Journal of the Royal Statistical Society Series B: Statistical Methodology, 2023, **00**, 1–28
https://doi.org/10.1093/jrsssb/qkad010

ROYAL STATISTICAL SOCIETY
DATA | EVIDENCE | DECISIONS
Journal of the Statistics Society
Series B
Statistical Methodology

Original Article

Bayesian Pyramids: identifiable multilayer discrete latent structure models for discrete data

Yuqi Gu[1] **and David B. Dunson**[2]

[1]Department of Statistics, Columbia University, New York, NY, USA
[2]Department of Statistical Science, Duke University, Durham, NC, USA

Address for correspondence: Yuqi Gu, Department of Statistics, Columbia University, New York, NY, USA.
Email: yuqi.gu@columbia.edu

Abstract

High-dimensional categorical data are routinely collected in biomedical and social sciences. It is of great importance to build interpretable parsimonious models that perform dimension reduction and uncover meaningful latent structures from such discrete data. Identifiability is a fundamental requirement for valid modeling and inference in such scenarios, yet is challenging to address when there are complex latent structures. In this article, we propose a class of identifiable multilayer (potentially deep) discrete latent structure models for discrete data, termed *Bayesian Pyramids*. We establish the identifiability of Bayesian Pyramids by developing novel transparent conditions on the pyramid-shaped deep latent directed graph. The proposed identifiability conditions can ensure Bayesian posterior consistency under suitable priors. As an illustration, we consider the two-latent-layer model and propose a Bayesian shrinkage estimation approach. Simulation results for this model corroborate the identifiability and estimatability of model parameters. Applications of the methodology to DNA nucleotide sequence data uncover useful discrete latent features that are highly predictive of sequence types. The proposed framework provides a recipe for interpretable unsupervised learning of discrete data and can be a useful alternative to popular machine learning methods.

Keywords: Bayesian inference, deep generative models, identifiability, interpretable machine learning, latent class, multivariate categorical data

1 Introduction

High-dimensional unordered categorical data are ubiquitous in many scientific disciplines, including the DNA nucleotides of A, G, C, T in genetics (Nguyen et al., 2016; Pokholok et al., 2005), occurrences of various species in ecological studies of biodiversity (Ovaskainen & Abrego, 2020; Ovaskainen et al., 2016), responses from psychological and educational assessments or social science surveys (Eysenck et al., 2020; Skinner, 2019), and document data gathered from huge text corpora or publications (Blei et al., 2003; Erosheva et al., 2004). Modeling and extracting information from multivariate discrete data require different statistical methods and theoretical understanding from those for continuous data. In an unsupervised setting, it is an important task to uncover reliable and meaningful latent patterns from the potentially high-dimensional and heterogeneous discrete observations. Ideally, the inferred lower-dimensional latent representations should not only provide scientific insights on their own, but also aid downstream statistical analyses through effective dimension reduction.

Recently, there has been a surge of interest in interpretable machine learning, see Doshi-Velez and Kim (2017), Rudin (2019), and Murdoch et al. (2019), among others. Latent variable approaches, however, have received limited attention in this emerging literature, likely due to the associated complexities. Indeed, deep learning models with many layers of latent variables are

Received: January 24, 2021. **Revised:** January 25, 2023. **Accepted:** January 29, 2023

usually considered as uninterpretable black boxes. For example, the deep belief network (Hinton et al., 2006; Lee et al., 2009) is a very popular deep learning architecture, but it is generally not reliable to interpret the inferred latent structure. However, for high-dimensional data, it is highly desirable to perform dimension reduction to extract the key signals in the form of lower-dimensional latent representations. If the latent representations are themselves reliable, then they can be viewed as surrogate features of the data and then passed along to existing interpretable machine learning methods for downstream tasks. A key to success in such modeling and analysis processes is the interpretability of the latent structure. This in turn relies crucially on the identifiability of the statistical latent variable model being used.

In statistical terms, a set of parameters for a family of statistical models are said to be identifiable if distinct values of the parameters correspond to distinct distributions of the observed data. Studies under such an identifiability notion date back to Koopmans and Reiersol (1950), Teicher (1961), and Goodman (1974). Model identifiability is a fundamental prerequisite for valid statistical estimation and inference. In the latent variable context, if one wishes to interpret the parameters and latent representations learned using a latent variable model, then identifiability is necessary for making such interpretation meaningful and reproducible. Early considerations of identifiability in the latent variable context can be traced to the seminal work of Anderson and Rubin (1956) for traditional factor analysis. For modern latent variable models potentially containing nonlinear, non-continuous, and even deep layers of latent variables, identifiability issues can be challenging to address.

Recent developments on the identifiability of continuous latent variable models include Drton et al. (2011), Anandkumar et al. (2013), and Chen, Li, et al. (2020). Discrete latent variable models are an appealing alternative to their continuous counterparts in terms of the combination of interpretability and flexibility. Finite mixture models (McLachlan & Basford, 1988) routinely used for model-based clustering are a canonical example involving a single discrete latent variable. Such relatively simple approaches are insufficiently flexible for complex data sets. Extensions with multiple latent variables and/or multilayer structure have distinct advantages in such settings but come with increasingly complex identifiability issues. In this work, we are motivated to build identifiable deep latent variable models, which are flexible enough to capture the complex dependencies in real-world data, yet also with appropriate restrictions and parsimony to yield identifiability.

We propose a family of multilayer, potentially deep, discrete latent variable models and propose novel identifiability conditions for them. We establish identifiability for hierarchical latent structures organized in a 'pyramid'—shaped Bayesian network. In such a *Bayesian Pyramid*, observed variables are at the bottom and multilayer latent variables above them describe the data generating process. Sparse graphical connections occur between layers, and our identifiability conditions impose structural and size constraints on these between-layer graphs. Technically, we tackle identifiability by first reformulating the Bayesian Pyramid as a *constrained* latent class model (LCM; Goodman, 1974) in a layerwise manner. Then, we derive a nontrivial algebraic property of LCMs under such parameter constraints (Proposition 2) and combine it with Kruskal's theorem (Allman et al., 2009; Kruskal, 1977) on tensor decompositions to establish identifiability. Our identifiability results are not only technically novel, but also provide insights into methodology development. Indeed, the identifiability theory directly inspires the specification of deep latent architecture in Bayesian Pyramids, which features fewer latent variables deeper up the hierarchy. The identifiability results offer a theoretical basis for learning potentially deep and interpretable latent structures from high-dimensional discrete data.

A nice consequence of the identifiability results is the posterior consistency of Bayesian procedures under suitable priors. As an illustration, we consider the two-latent-layer model and propose a Bayesian shrinkage estimation approach. We develop a Gibbs sampler with data augmentation for computation. Simulation studies corroborate identifiability and show good performance of Bayes estimators. We apply the proposed approach to two DNA nucleotide sequence datasets (Dua & Graff, 2017). For the splice junction data, when using latent representations learned from our two-latent-layer model in downstream classification of nucleotide sequence types, we achieve a remarkable accuracy comparable to fine-tuned convolutional neural networks (i.e., in Nguyen et al., 2016). This suggests that the developed recipe of unsupervised learning of discrete data may serve as a useful alternative to popular machine learning methods.

The rest of this paper is organized as follows. Section 2 proposes Bayesian Pyramids, a new family of pyramid-shaped deep latent variable models for discrete data and reformulates a Bayesian Pyramid into constrained latent class models (CLCMs) in a layerwise manner. Section 3 first considers the identifiability of the general CLCMs and then proposes identifiability conditions for the multilayer deep Bayesian Pyramids. To illustrate the proposed framework, Section 4 focuses on a two-latent-layer Bayesian Pyramid and proposes a Bayesian estimation approach. Section 5 provides simulation studies that examine the performance of the proposed methodology and corroborate the identifiability theory. Section 6 applies the method to real data on nucleotide sequences. Finally, Section 7 discusses implications and future directions. Technical proofs, posterior computation details, and additional data analyses are included in the Online Supplementary Material. Matlab code implementing the proposed method is available at https://github.com/yuqigu/BayesianPyramids.

2 Bayesian Pyramids: multilayer latent structure models

This section proposes Bayesian Pyramids to model the joint distribution of multivariate unordered categorical data. For an integer m, denote $[m] = \{1, 2, \ldots, m\}$. Suppose for each subject, there are p observed variables $\boldsymbol{y} = (y_1, \ldots, y_p)^\top$, where $y_j \in [d_j]$ for each variable $j \in [p]$. d_j is the number of categories that the jth observed variable can potentially take. We mainly consider multiple (potentially deep) layers of *binary* latent variables, motivated by better computational tractability and also the simpler interpretability, with each variable encoding presence or absence of a certain latent construct. The stack of multiple layers of binary latent variables induces a model resembling deep belief networks (Hinton et al., 2006). However, our proposed class of models is more general in terms of distributional assumptions. In the following, we first describe in detail our proposed Bayesian Pyramids in Section 2.1 and then connect them to a latent class model (Goodman, 1974) subject to certain constraints.

2.1 Multilayer Bayesian Pyramids

The proposed models belong to the broader family of Bayesian networks (Pearl, 2014), which are directed acyclic graphical models that can encode rich conditional independence information. We propose a 'pyramid'-like Bayesian network with one latent variable at the root and more and more latent variables in downward layers, where the bottom layer consists of the p observed variables $y_1, \ldots, y_p$. Denote the number of latent layers in this Bayesian network by D, which can be viewed as the depth. Specifically, let the layer of latent variables consecutive to the observed $\boldsymbol{y}$ be $\boldsymbol{\alpha}^{(1)} = (\alpha_1, \ldots, \alpha_{K_1}) \in \{0, 1\}^{K_1}$ with K_1 variables, and let a deeper layer of latent variables consecutive to $\boldsymbol{\alpha}^{(m)} \in \{0, 1\}^{K_m}$ be $\boldsymbol{\alpha}^{(m+1)} \in \{0, 1\}^{K_{m+1}}$ for $m = 1, 2, \ldots, D-1$. Finally, at the top and the deepest layer of the pyramid, we specify a single discrete latent variable z, or equivalently $z^{(D)} \in \{1, \ldots, B\}$. In this Bayesian network, all the directed edges are pointing in the top-down direction only between two consecutive layers, and there are no edges between variables within a layer. This gives the following factorization of the joint distribution of $\boldsymbol{y}$ and latent variables, where the subscript i denotes an index of a random subject,

$$\mathbb{P}(\boldsymbol{y}_i, \{\boldsymbol{\alpha}_i^{(m)}\}, z_i^{(D)}) = \mathbb{P}(\boldsymbol{y}_i \mid \boldsymbol{\alpha}_i^{(1)}) \prod_{m=1}^{D-2} \mathbb{P}(\boldsymbol{\alpha}_i^{(m)} \mid \boldsymbol{\alpha}_i^{(m+1)}) \mathbb{P}(\boldsymbol{\alpha}_i^{(D-1)} \mid z_i^{(D)}) \mathbb{P}(z_i^{(D)}), \tag{1}$$

$$\mathbb{P}(\boldsymbol{y}_i \mid \boldsymbol{\alpha}_i^{(1)}) = \prod_{j=1}^{p} \mathbb{P}(y_{i,j} \mid \boldsymbol{\alpha}_{i,\mathrm{pa}(j)}^{(1)}), \quad \mathbb{P}(\boldsymbol{\alpha}_i^{(m)} \mid \boldsymbol{\alpha}_i^{(m+1)}) = \prod_{k=1}^{K_m} \mathbb{P}(\alpha_{i,k}^{(m)} \mid \boldsymbol{\alpha}_{i,\mathrm{pa}(k^{(m)})}^{(m+1)}). \tag{2}$$

In the above display, for each $j \in [p]$, $\mathrm{pa}(j) \subseteq [K_1]$ is a collection of first-latent-layer variables, which are parents of y_j. Similarly, for each $k \in [K_m]$, $\mathrm{pa}(k^{(m)}) \subseteq [K_{m+1}]$ is a collection of $(m+1)$-latent-layer variables, which are parents of the m-latent-layer variable α_k.

Figure 1 gives a graphical visualization of the proposed model. To make clear the sparsity structure of this graph, we introduce binary matrices $\mathbf{G}^{(1)}, \mathbf{G}^{(2)}, \ldots, \mathbf{G}^{(D-1)}$, termed as *graphical matrices*, to encode the connecting patterns between consecutive layers. That is, $\mathbf{G}^{(d)}$ summarizes the

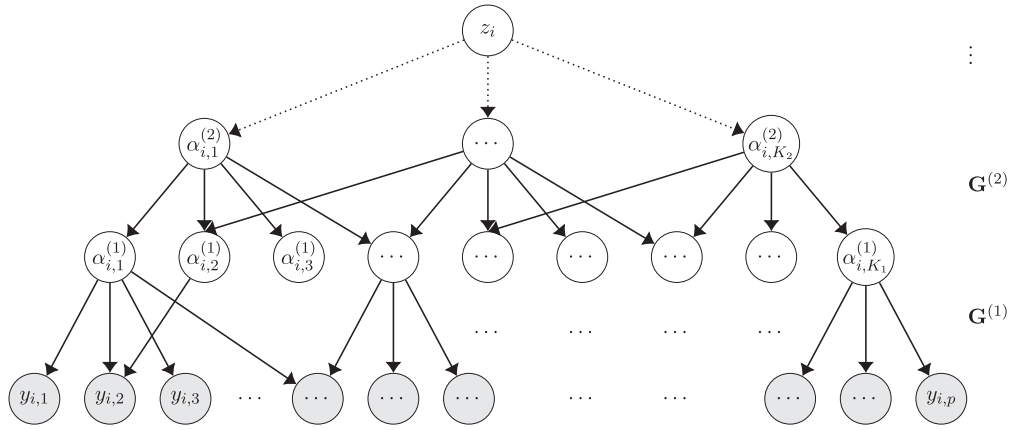

Figure 1. Multiple layers of binary latent traits $\boldsymbol{\alpha}_i^{(d)}$s model the distribution of observed $\boldsymbol{y}_i$. Binary matrices $\mathbf{G}^{(1)}$, $\mathbf{G}^{(2)}$, ... encode the sparse connection patterns between consecutive layers. Dotted arrows emanating from the root variable z_i summarize omitted layers $\{\boldsymbol{\alpha}_i^{(d)}\}$.

parent–child graphical patterns between the two consecutive layers d and $d+1$. Specifically, matrix $\mathbf{G}^{(1)} = (g_{j,k}^{(1)})$ has size $p \times K_1$ and matrix $\mathbf{G}^{(m)} = (g_{k,k'}^{(m)})$ has size $K_{m-1} \times K_m$ for $m = 2, \ldots, D-1$, with entries

$$g_{j,k}^{(1)} = 1, \quad \text{if } \alpha_k^{(1)} \text{ is a parent of } y_j,$$
$$g_{k,k'}^{(m)} = 1, \quad \text{if } \alpha_k^{(m)} \text{ is a parent of } \alpha_{k'}^{(m-1)}, \ 2 \le m \le D-1.$$

Each variable in the graph is subject-specific, implying that all the *circles* in Figure 1 represent subject-specific quantities. Namely, if there are n subjects in the sample, each of them would have its own realizations of $\boldsymbol{y}$ and $\boldsymbol{\alpha}^{(d)}$s. The proposed directed acyclic graph is not necessarily a tree, as shown in Figure 1. That is, each variable can have multiple parent variables in the layer above it, while in a directed tree, each variable can only have one parent. As a simpler illustration, we also provide a two-latent-layer Bayesian Pyramid in Figure 2, which features a simpler yet still quite expressive architecture. For example, we can specify the conditional distribution of each observed y_j using a multinomial or binomial logistic model with its parent variables as linear predictors and specify the distribution of $\boldsymbol{\alpha}$ given z using a latent class model. Namely, for a random subject i,

$$\mathbb{P}(y_{i,j} = c \mid \boldsymbol{\alpha}_i^{(1)} = \boldsymbol{\alpha}) = \frac{\exp\left(\beta_{j,c,0} + \sum_{k=1}^{K_1} \beta_{j,c,k} g_{j,k}^{(1)} \alpha_k\right)}{\sum_{m=1}^{d_j} \exp\left(\beta_{j,m,0} + \sum_{k=1}^{K_1} \beta_{j,m,k} g_{j,k}^{(1)} \alpha_k\right)}, \quad j \in [p], \ c \in [d_j], \ \boldsymbol{\alpha} \in \{0,1\}^{K_1},$$
$$\mathbb{P}(\boldsymbol{\alpha}_i^{(1)} = \boldsymbol{\alpha}) = \sum_{b=1}^{B} \mathbb{P}(z_i = b)\mathbb{P}(\boldsymbol{\alpha}_i^{(1)} = \boldsymbol{\alpha} \mid z_i = b) = \sum_{b=1}^{B} \tau_b \prod_{k=1}^{K_1} \eta_{k,b}^{\alpha_k} (1 - \eta_{k,b})^{1-\alpha_k}. \tag{3}$$

Later in Section 4, we will focus on the two-latent-layer model when describing the estimation methodology; please refer to that part for further details.

