Journal of Machine Learning Research 21 (2020) 1-49 Submitted 6/17; Revised 2/20; Published 4/20

General Latent Feature Models for Heterogeneous Datasets

Isabel Valera IVALERA@CS.UNI-SAARLAND.DE
Department of Computer Science
Saarland University
Saarbrücken, Germany; and
Max Planck Institute for Intelligent Systems
Tübingen, Germany;

Melanie F. Pradier MELANIE@SEAS.HARVARD.EDU
School of Engineering and Applied Sciences
Harvard University
Cambridge, USA

Maria Lomeli MARIA.LOMELI@ENG.CAM.AC.UK
Department of Engineering
University of Cambridge
Cambridge, UK

Zoubin Ghahramani ZOUBIN@ENG.CAM.AC.UK
Department of Engineering
University of Cambridge
Cambridge, UK; and
Uber AI,
San Francisco, California, USA

Editor: Samuel Kaski

Abstract

Latent variable models allow capturing the hidden structure underlying the data. In particular, feature allocation models represent each observation by a linear combination of latent variables. These models are often used to make predictions either for new observations or for missing information in the original data, as well as to perform exploratory data analysis. Although there is an extensive literature on latent feature allocation models for homogeneous datasets, where all the attributes that describe each object are of the same (continuous or discrete) type, there is no general framework for practical latent feature modeling for heterogeneous datasets. In this paper, we introduce a general Bayesian nonparametric latent feature allocation model suitable for heterogeneous datasets, where the attributes describing each object can be arbitrary combinations of real-valued, positive real-valued, categorical, ordinal and count variables. The proposed model presents several important properties. First, it is suitable for heterogeneous data while keeping the properties of conjugate models, which enables us to develop an inference algorithm that presents linear complexity with respect to the number of objects and attributes per MCMC iteration. Second, the Bayesian nonparametric component allows us to place a prior distribution on the number of features required to capture the latent structure in the data. Third, the latent features in the model are binary-valued, which facilitates the interpretability of the obtained latent features in exploratory data analysis. Finally, a software package, called GLFM toolbox, is made publicly available for other researchers to use and extend. It is available at https://ivaleram.github.io/GLFM/. We show the flexibility of the proposed model by solving both prediction and data analysis tasks on several real-world datasets.

1. Introduction

One of the aims of unsupervised learning is to recover the latent structure responsible for generating the observed properties or attributes of a set of objects. In particular, latent feature models (also called latent factor models) represent the attributes of each object with an unobserved vector of latent features, usually of lower dimensionality than the number of attributes which describe the object. It is assumed that the observations are generated from a distribution parameterized by those latent feature values. In other words, latent feature models allow us to represent, with only a few features, the immense redundant information present in the observed data, by capturing the statistical dependencies among the different objects and attributes. As a consequence, they have been used to make predictions either for new values of interest or missing information in the original data (Salakhutdinov and Mnih, 2008; Gopalan et al., 2014), as well as to perform exploratory data analysis in order to better understand the data (Blanco et al., 2013; Valera et al., 2016).

There is an extensive literature in latent feature models for homogeneous data, where all the attributes describing each object in the dataset are assumed to be of the same type, that is either continuous or discrete. Specifically, most of the existing literature assumes that datasets contain only either continuous data, often modeled as Gaussian random variables (Griffiths and Ghahramani, 2011; Todeschini et al., 2013), or discrete data, that can be modeled either with discrete likelihoods (Li, 2009; Ruiz et al., 2013; Gopalan et al., 2014) or simply treated as Gaussian variables (Salakhutdinov and Mnih, 2008; Blanco et al., 2013; Todeschini et al., 2013). However, to the best of our knowledge, only a few works consider mixed continuous and discrete variables (Khan et al., 2010, 2013; Klami et al., 2012; Collins et al., 2002)—either by assuming mixed Gaussian and categorical variables or mixed members of the exponential family—, which are very common in real world applications. For instance, Electronic Health Records of hospitals contain lab measurements (real-valued or positive real-valued data), diagnoses (categorical data) and genomic information (ordinal, count data and categorical data). Another example is surveys, which contain diverse information about the participants such as age (count data), gender (categorical data), salary (positive real data), among other types of data. Despite the diversity of data types, a standard approach for dealing with heterogeneous datasets is to model all attributes, either continuous or discrete, with a single member of the exponential family, e.g., using a Gaussian likelihood. Alternatively, some approaches consider mixed continuous and categorical variables, see Section 2 for a non-exhaustive overview of related contributions.

This paper presents a general latent feature model (GLFM) suitable for heterogeneous datasets, where the attributes describing each object might belong to mixed types of data, either discrete or continuous variables. Specifically, we simultaneously take into account real-valued and positive real-valued as examples of continuous variables, and categorical, ordinal and count data as examples of discrete variables. The proposed model relies on a Bayesian nonparametric prior, called the Indian Buffet Process (IBP), as a building block (Griffiths and Ghahramani, 2011; Teh and Görür, 2009; Broderick et al., 2013). The

IBP induces a prior distribution over binary matrices where the number of columns, corresponding to the number of latent features, is potentially infinite and can be learned from the data along with the other model parameters. The IBP presents several appealing properties. First, the nonparametric nature of the IBP allows to automatically infer the appropriate model complexity, i.e., the number of necessary latent features, from the data. Second, the IBP considers binary-valued latent features which have been shown to provide more interpretable results in data exploration than standard real-valued latent feature models (Ruiz et al., 2012, 2013). The standard linear-Gaussian IBP model assumes binary latent features, Gaussian weighting factors, that capture the influence of every latent feature in each observed attribute, and observations, leading to a conjugate model that permits the use of fast inference algorithms (Doshi-Velez and Ghahramani, 2009; Reed and Ghahramani, 2013; Doshi-Velez et al., 2009). In this paper, we extend the standard linear-Gaussian IBP model to handle heterogeneous datasets, where conjugacy is not straightforwardly preserved.

In order to make inference possible with such a general observation model together with the nonparametric prior for the number of features, we exploit two key ideas. First, we use a data augmentation scheme (Tanner and Wong, 1987) where we introduce an auxiliary real-valued variable, called a *pseudo-observation*, for each observed (continuous or discrete) attribute. Once we condition on the pseudo-observations, the model is the standard linear-Gaussian IBP from Griffiths and Ghahramani (2011). Second, we assume that there exists a function that transforms the pseudo-observation into an actual observation, mapping the real line into the (discrete or continuous) observation space of each attribute in the data. These two key ideas allow to derive an efficient inference algorithm based on the accelerated Gibbs sampler (Doshi-Velez and Ghahramani, 2009), which has linear complexity per MCMC iteration with respect to the number of objects and attributes in the data.

The flexibility and applicability of the proposed model is shown by tackling both prediction and data exploration tasks in several real-world datasets. In particular, we use the proposed model for missing data estimation in heterogeneous datasets. We assume that the missing data is missing completely at random, see (Seaman et al., 2013, Definition 4) for the definition of such concept. For the missing data estimation task, our scheme outperforms the Bayesian probabilistic matrix factorization model (BPMF) (Salakhutdinov and Mnih, 2008) and the standard linear-Gaussian IBP (Griffiths and Ghahramani, 2011), which assume Gaussian observations. These results have been previously discussed by Valera and Ghahramani (2014), where the main focus was missing data estimation or table completion. The extended version presented here focuses on the model itself, providing the necessary details on the GLFM, the corresponding inference scheme for latent feature modeling in heterogeneous datasets and a software toolbox. It provides a powerful tool not only for missing data estimation, but also for exploratory data analysis tasks. In the second part of the experiments we present several examples of how to use the proposed model for data exploration in real-world datasets from diverse application domains such as medicine, psychiatry, sociology and politics.

The source software package is publicly available at `https://github.com/ivaleraM/GLFM`, that provides users with the necessary functions and scripts to use the GLFM for both missing data estimation and data exploration tasks. The core inference algorithm is developed in C++, and the corresponding user interfaces are provided in Matlab, Python and R. A description of the GLFM implementation is provided in Appendix B.

The rest of the paper is organized as follows. In Section 2, a non-exhaustive overview of the existing literature is provided. In Section 3, we provide the details on the general Bayesian nonparametric latent feature model for heterogeneous datasets. In Section **??**, we develop the inference algorithm based on the Gibbs sampler, where the augmented pseudo-observation model is used to collapse the sampler. In Section 5, the model is used for two types of real-world tasks: missing data estimation and data analysis. Finally, in Section 6, potential applications and future work are suggested.

