1954.7
BIC (Blockmodeling): Advisory	3990.7	4481.8	4000.4	**3916.0**	3949.9	3996.9	3958.4
BIC (Blockmodeling): Friend	3127.0	3315.7	**3024.2**	3052.2	3100.5	3113.2	3076.5
BIC (Blockmodeling): Work	3972.4	4207.8	3918.9	3924.8	3924.6	**3917.1**	3917.9
AUC (Prediction): Advisory	0.7418	0.7026	0.7497	**0.7645**	0.7605	0.7507	0.7518
AUC (Prediction): Friend	0.7208	0.6954	**0.7715**	0.7547	0.7116	0.6857	0.7407
AUC (Prediction): Work	0.6752	0.6662	0.6265	0.6379	0.6629	0.6767	**0.6842**

Table 1. Performance comparison on the Lazega's Lawyer dataset.



(a) Input (b) BiLDA (c) MP (d) MDMP:Age (e) MDMP:Career (f) MDMP:Gender

Figure 7. Block structure visualization on the MovieLens dataset (five rating scores are denoted by the color ranging from green to purple; unobserved ratings are denoted by white entries).

Dataset: MovieLens	BiLDA	MP	MDMP:Age	MDMP:Career	MDMP:Gender
LL (Blockmodeling)	-1256.4	-1244.7	-1246.7	-1243.2	**-1225.4**
BIC (Blockmodeling)	2650.8	2493.5	2497.4	2491.4	**2456.8**
RMSE (Prediction)	0.8116	0.7916	0.7982	**0.7239**	0.7755

Table 3. Performance comparison on the MovieLens dataset.

References

Airoldi, Edoardo M., Blei, David M., Fienberg, Stephen E., and Xing, Eric P. Mixed membership stochastic blockmodels. In *NIPS*, pp. 33–40, 2009.

Blei, David M. and Frazier, Peter I. Distance dependent chinese restaurant processes. *Journal of Machine Learning Research*, 12:2383–2410, 2011.

Choi, David S., Wolfe, Patrick, and Airoldi, Edoardo M. Confidence sets for network structure. In *NIPS*, pp. 2097–2105, 2011.

Geman, Stuart and Geman, Donald. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

Gershman, Samuel, Frazier, Peter, and Blei, David. Distance dependent infinite latent feature models. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

Green, Peter J. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

Holland, Paul W., Laskey, Kathryn Blackmond, and Leinhardt, Samuel. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109–137, 1983.

Ishiguro, Katsuhiko, Iwata, Tomoharu, Ueda, Naonori, and Tenenbaum, Joshua B. Dynamic infinite relational model for time-varying relational data analysis. In *NIPS*, pp. 919–927, 2010.

Kemp, Charles, Tenenbaum, Joshua B., Griffiths, Thomas L., Yamada, Takeshi, and Ueda, Naonori. Learning systems of concepts with an infinite relational model. In *AAAI*, pp. 381–388, 2006.

Kim, Dae Il, Hughes, Michael, and Sudderth, Erik. The nonparametric metadata dependent relational model. In *ICML*, pp. 1559–1566, 2012.

Lazega, Emmanuel. The collegial phenomenon: The social mechanisms of cooperation among peers in a corporate law partnership. *Administrative Science Quarterly*, 48 (3):525–529, 2003.

Li, Bin, Yang, Qiang, and Xue, Xiangyang. Transfer learning for collaborative filtering via a rating-matrix generative model. In *ICML*, pp. 617–624, 2009.

Ma, Hao, Zhou, Dengyong, Liu, Chao, Lyu, Michael R., and King, Irwin. Recommender systems with social regularization. In *WSDM*, pp. 287–296, 2011.

MacEachern, Steven N. Dependent Dirichlet processes. *Unpublished manuscript, Department of Statistics, The Ohio State University*, 2000.

Miller, Kurt, Jordan, Michael I., and Griffiths, Thomas L. Nonparametric latent feature models for link prediction. In *NIPS*, pp. 1276–1284, 2009.

Muthukrishnan, S., Poosala, Viswanath, and Suel, Torsten. On rectangular partitionings in two dimensions: Algorithms, complexity and applications. In *ICDT*, pp. 236–256, 1999.

Nakano, Masahiro, Ishiguro, Katsuhiko, Kimura, Akisato, Yamada, Takeshi, and Ueda, Naonori. Rectangular tiling process. In *ICML*, pp. 361–369, 2014.

Porteous, Ian, Bart, Evgeniy, and Welling, Max. Multi-HDP: A non parametric Bayesian model for tensor factorization. In *AAAI*, pp. 1487–1490, 2008.

Ren, Lu, Wang, Yingjian, Carin, Lawrence, and Dunson, David B. The kernel beta process. In *NIPS*, pp. 963–971, 2011.

Roy, Daniel M. and Teh, Yee W. The Mondrian process. In *NIPS*, pp. 1377–1384, 2009.

Wang, Pu, Laskey, Kathryn B., Domeniconi, Carlotta, and Jordan, Michael I. Nonparametric Bayesian co-clustering ensembles. In *SDM*, pp. 331–342, 2011.

White, Harrison C., Boorman, Scott A., and Breiger, Ronald L. Social structure from multiple networks. i. blockmodels of roles and positions. *American Journa of Sociology*, pp. 730–780, 1976.

Williamson, Sinead, Orbanz, Peter, and Ghahramani, Zoubin. Dependent Indian buffet processes. In *AISTATS*, pp. 924–931, 2010.

Zhou, Mingyuan, Yang, Hongxia, Sapiro, Guillermo, Dunson, David B., and Carin, Lawrence. Dependent hierarchical beta process for image interpolation and denoising. In *AISTATS*, pp. 883–891, 2011.