We provide some discussions on the conceptual motivation for the proposed multilayer model. Intuitively, for each subject i, $z_i \in [B]$ in the deepest layer of the pyramid can encode some coarse-grained latent categories of subjects, while the vector $\boldsymbol{\alpha}_i^{(1)} \in \{0,1\}^{K_1}$ encodes more fine-grained latent features of subjects. The hierarchical multilayer structure is conceptually appealing as it can provide latent representations of data in *multiple resolutions*, where there are nonlinear compositions between different latent resolutions that boost model flexibility. Model (3) for generating

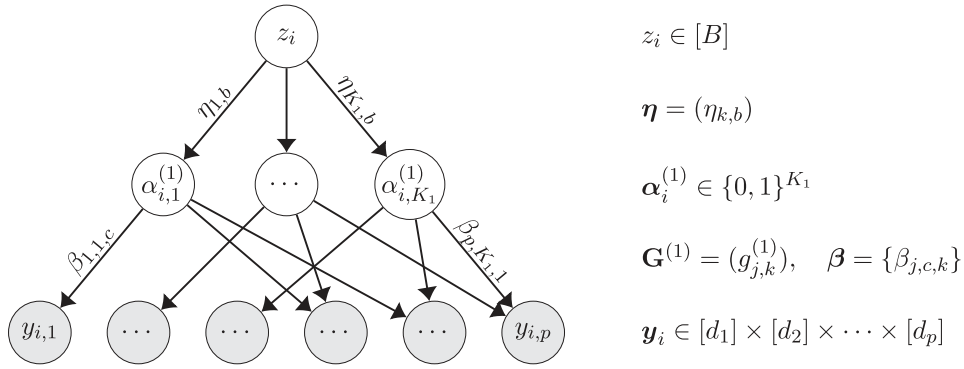

Figure 2. Two-latent-layer model. Latent variables z_i, $\alpha_{i,1}, \ldots, \alpha_{i,K_1}$ and observed variables $y_{i,1}, \ldots, y_{i,p}$ are subject-specific, and model parameters $\mathbf{G}^{(1)}$, $\boldsymbol{\beta}$, $\boldsymbol{\tau}$, $\boldsymbol{\eta}$ are population quantities.

$\boldsymbol{\alpha}^{(1)}$ given z is a nonlinear latent class model (LCM). Hence, z cannot be viewed as a linear projection of the $\boldsymbol{\alpha}^{(1)}$-layer; rather, the nonlinear LCM (with a deeper variable z and conditional independence of α_k's given z) can encode and explain rich and complex dependencies between the variables $\alpha^{(1)}, \ldots, \alpha^{(1)}_{K-1}$ using parsimonious parametrizations.

2.2 Reformulating the Bayesian Pyramid as a constrained latent class model

In this subsection, we reveal an interesting equivalence relationship between our Bayesian Pyramid models and latent class models (LCMs, Goodman, 1974) with certain equality constraints. Such equivalence will pave the way for investigating the identifiability of Bayesian Pyramids. The traditional latent class model (LCM) (Goodman, 1974; Lazarsfeld, 1950) posits the following joint distribution of $\boldsymbol{y}_i$ for a random subject i:

$$\mathbb{P}(y_{i,1} = c_1, \ldots, y_{i,p} = c_p) = \sum_{h=1}^{H} v_h \prod_{j=1}^{p} \lambda^{(j)}_{c_j,h}, \quad \forall c_j \in [d_j],\ j \in [p]. \tag{4}$$

The above equation specifies a finite mixture model with one discrete latent variable having H categories (i.e., H latent classes). In particular, given a latent variable $z \in [H]$, $v_h = \mathbb{P}(z = h)$ denotes the probability of z belonging to the hth latent class, and $\lambda^{(j)}_{c_j,h} = \mathbb{P}(y_j = c_j \mid z_j = h)$ denotes the conditional probability of response c_j for variable y_j given the latent class membership h. Expression (4) implies that p observed variables $y_1, \ldots, y_p$ are conditionally independent given the latent z. In the Bayesian literature, the model in Dunson and Xing (2009) is a nonparametric generalization of (4), which allows an infinite number of latent classes.

Latent class modeling under (4) is widely used in social and biomedical sciences, where researchers often hope to infer subpopulations of individuals having different profiles (Collins & Lanza, 2009). However, the overly-simplistic form of (4) can lead to poor performance in inferring distinct and interpretable subpopulations. In particular, the model assumes that individuals in different subpopulations have completely different probabilities $\lambda^{(j)}_{c_j,h}$ for all $c_j \in [d_j]$ and $j \in [p]$, and conditionally on subpopulation membership all the variables are independent. These restrictions can force the number of classes to increase in order to provide an adequate fit to the data, which can degrade interpretability of a plain latent class model.

We introduce some notation before proceeding. Denote a $p \times H$ all-one matrix by $\mathbf{1}_{p\times H}$. For a matrix $\mathbf{S}$ with p rows and a set $\mathcal{A} \subseteq [p]$, denote by $\mathbf{S}_{\mathcal{A},:}$ the submatrix of $\mathbf{S}$ consisting of rows indexed in $\mathcal{A}$. Consider a family of constrained latent class models, which enable learning of a potentially large number of interpretable, identifiable, and diverse latent classes. A key idea is sharing of parameters within certain latent classes for each observed variable. We introduce a binary *constraint matrix* $\mathbf{S} = (S_{j,h})$ of size $p \times H$, which has rows indexed by the p observed variables and columns indexed by the H latent classes. The binary entry $S_{j,h}$ indicates whether the conditional

probability table $\lambda^{(j)}_{1:d_j,h} = (\lambda^{(j)}_{1,h}, \ldots, \lambda^{(j)}_{d_j,h})$ is free or instead equal to some unknown baseline. Specifically, if $S_{j,h} = 1$ then $\lambda^{(j)}_{1:d_j,h}$ is free; while for those latent classes $h \in [H]$ such that $S_{j,h} = 0$, their conditional probability tables $\lambda^{(j)}_{1:d_j,h}$'s are constrained to be equal. Hence, $\mathbf{S}$ puts the following equality constraints on the parameters of (4):

$$\text{if } S_{j,h_1} = S_{j,h_2} = 0, \quad \text{then } \lambda^{(j)}_{1:d_j,h_1} = \lambda^{(j)}_{1:d_j,h_2}. \tag{5}$$

We also enforce a natural inequality constraint for identifiability,

$$\text{if } S_{j,h_1} \neq S_{j,h_2}, \quad \text{then } \lambda^{(j)}_{c_j,h_1} \neq \lambda^{(j)}_{c_j,h_2}. \tag{6}$$

If $\mathbf{S} = \mathbf{1}_{p\times H}$, then there are no active constraints and the original latent class model (4) is recovered. We generally denote the conditional probability parameters by $\boldsymbol{\Lambda} = \{\lambda^{(j)}_{c_j,h};\ j \in [p], c_j \in [d_j], h \in [H]\}$ where $\lambda_{c_j,h} := \lambda_{c_j,0}$ if $S_{j,h} = 0$. As will be revealed soon, such constrained latent class models are related to our proposed Bayesian Pyramids via a neat mathematical transformation.

Viewed from a different perspective, a latent class model (4) specifies a decomposition of the p-way probability tensor $\boldsymbol{\Pi} = (\pi_{c_1\ldots c_p})$, where $\pi_{c_1\cdots c_p} = \mathbb{P}(y_1 = c_1, \ldots, y_p = c_p \mid \boldsymbol{\Lambda}, \mathbf{S}, \boldsymbol{\nu})$. This corresponds to the PARAFAC/CANDECOMP (CP) decomposition in the tensor literature (Kolda & Bader, 2009), which can be used to factorize general real-valued tensors, while our focus is on probability tensors. The proposed equality constraint (5) induces a family of constrained CP decompositions.

This family is connected with the sparse CP decomposition of Zhou et al. (2015), with both having equality constraints summarized by a $p \times H$ binary matrix. However, Zhou et al. (2015) encourage different observed variables to share parameters, while our proposed model encourages different latent classes to share parameters through (5).

We have the following proposition linking the proposed multilayer Bayesian Pyramid to the constrained latent class model under equality constraint (5). For two vectors $\boldsymbol{a}, \boldsymbol{b}$ of the same length M, denote $\boldsymbol{a} \succeq \boldsymbol{b}$ if $a_i \geq b_i$ for all $i \in [M]$. Denote by $\mathbb{1}(\cdot)$ the binary indicator function, which equals one if the statement inside is true and zero otherwise.

Proposition 1 Consider the multilayer Bayesian Pyramid with binary graphical matrices $\mathbf{G}^{(1)}, \mathbf{G}^{(2)}, \ldots, \mathbf{G}^{(D-1)}$.

(a) In marginalizing out all the latent variables **except** $\boldsymbol{\alpha}^{(1)}$, the distribution of $\boldsymbol{y}$ is a constrained latent class model with 2^{K_1} latent classes, where each latent class can be characterized as one configuration of the K_1-dimensional binary vector $\boldsymbol{\alpha}^{(1)} \in \{0, 1\}^{K_1}$. The corresponding $p \times 2^{K_1}$ constraint matrix $\mathbf{S}^{(1)}$ is determined by the bipartite graph structure between the $\boldsymbol{\alpha}^{(1)}$-layer and the $\boldsymbol{y}$-layer, with entries being

$$\begin{aligned} S^{(1)}_{j,\boldsymbol{\alpha}^{(1)}} &= 1 - \mathbb{1}(\alpha^{(1)}_k \geq \mathbf{G}^{(1)}_{j,k} \text{ for all } k = 1, \ldots, K_1) \\ &= 1 - \mathbb{1}(\boldsymbol{\alpha}^{(1)} \succeq \mathbf{G}^{(1)}_{j,:}), \quad j \in [p],\ \boldsymbol{\alpha}^{(1)} \in \{0, 1\}^{K_1}. \end{aligned} \tag{7}$$

(b) Further, in considering the distribution of $\boldsymbol{\alpha}^{(m)} \in \{0, 1\}^{K_m}$ and marginalizing out all the latent variables deeper than $\boldsymbol{\alpha}^{(m)}$ **except** $\boldsymbol{\alpha}^{(m+1)}$, the distribution of $\boldsymbol{\alpha}^{(m)}$ is also a constrained latent class model with 2^{K_m+1} latent classes, where each latent class is characterized as one configuration of the K_{m+1}-dimensional binary vector $\boldsymbol{\alpha}^{(m+1)} \in \{0, 1\}^{K_{m+1}}$. Its corresponding constraint matrix $\mathbf{S}^{(m)}$ is determined by the bipartite

graph structure between the mth and the $(m+1)$th latent layers, with entries being

$$S^{(m+1)}_{k,\boldsymbol{\alpha}^{(m+1)}} = 1 - \mathbb{1}(\boldsymbol{\alpha}^{(m+1)} \succeq \mathbf{G}^{(m+1)}_{k,:}), \quad k \in [K_m], \ \boldsymbol{\alpha}^{(m+1)} \in \{0,1\}^{K_{m+1}}. \quad (8)$$

We present a toy example to illustrate Proposition 1 and discuss its implications.

Example 1 Consider a multilayer Bayesian Pyramid with $p = 6$ and $K_1 = 3$, with the graph between $\boldsymbol{\alpha}^{(1)}$ and $\boldsymbol{y}$ displayed in Figure 2. The 6×3 graphical matrix $\mathbf{G}^{(1)}$ is presented below. Proposition 1(a) states that there is a $p \times 2^{K_1}$ constraint matrix $\mathbf{S}^{(1)}$ taking the form

$$\mathbf{G}^{(1)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix},$$

$$\mathbf{S}^{(1)} = \begin{pmatrix} (000) & (100) & (010) & (001) & (110) & (101) & (011) & (111) \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

Entries of $\mathbf{S}^{(1)}$ are determined according to (7), for example, $S^{(1)}_{1,(000)} = 1 - \mathbb{1}((000) \succeq \mathbf{G}^{(1)}_{1,:}) = 1 - \mathbb{1}((000) \succeq (100)) = 1$; and $S^{(1)}_{6,(111)} = 1 - \mathbb{1}((111) \succeq \mathbf{G}^{(1)}_{6,:}) = 1 - \mathbb{1}((111) \succeq (011)) = 0$. Each column of the constraint matrix $\mathbf{S}^{(1)}$ is indexed by a latent class characterized by a configuration of a K_1-dimensional binary vector. This implies that if only considering the first latent layer of variables $\boldsymbol{\alpha}^{(1)}$, all the subjects are naturally divided into 2^{K_1} latent classes, each endowed with a binary pattern.

Proposition 1 gives a nice structural characterization of multilayer Bayesian Pyramids. This characterization is achieved by relating the multilayer sparse graph to the constrained latent class model in a layer-wise manner. This proposition provides a basis for investigating he identifiability of multilayer Bayesian Pyramids; see details in Section 3.

2.3 Connections to existing models and studies

We next briefly review connections between Bayesian Pyramids and existing models. In educational measurement research, Haertel (1989) first used the term restricted latent class models. Further developments along this line in the psychometrics literature led to a popular family of cognitive diagnosis models (de la Torre, 2011; Rupp & Templin, 2008; von Davier & Lee, 2019). These models are essentially binary latent skill models where each subject is endowed with a K-dimensional latent skill vector $\boldsymbol{\alpha} \in \{0,1\}^K$ indicating the mastery/deficiency statuses of K skills, and each test item j is designed to measure a certain configuration of skills, summarized by a loading vector $\boldsymbol{q}_j \in \{0,1\}^K$. The matrix $\mathbf{Q} \in \{0,1\}^{J \times K}$ collecting all of the J skill loading vectors $\boldsymbol{q}_1, \ldots, \boldsymbol{q}_J$ as row vectors is often pre-specified by educational experts. The observed data for each subject are a J-dimensional binary vector $\boldsymbol{r} \in \{0,1\}^J$, indicating the correct/wrong answers to J test questions in the assessment. Recently, there have been emerging studies on the identifiability and estimation of such cognitive diagnosis models (e.g., Chen, Culpepper, et al., 2020; Chen et al., 2015; Fang et al., 2019; Gu & Xu, 2019, 2020; Xu, 2017). However, to our knowledge,

there have not been works that model multilayer (i.e., deep) latent structure behind the data and investigate identifiability in such scenarios.

Bayesian Pyramids are also related to deep belief networks (DBNs, Hinton et al., 2006), sum-product networks (SPNs, Poon & Domingos, 2011), and latent tree graphical models (LTMs, Mourad et al., 2013) in the machine learning literature. DBNs have undirected edges between the deepest two layers designed based on computational considerations (Hinton, 2009), while a Bayesian Pyramid is a fully generative directed graphical model (Bayesian network) with all the edges pointing top down. Such generative modeling is naturally motivated by the identifiability considerations and also provides a full description of the data generating process. Also, DBNs in their popular form are models for multivariate binary data, feature a fully connected graph structure, and use logistic link functions between layers. In contrast, Bayesian Pyramids accommodate general multivariate categorical data and allow flexible forms of layerwise conditional distributions and sparse connections between consecutive layers. An SPN is a rooted directed acyclic graph consisting of sum nodes and product nodes. Zhao et al. (2015) show that an SPN can be converted to a bipartite Bayesian network, where the number of discrete latent variables equals the number of sum nodes in the SPN. Our model is more general in that in addition to modeling a bipartite network between the latent layer and the observed layer, we further model the dependence of the latent variables instead of assuming them independent. LTMs are a special case of our model (3) because, while all the variables in an LTM form a tree, we allow for more general DAGs beyond trees. Although the above models are extremely popular, identifiability has received essentially no attention; an exception is Zwiernik (2018), which discussed identifiability for relatively simple LTMs.