2. Related work

For this reason, Latent variable models are useful for capturing the underlying statistical dependencies via the unobserved random variables. In particular, probabilistic matrix factorization models (Singh and Gordon, 2008) decompose the data as a product of a weight matrix and a feature matrix. Some examples include principal component analysis (PCA) (Pearson, 1901), probabilistic PCA (Tipping and Bishop, 1997; Roweis, 1997), independent component analysis (Hyvärinen, 1997) and factor analysis (Thurstone, 1931), among others. Matrix factorization models are used in a wide range of applications which include, e.g., recommender systems (Gopalan et al., 2014), matrix completion (Salakhutdinov and Mnih, 2007) and dimensionality reduction (Tipping and Bishop, 1997).

Next, we briefly review two basic matrix factorization models: probabilistic PCA and factor analysis. Probabilistic PCA (pPCA) assumes that for each observation vector $\mathbf{x}_n \in \mathbb{R}^D$, there is a latent vector $\mathbf{z}_n \in \mathbb{R}^K$, such that the observation vector can be written as a noisy linear combination of the latent vector, namely, $\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$. The latent vector and the noise are assumed to be Gaussian, $\mathbf{z}_n \sim \mathcal{N}(0, \mathbf{I}_K)$ and $\epsilon_n \sim \mathcal{N}(0, \psi\mathbf{I}_D)$ and $K < D$. In pPCA: i) the maximum likelihood solution for $\mathbf{W}$ given a fixed ψ lies in the K-principal subspace of the data; ii) when $\psi = 0$, we recover the classical PCA; and iii) the marginal likelihood is given by $p(\mathbf{x}_n) = \mathcal{N}(\mathbf{x}_n \mid \mathbf{0}, \mathbf{W}\mathbf{W}^T + \psi\mathbf{I}_D)$. Note that pPCA assumes that the noise variance ψ is shared across all the dimensions in the data, which in general is a restrictive assumption. Factor analysis (FA) generalizes pPCA by assuming a more general distribution for ϵ_n such that $\epsilon_n \sim \mathcal{N}(0, \mathbf{\Psi})$, where $\mathbf{\Psi}$ is a diagonal matrix. As a consequence, the marginal likelihood is distributed according to $p(\mathbf{x}_n) = \mathcal{N}(\mathbf{x}_n \mid \mathbf{0}, \mathbf{W}\mathbf{W}^T + \mathbf{\Psi})$. These two basic matrix factorization models assume that the data lies in a lower dimensional manifold and that the corresponding marginal distribution is Gaussian.

Both pPCA and FA are known to be unidentifiable, being in general difficult to establish the identifiability of the parameters of latent variable models. In order to ensure identifiability, most Bayesian factor analysis models rely on a lower-triangular specification for the factor loading matrix. This idea was originally proposed by Anderson and Rubin (1956) (see also Geweke and Zhou (1996); Aguilar and West (2000); Lopes and West (2004); Frühwirth-Schnatter and Lopes (2010), and references therein). Typically, in order to ensure identifiability one can pick a specific ordering of the factor loadings by setting the upper triangle of the loadings matrix to be zero a priori. Alternatively, Conti et al. (2014) introduce a novel way to ensure identifiability of a Bayesian exploratory factor analysis model by incorporating identifying criteria into the factor distribution of model parameters.

In order to have more general assumptions about the data's generating mechanism, some alternatives have been proposed such as mixtures of factor analyzers (Ghahramani and

Hinton, 1996), nonparametric mixtures of factor analyzers (Görür and Rasmussen, 2009) and nonparametric factor analyzers with Beta process priors (Paisley and Carin, 2009). In the Bayesian nonparametric context, most methods usually focus on covariance structure, variable selection or prediction, and identification is not strictly required to achieve these goals from a Bayesian perspective. All of these models assume that the distribution of the data can be approximated arbitrarily well with a mixture distribution given enough number of mixture components. In order for this to be true, the assumption of continuous support is crucial and a Gaussian distribution for each mixture component is typically used.

Relaxing the assumption of Gaussianity might lead to potential benefits in terms of predictive accuracy or more meaningful latent representations, but it makes the inference problem more challenging due to the presence of non-analytic likelihood functions. Non-Gaussian likelihood models have been considered before, for instance, discrete component analysis (Buntine and Jakulin, 2006), exponential family PCA (Collins et al., 2002; Mohamed et al., 2009), exponential family partial least squares (Klami et al., 2012), and latent variable models with non-conjugate likelihoods (Wang and Blei, 2013; Khan et al., 2013). Nonparametric counterparts of some of these models have also been proposed (Miller et al., 2009; Ruiz et al., 2014; Hoffman et al., 2010; Lee et al., 2015; Hannah et al., 2011). All of these works consider homogeneous datasets, since they assume the same likelihood model, often from the exponential family, for all the observed variables in a dataset.

In addition, only a few works consider mixed continuous and discrete variables (Gunasekar et al., 2014; Khan et al., 2010). Khan et al. (2010) proposes a variational EM algorithm to perform fast inference in factor analysis models with mixed continuous and categorical observations. The performance of the proposed method is evaluated in a missing data estimation task, for which code is provided by the authors in `https://emtiyaz.github.io/software/mixedDataFA.html`. In Section 6, we compare the performance of this approach with the proposed GLFM for the missing data imputation task. More recently, Gunasekar et al. (2014) introduced an exponential matrix factorization model under structural constraints, which accounts for heterogeneous noise within the exponential family. In this rather theoretical work, the model parameters are learned by convex optimization and data imputation is done using maximum likelihood. Remarkably, none of the existing contributions can handle discrete data of ordinal type. The reason for this is that the only member of the exponential family suitable for ordinal data is the multinomial, which does not take into account the inherent order between categories. In many situations, ordinal data are part of real datasets, for example, the severity scores of a disease, the quality levels of a product, or the frequency of an action.

Our proposed model provides a general framework for heterogeneous datasets: it is useful for mixed continuous, real-valued and positive real-valued attributes, and for discrete attributes, categorical, ordinal and count data. It performs dimensionality reduction in the same way as in matrix factorization models. It uses a Bayesian nonparametric prior to infer the number of latent features from the data and uses a variety of generalized linear model link functions to handle heterogeneous datasets. Importantly, this work is accompanied by a software package that implements the proposed GLFM, allowing users to perform a variety of tasks. In the next sections, we detail the proposed model and inference algorithm and analyze in detail several real-world applications of the proposed GLFM.

3. Latent feature model for heterogeneous data

We assume that the data can be stored in an observation matrix $\mathbf{X}$ of size $N \times D$, each of the N objects is defined by a set of D attributes. Let x_n^d denote each entry of the observation matrix $\mathbf{X}$, which might be of the following types:

- Continuous variables:
  1. Real-valued, $x_n^d \in \Re$
  2. Positive real-valued, $x_n^d \in \Re_+$.
- Discrete variables:
  1. Categorical data, x_n^d takes a value in a finite unordered set, e.g., $x_n^d \in$ {'blue', 'red', 'black'}.
  2. Ordinal data, x_n^d takes values in a finite ordered set, e.g., $x_n^d \in$ {'never', 'sometimes', 'often', 'usually', 'always'}.
  3. Count data, $x_n^d \in \{0, \ldots, \infty\}$.

As in standard latent feature allocation models, we assume that x_n^d can be explained by a K-length vector of latent features associated to the n-th data point, $\mathbf{z}_n = [z_{n1}, \ldots, z_{nK}]$, and a weight vector[1] $\mathbf{B}^d = [b_1^d, \ldots, b_K^d]^T$ (K is the number of latent variables), whose elements b_k^d weight the contribution of the k-th latent feature to the d-th dimension of $\mathbf{X}$. The corresponding likelihood can be factorized as follows

$$p(\mathbf{X}|\mathbf{Z}, \{\mathbf{B}^d, \Psi^d\}_{d=1}^D) = \prod_{d=1}^D \prod_{n=1}^N p(x_n^d|\mathbf{z}_n, \mathbf{B}^d, \Psi^d),$$

where Ψ^d denotes the set of random variables necessary to define the distribution of the d-th attribute. The binary-valued latent binary feature vectors $\mathbf{z}_n$ are stored in an $N \times K$ matrix $\mathbf{Z}$ that follows an IBP prior with concentration parameter α, denoted by $\mathbf{Z} \sim \text{IBP}(\alpha)$ (Griffiths and Ghahramani, 2011). Additionally, a Gaussian distribution with zero mean and covariance matrix, given by $\sigma_B^2 \mathbf{I}_K$, is assumed for the weight vectors $\mathbf{B}^d$.