Our two-latent-layer Bayesian Pyramid shares a similar structure with the nonparametric latent feature models in Doshi-Velez and Ghahramani (2009). Both works consider a mixture of binary latent feature models, with each data point associated with both a deep latent cluster (z in our notation) and a binary vector of latent features [$\boldsymbol{\alpha}^{(1)}$ in our notation]. One distinction is that we adopt very flexible probabilistic distributions (see Examples 2 and 3) as conditional distributions in the DAG, while Doshi-Velez and Ghahramani (2009) posits that the latent features are a deterministic function of the products of z and $\boldsymbol{\alpha}^{(1)}$. In addition, Bayesian Pyramids are directly inspired by identifiability considerations, and in the next section, we provide explicit conditions that guarantee the model is identifiable—not only in the two-latent-layer architecture similar to Doshi-Velez and Ghahramani (2009) but also in general deep latent hierarchies with $\boldsymbol{\alpha}^{(2)}, \ldots, \boldsymbol{\alpha}^{(D-1)}, z$. Interestingly, Doshi-Velez and Ghahramani (2009) conjecture heuristically that their specification 'likely resolves the identifiability issues'.

3 Identifiability and constrained latent class structure behind Bayesian Pyramids

3.1 Identifiability of the constrained latent class model and posterior consistency

In Section 2, we proposed a new class of multilayer latent variable models deemed Bayesian Pyramids and showed that these models can be formulated as a type of constrained latent class model (CLCM) defined in (4)–(5). In this section, we study theoretical properties of model (4)–(5).

The classic latent class model in (4) was shown by Gyllenberg et al. (1994) to be not strictly identifiable. Strict identifiability generally requires one to establish a parameterization in which the parameters can be expressed as one-to-one functions of observables. As a weaker notion, generic identifiability requires that the map is one-to-one except on a Lesbesgue measure zero subset of the parameter space. In a seminal paper, Allman et al. (2009) leveraged Kruskal's theorem (Kruskal, 1977) to show generic identifiability for an unconstrained latent class model. However, Allman et al. (2009)'s approach is not sufficient for establishing identifiability in constrained LCMs or Bayesian Pyramids, due to the complex parameter constraints in these models. Indeed, Allman et al. (2009)'s generic identifiability results imply that in the latent class model with an *unconstrained* parameter space, there exists a measure-zero subset of parameters where identifiability breaks down. In constrained LCMs, the equality constraints (5) exactly enforce parameters to fall into a measure-zero subset. In Bayesian Pyramids, these equality constraints arise from between-layer potentially sparse graphs. So without a careful and thorough investigation into the relationship between the parameter constraints and the graphical structure, Kruskal's theorem is

not directly applicable to investigating the identifiability of constrained LCMs or Bayesian Pyramids.

Below, we establish strict identifiability of model (4)–(5), by carefully examining the algebraic structure imposed by the constraint matrix $\mathbf{S}$. We first introduce some notation. Denote by $\otimes$ the Kronecker product of matrices and by $\odot$ the Khatri–Rao product of matrices. In particular, consider matrices $\mathbf{A} = (a_{i,j}) \in \mathbb{R}^{m\times r}$, $\mathbf{B} = (b_{i,j}) \in \mathbb{R}^{s\times t}$; and matrices $\mathbf{C} = (c_{i,j}) = (\boldsymbol{c}_{:,1} \mid \cdots \mid \boldsymbol{c}_{:,k}) \in \mathbb{R}^{n\times k}$, $\mathbf{D} = (d_{i,j}) = (\boldsymbol{d}_{:,1} \mid \cdots \mid \boldsymbol{d}_{:,k}) \in \mathbb{R}^{\ell\times k}$, then there are $\mathbf{A}\otimes\mathbf{B} \in \mathbb{R}^{ms\times rt}$ and $\mathbf{C}\odot\mathbf{D} \in \mathbb{R}^{n\ell\times k}$ with

$$\mathbf{A}\otimes\mathbf{B} = \begin{pmatrix} a_{1,1}\mathbf{B} & \cdots & a_{1,r}\mathbf{B} \\ \vdots & \vdots & \vdots \\ a_{m,1}\mathbf{B} & \cdots & a_{m,r}\mathbf{B} \end{pmatrix}, \quad \mathbf{C}\odot\mathbf{D} = (\boldsymbol{c}_{:,1}\otimes\boldsymbol{d}_{:,1} \mid \cdots \mid \boldsymbol{c}_{:,k}\otimes\boldsymbol{d}_{:,k}).$$

The above definitions show the Khatri–Rao product is a column-wise Kronecker product; see more in Kolda and Bader (2009). We first establish the following technical proposition, which is useful for the later theorem on identifiability.

Proposition 2 For the constrained latent class model, define the following p parameter matrices subject to constraints (5) and (6) with some constraint matrix $\mathbf{S}$,

$$\mathbf{\Lambda}^{(j)} = \begin{pmatrix} \lambda^{(j)}_{1,1} & \lambda^{(j)}_{1,2} & \cdots & \lambda^{(j)}_{1,H} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda^{(j)}_{d_j,1} & \lambda^{(j)}_{d_j,2} & \cdots & \lambda^{(j)}_{d_j,H} \end{pmatrix}, \quad j = 1, \ldots, p.$$

Denote the Khatri–Rao product of the above p matrices by $\mathbf{K} = \odot_{j=1}^{p}\mathbf{\Lambda}^{(j)}$, which has size $\prod_{j=1}^{p} d_j \times H$. The following two conclusions hold.

(a) If the H column vectors of the constraint matrix $\mathbf{S}$ are distinct, then $\mathbf{K}$ must have full column rank H.
(b) If $\mathbf{S}$ contains identical column vectors, then $\mathbf{K}$ can be rank-deficient.

Remark 1 Proposition 2 implies that $\mathbf{S}$ having distinct columns is sufficient [in part (a)] and almost necessary [in part (b)] for the Khatri–Rao product $\mathbf{K} = \odot_{j=1}^{p}\mathbf{\Lambda}^{(j)}$ to be full rank. To see the 'almost necessary' part, consider a special case where besides constraint (5) that $\lambda^{(j)}_{1:d_j,h_1} = \lambda^{(j)}_{1:d_j,\,h_2}$ if $S_{j,h_1} = S_{j,h_2} = 0$, the parameters also satisfy $\lambda^{(j)}_{1:d_j,h_1} = \lambda^{(j)}_{1:d_j,h_2}$ if $S_{j,h_1} = S_{j,h_2} = 1$. In this case, our proof shows that whenever the binary matrix $\mathbf{S}$ contains identical column vectors in columns h_1 and h_2, the matrix $\mathbf{K}$ also contains identical column vectors in columns h_1 and h_2 and hence is surely rank-deficient.

In the Khatri–Rao product matrix $\mathbf{K}$ defined in Proposition 2, each column characterizes the conditional distribution of vector $\boldsymbol{y}$ given a particular latent class. Therefore, Proposition 2 reveals an nontrivial algebraic property: whether $\mathbf{S} \in \{0, 1\}^{p\times H}$ has distinct column vectors is linked to whether the H conditional distributions of $\boldsymbol{y}$ given each latent class are linearly independent. The matrix $\mathbf{S}$ does not need to have full column rank in order to have distinct column vectors. For example, a 3×3 matrix $\mathbf{S}$ with three columns being $(1, 0, 0)^\top$, $(0, 1, 1)^\top$, and $(1, 1, 1)^\top$ is rank-deficient but has distinct column vectors. Indeed, it is not hard to see that a binary matrix $\mathbf{S}$ with m rows can have as many as 2^m distinct column vectors.

Proposition 1 reformulates a Bayesian Pyramid into a *constrained LCM* with a constraint matrix $\mathbf{S}$, and then Proposition 2 establishes a nontrivial algebraic property of such constrained LCMs. Propositions 1 and 2 together pave the way for the development of the identifiability theory of Bayesian Pyramids. In particular, the sufficiency part of Proposition 2 uncovers a nontrivial inherent algebraic structure of the considered models. To prove Proposition 2, we leveraged a novel

proof technique, the marginal probability matrix, in order to find a sufficient and almost necessary condition for the Khatri–Rao product of conditional probability matrices to have full rank. We anticipate that the conclusion in Proposition 2 can be useful in other graphical models with discrete latent variables, even beyond the Bayesian Pyramids considered in this paper. This is because graphical models involving discrete latent structure can often be formulated as a latent class model with equality constraints determined by the graph. Therefore, the conclusion of Proposition 2 might be of independent interest.

Although the linear independence of $\mathbf{K}$'s columns itself does not lead to identifiability, it provides a basis for investigating strict identifiability of our model. We introduce the definition of strict identifiability under the current setup and then give the strict identifiability result.

Definition 1 (Strict Identifiability). The constrained latent class model with (5) and (6) is said to be strictly identifiable if for any valid parameters $(\mathbf{\Lambda}, \mathbf{S}, \boldsymbol{\nu})$, the following equality holds if and only if $(\overline{\mathbf{\Lambda}}, \overline{\mathbf{S}}, \overline{\boldsymbol{\nu}})$ and $(\mathbf{\Lambda}, \mathbf{S}, \boldsymbol{\nu})$ are identical up to a latent class permutation:

$$\mathbb{P}(\boldsymbol{y} = \boldsymbol{c} \mid \mathbf{\Lambda}, \mathbf{S}, \boldsymbol{\nu}) = \mathbb{P}(\boldsymbol{y} = \boldsymbol{c} \mid \overline{\mathbf{\Lambda}}, \overline{\mathbf{S}}, \overline{\boldsymbol{\nu}}), \quad \forall \boldsymbol{c} \in \times_{j=1}^{p}[d_j]. \tag{9}$$

Remark 2 When the constraint matrix $\mathbf{S}$ is unknown and needs to be identified together with unknown continuous parameters $\mathbf{\Lambda}$, there is a trivial nonidentifiability issue that needs to be resolved. To see this, continue to consider the special case mentioned in Remark 1 where $\lambda^{(j)}_{1:d_j,h_1} = \lambda^{(j)}_{1:d_j,h_2}$ whenever $S_{j,h_1} = S_{j,h_2}$, then given a matrix $\mathbf{S}$ we can generally denote $\lambda^{(j)}_{1:d_j,h} =: \lambda^{(j)}_{1:d_j,+}$ if $S_{j,h} = 1$ and $\lambda^{(j)}_{1:d_j,h} =: \lambda^{(j)}_{1:d_j,-}$ if $S_{j,h} = 0$. Then without further restrictions, the following alternative $(\tilde{\mathbf{S}}, \tilde{\mathbf{\Lambda}})$ will be indistinguishable from the true $(\mathbf{S}, \mathbf{\Lambda})$, where $\tilde{\mathbf{S}} = \mathbf{1}_{p\times H} - \mathbf{S}$, $\tilde{\lambda}_{1:d_j,+} = \lambda^{(j)}_{1:d_j,-}$ and $\tilde{\lambda}_{1:d_j,-} = \lambda^{(j)}_{1:d_j,+}$. One straightforward way to resolve such trivial nonidentifiability of $\mathbf{S}$ is to simply enforce that whenever $s_{j,h_1} > s_{j,h_2}$ the order of λ_{j,c,h_1} and λ_{j,c,h_2} is fixed for every possible category $c \in [d_j]$. In the following studies of identifiability, we always assume such orderings of $\mathbf{\Lambda}$ with respect to $\mathbf{S}$ has been fixed.

For an arbitrary subset $\mathcal{A} \subseteq [p]$, denote by $\mathbf{S}_{\mathcal{A},:}$ the submatrix of $\mathbf{S}$ that consists of those rows indexed by variables belonging to $\mathcal{A}$. $\mathbf{S}_{\mathcal{A},:}$ has size $|\mathcal{A}| \times H$.

Theorem 1 (Strict Identifiability). Consider the proposed constrained latent class model under (5) and (6) with true parameters $\boldsymbol{\nu} = \{\nu_h\}_{h\in[H]}$, $\mathbf{\Lambda} = \{\mathbf{\Lambda}^{(j)}\}_{j\in[p]}$, and $\mathbf{S}$. Suppose there exists a partition of the p variables $[p] = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3$ such that

(a) the submatrix $\mathbf{S}_{\mathcal{A}_i,:}$ has distinct column vectors for $i = 1$ and 2; and
(b) for any $h_1 \neq h_2 \in [H]$, there is $\lambda^{(j)}_{c,h_1} \neq \lambda^{(j)}_{c,h_2}$ for some $j \in \mathcal{A}_3$ and some $c \in [d_j]$.

Also suppose $\nu_h > 0$ for each $h \in [H]$. Then, $(\mathbf{\Lambda}, \mathbf{S}, \boldsymbol{\nu})$ are strictly identifiable up to a latent class permutation.

In the above theorem, the constraint matrix $\mathbf{S}$ is not assumed to be fixed and known. This implies that both the matrix $\mathbf{S}$ and the parameters can be uniquely identified from data. We next give a corollary of Theorem 1, which replaces the condition on parameters $\mathbf{\Lambda}$ in part (b) with a slightly more stringent but also more transparent condition on the binary matrix $\mathbf{S}$.

Corollary 1 Consider the proposed constrained latent class model under (5) and (6). Suppose $\nu_h > 0$ for each $h \in [H]$. If there is a partition of the p variables $[p] = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3$ such that each submatrix $\mathbf{S}_{\mathcal{A}_i,:}$ has distinct column vectors for $i = 1, 2, 3$, then parameters $(\boldsymbol{\nu}, \mathbf{\Lambda}, \mathbf{S})$ are strictly identifiable up to a latent class permutation.

The conditions of Corollary 1 are more easily checkable than those in Theorem 1 because they only depend on the structure of the constraint matrix $\mathbf{S}$. It requires that after some column rearrangement, matrix $\mathbf{S}$ should vertically stack three submatrices, each of which has distinct column vectors.

The conclusions of Theorem 1 and Corollary 1 both regard strict identifiability, which is the strongest possible conclusion on parameter identifiability up to label permutation. If we consider the slightly weaker notion of generic identifiability as proposed in Allman et al. (2009), the conditions in Theorem 1 and Corollary 1 can be relaxed. Given a constraint matrix $\mathbf{S}$, denote the constrained parameter space for $(\boldsymbol{\nu}, \boldsymbol{\Lambda})$ by

$$\mathcal{T}^{\mathbf{S}} = \{(\boldsymbol{\nu}, \boldsymbol{\Lambda}) : \boldsymbol{\Lambda} \text{ satisfies the constraints specified by } \mathbf{S}\}; \tag{10}$$

and define the following subset of $\mathcal{T}_{\mathbf{S}}$ as

$$\begin{aligned}\mathcal{N}^{\mathbf{S}} = \{(\boldsymbol{\nu}, \boldsymbol{\Lambda}) \in \mathcal{T}^{\mathbf{S}} : \exists(\overline{\boldsymbol{\nu}}, \overline{\boldsymbol{\Lambda}}) \text{ satisfying the constraints specified by some } \overline{\mathbf{S}} \\ \text{such that } \mathbb{P}(\boldsymbol{y} \mid \boldsymbol{\nu}, \boldsymbol{\Lambda}) = \mathbb{P}(\boldsymbol{y} \mid \overline{\boldsymbol{\nu}}, \overline{\boldsymbol{\Lambda}})\}.\end{aligned} \tag{11}$$

With the above notation, the generic identifiability of the proposed constrained latent class model is defined as follows.

Definition 2 (Generic Identifiability). Parameters $(\boldsymbol{\Lambda}, \mathbf{S}, \boldsymbol{\nu})$ are said to be generically identifiable if $\mathcal{N}_{\mathbf{S}}$ defined in (11) has measure zero with respect to the Lebesgue measure on $\mathcal{T}_{\mathbf{S}}$ defined in (10).

Theorem 2 (Generic Identifiability). Consider the constrained latent class model under (5) and (6). Suppose for $i = 1, 2$, changing some entries of $S_{\mathcal{A}_i,:}$ from '1' to '0' yields an $\widetilde{\mathbf{S}}_{\mathcal{A}_i,:}$ having distinct columns. Also suppose for any $h_1 \neq h_2 \in [H]$, there is $\lambda^{(j)}_{h_1,c} \neq \lambda^{(j)}_{h_2,c}$ for some $j \in \mathcal{A}_3$ and some $c \in [d_j]$. Then $(\boldsymbol{\Lambda}, \mathbf{S}, \boldsymbol{\nu})$ are generically identifiable.

Remark 3 Note that altering some $S_{j,h}$ from one to zero corresponds to adding one more equality constraint that the distribution of the jth variable given the hth latent class is set to the baseline through (5). Therefore, Theorem 2 intuitively implies that if enforcing more parameters in $\mathcal{T}$ to be equal can give rise to a strictly identifiable model, then the parameters that make the original model unidentifiable only occupy a negligible set in $\mathcal{T}$.

Theorem 2 relaxes the conditions on $\mathbf{S}$ for strict identifiability presented earlier. In particular, here the submatrices $\mathbf{S}_{\mathcal{A}_i,:}$ need not have distinct column vectors; rather, it would suffice if altering some entries of $\mathbf{S}_{\mathcal{A}_i,:}$ from one to zero yield distinct column vectors. As pointed out by Allman et al. (2009), generic identifiability is often sufficient for real data analyses.

So far, we have focused on discussing model identifiability. Next, we show that our identifiability results guarantee Bayesian posterior consistency under suitable priors. Given a sample of size n, denote the observations by $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$, which are n vectors each of dimension p. Recall that under (4), the distribution of the vector $\boldsymbol{y}$ under the considered model can be denoted by a p-way probability tensor $\boldsymbol{\Pi} = (\pi_{c_1 \cdots c_p})$. When adopting a Bayesian approach, one can specify prior distributions for the parameters $\boldsymbol{\Lambda}$, $\mathbf{S}$, and $\boldsymbol{\nu}$, which induce a prior distribution for the probability tensor $\boldsymbol{\Pi}$. Within this context, we are now ready to state the following theorem.

Theorem 3 (Posterior Consistency). Denote the collection of model parameters by $\Theta = (\boldsymbol{\Lambda}, \mathbf{S}, \boldsymbol{\nu})$. Suppose the prior distributions for the parameters $\boldsymbol{\Lambda}$, $\mathbf{S}$, and $\boldsymbol{\nu}$ all have full support around the true values. If the true latent structure $\mathbf{S}^0$ and model parameters $\boldsymbol{\Lambda}^0$ satisfy the proposed strict identifiability conditions

in Theorem 1 or Corollary 1, we have

$$\mathbb{P}(\Theta \in \mathcal{N}_\epsilon^c(\Theta^0) \mid \boldsymbol{y}_1, \ldots, \boldsymbol{y}_n) \to 0 \quad \text{almost surely},$$

where $\mathcal{N}_\epsilon^c(\Theta^0)$ is the complement of an ϵ-neighborhood of the true parameters Θ^0 in the parameter space.

Theorem 3 implies that under an identifiable model and with appropriately specified priors, the posterior distribution places increasing probability in arbitrarily small neighborhoods of the true parameters of the constrained latent class model as sample size increases. These parameters include the mixture proportions and the class specific conditional probabilities.

3.2 Identifiability of multilayer Bayesian Pyramids

According to Proposition 1, for a multilayer Bayesian Pyramid with a $p \times K_1$ binary graphical matrix $\mathbf{G}^{(1)}$ between the two bottom layers, one can follow (7) to construct a $p \times 2^{K_1}$ constraint matrix $\mathbf{S}^{(1)}$ as illustrated in Example 1. We next provide transparent identifiability conditions that directly depend on the binary graphical matrices $\mathbf{G}^{(m)}$s. With the next theorem, one only needs to examine the structure of the between layer connecting graphs to establish identifiability.

Theorem 4 Consider the multilayer latent variable model specified in (1)–(2). Suppose the numbers $K_1, \ldots, K_{D-1}$ are known. Suppose each binary graphical matrix $\mathbf{G}^{(m)}$ of size $K_{m-1} \times K_m$ (size $p \times K_1$ if $m = 1$) takes the following form after some row permutation:

$$\mathbf{G}^{(m)} = \begin{pmatrix} \mathbf{I}_{K_m} \\ \mathbf{I}_{K_m} \\ \mathbf{I}_{K_m} \\ \mathbf{G}^{(m),\star} \end{pmatrix}, \quad m = 1, \ldots, D-1, \tag{12}$$

where $\mathbf{G}^{(m),\star}$ generally denotes a submatrix of $\mathbf{G}^{(m)}$ that can take an arbitrary form. Further suppose that the conditional distributions of variables satisfy the inequality constraint in (6). Then, the following parameters are identifiable up to a latent variable permutation within each layer: the probability distribution tensor $\boldsymbol{\tau}$ for the deepest latent variable $z^{(D)}$, the conditional probability table of each variable (including observed and latent) given its parents, and also the binary graphical matrices $\{\mathbf{G}^{(m)}; m = 1, \ldots, D-1\}$.

Remark 4 The proof of Theorem 4 provides a nice layer-wise argument on identifiability, that is, one can examine the structure of the Bayesian Pyramid in the bottom-up direction. As long as for some ℓ there are $\mathbf{G}^{(1)}, \ldots, \mathbf{G}^{(\ell)}$ taking the form of (12), then the parameters associated with the conditional distribution of $\boldsymbol{y}$ and $\boldsymbol{\alpha}^{(1)}, \ldots, \boldsymbol{\alpha}^{(\ell-1)}$ are identifiable and the marginal distributions of $\boldsymbol{\alpha}^{(\ell)}$ are also identifiable.

Theorem 4 implies a requirement that $p \geq 3K_1$ and that $K_{m-1} \geq 3K_m$ for every $m = 2, \ldots, D-1$, through the form of $\mathbf{G}^{(m)}$ in (12). That is, the number of latent variables per layer decreases as the layer goes deeper up the pyramid. Condition (12) in Theorem 4 requires that each latent variable $\alpha_k^{(m)}$ in the mth latent layer has at least three children in the $(m+1)$th layer that do not have any other parents. Our identifiability conditions hold regardless of the specific models chosen for the conditional distributions, $y_j \mid \alpha_k^{(1)}$ and each $\alpha_k^{(m)} \mid \alpha_{k'}^{(m+1)}$, as long as the graphical structure is enforced and these component models are not over-parameterized in a naive manner. We next give two concrete examples which differently model the distribution of $\boldsymbol{\alpha}^{(m)} \mid \boldsymbol{\alpha}^{(m+1)}$ but both respect the graph given by $\mathbf{G}^{(m+1)}$.

Example 2 We first consider modeling the effects of the parent variables of each $\alpha_k^{(m)}$ as

$$\mathbb{P}(\alpha_k^{(m)} = 1 \mid \boldsymbol{\alpha}^{(m+1)}, \boldsymbol{\beta}^{(m+1)}, \mathbf{G}^{(m+1)}) = f\left(\beta_{k,0}^{(m+1)} + \sum_{k':g_{k,k'}^{(m+1)}=1} \beta_{k,k'}^{(m+1)} \alpha_{k'}^{(m+1)}\right), \tag{13}$$

where $f : \mathbb{R} \to (0, 1)$ is a link function. The number of β-parameters in (13) equals $\sum_{k'=1}^{K_{m+1}} g_{k,k'}^{(m+1)}$, which is the number of edges pointing to $\alpha_k^{(m)}$. Choosing $f(x) = 1/(1 + \exp(-x))$ leads to a model similar to sparse deep belief networks (Hinton et al., 2006; Lee et al., 2007).

Example 3 To obtain a more parsimonious alternative to (13), let

$$\mathbb{P}(\alpha_k^{(m)} = 1 \mid \boldsymbol{\alpha}^{(m+1)}, \boldsymbol{\theta}^{(m+1)}, \mathbf{G}^{(m+1)}) = \begin{cases} \theta_{k,1}^{(m+1)}, & \text{if } \mathbb{1}(\alpha_{k'}^{(m+1)} = g_{k,k'}^{(m+1)} = 1 \text{ for at least one } k' \in [K_{m+1}]); \\ \theta_{k,0}^{(m+1)}, & \text{otherwise.} \end{cases} \tag{14}$$

Model (14) satisfies the conditional independence encoded by $\mathbf{G}^{(m+1)}$, since $I(\alpha_{k'}^{(m+1)} = g_{k,k'}^{(m+1)} = 1 \text{ for at least one } k' \in [K_{m+1}]) \equiv \prod_{k':g_{k,k'}^{(m+1)}=1} (1 - \alpha_{k'}^{(m+1)})$, implying that the distribution of $\alpha_k^{(m)}$ only depends on its parents in the $(m+1)$th latent layer. This model provides a probabilistic version of Boolean matrix factorization (Miettinen & Vreeken, 2014). The binary indicator equals the Boolean product of two binary vectors $\mathbf{G}_{k,:}^{(m+1)}$ and $\boldsymbol{\alpha}^{(m+1)}$. The $1 - \theta_{k,1}^{(m+1)}$ and $\theta_{k,0}^{(m+1)}$ quantify the two probabilities that the entry $\alpha_k^{(m)}$ does not equal the Boolean product.

Since Examples 2 and 3 satisfy the conditional independence constraints encoded by graphical matrices $\mathbf{G}^{(m+1)}$s, they satisfy the equality constraint in (5) with the constraint matrix $\mathbf{S}^{(m+1)}$. Therefore, our identifiability conclusion in Theorem 4 applies to both examples with appropriate inequality constraints on the β—parameters or the θ—parameters; for example, see Proposition 3 in Section 4.

Besides Examples 2 and 3, there are many other models that respect the graphical structure. For example, (13) can be extended to include interaction effects of the parents of $\alpha_k^{(m)}$ as follows:

$$\begin{aligned} \mathbb{P}(\alpha_k^{(m)} = 1 \mid \boldsymbol{\alpha}^{(m+1)}, \boldsymbol{\theta}^{(m+1)}, \mathbf{G}^{(m+1)}) &= f\Bigg(\beta_{k,0}^{(m+1)} + \sum_{k':g_{k,k'}^{(m+1)}=1} \beta_{k,k'}^{(m+1)} \alpha_{k'}^{(m+1)} \\ &\quad + \sum_{\substack{k_1 \neq k_2: \\ g_{k,k_1}^{(m+1)} = g_{k,k_2}^{(m+1)} = 1}} \beta_{k,k_1k_2}^{(m+1)} \alpha_{k_1}^{(m+1)} \alpha_{k_2}^{(m+1)} + \cdots + \beta_{k,\text{all}}^{(m+1)} \prod_{\ell:g_{k,\ell}^{(m+1)}=1} \alpha_\ell^{(m+1)}\Bigg). \end{aligned} \tag{15}$$

In (15) if $\alpha_k^{(m)}$ has $M := \sum_{k'=1}^{K_{m+1}} g_{k,k'}^{(m+1)}$ parents, then the number of β-parameters equals 2^M.

Anandkumar et al. (2013) considered the identifiability of linear Bayesian networks. Although both Anandkumar et al. (2013) and this work address identifiability issues of Bayesian networks, their results are not applicable to our highly nonlinear models. The nonlinearity requires techniques that look into the inherent tensor decomposition structures (in particular, a *constrained* CP decomposition) caused by the discrete latent variables and graphical constraints imposed on the discrete latent distribution. Such inherent constrained tensor structures are specific to discrete and graphical latent structures and are not present in the settings considered in Anandkumar et al. (2013).

4 Bayesian inference for two-layer Bayesian Pyramids

As discussed earlier, the proposed multilayer Bayesian Pyramid in Section 2 has universal identifiability arguments for many different model structures. In this section, as an important special case, we focus on a two-latent-layer model with depth $D = 2$ and use a Bayesian approach to infer the latent structure and model parameters. Recall our two-latent-layer Bayesian Pyramid specified earlier in (3) takes the form

$$\mathbb{P}(y_{i,j} = c \mid \boldsymbol{\alpha}_i^{(1)} = \boldsymbol{\alpha}) = \frac{\exp\left(\beta_{j,c,0} + \sum_{k=1}^{K_1} \beta_{j,c,k} g_{j,k}^{(1)} \alpha_k\right)}{\sum_{m=1}^{d_j} \exp\left(\beta_{j,m,0} + \sum_{k=1}^{K_1} \beta_{j,m,k} g_{j,k}^{(1)} \alpha_k\right)}, \quad j \in [p], \ \ c \in [d_j];$$

$$\mathbb{P}(\boldsymbol{\alpha}_i^{(1)} = \boldsymbol{\alpha}) = \sum_{b=1}^{B} \tau_b \prod_{k=1}^{K_1} \eta_{k,b}^{\alpha_k} (1 - \eta_{k,b})^{1-\alpha_k}, \quad \boldsymbol{\alpha} \in \{0, 1\}^{K_1}.$$

Here, we assume $\beta_{j,d_j,0} \equiv \beta_{j,d_j,k} \equiv 0$ for all $k \in [K_1]$, as conventionally done in logistic models. The parameter $\beta_{j,c,k}$ can be viewed as the weight associated with the potential directed edge from latent variable $\alpha_k^{(1)}$ to observed variable y_j, for response category c. The $\beta_{j,c,k}$ only impacts the likelihood if there is an edge from $\alpha_k^{(1)}$ to y_j with $g_{j,k}^{(1)} = 1$. Denote the collection of β parameters as $\boldsymbol{\beta}$. (3) specifies a usual latent class model in the second latent layer with B latent classes for the K_1-dimensional latent vector $\boldsymbol{\alpha}^{(1)}$. This layer of the model has latent class proportion parameters $\boldsymbol{\tau} = (\tau_1, \ \ldots, \tau_B)^\top$ and conditional probability parameters $\boldsymbol{\eta} = (\eta_{k,b})_{K_1 \times B}$. The full set of model parameters across layers is $(\mathbf{G}^{(1)}, \boldsymbol{\beta}, \boldsymbol{\tau}, \boldsymbol{\eta})$, and the model structure is shown in Figure 2. We can denote the conditional probability $\mathbb{P}(y_j = c \mid \boldsymbol{\alpha}^{(1)} = \boldsymbol{\alpha})$ in (3) by $\lambda_{j,c,\boldsymbol{\alpha}}$.

4.1 Identifiability theory adapted to two-layer Bayesian Pyramids defined in (3)

For the two-latent-layer model in (3), the following proposition presents strict identifiability conditions in terms of explicit inequality constraints for the $\boldsymbol{\beta}$ parameters.

Proposition 3 Consider model (3) with true parameters $(\mathbf{G}^{(1)}, \boldsymbol{\beta}, \boldsymbol{\tau}, \boldsymbol{\eta})$.

(a) Suppose $\mathbf{G}^{(1)} = (\mathbf{I}_{K_1}; \mathbf{I}_{K_1}; \mathbf{I}_{K_1}; (\mathbf{G}^\star)^\top)^\top$, $\beta_{j,j,c} \neq 0$, $\beta_{j,j+K_1,c} \neq 0$, and $\beta_{j,j+2K_1,c} \neq 0$ for $j \in [K_1]$, $c \in [d_j - 1]$. Then $\mathbf{G}^{(1)}$, $\boldsymbol{\beta}$, and probability tensor $\boldsymbol{\nu}^{(1)}$ of $\boldsymbol{\alpha} \in \{0, 1\}^K$ are strictly identifiable.

(b) Under the conditions of part (a), if further there is $K_1 \geq 2\log_2 B + 1$, then the parameters $\boldsymbol{\tau}$ and $\boldsymbol{\eta}$ are generically identifiable from $\boldsymbol{\nu}^{(1)}$.