If $x_n^d \in \Re$ is assumed to be Gaussian, for each of the $d = 1, \ldots, D$ attributes, with mean $\mathbf{z}_n\mathbf{B}^d$, where $\mathbf{z}_n\mathbf{B}^d$ denotes the usual vector multiplication, then, the above model is equivalent to the standard IBP with Gaussian observations (Griffiths and Ghahramani, 2011). This model can be efficiently learnt using the properties of the Gaussian distribution (Doshi-Velez and Ghahramani, 2009). However, if the observation matrices are heterogeneous or non-Gaussian then, the inference algorithm from Doshi-Velez and Ghahramani (2009) cannot be used directly. The reason is that the priors are no longer conjugate and the model becomes intractable.

We propose an augmentation of the original model to solve the intractability due to non-conjugacy. An auxiliary Gaussian variable y_n^d is introduced per entry x_n^d in the observation matrix, called *pseudo-observation*. We assume that there exists a link function $f_d(\cdot)$ over the variables y_n^d to obtain the observations x_n^d, mapping the real line $\Re$ into the observation space of the d-th attribute in the observation matrix Ω_d, i.e.,

$$\begin{array}{rccc} f_d: & \Re & \mapsto & \Omega_d \\ & y_n^d & \to & x_n^d \end{array} . \qquad (1)$$

[1] For convenience, we capitalize here the notation for the weight vectors $\mathbf{B}^d$.

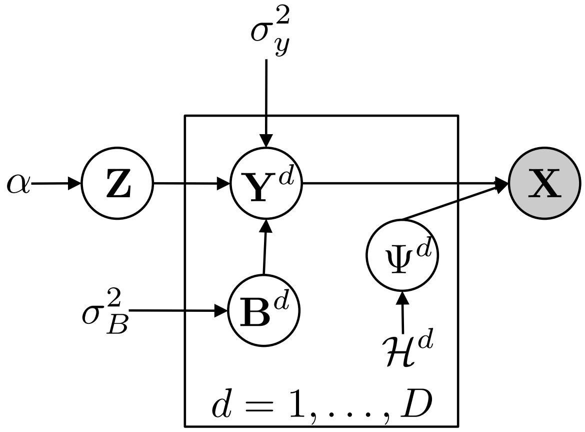


Figure 1: **Graphical Model for the Generalized Latent Feature Model.** Grey nodes represent observed variables, white nodes correspond to latent variables. Introducing the pseudo-observations $\mathbf{Y}^d$ allow us to deal with heterogeneous data.

Each *pseudo-observation* y_n^d is Gaussian distributed with mean $\mathbf{z}_n\mathbf{B}^d$ and variance σ_y^2, i.e.,

$$p(y_n^d|\mathbf{z}_n, \mathbf{B}^d) = \mathcal{N}(y_n^d|\mathbf{z}_n\mathbf{B}^d, \sigma_y^2),$$

such that, when we conditioned on the pseudo-observations, the latent variable model behaves as the standard linear-Gaussian IBP. In Section 3.1, more details are provided about the functions which map the real line $\Re$ into each of the discrete and continuous spaces of the original attributes. In previous work, auxiliary Gaussian variables have been used to link a latent variable model to the original datapoints for multi-class classification (Girolami and Rogers, 2005) and for ordinal regression (Chu and Ghahramani, 2005). Such approach has not been used to account for mixed continuous and discrete data simultaneously. Furthermore, the existing approaches for the IBP with discrete observations propose non-conjugate likelihood models and approximate inference algorithms (Ruiz et al., 2012, 2013; Valera et al., 2016) which make inference more costly.

The generative model is shown in Figure 4, where $\mathbf{Z}$ is the IBP latent matrix, and $\mathbf{Y}^d$ and $\mathbf{B}^d$ contain, respectively, the pseudo-observations y_n^d and the weight factors b_k^d for the d-th dimension of the data. Additionally, Ψ^d denotes the set of auxiliary random variables needed to obtain the observation vector $\mathbf{x}^d$ given $\mathbf{Y}^d$, and $\mathcal{H}^d$ contains the hyper-parameters associated to the random variables in Ψ^d. It is also possible to extend the latent feature matrix $\mathbf{Z}$ so it contains an extra latent feature that is active for every object in the data. This can be used as a bias term, similar to (Ruiz et al., 2012, 2013; Valera et al., 2016), it allows to obtain more interpretable results while performing data exploration.

3.1. Mapping Functions

In this section, we define the set of functions that transforms the pseudo-observations y_n^d into the corresponding observations x_n^d. These functions map from the real line $\Re$ to the (continuous or discrete) observation space of the d-th attribute describing the data. Since each attribute (dimension) in $\mathbf{X}$ may contain any discrete or continuous data types, we provide a mapping function for each kind of data and the corresponding likelihood function for heterogeneous data.

3.1.1. Continuous Variables

In the case of continuous variables, we assume that the mapping functions are of the form $x = f(y+u)$, where $f(\cdot)$ is a continuous invertible and differentiable function and u corresponds to additive Gaussian noise with variance σ_u^2. The corresponding likelihood function, after integrating out the pseudo-observation y_n^d, is given as follows

$$p(x_n^d|\mathbf{z}_n, \mathbf{B}^d) = \frac{1}{\sqrt{2\pi(\sigma_y^2+\sigma_u^2)}} \exp\left\{-\frac{1}{2(\sigma_y^2+\sigma_u^2)}(f^{-1}(x_n^d) - \mathbf{z}_n\mathbf{B}^d)^2\right\} \left|\frac{d}{dx_n^d} f^{-1}(x_n^d)\right|, \quad (2)$$

where $f^{-1}(\cdot)$ is the inverse of the function $f(\cdot)$, i.e., $f^{-1}(f(v)) = v$. Next, we provide examples of mapping functions for the real-valued and positive real-valued data cases.

Real-valued Data. In order to obtain real-valued observations, i.e., $x_n^d \in \Re$, we need a transformation over y_n^d that maps from the real numbers to the real numbers, i.e., $f_d : \Re \rightarrow \Re$. The simplest case is to assume that $x = f_d(y+u) = y+u$. Therefore, each observation is distributed as $x_n^d \sim \mathcal{N}(\mathbf{z}_n\mathbf{b}_{\Re}^d, \sigma_y^2 + \sigma_u^2)$. Other mapping functions can be used, e.g., one might opt for the following transformation

$$x = f_d(y+u) = w(y+u) + \mu,$$

where w and μ are parameters which allow the user to scale or shift the attribute. A common choice is to choose $w = 1/\text{Var}[\mathbf{x}^d]$ and $\mu = \mathbb{E}[\mathbf{x}^d]$, which normalize the data. The corresponding auxiliary variables and hyper-parameters are $\Psi^d = \{u_d^n\}$ and $\mathcal{H}^d = \{\sigma_u^2, w, \mu\}$.

Positive Real-valued Data. In order to obtain positive real-valued observations, i.e., $x_n^d \in \Re_+$, we can apply any transformation over y_n^d that maps the real numbers to the positive real numbers, i.e., $f_d : \Re \rightarrow \Re_+$, as long as f_d is an invertible and differentiable function. An example of such function is

$$f_d(y) = \log(\exp(wy + \mu) + 1),$$

where w and μ are hyper-parameters. Similarly to the case of real-valued attributes, we also use the Gaussian variable u_n^d to obtain x_n^d from y_n^d, therefore, $\Psi^d = \{u_d^n\}$ and $\mathcal{H}^d = \{\sigma_u^2, w, \mu\}$.

3.1.2. Discrete Variables

In the case of discrete variables, there is no general way to map the real line into a generic type of discrete variable. Therefore, we derive a different transformation that is tailored for each of the specific types of discrete variables, i.e., categorical, ordinal and count data.