As mentioned in the last paragraph of Section 2, the identifiability results in that section apply to general distributional assumptions for variables organized in a sparse multilayer Bayesian Pyramid. When considering specific models, properties of the model can be leveraged to weaken the identifiability conditions. The next proposition illustrates this, establishing generic identifiability for the model in (3). Before stating the identifiability result, we formally define the allowable

constrained parameter space for $\boldsymbol{\beta}$ under a graphical matrix $\mathbf{G}^{(1)}$ as

$$\Omega(\boldsymbol{\beta}; \mathbf{G}^{(1)}) = \{\beta_{1:p,1:K_1,1:(d_j-1)}; \beta_{j,c,k} \neq 0 \text{ if } g^{(1)}_{j,k} = 1; \text{ and } \beta_{j,c,k} = 0 \text{ if } g^{(1)}_{j,k} = 0\}. \tag{16}$$

Proposition 4 Consider model (3) with $\boldsymbol{\beta}$ belonging to $\Omega(\boldsymbol{\beta}; \mathbf{G}^{(1)})$ in (16). Suppose the graphical matrix $\mathbf{G}^{(1)}$ can be rewritten as $\mathbf{G}^{(1)} = (\mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3, (\mathbf{G}^{\star})^{\top})^{\top}$, where each $\mathbf{G}_m$ has size $K_1 \times K_1$ and

$$\mathbf{G}_m = \begin{pmatrix} 1 & * & \cdots & * \\ * & 1 & \cdots & * \\ \vdots & \vdots & \ddots & \vdots \\ * & * & \cdots & 1 \end{pmatrix}, \quad m = 1, 2, 3;$$

that is, each of $\mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3$ has all the diagonal entries equal to one while any off-diagonal entry is free to be either one or zero. Also suppose $K_1 \geq 2\log_2 B + 1$. Then $(\mathbf{G}^{(1)}, \boldsymbol{\beta}, \boldsymbol{\tau}, \boldsymbol{\eta})$ are generically identifiable.

The two different identifiability conditions on the binary graphical matrix $\mathbf{G}^{(1)}$ stated in Propositions 3 and 4 correspond to different identifiability notions—strict and generic identifiability, respectively. The *generic* identifiability notion is slightly less stringent than *strict* identifiability, by allowing a Lebesgue-measure-zero subset $\mathcal{N}$ of the parameter space $\mathcal{T}$ where identifiability does not hold. Our sufficient generic identifiability conditions in Proposition 4 are much less stringent than conditions in Proposition 3.

4.2 Bayesian inference for the latent sparse graph and number of binary latent traits

We propose a Bayesian inference procedure for two-latent-layer Bayesian Pyramids. We apply a Gibbs sampler by employing the Polya-Gamma data augmentation in Polson et al. (2013) together with the auxiliary variable method for multinomial logit models in Holmes and Held (2006). Such Gibbs sampling steps can handle general multivariate categorical data.

4.2.1 Inference with a Fixed K_1

First consider the case where the true number of binary latent variables K_1 in the middle layer is fixed. Inferring the latent sparse graph $\mathbf{G}^{(1)}$ is equivalent to inferring the sparsity structure of the continuous parameters β_{jkc}'s. Let the prior for $\beta_{j,k,0}$ be $N(\mu_0, \sigma_0^2)$, where hyperparameters μ_0, σ_0^2 can be set to give weakly informative priors for the logistic regression coefficients (Gelman et al., 2008). Specify priors for other parameters as

$$\begin{aligned} \text{(i) for fixed } K_1, k = 1, \ldots, K_1: \quad & \beta_{jck} \mid (\sigma^2_{ck}, g_{j,k} = 1) \sim N(0, \sigma^2_{ck}); \\ & \beta_{jck} \mid (\sigma^2_{ck}, g_{j,k} = 0) \sim N(0, \sigma_0^2); \\ & \sigma^2_{ck} \sim \text{InvGa}(b_{1\sigma}, b_{2\sigma}); \\ & \mathbb{P}(g_{j,k} = 1) = 1 - \mathbb{P}(g_{j,k} = 0) = \gamma; \end{aligned} \tag{17}$$

Here σ_0 is a small positive number specifying the 'pseudo'-variance, and we take $\sigma_0 = 0.1$ in the numerical studies. Our adoption of a prior variance for $\beta_{j,c,k}$ when $g_{j,k} = 0$ follows a similar spirit as the 'pseudo-prior' approach in the Bayesian variable selection literature. In Bayesian variable selection for regression analysis, Dellaportas et al. (2002) first proposed using a pseudo-prior for the variance when one variable is not included in the model to facilitate convenient Gibbs sampling steps. Specifically, a binary variable ρ_j encodes whether or not the jth predictor is included in the regression, and the regression coefficient is $\beta_j\rho_j$, with $\beta_j \sim \pi_j N(0, \sigma^2) + (1 - \pi_j)N(0, \sigma_0^2)$ and

$\mathbb{P}(\rho_j = 1) = \pi_j$. The pseudo-prior variance σ_0^2 does not affect the posterior but may influence mixing of the MCMC algorithm.

The γ in (17) is further given a noninformative prior $\gamma \sim \text{Beta}(1, 1)$. The hyperparameters for the Inverse-Gamma distribution can be set to $b_{1\sigma} = b_{2\sigma} = 2$. In the data augmentation part, for each subject $i \in [n]$, each observed variable $j \in [p]$, and each non-baseline category $c = 1, \ldots, d_j - 1$, we introduce Polya-Gamma random variable $w_{ijc} \sim \text{PG}(1, 0)$. We use the auxiliary variable approach in Holmes and Held (2006) for multinomial regression to derive the conditional distribution of each β_{jck}. Given data $y_{ij} \in [d_j]$, introduce binary indicators $y_{ijc} = \mathbb{1}(y_{ij} = c)$. The posteriors of $\boldsymbol{\beta}$ satisfy that

$$p(\boldsymbol{\beta}_{j,:,:} \mid \boldsymbol{y}_{1:n}) \propto p(\boldsymbol{\beta}_{j,:,:}) \prod_{i=1}^{n} \prod_{c=1}^{d_j - 1} \frac{\left[\exp\left(\beta_{jc0} + \sum_{k=1}^{K_1} \beta_{jck} g_{j,k} \alpha_{i,k}\right)\right]^{y_{ijc}}}{\sum_{c'=1}^{d_j} \exp\left(\beta_{jc'0} + \sum_{k=1}^{K_1} \beta_{jc'k} g_{j,k} \alpha_{i,k}\right)},$$

and we introduce notation

$$\begin{aligned} \phi_{ijc} &= \beta_{jc0} + \sum_{k=1}^{K_1} \beta_{jck} g_{j,k} \alpha_{i,k} - C_{ij(c)}; \\ C_{ij(c)} &= \log\left\{ \sum_{1 \le \ell \le d_j, \ell \ne c} \exp\left(\beta_{j\ell 0} + \sum_{k=1}^{K_1} \beta_{j\ell k} g_{j,k} \alpha_{i,k}\right) \right\}. \end{aligned} \tag{18}$$

Next by the property of the Polya-Gamma random variables (Polson et al., 2013), we have

$$\frac{\exp(\phi_{ijc})^{y_{ijc}}}{1 + \exp(\phi_{ijc})} = 2 \exp\{(y_{ijc} - 1/2)\phi_{ijc}\} \int_0^\infty \exp\{-w_{ijc}\phi_{ijc}^2/2\} p^{\text{PG}}(w_{ijc} \mid 1, 0)\, \mathrm{d}w_{ijc},$$

where $p^{\text{PG}}(w_{ijc} \mid 1, 0)$ denotes the density function of PG(1, 0). Based on the above identity, the conditional posterior of the β_{jc0}'s and β_{jck}'s are still Gaussian, and the conditional posterior of each w_{ijc} is still Polya-Gamma with $(w_{ijc} \mid -) \sim \text{PG}(1, \beta_{jc0} + \sum_{k=1}^{K_1} \beta_{jck} g_{j,k} \alpha_{i,k} - C_{ij(c)})$; these full conditional distributions are easy to sample from. As for the binary entries $g_{j,k}$'s indicating the presence or absence of edges in the Bayesian Pyramid, we sample each $g_{j,k}$ individually from its posterior Bernoulli distribution. The detailed steps of such a Gibbs sampler with known K_1 are presented in the Online Supplementary Material.

4.2.2 *Inferring an Unknown* K_1

On top of the sampling algorithm described above for fixed K_1, we propose a method for simultaneously inferring K_1 and other parameters. In the context of mixture models, Rousseau and Mengersen (2011) defined over-fitted mixtures with more than enough latent components and used shrinkage priors to effectively delete the unnecessary ones. In a similar spirit but motivated by Gaussian linear factor models, Legramanti et al. (2020) proposed the cumulative shrinkage process (CSP) prior, which has a spike and slab structure. We use a CSP prior on the variances $\{\sigma_{ck}^2\}$ of $\{\beta_{jck}\}$ to infer the number of latent binary variables K_1 in a two-layer Bayesian Pyramid. The rationale for using such an *increasing shrinkage prior* for latent dimension selection is that, it is natural to expect additional latent dimensions to play a progressively less important role in characterizing the data, so the associated parameters should have a stochastically decreasing effect. Specifically, under the CSP prior, the kth latent dimension is controlled by a scalar θ_k that follows a spike-and-slab distribution. Redundant dimensions will be essentially deleted by progressively shrinking the sequence $\{\theta_1, \theta_2, \ldots\}$ towards an appropriate value θ_∞ (the spike). In particular, Legramanti et al. (2020) considered the *continuous* factor model where $\theta_k > 0$ denotes the variance of the factor loadings for the kth factor and θ_∞ is a small positive number indicating the variance of redundant latent factors.

We next describe in detail the prior specifications with an unknown number of binary latent variables K_1. Consider an upper bound K_{upper} for K_1, with $K_1 < K_{\text{upper}}$. Based on the identifiability conditions in Theorem 4 about the shape of $\mathbf{G}^{(1)}_{p\times K_1}$, K_1 is naturally constrained to be at most $p/3$, therefore K_{upper} can be set to $p/3$ or smaller in practice. We adopt a prior that detects redundant binary latent variables by increasingly shrinking the variance of β_{jck}'s as k grows from 1 to K_{upper}. Specifically, letting $\beta_{jck} \sim N(0, \sigma^2_{ck})$, we put a CSP prior on variances $\{\sigma^2_{c1}, \sigma^2_{c2}, \ldots, \sigma^2_{cK_{\text{upper}}}\}$ for each category $c \in [d-1]$, where σ^2_∞ is a prespecified small positive number indicating the spike variance for redundant binary latent variables:

$$\begin{aligned}&\text{(ii) for unknown } K_1 < K_{\text{upper}}, \quad k = 1, \ldots, K_{\text{upper}}:\\ &\qquad \sigma^2_{ck} \mid \pi_k \sim (1-\pi_k)\text{InvGa}(b_{1\sigma}, b_{2\sigma}) + \pi_k \delta_{\sigma^2_\infty};\end{aligned} \tag{19}$$

$$\pi_k = \sum_{\ell=1}^{k} \nu_\ell \prod_{m=1}^{\ell-1}(1-\nu_m), \tag{20}$$

where $\text{InvGa}(b_{1\sigma}, b_{2\sigma})$ refers to the inverse gamma distribution with shape $b_{1\sigma}$ and scale $b_{2\sigma}$, and π_k has a stick-breaking representation as in (20) with $\nu_1, \nu_2, \ldots, \nu_{K_{\text{upper}}-1}$ independently following the Beta distribution $\text{Beta}(1, \alpha_0)$. We set $\nu_{K_{\text{upper}}} \equiv 1$ to truncate the stick-breaking representation at K_{upper}, similarly to Legramanti et al. (2020). The $\delta_{\sigma^2_\infty}$ represents the Dirac spike distribution with σ^2_∞ serving as the variance of redundant latent variables, while $\text{InvGa}(a_\sigma, b_\sigma)$ represents the more diffuse slab distribution for the variances corresponding to active latent variables. The 'increasing shrinkage' comes from the fact that as the latent variable index k increases, the probability of α_k belonging to the spike, π_k, stochastically increases, because

$$\mathbb{E}[\pi_k] = \sum_{\ell=1}^{k} \mathbb{E}[\nu_\ell] \prod_{m=1}^{\ell-1} \mathbb{E}[1-\nu_m] = 1 - \frac{1}{(1/\alpha_0 + 1)^k}$$

increases as the index k increases. Therefore, the CSP prior features an increasing amount of shrinkage for larger k. We introduce auxiliary variables $\{h_k;\ k = 1, \ldots, K_{\text{upper}}\}$ with $h_k \in [K_{\text{upper}}]$ to help with understanding and facilitate posterior computation. Specifically, the prior in (19) can be obtained by marginalizing out a discrete auxiliary variable h_k with $\mathbb{P}(h_k = \ell) = \nu_\ell \prod_{m=1}^{\ell-1}(1-\nu_m)$, so (19)–(20) can be reformulated in terms of h_k as

$$\begin{aligned}&(\sigma^2_{ck} \mid h_k) \sim \mathbb{1}(h_k > k) \cdot \text{InvGa}(b_{1\sigma}, b_{2\sigma}) + \mathbb{1}(h_k \le k) \cdot \delta_{\sigma^2_\infty};\\ &\mathbb{P}(h_k \le k) = \sum_{\ell=1}^{k} \nu_\ell \prod_{m=1}^{\ell-1}(1-\nu_m) = \pi_k, \quad k = 1, \ldots, K_{\text{upper}}.\end{aligned}$$

Therefore, the auxiliary variables h_k determine whether the kth latent dimension is in the spike and hence corresponds to a redundant latent variable α_k; specifically, if $h_k > k$ then σ^2_{ck} follows the slab distribution $\text{InvGa}(b_{1\sigma}, b_{2\sigma})$ and α_k is *active*, otherwise $\sigma^2_{ck} = \sigma^2_\infty$ is in the spike and α_k is *redundant*. Given (19), the largest possible number of active latent variables is $K_{\text{upper}} - 1$, because $\pi_{K_{\text{upper}}} \equiv 1 = \mathbb{P}(h_{K_{\text{upper}}} \le K_{\text{upper}})$ and the last latent variable is always redundant. Since the event $\mathbb{1}(h_k > k)$ indicates that the kth component is in the slab and hence active, we can write the total number of active latent dimensions $K^\star$ as

$$K^\star = \sum_{k=1}^{K_{\text{upper}}} \mathbb{1}(h_k > k). \tag{21}$$

Tracking the posterior samples of all the h_k can give a posterior estimate of $K^\star$. The above data augmentation leads to Gibbs updating steps. We present the details of our Gibbs sampler in the Online Supplementary Material.

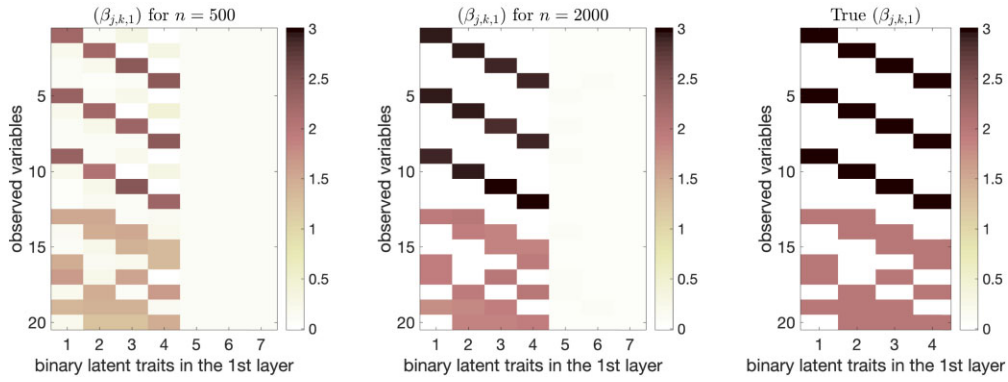

Figure 3. Posterior means of the main-effect parameters $\{\beta_{j1k}\}$ for response category $c = 1$ averaged across 50 independent replications, for $n = 500$ or 2,000. $K_{\text{upper}} = 7$ is taken in the CSP prior for posterior inference, while the true K_1 is 4 as shown in the rightmost plot.