Categorical Data. Let x_n^d be a categorical observation, namely, it can take values in the index set given by $\{1, \ldots, R_d\}$. Hence, assuming a multinomial probit model, we can then write

$$x_n^d = f_d(y_n^d) = \arg\max_{r\in\{1,\ldots,R_d\}} y_{nr}^d, \quad (3)$$

with $y_{nr}^d \sim \mathcal{N}(y_{nr}^d|\mathbf{z}_n\mathbf{b}_r^d, \sigma_y^2)$ where $\mathbf{b}_r^d$ denotes the K-length weight (column) vector, in which each b_{kr}^d measures the influence of the k-th feature for the observation x_n^d taking value r. Under this likelihood model, we have as many pseudo-observations y_{nr}^d and weight vectors $\mathbf{b}_r^d$ per observation as number of categories in the d-th attribute, i.e., $r \in \{1, \ldots, R_d\}$. In this case, the pseudo-observations can be stored in the $N \times R_d$ matrix $\mathbf{Y}^d$ and the weight factors in a $K \times R_d$ matrix $\mathbf{B}^d$. Under this observation model, we can write $y_{nr}^d = \mathbf{z}_n\mathbf{b}_r^d + u_{nr}^d$, where u_{nr}^d is Gaussian noise with variance σ_y^2. Analogously to Girolami and Rogers (2005), the probability of each element x_n^d taking a value $r \in \{1, \ldots, R_d\}$ is obtained as follows

$$p(x_n^d = r|\mathbf{z}_n, \mathbf{B}^d) = \mathbb{E}_{p(u)}\left[\prod_{\substack{j=1\\j\neq r}}^{R_d} \Phi\Big(u + \mathbf{z}_n(\mathbf{b}_r^d - \mathbf{b}_j^d)\Big)\right], \tag{4}$$

where the subscript r in $\mathbf{b}_r^d$ refers to the column in $\mathbf{B}^d$ ($r \in \{1, \ldots, R_d\}$), $\Phi(\cdot)$ denotes the cumulative distribution function of the standard normal distribution and $\mathbb{E}_{p(u)}[\cdot]$ denotes expectation with respect to the distribution $p(u) = \mathcal{N}(0, \sigma_y^2)$. Then, the auxiliary variables and hyper-parameters are defined as $\Psi^d = \{u_d^n\}$ and $\mathcal{H}^d = \{\sigma_u^2\}$.

Ordinal Data. Consider ordinal data, in which each element x_n^d takes values in the ordered index set $\{1, \ldots, R_d\}$. Then, assuming an ordered probit model, we can write

$$x_n^d = f_d(y_n^d) = \begin{cases} 1 & \text{if } y_n^d \leq \theta_1^d \\ 2 & \text{if } \theta_1^d < y_n^d \leq \theta_2^d \\ & \vdots \\ R_d & \text{if } \theta_{R_d-1}^d < y_n^d \end{cases} \tag{5}$$

where again y_n^d is Gaussian distributed with mean $\mathbf{z}_n\mathbf{B}^d$ and variance σ_y^2, and θ_r^d for $r \in \{1, \ldots, R_d - 1\}$ are the thresholds that divide the real line into R_d regions. We assume that the thresholds θ_r^d are sequentially generated from the truncated Gaussian distribution $\theta_r^d \propto \mathcal{N}(\theta_r^d|0, \sigma_\theta^2)\mathbb{I}(\theta_r^d > \theta_{r-1}^d)$, where $\theta_0^d = -\infty$ and $\theta_{R_d}^d = +\infty$. In this case, the value of x_n^d is determined by the region in which y_n^d falls and, as opposed to the categorical case. A unique weight vector $\mathbf{B}^d$ and a unique Gaussian variable y_n^d are obtained for each observation x_n^d.

Under the ordered probit model (Chu and Ghahramani, 2005), the probability of each element x_n^d taking value $r \in \{1, \ldots, R_d\}$ can be written as

$$p(x_n^d = r|\mathbf{z}_n, \mathbf{B}^d) = \Phi\left(\frac{\theta_r^d - \mathbf{z}_n\mathbf{B}^d}{\sigma_y}\right) - \Phi\left(\frac{\theta_{r-1}^d - \mathbf{z}_n\mathbf{B}^d}{\sigma_y}\right). \tag{6}$$

As a final remark, if the d-th dimension of the observation matrix contains ordinal data, the set of auxiliary variables reduces to the thresholds $\Psi^d = \{\theta_1^d, \ldots, \theta_{R_d-1}^d\}$, and thus, $\mathcal{H}^d = \{\sigma_\theta^2\}$.

Count Data. In the case of count data, each observation x_n^d takes non-negative integer values, , $x_n^d \in \{0, \ldots, \infty\}$. Then, we assume that the observations are given by

$$x_n^d = f_d(y_n^d) = \lfloor f_{\Re_+}(y_n^d) \rfloor, \tag{7}$$

where $\lfloor v \rfloor$ returns the floor of v, that is the largest integer that does not exceed v, and $f_{\Re_+}: \Re \to \Re_+$ is an invertible function that maps the real numbers to the positive real numbers. We can therefore write the likelihood function as

$$p(x_n^d|\mathbf{z}_n, \mathbf{B}^d) = \Phi\left(\frac{f^{-1}(x_n^d + 1) - \mathbf{z}_n\mathbf{B}^d}{\sigma_y}\right) - \Phi\left(\frac{f^{-1}(x_n^d) - \mathbf{z}_n\mathbf{B}^d}{\sigma_y}\right), \tag{8}$$

where $f_{\Re_+}^{-1}: \Re_+ \to \Re$ is the inverse function of the transformation $f_{\Re_+}(\cdot)$. In this case, there are no auxiliary random variables Ψ^d or hyper-parameters $\mathcal{H}^d$ and both these sets are empty.

4. Inference

In this section, we describe the algorithm for learning the latent variables given the observation matrix. In order to jointly learn the latent vectors $\mathbf{z}_n$, the weight factors $\mathbf{B}^d$, and the auxiliary variables Ψ^d, we use a Markov Chain Monte Carlo (MCMC) inference scheme. MCMC methods have been broadly applied to infer the IBP matrix (Griffiths and Ghahramani, 2011; Williamson et al., 2010; Titsias, 2007). The proposed inference algorithm, summarized in Algorithm 1, exploits the information in the available data to learn similarities among objects, captured by the latent feature matrix $\mathbf{Z}$. The inference scheme identifies how the latent features show up in the attributes that describe the objects, captured by $\mathbf{B}^d$.

In Algorithm 1, we first update the latent matrix $\mathbf{Z}$. Note that conditioned on $\{\mathbf{Y}^d\}_{d=1}^D$, both the latent matrix $\mathbf{Z}$ and the weight matrices $\{\mathbf{B}^d\}_{d=1}^D$ are independent of the observation matrix $\mathbf{X}$. Additionally, since $\{\mathbf{B}^d\}_{d=1}^D$ and $\{\mathbf{Y}^d\}_{d=1}^D$ are Gaussian distributed, we can marginalize out the weight matrices $\{\mathbf{B}^d\}_{d=1}^D$ to obtain $p(\{\mathbf{Y}^d\}_{d=1}^D|\mathbf{Z})$. In order to learn matrix $\mathbf{Z}$, we apply the collapsed Gibbs sampler which presents better mixing properties than the uncollapsed version. For this reason, it is the common method of choice in the context of the standard linear-Gaussian IBP (Griffiths and Ghahramani, 2011). However, using an MCMC algorithm with such representation of the model has a high computational cost: it is cubic in the number of data points N at every iteration, which is a prohibitive cost when the dataset is big. Instead, we use the accelerated Gibbs sampler (Doshi-Velez and Ghahramani, 2009), a fast, albeit approximate, scheme for inference. This algorithm presents linear complexity with respect to the number of objects N in the observation matrix per MCMC iteration.