Remark 5 It is methodologically straightforward to extend our Gibbs sampler to deep Bayesian Pyramids with more than two latent layers. To see this, note that in deep Bayesian Pyramids with $m \geq 2$, for each m, the conditional distribution of $\boldsymbol{\alpha}^{(m)}$ given $\boldsymbol{\alpha}^{(m+1)}$ in Example 2 is a special case of the conditional distribution of $\boldsymbol{y}$ given $\boldsymbol{\alpha}^{(1)}$. Indeed, both of these conditionals follow generalized linear models with the (multinomial) logit link, with the parent variables serving as predictors for the child. Under such a formulation, introducing additional Polya-Gamma auxiliary variables for the $\boldsymbol{\alpha}^{(1)}$-layer similar to those for the $\boldsymbol{y}$-layer would allow Gibbs updates in a three-latent-layer Bayesian Pyramid. In this work, we focus on two-latent-layer models for computational efficiency.

Remark 6 We also remark that it is not hard to derive and implement a Gibbs sampler for constrained latent class models (CLCMs) mentioned in Section 3.1. Performing Gibbs sampling for CLCMs is a relatively straightforward extension to the current Gibbs sampler for Bayesian Pyramids. The reason is that one can similarly use the data augmentation strategies in Holmes and Held (2006) and Polson et al. (2013) to deal with $y_j \in [d_j]$; and one can also adopt similar priors for the binary constraint matrix $\mathbf{S} \in \{0, 1\}^{p \times H}$ as those adopted for the graphical matrix $\mathbf{G} \in \{0, 1\}^{p \times K_1}$ in a Bayesian Pyramid. Our preliminary simulations for such a Gibbs sampler under CLCMs showed that the recovery of parameters and the constraint matrix are not as stable and accurate as that for Bayesian Pyramids (shown in the later Figures 3–5). One explanation is that compared to multilayer Bayesian Pyramids, the parametrization of a CLCM is less parsimonious and requires many more mixture proportion parameters in order to describe the same joint distribution of the observed variables. Therefore, we have chosen to focus on the method for Bayesian Pyramids in this work, and treat CLCMs mainly as intermediate tools to help establish identifiability of Bayesian Pyramids.

5 Simulation studies

We conducted replicated simulation studies to assess the Bayesian procedure proposed in Section 4 and examine whether the model parameters are indeed estimable as implied by Theorem 3. Consider a two-latent-layer Bayesian Pyramid with $p = 20$ observed variables, $d = 4$ response categories for each observed variable, and $K_1 = 4$ binary latent variables in the middle layer and one binary latent variable in the deep layer. Let the true $p \times K_1$ binary graphical matrix be

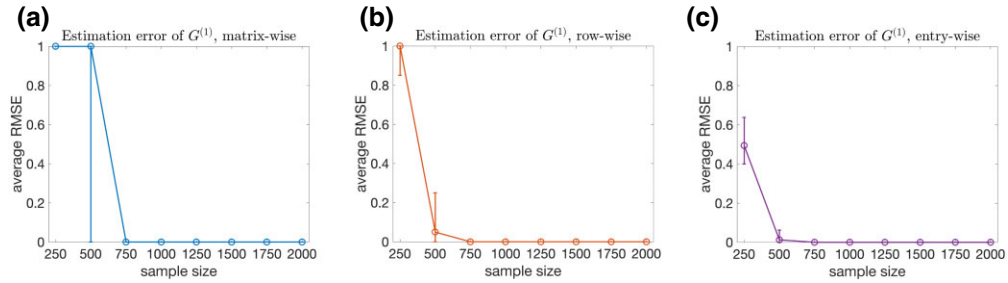

Figure 4. Mean estimation errors of the binary graphical matrix $\mathbf{G}^{(1)}$ at the matrix level (in (a)), row level (in (b)), and entry level (in (c)). The median, 25% quantile, and 75% quantile based on the 50 independent simulation replications are shown by the circle, lower bar, and upper bar, respectively.

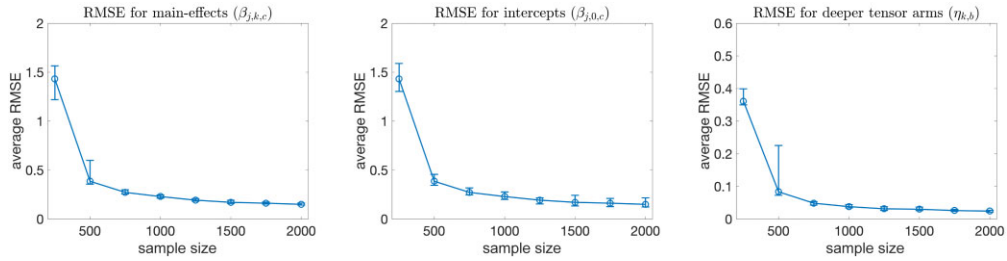

Figure 5. Average root-mean-square errors (RMSE) of the posterior means of the main-effects $\boldsymbol{\beta}$, intercepts $\boldsymbol{\beta}_0$, and deeper tensor arms $\boldsymbol{\eta}$ versus sample size $n = 250 \cdot i$ for $i \in \{1, 2, \ldots, 8\}$. The median, 25% quantile, and 75% quantile based on the 50 independent simulation replications are shown by the circle, lower bar, and upper bar, respectively.

$\mathbf{G}^{(1)} = (\mathbf{I}_4; \mathbf{I}_4; \mathbf{I}_4; 1100; 0110; 0011; 1001; 1010; 1001; 0101; 1110; 0111)$. Such a $\mathbf{G}^{(1)}$ satisfies the conditions for strict identifiability in Theorem 4, since it contains three copies of the identity matrix $\mathbf{I}_4$ as submatrices. Let the true intercept parameters for categories $c = 1, 2, 3$ be $(\beta_{j,1,0}, \beta_{j,2,0}, \beta_{j,3,0}) = (-3, -2, -1)$ for each j; and for any $g^{(1)}_{j,k} = 1$, let the corresponding true main-effect parameters of the binary latent variables be $\beta_{j,c,k} = 3$ if variable y_j has a single parent and $\beta_{j,c,k} = 2$ if y_j has multiple parents. See Figure 3c for a heatmap of the sparse matrix of the main-effect parameters $(\beta_{j1k}; j \in [p], k \in [K_1])$ for category $c = 1$.

We use the method developed in Section 4 with the CSP prior for posterior inference under an unknown K_1. We specify an upper bound for K_1 as $K_{\text{upper}} = 7$, because $\lceil p/20 \rceil = 7$ is a natural upper bound here based on the identifiability considerations mentioned before. As for the hyperparameters in the CSP prior, we mainly follow the default setting and suggestion in Legramanti et al. (2020). Specifically, we set α_0 to the same value in Legramanti et al. (2020), $\alpha_0 = 5$; we also follow the suggestion in Legramanti et al. (2020) that the spike variance σ^2_∞ should be a small positive number but should avoid to take excessively low values by setting $\sigma^2_\infty = 0.07$. Our simulations adopting these choices turn out to produce accurate estimation results under different settings (see the later Figures 3–5). In our preliminary simulations, we also varied these hyperparameters around the above values, and did not observe the algorithmic performance to be sensitive to them. Throughout the Gibbs sampling iterations, we enforce an identifiability constraint on $\boldsymbol{\beta}$ that $\beta_{jck} > 0$ as long as $g_{j,k} = 1$. For each sample size n we conduct 50 independent simulation replications. Since the model is identifiable up to a permutation of the latent variables in each layer, we post-process the posterior samples to find a column permutation of $\mathbf{G}^{(1)}$ to best match the simulation truth; then the columns of other parameter matrices are permuted accordingly.

We have used the Gelman-Rubin convergence diagnostic (Gelman et al., 2013; Gelman & Rubin, 1992) to assess the convergence of the MCMC output from multiple random

initializations. In particular, for the simulation setting corresponding to $n = 1{,}000$, we randomly initialize the parameters from their prior distributions $J = 5$ times and then run five MCMC chains for each simulated dataset. For each MCMC chain, we ran the chain for 15,000 iterations, discarding the first 10,000 iterations as burn-in, and retaining every fifth sample post burn-in to thin the chain. After collecting the posterior samples, we calculate the *potential scale reduction factor* R^2 (i.e., the Gelman-Rubin statistic) of the model parameters. The median Gelman-Rubin statistics for the deep conditional probabilities $\boldsymbol{\eta} = (\eta_{kb})_{K\times B}$ and that for the deep latent class proportions $\boldsymbol{\tau} = (\tau_b)_{B\times 1}$ across the five chains are as follows:

$$\text{median GR}(\boldsymbol{\eta}_{K\times B}) = \begin{pmatrix} 1.0009 & 1.0013 \\ 1.0016 & 1.0020 \\ 1.0018 & 1.0015 \\ 1.0016 & 1.0014 \end{pmatrix}, \quad \text{median GR}(\boldsymbol{\tau}_{B\times 1}) = \begin{pmatrix} 1.0021 \\ 1.0021 \end{pmatrix}.$$

The Gelman–Rubin statistics for other parameters are similarly well controlled and are omitted. We also inspected the traceplots of the MCMC outputs and observed fast mixing of the MCMC chain after convergence. These observations justify running MCMC for 15,000 iterations and discarding the first 10,000 as burn-in, therefore we adopt these settings in all the numerical experiments. When applying the proposed method to other datasets, we also recommend calculating the Gelman–Rubin statistics and inspecting the traceplots to determine the appropriate number of overall MCMC iterations and burn-in iterations.

Our Bayesian modeling of $\mathbf{G}$ adopts rather uninformative priors with each entry $g_{j,k} \sim \text{Bernoulli}(\gamma)$ and we do not force $\mathbf{G}$ to take a specific form (such as to include any identity submatrix as described in Proposition 3 for strict identifiability) but rather let the sampler freely explore the space of all binary matrices to estimate $\mathbf{G}$. Our theory on identifiability and posterior consistency implies that the posteriors of both $\mathbf{G}$ and other parameters concentrate around their true values as sample size grows. This is empirically verified in our simulation studies; Figures 3–5 show that as sample size n grows, both the discrete $\mathbf{G}$ and the continuous parameters are estimated consistently. We next elaborate on these findings from simulations.

Figure 3 presents posterior means of the main-effect parameters $(\beta_{j1k}; j \in [p], k \in [K_1])$ for category $c = 1$, averaged across the 50 independent replications. The leftmost plot of Figure 3 shows that for a relatively small sample size $n = 500$, the posterior means of (β_{j1k}) already exhibit similar structure as the ground truth in the rightmost plot. Also, the true number of $K_1 = 4$ binary latent variables in the middle latent layer are revealed in Figures 3a and b. For the sample size $n = 2{,}000$, Figure 3b shows that the posterior means of (β_{j1k}) are very close to the ground truth. The posterior means for other categories $c = 2, 3, 4$ show similar patterns to those for $c = 1$. The estimated (β_{j1k}) are slightly biased toward zero for the smaller sample size ($n = 500$) with bias less for the larger sample size ($n = 2, 000$). Our results suggest that the binary graphical matrix that underlies $\{\beta_{jck}\}$ (that is, the sparsity structure of the continuous parameters) is easier to estimate than the nonzero $\{\beta_{jck}\}$ themselves. That is, with finite samples, the existence of a link between each observed variable y_j and each binary latent variable α_k is easier to estimate than the strength of such a link (if it exists).

To assess how the approach performs with an increasing sample size, we consider eight sample sizes $n = 250 \cdot i$ for $i = 1, 2, \ldots, 8$ under the same settings as above. To compare the estimated structures to the simulation truth with $K_1 = 4$, we retain exactly four binary latent variables $k \in [K_{\text{upper}}]$; choosing those having the largest average posterior variance $1/d \sum_{c=1}^{d} \sigma_{ck}^2$. For the discrete structure of the binary graphical matrix $\mathbf{G}^{(1)}$, we present mean estimation errors in Figure 4. Specifically, Figure 4a plots the errors of recovering the entire matrix $\mathbf{G}^{(1)}$, Figure 4b plots the errors of recovering the row vectors of $\mathbf{G}^{(1)}$, and Figure 4b plots the errors of recovering the individual entries of $\mathbf{G}^{(1)}$. For sample size as small as $n = 750$, estimation of $\mathbf{G}^{(1)}$ is very accurate across the simulation replications. Notably, $n = 750$ is much smaller than the total number of cells $d^p = 4^{20} \approx 1.1 \times 10^{12}$ in the contingency table for the observed variables $\boldsymbol{y}$. For continuous parameters $\boldsymbol{\beta}_0 = \{\beta_{jc0}\}$, $\boldsymbol{\beta} = \{\beta_{jck}\}$, and $\boldsymbol{\eta} = \{\eta_{kb}\}$, respectively, in Figure 5 we plot average root-mean-square errors (RMSE) of the posterior means versus sample size n. For each $\boldsymbol{\beta}_0 = \{\beta_{jc0}\}$, $\boldsymbol{\beta} = \{\beta_{jck}\}$, and $\boldsymbol{\eta} = \{\eta_{kb}\}$, Figure 5 shows RMSE decreases with sample size, which is as expected given our posterior consistency result in Theorem 3.

In simulations (and also the later real data analysis), we have checked the posterior samples collected from the MCMC algorithm and obtained the traceplots of various parameters in the model. By examining these, we have not observed label-switching issues in our numerical studies. In general, we would suggest one uses the traceplots of those mixture-component-specific parameters to examine whether there is a label switching issue. If there exists such issues, we recommend using the `R` package `label.switching` (Papastamoulis, 2016) for MCMC outputs to address the issue.

6 Application to DNA nucleotide sequence data

We apply our proposed two-latent-layer Bayesian Pyramid with the CSP prior to the Splice Junction dataset of DNA nucleotide sequences; the data are available from the UCI machine learning repository. We also analyze another dataset of nucleotide sequences, the Promoter data, and present the results in the Online Supplementary Material. The Splice Junction data consist of A, C, G, T nucleotides ($d = 4$) at $p = 60$ positions for $n = 3{,}175$ DNA sequences. There are two types of genomic regions called introns and exons; junctions between these two are called splice junctions (Nguyen et al., 2016). Splice junctions can be further categorized as (a) exon-intron junction; and (b) intron-exon junction. The $n = 3{,}175$ samples in the Splice dataset each belong to one of three types: Exon-Intron junction ('EI', 762 samples); Intron-Exon junction ('IE', 765 samples); and Neither EI or IE ('N', 1648 samples). Previous studies have used supervised learning methods for predicting sequence type (e.g., Li & Wong, 2003; Nguyen et al., 2016). Here we fit the proposed two-latent-layer Bayesian Pyramid to the data in a completely unsupervised manner, with the sequence type information held out. We use the nucleotide sequences to learn discrete latent representations of each sequence, and then investigate whether the learned lower dimensional discrete latent features are interpretable and predictive of the sequence type.

We let the variable z in the deepest latent layer have $B = 3$ categories, inspired by the fact that there are three types of sequences: EI, IE, and N; but we do not use any information of which sequence belongs to which type when fitting the model to data. As mentioned earlier, the upper bound for the binary latent variables K_{upper} can be set to $\lceil p/3 \rceil$ or smaller in order to yield an identifiable Bayesian Pyramid model. In practice, when p is large, we recommend starting with a relatively small K_{upper}, inspecting the estimated active/redundant latent dimensions, and only increasing K_{upper} if all the latent dimensions are estimated to be active *a posteriori* (that is, if the posterior model of $K^\star$ defined in (21) equals $K_{\text{upper}} - 1$). For this splice junction dataset with $p = 60$, we start with $K_{\text{upper}} = 7$ for better computational efficiency; this K_{upper} is the same as that used in the simulations and already allows for $2^{K_{\text{upper}}-1} = 64$ distinct latent binary profiles. We find that the posteriors select only $K^\star = 5$ active latent dimensions; this suggests that it is not necessary to increase K_{upper} to a larger number for this dataset.

We still run the Gibbs sampler for 15,000 iterations, discarding the first 10,000 as burn-in, and retaining every fifth sample post burn-in to thin the chain. Based on examining traceplots, the sampler has good mixing behavior. As mentioned in the last paragraph, our method selects $K^\star = 5$ binary latent variables. We index the saved posterior samples by $r \in \{1, \ldots, R = 2{,}000\}$, for each r denote the samples of $\mathbf{G}$ by $(g_{j,k}^{(r)}; j \in [60], k \in [5])$. Similarly for each r, denote the posterior samples of the nucleotides sequences' latent binary traits by $(a_{i,k}^{(r)}; i \in [3{,}175], k \in [5])$, and denote those of the nucleotide sequences' deep latent category by $(z_i^{(r)}; i \in [3{,}175])$ where $z_i^{(r)} \in \{1, 2, 3\}$. Define our final estimators $\widehat{\mathbf{G}} = (\widehat{g}_{j,k})$, $\widehat{\mathbf{A}} = (\widehat{a}_{i,k})$, and $\widehat{\mathbf{Z}} = (\widehat{z}_{i,b})$ to be

$$\widehat{g}_{j,k} = \mathbb{1}\left(\frac{1}{R}\sum_{r=1}^{R} g_{j,k}^{(r)} > \frac{1}{2}\right), \quad \widehat{a}_{i,k} = \mathbb{1}\left(\frac{1}{R}\sum_{r=1}^{R} a_{i,k}^{(r)} > \frac{1}{2}\right),$$

$$\widehat{z}_{i,b} = \begin{cases} 1, & \text{if } b = \arg\max_{b \in B} \frac{1}{R}\sum_{r=1}^{R} \mathbb{1}(z_i^{(r)} = b); \\ 0, & \text{otherwise} \end{cases}.$$

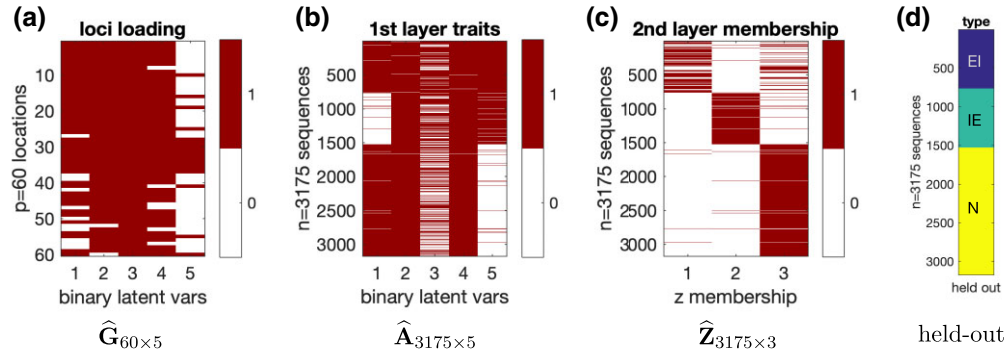

Figure 6. Splice junction data analysis under the CSP prior with $K_{\text{upper}} = 7$. Plots are presented with the $K^\star = 5$ binary latent traits selected by our proposed method a posteriori. After applying the rule-lists approach to deterministically match the latent features to the gene types as in (22), the accuracies for predicting the gene types EI, IE, N are all above 95%.

$\widehat{g}_{j,k}$, $\widehat{\alpha}_{i,k}$, and $\widehat{z}_{i,b}$ summarize information of the element-wise posterior modes of the discrete latent structures in our model. The 60×5 matrix $\widehat{\mathbf{G}}$ depicts how the loci load on the binary latent traits, the $3{,}175 \times 5$ matrix $\widehat{\mathbf{A}}$ depicts the presence or absence of each binary latent trait in each nucleotide sequence, and the $3{,}175 \times 3$ matrix $\widehat{\mathbf{Z}}$ depicts which deep latent group each nucleotide sequence belongs to. $\widehat{\mathbf{G}}$, $\widehat{\mathbf{A}}$, and $\widehat{\mathbf{Z}}$ are all binary matrices, but the first two are binary feature matrices while the last one $\widehat{\mathbf{Z}}$ has each row having exactly one entry of '1' indicating group membership. In Figure 6, the first three plots display the three estimated matrices $\widehat{\mathbf{G}}$, $\widehat{\mathbf{A}}$, and $\widehat{\mathbf{Z}}$, respectively; and the last plot shows the held-out gene type information for reference. As for the estimated loci loading matrix $\widehat{\mathbf{G}}$, Figure 6a provides information on how the $p = 60$ loci depend on the five binary latent traits. Specifically, we found that the first 27 loci show somewhat similar loading patterns and mainly load on the first four binary traits. Also, the middle 10 loci (from locus 28 to locus 37) are similar in loading on all five traits, and the last 23 loci (from locus 38 to locus 60) are similar in exhibiting sparser loading structures. Figure 6b–d shows that the two matrices $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{Z}}$ corresponding to the $n = 3{,}175$ nucleotide sequences exhibit a clear pattern of clustering, which aligns well with the known but held-out junction types EI, IE, and N.

To formally assess how the latent discrete features learned by the proposed method perform in downstream prediction, we apply the 'rule lists' classification approach in Angelino et al. (2017) to the estimated latent features in $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{Z}}$ for $n = 3{,}175$ nucleotide sequences. The rule-lists approach is an interpretable classification method based on a categorical feature space, and it finds simple and deterministic rules of the categorical features in predicting a (binary) class label. For each instance $i \in \{1, \ldots, n = 3{,}175\}$, we define the categorical features to be the eight-dimensional vector $\widehat{\boldsymbol{x}}_i = (\widehat{z}_{i,1}, \widehat{z}_{i,2}, \widehat{z}_{i,3}, \widehat{a}_{i,1}, \ldots, \widehat{a}_{i,5})$. $\widehat{\boldsymbol{x}}_i$ is concatenated from $\widehat{\boldsymbol{z}}_i$ and $\widehat{\boldsymbol{a}}_i$ and is a feature vector of binary entries. Denote the ground-truth nucleotide sequence types by $\boldsymbol{t} = (t_i;\ 1 \le i \le 3{,}175)$, then $t_i = \text{EI}$ for $i = 1, \ldots, 762$, $t_i = \text{IE}$ for $i = 763, \ldots, 1527$, and $t_i = \text{N}$ for $i = 1{,}528, \ldots, 3{,}175$. Recall that the information of t_i's are not used in fitting our Bayesian Pyramid to obtain the latent features $\widehat{\boldsymbol{x}}_i$'s. We use the Python package `CORELS` for the rule-lists method with $\boldsymbol{x}_i$'s and t_i's as input, and find rules that match $\widehat{\boldsymbol{x}}_i$'s to t_i's. Specifically, these deterministic rules given by `CORELS` are:

$$\begin{aligned} \widehat{t}_i^{\,\text{EI}} &= \mathbb{1}(\widehat{a}_{i,1} = 1 \text{ and } \widehat{a}_{i,5} = 1), \\ \widehat{t}_i^{\,\text{IE}} &= \mathbb{1}(\widehat{z}_{i,2} = 1), \\ \widehat{t}_i^{\,\text{N}} &= \mathbb{1}(\widehat{z}_{i,2} = 0 \text{ and } \widehat{a}_{i,5} = 0). \end{aligned} \tag{22}$$

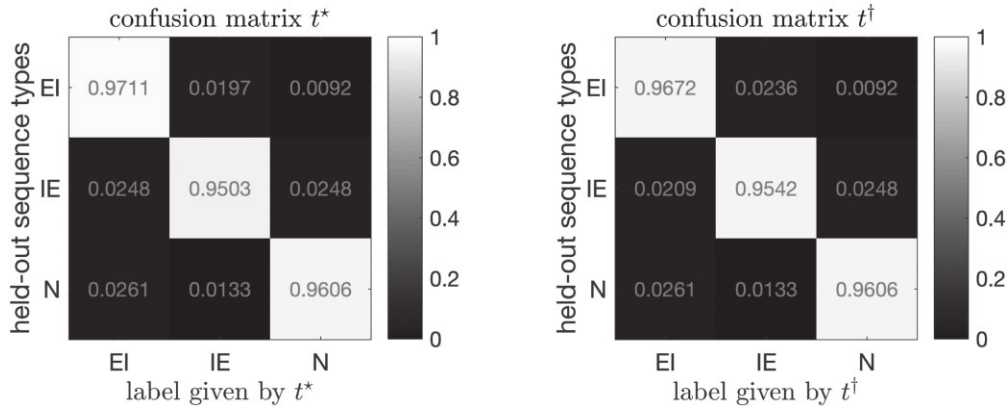

Figure 7. Downstream interpretable classification results for splice junction data following fitting the two-latent-layer Bayesian Pyramid. Prediction performance of the learned lower-dimensional discrete latent features are summarized in two confusion matrices, corresponding to $\widehat{\boldsymbol{t}}^\star$ and $\widehat{\boldsymbol{t}}^\dagger$ defined in (23).

Equation (22) gives a very simple and explicit rule depending on $\widehat{z}_{i,2}$, $\widehat{a}_{i,1}$, and $\widehat{a}_{i,5}$ for each nucleotide sequence i. This simple rule can be empirically verified by comparing the middle two plots to the rightmost plot in Figure 6. In (22), the rules for EI and IE are not mutually exclusive, but those for EI and N are mutually exclusive and the same holds for IE and N. Therefore, to obtain mutually exclusive rules for the three class labels EI, IE, and N based on (22), we can simply define the following two types of labels $\widehat{\boldsymbol{t}}^\star = (\widehat{t}_i^\star;\ i \in [n])$ and $\widehat{\boldsymbol{t}}^\dagger = (\widehat{t}_i^\dagger;\ i \in [n])$:

$$\widehat{t}_i^\star = \begin{cases} \text{EI}, & \text{if } \widehat{t}_i^{\text{EI}} = 1 \text{ and } \widehat{t}_i^{\text{IE}} = 0; \\ \text{EI}, & \text{if } \widehat{t}_i^{\text{IE}} = 1; \\ \text{N}, & \text{if } \widehat{t}_i^{\text{N}} = 1; \end{cases} \quad \text{or} \quad \widehat{t}_i^\dagger = \begin{cases} \text{EI}, & \text{if } \widehat{t}_i^{\text{EI}} = 1; \\ \text{EI}, & \text{if } \widehat{t}_i^{\text{IE}} = 1 \text{ and } \widehat{t}_i^{\text{EI}} = 0; \\ \text{N}, & \text{if } \widehat{t}_i^{\text{N}} = 1. \end{cases} \tag{23}$$

For the two types of labels $\widehat{\boldsymbol{t}}^\star$ and $\widehat{\boldsymbol{t}}^\dagger$ in (23), we provide their corresponding normalized confusion matrices in Figure 7 along with their heatmaps. Figure 7 shows that both $\widehat{\boldsymbol{t}}^\star$ and $\widehat{\boldsymbol{t}}^\dagger$ have very high prediction accuracies for the sequence types $\boldsymbol{t}$, with all the diagonal entries above 95%. The superb prediction performance as shown in Figure 7 implies that the learned discrete latent features of nucleotide sequences are interpretable and very useful in the downstream task of classification. For the Splice Juntion dataset, such accuracies are even comparable to the state-of-the-art prediction performance given by fine-tuned convolutional neural networks given in Nguyen et al. (2016), which are also between 95% and 97%.

In the above data analysis, we have fixed $B = 3$ inspired by the fact that there are three types of splice junctions. Alternatively, we can use the *overfitted mixture framework* of Rousseau and Mengersen (2011) to accommodate an unknown B. Overfitted mixtures are finite mixture models with an unknown number of mixture components B, where only an upper bound B_{upper} of B is known. In such scenarios, Rousseau and Mengersen (2011) proposed to use shrinkage priors for the mixture proportion parameters, such as Dirichlet priors with small Dirichlet hyperparameters. More specifically, denote the mixture proportion parameters corresponding to the B_{upper} 'overfitted' mixture components by $\boldsymbol{\pi} := (\pi_1, \ldots, \pi_{B_{\text{upper}}})$, then $\boldsymbol{\pi}$ lives on the $(B_{\text{upper}} - 1)$-dimensional probability simplex. The overfitted mixture framework in Rousseau and Mengersen (2011) guarantees that with the prior $\boldsymbol{\pi} \sim \text{Dirichlet}(a_1, \ldots, a_{B_{\text{upper}}})$ where the Dirichlet parameters are small enough with respect to the dimension of the observations, the redundant mixture components will be 'emptied out' in the posterior asymptotically. Hence, the B true mixture components underlying the data will be identified from the posterior distribution. In the Online Supplementary Material, we reanalyze the real data specifying $B = 5$ and simulation studies with an overfitted B, and achieve favorable results.

Table 1. Downstream classification accuracy for splice data

	EI (train)	IE (train)	N (train)	EI (test)	IE (test)	N (test)
Raw nucleotide nucleotides	0.909	0.854	0.840	0.916	0.866	0.839
DX2009	0.946	0.955	0.918	0.950	0.965	0.929
IndepBinary	0.964	0.955	0.956	0.965	0.957	0.965
BayesPyramid	0.971	0.975	0.970	0.984	0.983	0.976

For comparison, we also applied/implemented two alternative discrete latent structure models, including: (a) the latent class model, with a single univariate discrete latent variable behind the observed data layer; and (b) a single-latent-layer model, with one layer of *independent* binary latent variables behind the data layer. For model (a), we applied the nonparametric Bayesian method in Dunson and Xing (2009) to the splice junction dataset and extracted $k_{DX} = 10$ latent classes (default in Dunson & Xing, 2009; increasing k_{DX} beyond 10 is not considered because in the current 10 latent classes there are already empty classes not occupied by any nucleotide sequence). For model (b), we implemented a new Gibbs sampler (see description in Section 9.3 in the Online Supplementary Material), still employing the cumulative shrinkage process prior as used for Bayesian Pyramids. Then based on the learned lower-dimensional latent representations, we still apply the 'rule-lists' classification approach in Angelino et al. (2017) (via the Python package `CORELS`) for each of the three labels EI, IE, and N via binary classification. In addition to the above two competitors (abbreviated as 'DX2009' and 'IndepBinary', respectively), we also consider a benchmark approach of directly applying the rule-lists classifier to raw DNA nucleotide sequences. Since the rule-lists classifier in the `CORELS` package only applies to binary predictors, we first convert the DNA nucleotide sequences of A, G, C, T into longer sequences containing binary indicators of whether each loci is 'A', or 'G', or 'C'; after this, each original 60-dimensional nucleotide sequence is converted to a binary vector of length 180. We then apply the rule-lists classifier using the same setting of learning at most three 'rules' (i.e., `max_card` = 3 in the `CORELS` package) as all the other three latent variable methods: DX2009, IndepBinary, and BayesPyramid. We obtain the classification accuracy in Table 1.

The left three columns in Table 1 list accuracies on the training set containing 80% of the samples, and the right three columns on the test set containing the remaining 20% of the samples. The test accuracies are very close to the training ones, indicating satisfactory generalizability. The reason is that the maximum number of rules learned for each model is limited to be at most three, i.e., `max_card` = 3 in the `CORELS` package. Such a small number of rules in a classifier do not overfit the training data and hence generalize well to the test data. We remark that when increasing `max_card` beyond three, the downstream classification accuracy of BayesPyramid remains the same as those in Table 1; however, for the raw nucleotide sequences, the `CORELS` package cannot complete the execution of the classifier with `max_card` = 4, likely due to the high-dimensionality of the search space of rules. Therefore, we present the results of `CORELS` classification with `max_card` = 3 in Table 1.

Remarkably, Table 1 shows that all the three unsupervised learning methods learn latent features that are more predictive of the sequence type than the raw sequences themselves, among which the Bayesian Pyramid is the apparent top performer. This observation indicates that the latent variable approaches, especially our multilayer Bayesian Pyramids, can 'denoise' the raw sequence data and return more useful features for downstream tasks. The model in Dunson and Xing (2009) ('DX2009') and the single-layer-independent-latent model ('IndepBinary') are special cases of Bayesian Pyramids, and both have excellent performance on the splice data. However, Bayesian Pyramids significantly decreased the test set misclassification rate of IndepBinary by 31%–60%, of DX2009 by 51%–68%, and of raw nucleotide sequences by 81%–87%.