Second, we sample the weight factors in $\mathbf{B}^d$, which is a $K \times R_d$ matrix in the case of categorical attributes, and a K-length column vector, otherwise. We denote each column vector in $\mathbf{B}^d$ by $\mathbf{b}_r^d$. The posterior over the weight vectors is given by

$$p(\mathbf{b}_r^d|\mathbf{y}_r^d, \mathbf{Z}) = \mathcal{N}(\mathbf{b}_r^d|\mathbf{P}^{-1}\boldsymbol{\lambda}_r^d, \mathbf{P}^{-1}), \tag{9}$$

where $\mathbf{P} = \mathbf{Z}^\top\mathbf{Z} + 1/\sigma_B^2\mathbf{I}_k$ and $\boldsymbol{\lambda}_r^d = \mathbf{Z}^\top\mathbf{y}_r^d$, with $\mathbf{y}_r^d$ the r-th column of $\mathbf{Y}^d$. Here, r takes values in $\{1, \ldots, R_d\}$ in the case of categorical observations, while $r = 1$ for the rest of the variable types. Since the covariance matrix $\mathbf{P}^{-1}$ does not depend on the dimension d or on r, we only need to invert the $K \times K$ matrix $\mathbf{P}$ once per iteration. In Section 4.1, we describe how to efficiently sample $\mathbf{Z}$, as well as how to efficiently compute $\mathbf{P}$ after the corresponding changes are made to the matrix $\mathbf{Z}$ by rank one updates. For this reason, we managed to bypass the computation of the matrix product $\mathbf{Z}^\top\mathbf{Z}$. Once we have updated $\mathbf{Z}$ and $\mathbf{B}^d$, we sample each element in $\mathbf{Y}^d$ from the distribution $\mathcal{N}(y_{nr}^d|\mathbf{z}_n\mathbf{b}_r^d, \sigma_y^2)$ if the observation x_n^d is missing, or from the posterior $p(y_{nr}^d|x_n^d, \mathbf{z}_n, \mathbf{B}^d)$ specified in Section 4.2, otherwise. Finally, we sample the auxiliary variables in Ψ^d from their posterior distributions if necessary.[2] See Section 4.2 for more details about the posterior distributions in Ψ^d. In the worst case, the last two steps consist of sampling from a doubly truncated univariate normal distribution, we used the algorithm in Robert (1995).

Algorithm 1 Inference Algorithm.

```
Input: X
Initialize: $\mathbf{Z}$ and $\{\mathbf{Y}^d\}_{d=1}^D$
for each iteration do
    Update $\mathbf{Z}$ given $\{\mathbf{Y}^d\}_{d=1}^D$ as detailed in Section 4.1.
    for $d = 1, \dots, D$ do
        Sample $\mathbf{B}^d$ given $\mathbf{Z}$ and $\mathbf{Y}^d$ according to (9).
        Sample $\mathbf{Y}^d$ given $\mathbf{X}$, $\mathbf{Z}$ and $\mathbf{B}^d$ as shown in Section 4.2.
        Sample $\Psi^d$ (if needed) as shown in Section 4.2.
    end for
end for
Output: $\mathbf{Z}$, $\{\mathbf{B}^d\}_{d=1}^D$ and $\{\Psi^d\}_{d=1}^D$
```

4.1. Details on the Accelerated Gibbs Sampler

In this section, we review and adapt the sampler in Doshi-Velez and Ghahramani (2009). The authors introduce a linear-time accelerated Gibbs sampler for conjugate IBP models that effectively marginalizes out the weight factors. The per-iteration complexity of this algorithm is $\mathcal{O}(N(K^2 + KD))$, which is comparable to the uncollapsed linear-Gaussian IBP sampler that has per-iteration complexity of $\mathcal{O}(NDK^2)$ but does not marginalize out the weight factors. The uncollapsed version of the algorithm presents a slower convergence rate. In this paper, we adapt this algorithm for the proposed IBP model for heterogeneous data.

The accelerated Gibbs sampling algorithm exploits the Bayes rule to avoid the cubic complexity per MCMC iteration with respect to N due to the computation of the marginal likelihood in the collapsed Gibbs sampler. In particular, it uses the Bayes rule to obtain the probability of each element in the latent feature matrix $\mathbf{Z}$ of being active as

$$p(z_{nk} = 1|\{\mathbf{Y}^d\}_{d=1}^D, \mathbf{Z}_{\neg nk}) \propto \frac{m_{\neg n,k}}{N} \prod_{d=1}^{D} \prod_{r=1}^{S_d} \int_{\mathbf{b}_r^d} p(y_{nr}^d|\mathbf{z}_n, \mathbf{b}_r^d) p(\mathbf{b}_r^d|\mathbf{y}_{\neg nr}^d \mathbf{Z}_{\neg n}) d\mathbf{b}_r^d, \qquad (10)$$

[2] The set of auxiliary variables for the d-dimension, Ψ^d, can be augmented to contain the variance of the pseudo-observations $\mathbf{Y}^d$ associated to the d-th attribute, which we denote by σ_d^2 and for which we assume an inverse-gamma prior with parameters β_1 and β_2. Under this prior distribution, the posterior of σ_d^2 is an inverse-gamma with parameters $\beta_1 + NS_d/2$ and $\beta_2 + \sum_{n=1}^N \sum_{r=1}^{S_d} (y_{nr}^d - \mathbf{z}_n\mathbf{b}_r^d)/2$, where S_d is equal to the number of categories R_d for those dimensions d that contain categorical attributes, and it is equal to $S_d = 1$, otherwise.

where S_d is the number of columns in matrices $\mathbf{Y}^d$ and $\mathbf{B}^d$, $\mathbf{Z}_{\neg n}$ corresponds to matrix $\mathbf{Z}$ after removing the n-th row, vector $\mathbf{y}^d_{\neg nr}$ is the r-th column of matrix $\mathbf{Y}^d$ without the element y^d_{nr}. Specifically, S_d is the number of categories R_d for those dimensions d that contain categorical attributes, and it is equal to one, otherwise. The conditional distribution, denoted by $p(\mathbf{b}^d_r|\mathbf{x}^d_{\neg n}, \mathbf{Z}_{\neg n})$, corresponds to the posterior of $\mathbf{b}^d_r$ computed without taking the n-th datapoint into account, i.e.,

$$p(\mathbf{b}^d_r|\mathbf{y}^d_{\neg nr}, \mathbf{Z}_{\neg n}) = \mathcal{N}(\mathbf{b}^d_r|\mathbf{P}^{-1}_{\neg n}\boldsymbol{\lambda}^d_{\neg nr}, \mathbf{P}^{-1}_{\neg n}), \tag{11}$$

where $\mathbf{P}_{\neg n} = \mathbf{Z}^\top_{\neg n}\mathbf{Z}_{\neg n} + 1/\sigma^2_B\mathbf{I}_K$ and $\boldsymbol{\lambda}^d_{\neg ny} = \mathbf{Z}^\top_{\neg n}\mathbf{y}^d_{\neg nr}$ are the natural parameters of the Gaussian distribution. In this case, we condition on the Gaussian pseudo-observations $\{\mathbf{Y}^d\}^D_{d=1}$, instead of the actual observations $\mathbf{X}$, to compute the conditional distribution $p(z_{nk} = 1|\{\mathbf{Y}^d\}^D_{d=1}, \mathbf{Z}_{\neg nk})$.

We use the natural parameterization for the Gaussian distribution over the posterior of $\mathbf{b}^d_r$ in contrast to Doshi-Velez and Ghahramani (2009), who use the mean and covariance matrix. This formulation allows to compute the full posterior over the weight factors as

$$p(\mathbf{b}^d_r|\mathbf{y}^d_r, \mathbf{Z}) = \mathcal{N}(\mathbf{b}^d_r|\mathbf{P}^{-1}\boldsymbol{\lambda}^d_r, \mathbf{P}^{-1}). \tag{12}$$

$\mathbf{P} = \mathbf{P}_{\neg n} + \mathbf{z}^\top_n\mathbf{z}_n$ and $\boldsymbol{\lambda}^d_r = \boldsymbol{\lambda}^d_{\neg nr} + \mathbf{z}^\top_n y^d_{nr}$ are the natural parameters of the Gaussian distribution.

In the accelerated Gibbs sampling scheme, we iteratively sample the value of each element z_{nk}, after marginalizing out the weight factors $\mathbf{B}^d$, according to

$$p(z_{nk} = 1|\{\mathbf{Y}^d\}^D_{d=1}, \mathbf{Z}_{\neg nk}) \propto \frac{m_{\neg n,k}}{N} \prod_{d=1}^{D}\prod_{r=1}^{S_d} \mathcal{N}(y^d_{nr}|\mathbf{z}_n\boldsymbol{\lambda}^d_{\neg nr}, \mathbf{z}_n\mathbf{P}_{\neg n}\mathbf{z}^\top_n + \sigma^2_y). \tag{13}$$

For each object n, we first sample the existing latent features z_{nk} for $k = 1, \ldots, K_+$, where K_+ is the number of non-zero columns in $\mathbf{Z}$, or number of active features up to this iteration. Successively, we sample the number of new features necessary to explain that data point from a Poisson distribution with mean α/N, as proposed by Griffiths and Ghahramani (2011).

4.2. Posterior Distribution over the Pseudo-observations

In Algorithm 1, we sample the pseudo-observations y^d_{nr} and the auxiliary variables in Ψ^d from their corresponding posterior distributions. The posterior distributions for y^d_{nr}, and for Ψ^d if needed, for all the considered types of data are given by

1. For real-valued observations

$$p(y^d_{n1}|x^d_n, \mathbf{z}_n, \mathbf{B}^d) = \mathcal{N}\left(y^d_{n1}\middle|\left(\frac{\mathbf{z}_n\mathbf{b}^d_1}{\sigma^2_y} + \frac{f^{-1}_d(x^d_n)}{\sigma^2_u}\right)\left(\frac{1}{\sigma^2_y} + \frac{1}{\sigma^2_u}\right)^{-1}, \left(\frac{1}{\sigma^2_y} + \frac{1}{\sigma^2_u}\right)^{-1}\right), \tag{14}$$

where $f^{-1}_d : \Re \to \Re$.

2. For positive real-valued observations

$$p(y_{n1}^d|x_n^d, \mathbf{z}_n, \mathbf{B}^d) = \mathcal{N}\left(y_{n1}^d \middle| \left(\frac{\mathbf{z}_n \mathbf{b}_1^d}{\sigma_y^2} + \frac{f_d^{-1}(x_n^d)}{\sigma_u^2}\right)\left(\frac{1}{\sigma_y^2} + \frac{1}{\sigma_u^2}\right)^{-1}, \left(\frac{1}{\sigma_y^2} + \frac{1}{\sigma_u^2}\right)^{-1}\right), \tag{15}$$

where $f_d^{-1} : \Re_+ \rightarrow \Re$.

3. For categorical observations

$$p(y_{nr}^d|x_n^d = T, \mathbf{z}_n, \mathbf{B}^d) = \begin{cases} \mathcal{N}(y_{nr}^d|\mathbf{z}_n\mathbf{b}_r^d, \sigma_y^2)\mathbb{I}(y_{nr}^d > \max_{j\neq r}(y_{nj}^d)) & \text{If} \quad r = T \\ \mathcal{N}(y_{nr}^d|\mathbf{z}_n\mathbf{b}_r^d, \sigma_y^2)\mathbb{I}(y_{nr}^d < y_{nT}^d) & \text{If} \quad r \neq T \end{cases} \tag{16}$$

In Equation (16), if $x_n^d = T = r$, we sample y_{nr}^d from a Gaussian left-truncated by $\max_{j\neq r}(y_{nj}^d)$. Otherwise, we sample from a Gaussian right-truncated by y_{nr}^d with $r = x_n^d$. Sampling the variables y_{nr}^d corresponds to solving a multinomial probit regression problem. In order for the model to be identifiable, without loss of generality, we assume that the regression function $f_{R_d}(\mathbf{z}_n)$ is identically zero. Therefore, we fix $b_{kR_d}^d = 0$ for all k.

4. For ordinal observations:

$$p(y_{n1}^d|x_n^d = r, \mathbf{z}_n, \mathbf{B}^d) \sim \mathcal{N}(y_{n1}^d|\mathbf{z}_n\mathbf{b}_1^d, \sigma_y^2)\mathbb{I}(\theta_{r-1}^d < y_{n1}^d \leq \theta_r^d). \tag{17}$$

In this case, we sample y_{n1}^d form a Gaussian left-truncated by θ_{r-1}^d and right-truncated by θ_r^d. In this case, we also need to sample the threshold values θ_r^d with $r = 1, \ldots, R_d-1$ as

$$\begin{aligned} p(\theta_r^d|y_{n1}^d) \sim & \mathcal{N}(\theta_r^d|0, \sigma_\theta^2)\mathbb{I}(\theta_r^d > \max(\theta_{r-1}^d, \max_n(y_{n1}^d|x_n^d = r)) \\ & \times \mathbb{I}(\theta_r^d < \min(\theta_r^d, \min_n(y_{n1}^d|x_n^d = r+1)). \end{aligned} \tag{18}$$

In this case, sampling the variables y_{n1}^d corresponds to solving an ordered probit regression problem, where the thresholds $\{\theta_r\}_{r=1}^{R_d}$ are unknown. In order for this part of the model to be identifiable, we set one of the thresholds, θ_1, to zero.

5. For count observations

$$p(y_{n1}^d|x_n^d, \mathbf{z}_n, \mathbf{B}^d) = \mathcal{N}(y_{n1}^d|\mathbf{z}_n\mathbf{b}_1^d, \sigma_y^2)\mathbb{I}(f_{\Re_+}^{-1}(x_n^d) \leq y_{n1}^d < f^{-1}(x_n^d + 1)), \tag{19}$$

where $f_{\Re_+}^{-1} : \Re_+ \rightarrow \Re$. We sample y_{n1}^d from a Gaussian left-truncated by $f_{\Re_+}^{-1}(x_n^d)$ and right-truncated by $f_{\Re_+}^{-1}(x_n^d + 1)$.

5. Applications

In this section, we apply the proposed model to solve two different tasks on several real-world datasets. In Section 5.1, we focus on a prediction task in which we aim to estimate and replace the missing data, which is assumed to be missing completely at random. These results have been previously introduced in Valera and Ghahramani (2014). In Section 5.2, we focus on a data analysis task on several real-world datasets from different application domains including psychiatry, clinical trials and politics. We show how to use the proposed model to perform exploratory data analysis, i.e., to find the latent structure in the data and capture the statistical dependencies among the objects and their attributes in the data.

5.1. Missing Data Estimation

In this section, we use the proposed model to estimate missing data in heterogeneous datasets, where we assume that the data is missing completely at random (MCAR) (Seaman et al., 2013). Missing data may occur in diverse applications due to different reasons. For example, participants of a survey may decide not to respond or skip some questions of the survey; participants in a clinical study may drop out during the course of the study; or users of a recommendation system might only be able to rate a small fraction of the available books, movies, or songs, due to time constraints. The presence of missing values is challenging when the data is used for reporting, information sharing and decision support. As a consequence, handling missing data has captured attention in diverse areas of data science such as machine learning, data mining, and data warehousing and management (Schafer and Graham, 2002; Mazumder et al., 2010).

Most of the extensive literature in probabilistic missing data estimation and imputation focuses on homogeneous datasets which contain only either continuous data, usually modeled as Gaussian variables (Todeschini et al., 2013), or discrete data, that can be either modeled by discrete likelihoods (Li, 2009) or simply treated as Gaussian variables (Salakhutdinov and Mnih, 2008; Todeschini et al., 2013). Only a few works consider mixed continuous and discrete variables Khan et al. (2010, 2013). However, to the best of our knowledge, none of the previous approaches consider ordinal data in their likelihood models.

Experimental Setup. We evaluate the predictive power of the proposed model at estimating missing data on five real datasets, which are summarized in Table 1. The datasets contain different numbers of objects and attributes, which cover all the discrete and continuous variables described in Section 3. According to our model, the probability distribution of the observation matrix is fully characterized by the latent matrices $\mathbf{Z}$ and $\{\mathbf{B}^d\}_{d=1}^D$ as well as the auxiliary variables Ψ^d. Hence, if we assume the latent vector $\mathbf{z}_n$ for the n-th datapoint, the weight factors $\mathbf{B}^d$ and the auxiliary variables Ψ^d to be known, we have a probability distribution over missing observations $\mathbf{x}_n^d$. The probability distribution for missing observations can be used to obtain estimates for $\mathbf{x}_n^d$ by sampling from this distribution,[3] or by taking a summary statistic such as mean, mode or median value, once the latent matrix $\mathbf{Z}$ and the latent weight factors $\mathbf{B}^d$ (and Ψ^d) are learnt.

Here, we consider the following benchmark methods for missing data estimation to compare to our proposed general table completion approach, denoted by GLFM:

- The standard linear-Gaussian IBP (Griffiths and Ghahramani, 2011) denoted by SIBP, treating all attributes as Gaussian.

[3] Note that sampling from this distribution might be computationally expensive. In this case, we can easily obtain samples of $\mathbf{x}_n^d$ by exploiting the structure of our model. In particular, we can simply sample the auxiliary Gaussian variables y_n^d given z_n and $\mathbf{B}^d$, and then obtain an estimate for $\mathbf{x}_n^d$ by applying the corresponding transformation, detailed in Section 3.1.