7 Discussion

In this article, we have proposed Bayesian Pyramids, a general family of discrete multilayer latent variable models for multivariate categorical data. Bayesian Pyramids cover multiple existing

statistical and machine learning models as special cases, while allowing for various different assumptions on the conditional distributions. Our identifiability theory is key in providing reassurance that one can reliably and reproducibly learn the latent structure, guarantees that are lacking from almost all of the related literature. The proposed Bayesian approach has excellent performance in the simulations and data analyses, and shows good computational performance and a surprising ability to accurately infer meaningful and useful latent structure. There are immediate applications of the proposed Bayesian Pyramid approach in many other disciplines. For example, in ecology and biodiversity research, joint species distribution modeling is a very active area (see a recent book of Ovaskainen & Abrego, 2020). The data consist of high-dimensional binary indicators of presence or absence of different species at each sampling location. In this contest, Bayesian Pyramids provide a useful approach for inferring latent community structure of biologic and ecological interest.

This work focuses on the unsupervised setting and uses latent variable approaches. In unsupervised cases, there is not a specific outcome/response associated with each data point, and the general goal is to discover 'interesting' patterns in data (Murphy, 2012). Such unsupervised learning problems are generally considered more challenging and less well studied than supervised ones. Latent variables can capture semantically interpretable constructs, and marginalizing out latents induces great flexibility in the resulting marginal distribution of observables. Bayesian Pyramids provide unsupervised feature discovery tools for learning latent architectures underlying multivariate data, yielding insight into data generating processes, performing nonlinear dimension reduction, and extracting useful latent representations.

In order to realize the great potential of latent variable approaches to unsupervised learning, identifiability issues must be addressed so that reliable latent constructs can be reproducibly extracted from data. Especially, if one wishes to interpret the latent representations and parameters estimated from a latent variable model and use them in downstream analyses, then identifiability is a prerequisite for making such interpretation meaningful and reproducible. In this sense, identifiability is necessary for interpretability of the latent structures. We remark that such interpretability considerations differ from the goals of learning interpretable treatment rules for some outcome in complex supervised learning settings (see, e.g., Kitagawa & Tetenov, 2018; Semenova & Chernozhukov, 2021). In modern machine learning, although powerful deep latent variable models have been proposed, identifiability issues have rarely been considered and the design of the latent architecture is often guided by heuristics. In this work, our identifiability theory directly inspires how to specify the deep latent architecture in a Bayesian Pyramid: the identifiability conditions on matrices $\mathbf{G}^{(m)}$ (in Theorem 4) inspire us to specify a 'pyramid' structure featuring fewer and fewer latent variables deeper up the hierarchy ($J \geq 3K_1$; $K_1 \geq 3K_2$, ...) and sparse graphical connections between layers.

In the future, it is worth further advancing and refining scalable algorithms for Bayesian Pyramids beyond the two-latent-layer case. Methodologically, our current Gibbs sampling procedure can be readily extended to multiple latent layers, because non-adjacent layers in a Bayesian Pyramid are conditionally independent and the current Gibbs updates for shallower layers can be similarly applied to deeper ones. In terms of the computational performance, however, efficiently sampling discrete latent variables via MCMC is generally more challenging than sampling continuous ones. In a Bayesian Pyramid, binary entries of the graphical matrices $\mathbf{G}^{(1)}$, $\mathbf{G}^{(2)}$, ... act similarly as covariate selection indicators in (logistic) regression problems, yet are more difficult to estimate because the 'covariates' $\boldsymbol{\alpha}^{(1)}$, $\boldsymbol{\alpha}^{(2)}$, ... themselves are latent and discrete. This fact can cause unsatisfactory mixing behavior of naive extensions of the Gibbs sampler to deeper models. For example, sampling to update an entry in a deeper graphical matrix $\mathbf{G}^{(d)}$ where $d \geq 2$ can cause a substantial shift of the posterior space for the associated continuous parameters in downward layers. Future research is warranted to advance the MCMC methodology or explore complementary methods for estimating the discrete latent structures in deeper models.

Acknowledgments

The authors thank the Editor, Associate Editor, and three referees for helpful and constructive comments.

Conflict of interest: None declared.

Funding

This work was partially supported by the National Science Foundation grant DMS-2210796. This work was also partially supported by grants R01ES027498 and R01ES028804 from National Institutes of Environmental Health Sciences of the National Institutes of Health, and received funding from European Research Council under the European Union's Horizon 2020 research and innovation program (grant agreement No 856506).

Data availability

Matlab code implementing the proposed method and data are available at this GitHub repository: https://github.com/yuqigu/BayesianPyramids.

Supplementary material

Supplementary material are available at *Journal of the Royal Statistical Society: Series B* online.

References

Allman E. S., Matias C., & Rhodes J. A. (2009). Identifiability of parameters in latent structure models with many observed variables. *The Annals of Statistics*, *37*(6A), 3099–3132. https://doi.org/10.1214/09-AOS689

Anandkumar A., Hsu D., Javanmard A., & Kakade S. (2013). Learning linear Bayesian networks with latent variables. In *International Conference on Machine Learning* (pp. 249–257). PMLR.

Anderson T. W., & Rubin H. (1956). Statistical inference in factor analysis. In *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability* (Vol. 5. pp. 111–150). Univ of California Press.

Angelino E., Larus-Stone N., Alabi D., Seltzer M., & Rudin C. (2017). Learning certifiably optimal rule lists for categorical data. *The Journal of Machine Learning Research*, *18*(1), 8753–8830.

Blei D. M., Ng A. Y., & Jordan M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Chen Y., Culpepper S., & Liang F. (2020). A sparse latent class model for cognitive diagnosis. *Psychometrika*, *85*(1), 121–153. https://doi.org/10.1007/s11336-019-09693-2

Chen Y., Li X., & Zhang S. (2020). Structured latent factor analysis for large-scale data: Identifiability, estimability, and their implications. *Journal of the American Statistical Association*, *115*(532), 1756–1770. https://doi.org/10.1080/01621459.2019.1635485

Chen Y., Liu J., Xu G., & Ying Z. (2015). Statistical analysis of Q-matrix based diagnostic classification models. *Journal of the American Statistical Association*, *110*(510), 850–866. https://doi.org/10.1080/01621459.2014.934827

Collins L. M., & Lanza S. T. (2009). *Latent class and latent transition analysis: With applications in the social, behavioral, and health sciences*. (Vol. *718*). John Wiley & Sons.

de la Torre J. (2011). The generalized DINA model framework. *Psychometrika*, *76*(2), 179–199. https://doi.org/10.1007/s11336-011-9207-7

Dellaportas P., Forster J. J., & Ntzoufras I. (2002). On Bayesian model and variable selection using MCMC. *Statistics and Computing*, *12*(1), 27–36. https://doi.org/10.1023/A:1013164120801

Doshi-Velez F., & Ghahramani Z. (2009). Correlated non-parametric latent feature models. In *Uncertainty in artificial intelligence*. PMLR.

Doshi-Velez F., & Kim B. (2017). 'Towards a rigorous science of interpretable machine learning', arXiv, arXiv:1702.08608, preprint: not peer reviewed.

Drton M., Foygel R., & Sullivant S. (2011). Global identifiability of linear structural equation models. *The Annals of Statistics*, *39*(2), 865–886. https://doi.org/10.1214/10-AOS859

Dua D., & Graff C. (2017). UCI machine learning repository.

Dunson D. B., & Xing C. (2009). Nonparametric Bayes modeling of multivariate categorical data. *Journal of the American Statistical Association*, *104*(487), 1042–1051. https://doi.org/10.1198/jasa.2009.tm08439

Erosheva E., Fienberg S., & Lafferty J. (2004). Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences*, *101*(suppl_1), 5220–5227. https://doi.org/10.1073/pnas.0307760101

Eysenck S. B., Barrett P. T., & Saklofske D. H. (2020). The junior Eysenck personality questionnaire. *Personality and Individual Differences*, *169*(5), 109974. https://doi.org/10.1016/j.paid.2020.109974

Fang G., Liu J., & Ying Z. (2019). On the identifiability of diagnostic classification models. *Psychometrika*, *84*(1), 19–40. https://doi.org/10.1007/s11336-018-09658-x

Gelman A., Carlin J. B., Stern H. S., Dunson D. B., Vehtari A., & Rubin D. B. (2013). *Bayesian data analysis*. CRC Press.

Gelman A., Jakulin A., Pittau M. G., & Su Y.-S. (2008). A weakly informative default prior distribution for logistic and other regression models. *The Annals of Applied Statistics*, *2*(4), 1360–1383. https://doi.org/10.1214/08-AOAS191

Gelman A., & Rubin D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, *7*(4), 457–472. https://doi.org/10.1214/ss/1177011136

Goodman L. A. (1974). Exploratory latent structure analysis using both identifiable and unidentifiable models. *Biometrika*, *61*(2), 215–231. https://doi.org/10.1093/biomet/61.2.215

Gu Y., & Xu G. (2019). Learning attribute patterns in high-dimensional structured latent attribute models. *Journal of Machine Learning Research*, *20*(115), 1–58. https://jmlr.org/papers/v20/19-197.html

Gu Y., & Xu G. (2020). Partial identifiability of restricted latent class models. *Annals of Statistics*, *48*(4), 2082–2107. https://doi.org/10.1214/19-AOS1878

Gyllenberg M., Koski T., Reilink E., & Verlaan M. (1994). Non-uniqueness in probabilistic numerical identification of bacteria. *Journal of Applied Probability*, *31*(2), 542–548. https://doi.org/10.2307/3215044

Haertel E. H. (1989). Using restricted latent class models to map the skill structure of achievement items. *Journal of Educational Measurement*, *26*(4), 301–321. https://doi.org/10.1111/j.1745-3984.1989.tb00336.x

Hinton G. E. (2009). Deep belief networks. *Scholarpedia*, *4*(5), 5947. https://doi.org/10.4249/scholarpedia.5947

Hinton G. E., Osindero S., & Teh Y. -W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, *18*(7), 1527–1554. https://doi.org/10.1162/neco.2006.18.7.1527

Holmes C. C., & Held L. (2006). Bayesian auxiliary variable models for binary and multinomial regression. *Bayesian Analysis*, *1*(1), 145–168. https://doi.org/10.1214/06-BA105

Kitagawa T., & Tetenov A. (2018). Who should be treated? Empirical welfare maximization methods for treatment choice. *Econometrica*, *86*(2), 591–616. https://doi.org/10.3982/ECTA13288

Kolda T. G., & Bader B. W. (2009). Tensor decompositions and applications. *SIAM Review*, *51*(3), 455–500. https://doi.org/10.1137/07070111X

Koopmans T. C., & Reiersol O. (1950). The identification of structural characteristics. *The Annals of Mathematical Statistics*, *21*(2), 165–181. https://doi.org/10.1214/aoms/1177729837

Kruskal J. B. (1977). Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra and its Applications*, *18*(2), 95–138. https://doi.org/10.1016/0024-3795(77)90069-6

Lazarsfeld P. F. (1950). The logical and mathematical foundation of latent structure analysis. In *Studies in social psychology in world war II Vol. IV: Measurement and prediction* (pp. 362–412). Princeton University Press.

Lee H., Ekanadham C., & Ng A. (2007). Sparse deep belief net model for visual area V2. *Advances in Neural Information Processing Systems*, *20*, 873–880.

Lee H., Grosse R., Ranganath R., & Ng A. Y. (2009). Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proceedings of the 26th Annual International Conference on Machine Learning* (pp. 609–616).

Legramanti S., Durante D., & Dunson D. B. (2020). Bayesian cumulative shrinkage for infinite factorizations. *Biometrika*, *107*(3), 745–752. https://doi.org/10.1093/biomet/asaa008

Li J., & Wong L. (2003). Using rules to analyse bio-medical data: A comparison between C4. 5 and PCL. In *International Conference on Web-Age Information Management* (pp. 254–265). Springer.

McLachlan G. J., & Basford K. E. (1988). *Mixture models: Inference and applications to clustering*. (Vol. *38*). M. Dekker.

Miettinen P., & Vreeken J. (2014). Mdl4bmf: Minimum description length for Boolean matrix factorization. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *8*(4), 1–31. https://doi.org/10.1145/2601437

Mourad R., Sinoquet C., Zhang N. L., Liu T., & Leray P. (2013). A survey on latent tree models and applications. *Journal of Artificial Intelligence Research*, *47*(1), 157–203. https://doi.org/10.1613/jair.3879

Murdoch W. J., Singh C., Kumbier K., Abbasi-Asl R., & Yu B. (2019). Interpretable machine learning: Definitions, methods, and applications. *Proceedings of the National Academy of Sciences*, *116*(44), 22071–22080. https://doi.org/10.1073/pnas.1900654116

Murphy K. P. (2012). *Machine learning: A probabilistic perspective*. MIT Press.

Nguyen N. G., Tran V. A., Ngo D. L., Phan D., Lumbanraja F. R., Faisal M. R., Abapihi B., Kubo M., & Satou K. (2016). DNA sequence classification by convolutional neural network. *Journal of Biomedical Science and Engineering*, *9*(5), 280–286. https://doi.org/10.4236/jbise.2016.95021

Ovaskainen O., & Abrego N. (2020). *Joint species distribution modelling: With applications in R*. Ecology, Biodiversity and Conservation. Cambridge University Press.

Ovaskainen O., Abrego N., Halme P., & Dunson D. B. (2016). Using latent variable models to identify large networks of species-to-species associations at different spatial scales. *Methods in Ecology and Evolution*, *7*(5), 549–555. https://doi.org/10.1111/2041-210X.12501

Papastamoulis P. (2016). Label.switching: An R package for dealing with the label switching problem in MCMC outputs. *Journal of Statistical Software*, *69*(1), 1–24. https://doi.org/10.18637/jss.v069.c01

Pearl J. (2014). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Elsevier.
Pokholok D. K., Harbison C. T., Levine S., Cole M., Hannett N. M., Lee T. I., Bell G. W., Walker K., Rolfe P. A., Herbolsheimer E., Zeitlinger J., Lewitter F., Gifford D. K., & Young R. A. (2005). Genome-wide map of nucleosome acetylation and methylation in yeast. *Cell*, *122*(4), 517–527. https://doi.org/10.1016/j.cell.2005.06.026
Polson N. G., Scott J. G., & Windle J. (2013). Bayesian inference for logistic models using Pólya–Gamma latent variables. *Journal of the American Statistical Association*, *108*(504), 1339–1349. https://doi.org/10.1080/01621459.2013.829001
Poon H., & Domingos P. (2011). Sum-product networks: A new deep architecture. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)* (pp. 689–690). IEEE.
Rousseau J., & Mengersen K. (2011). Asymptotic behaviour of the posterior distribution in overfitted mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *73*(5), 689–710. https://doi.org/10.1111/j.1467-9868.2011.00781.x
Rudin C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, *1*(5), 206–215. https://doi.org/10.1038/s42256-019-0048-x
Rupp A. A., & Templin J. L. (2008). Unique characteristics of diagnostic classification models: A comprehensive review of the current state-of-the-art. *Measurement*, *6*(4), 219–262.
Semenova V., & Chernozhukov V. (2021). Debiased machine learning of conditional average treatment effects and other causal functions. *The Econometrics Journal*, *24*(2), 264–289. https://doi.org/10.1093/ectj/utaa027
Skinner C. (2019). Analysis of categorical data for complex surveys. *International Statistical Review*, *87*(S1), S64–S78. https://doi.org/10.1111/insr.12285
Teicher H. (1961). Identifiability of mixtures. *The Annals of Mathematical Statistics*, *32*(1), 244–248. https://doi.org/10.1214/aoms/1177705155
von Davier M., & Lee Y.-S. (2019). *Handbook of diagnostic classification models*. Springer.
Xu G. (2017). Identifiability of restricted latent class models with binary responses. *The Annals of Statistics*, *45*(2), 675–707. https://doi.org/10.1214/16-AOS1464
Zhao H., Melibari M., & Poupart P. (2015). On the relationship between sum-product networks and Bayesian networks. In *International Conference on Machine Learning* (pp. 116–124). PMLR.
Zhou J., Bhattacharya A., Herring A. H., & Dunson D. B. (2015). Bayesian factorizations of big sparse tensors. *Journal of the American Statistical Association*, *110*(512), 1562–1576. https://doi.org/10.1080/01621459.2014.983233
Zwiernik P. (2018). Latent tree models. In *Handbook of graphical models* (pp. 283–306). CRC Press.