- The Bayesian probabilistic matrix factorization approach (Salakhutdinov and Mnih, 2008) denoted by BPMF, that also treats all attributes in $\mathbf{X}$ as Gaussian distributed.
- The Mixed-Data Factor Analysis approach (Khan et al., 2010) denoted by MFDA, that accounts for mixed Gaussian and categorical variables. Here we model all the numerical variables, i.e., real-valued, positive real-valued and count data, as continuous variables, and all nominal variables including both categorical and ordinal data, as categorical.

We compare the above methods in terms of average imputation error computed as $Err = 1/D\sum_d err(d)$. For numerical variables, we use the normalized root mean squared error (NRMSE), normalized by the range of the variable. For categorical variables, we compute the accuracy error which counts the number of times when the true and the imputed value disagree. For ordinal variables, we compute the displacement error, which computes the difference between the true and the imputed value, divided by the range of the variable (i.e., the total number of ordinal categories minus one). Additional results in terms of predictive log-likelihood are provided in Appendix A.2.

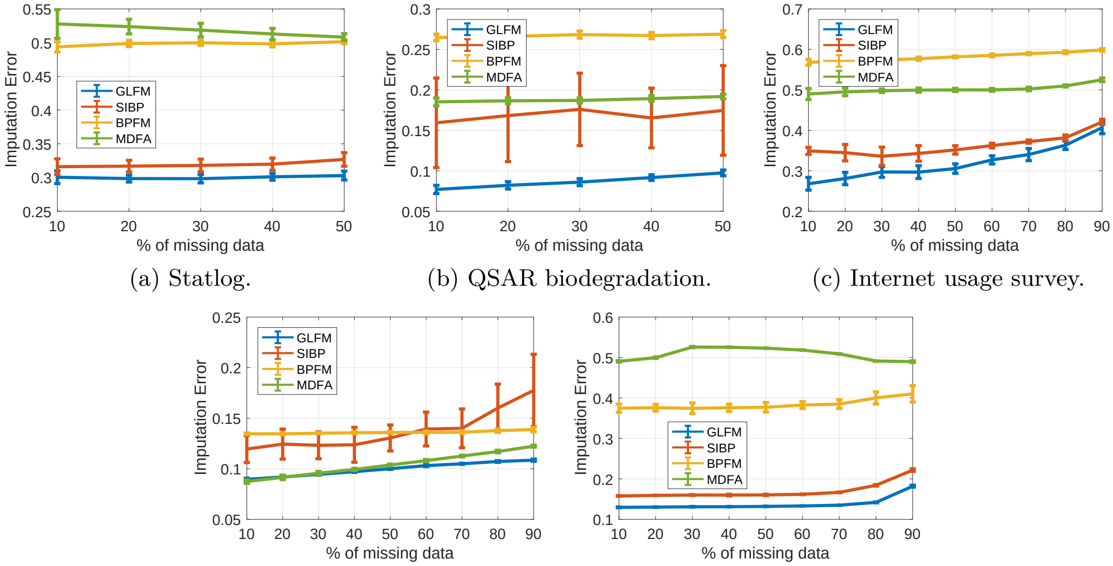
In the GLFM model, for real positive and/or count data, we consider the following transformation that maps from the real numbers to the real positive numbers, $f(x) = \log(\exp(wx) + 1)$. We select the parameter w such that the data is scaled to a common range. For each dataset we run 5,000 iterations of the proposed MCMC sampler from Section 4. The trace plots of the likelihood per iteration to evaluate the convergence of the method are provided in Appendix A.1). Before running SIBP and BPMF, we normalized each column in matrix $\mathbf{X}$ to have zero-mean and unit-variance. This normalization ensures that the Gaussian likelihood evaluations of all the attributes describing the objects in each dataset are comparable, regardless of their discrete or continuous nature. As a consequence, it provides more accurate and fair results than applying the SIBP and BPMF directly on the data without prior normalization.

Additionally, since both SIBP and BPMF assume continuous observations, when dealing with discrete data, we estimate each missing value as the closest integer value to the (denormalized) Gaussian variable. Similarly, when dealing with count data in MFDA, we estimate each missing value as the closest integer value.

Results. Figure 2 shows the average imputation error per missing value as a function of the percentage of missing data. Each value in Figure 2 was obtained by averaging the results across 20 independently split sets where the missing values were randomly chosen. In Figures 2c and 2b, we cut the plot at a missing percentage of 50%. The reason for this cut is that in these two datasets, the discrete attributes present a mode value that appears for more than 80% of the instances. As a consequence, SIBP and BPMF assign probability close to one to the mode, which results in an artificial decrease in the imputation error when larger percentages of missing data are present. We used different numbers of latent features for BPMF and MDFA models: 10, 20 and 50, respectively. We only show the best results for each dataset. Specifically, for BPMF we depict $K = 10$ for the Nesarc and the Wine datasets, and $K = 50$ for the remainder; and for MDFA, we show $K = 50$ for the Wine and Internet dataset, $K = 20$ for the Statlog and Nesarc dataset, and $K = 10$ for the biodegradation dataset. Both GLFM and SIBP have not learnt a number of binary latent features above 25 in any case. As expected, in Figure 2 we observe that the average imputation error tends to increase for the four models as the number of missing values

Dataset	N	D	Description
Statlog German credit dataset (Eggermont et al., 2004)	1,000	20 (10 C + 4 O + 6 N)	Information about the credit risks of the applicants.
QSAR biodegradation dataset (Mansouri et al., 2013)	1,055	41 (2 R + 17 P + 4 C + 18 N)	Molecular descriptors of biodegradable and non-biodegradable chemicals.
Internet usage survey dataset (Centre, 2014)	1,006	32 (23 C + 8 O + 1 N)	Responses of the participants to a survey related to the usage of internet.
Wine quality dataset (Cortez et al., 2009)	6,497	12 (11 P + 1 N)	Results of physicochemical tests realized to different wines.
Nesarc dataset (Ruiz et al., 2013)	43,000	55 C	Responses of the participants to a survey related to personality disorders.

Table 1: **Description of datasets.** 'R' stands for real-valued variables, 'P' for positive real-valued variables, 'C' for categorical variables, 'O' for ordinal variables and 'N' for count variables



(a) Statlog.

(b) QSAR biodegradation.

(c) Internet usage survey.

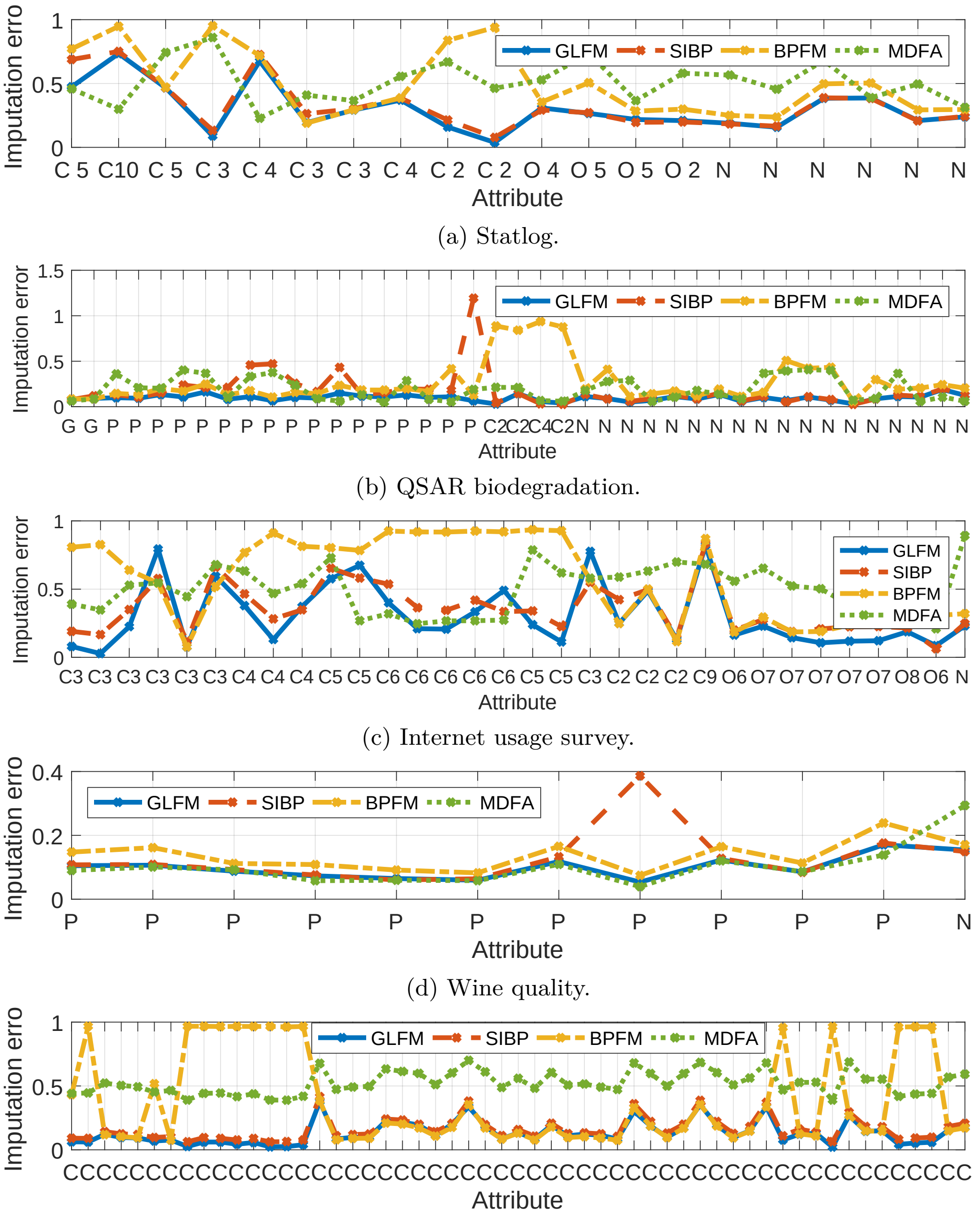
(d) Wine quality.

(e) Nesarc dataset.

Figure 2: **Average test imputation error per missing datum versus percentage of missing data.** The 'whiskers' show one standard deviation away from the average test log-likelihood.

increases. Figure 2 also shows that the proposed GLFM clearly outperforms the other three models for the five datasets. The comparison among SIBP, BPMF and MDFA depends on the dataset. BPMF presented poorer performance in general since it assumes a fixed number of latent features, and Gaussian observations with fix variance.

Successively, we analyzed the performance of the three models for each kind of discrete and continuous variables. Figure 3 shows the average imputation error per missing value for each attribute in the table, which corresponds to each dimension in **X**. In this figure,



(a) Statlog.

(b) QSAR biodegradation.

(c) Internet usage survey.

(d) Wine quality.

(e) Nesarc dataset.

Figure 3: **Average test imputation error per missing datum in each dimension.** Here, we consider 50% of missing data. In the x-axis 'R' stands for real-valued variables, 'P' for positive real-valued variables, 'C' for categorical variables, 'O' for ordinal variables and 'N' for count variables. The number that accompanies 'C' or 'O' corresponds to the number of categories. In the Nesarc dataset, all the variables are binary, i.e., 'C2'.

we have grouped the dimensions according to the kind of data that they contain, the x-axis shows the number of considered categories for the case of categorical and ordinal data. In the case of the Nesarc dataset, all the attributes are binary. In this figure, we observe that while the proposed GLFM leads to low imputation error for all the variables, for the competing models, the imputation error increases drastically for some of the attributes, independently on they statistical data type. Using the Wine data set as an example, we can observe that while the SIBP and MDFA presents similar imputation error than the proposed GLFM for most of the variables, for one of the positive real-valued variables SIBP present imputation error close to one, and MDFA fails at imputing the count variable, In contrast, BPMF provides systematically slightly larger error than GLFM for all the variables. In summary, Figures 2 and 3 shows that the two main properties of GLFM, i.e., unbounded number of latent features and a heterogeneous likelihood model, results in a consistent improvement at imputing missing values with respect to models that cannot handle the heterogenous nature of the datasets or assume a fixed model complexity.

5.2. Data Exploration

In this section, we describe how to use GLFM for data exploration tasks. The main objective of data exploration is to find and analyze the latent structure to *understand* the observed data. A typical approach for data exploration is usually based on: i) applying a dimensionality reduction algorithm, such as for example PCA, factor analysis or clustering, which provides a *summary* of the data; and ii) visualize and analyze this summary. However, as discussed in Section 2, most existing approaches for dimensionality reduction assume homogeneous, and often continuous, observations. We propose the GLFM model as an alternative method for dimensionality reduction that can handle heterogeneous data and is also easily interpretable due to the binary nature of the feature activation vectors. Our approach brings a complementary view to other methods, either by shedding light on novel pieces of knowledge, or by validating previous results in the literature. See Ruiz et al. (2012, 2013); Valera et al. (2016); Utkovski et al. (2018); Pradier et al. (2018) for examples of data exploration using an IBP prior on homogeneous datasets.

An extension of GLFM for biomarker discovery has been successfully used to analyze biomedical data from a phase II of a clinical trial, where the goal was to test the efficacy of a new immunotherapy treatment against hepatocellular carcinoma (Pradier et al., 2019).

First, we provide some general guidelines about how to use the proposed GLFM for data exploration. Then, we provide three showcase examples in the context of i) clinical trials, to discover the effects of a new drug for prostate cancer; ii) psychiatry, to capture the impact of social background in the development of mental disorders; and iii) politics, to identify meaningful demographic profiles, together with their geographic location, and voting tendencies in the United States. To run these experiments, we make use of the GLFM software package which, as detailed in Appendix B, does not only provide functions for dimensionality reduction, for the inference part in the GLFM, but also for the visualization of results.

$$\begin{array}{ccccccc}
\mathbf{z}_n & & \mathbf{B}^d & & y_n^d & & x_n^d \\
\begin{array}{c}[1\;\;0\;\;1\;\;0\;\;\cdots]\\[3ex][0\;\;0\;\;0\;\;1\;\;\cdots]\\[3ex][1\;\;0\;\;0\;\;1\;\;\cdots]\end{array} & \times & \begin{bmatrix}1.2\\3.2\\-2.4\\0.15\\\vdots\end{bmatrix} & \approx & \begin{array}{c}-1.5\\[3ex]0.15\\[3ex]1.35\end{array} & \xrightarrow{f_d(\cdot)} & \begin{array}{l}1\;\;(\text{``}\mathit{low}\text{''})\\[3ex]2\;\;(\text{``}\mathit{medium}\text{''})\\[3ex]2\;\;(\text{``}\mathit{medium}\text{''})\end{array}
\end{array}$$

Figure 4: Example of GLFM model for an ordinal variable.

5.2.1. GLFM for Data Exploration

As detailed in previous sections, the proposed GLFM assumes that each observation x_n can be explained by a potentially infinitely long binary vector $\mathbf{z}_n$ whose elements indicate whether a latent feature is active or not for the n-th object; and a real-valued weight vector $\mathbf{B}^d$ (dictionary element), whose elements weight the influence of each latent feature in the d-th attribute. Since the product of the latent feature vector and the dictionary element leads to a real-valued variable, the GLFM then applies the link function $f_d(\cdot)$ to map the real-valued pseudo-observation y_n^d into the observation x_n^d. Figure 4 illustrates the GLFM for an ordinal attribute taking values in the ordered set $\{$*low, medium, high*$\}$.

In order to perform data exploration, given the dictionary elements $\mathbf{B}^d$ for each d, one can visualize the distribution of each attribute d given a latent feature allocation vector $\mathbf{z}$ (containing either none, one or several features active) by depicting the corresponding marginal likelihood after integrating out the pseudo-observations, $p(x^d|\mathbf{z}, \mathbf{B}^d)$. See Section 3.1 for details on the analytic form of the marginal likelihood for each data type.

Each latent feature vector $\mathbf{z}$ present in the data can be interpreted as a *pattern* which leads to a particular distribution for all the attributes, capturing the statistical dependencies or correlations between the different attributes. Furthermore, similarly to the linear-Gaussian IBP, GLFM also assumes a linear combination of the latent features, while the non-linearity only comes at the level of the likelihood model, due to the transformation $f_d(\cdot)$. As a consequence, the contribution of each latent feature is additive in the observations: if two features increase the activation probability of a value in an attribute (e.g., value 'high' in an ordinal attribute), the joint activation of these two features in a pattern will lead to a higher probability (for that value) than under the activation of only one of the features. Hence, patterns with more than one active latent feature can be seen as combinations of patterns with only one active feature.

Additionally, similarly to Ruiz et al. (2012, 2013); Valera et al. (2016); Utkovski et al. (2018); Pradier et al. (2018), the latent feature vectors in the GLFM can include a bias term. The bias term is a latent feature that is active for every object in the data and may ease the interpretability of results. In the following sections, we activate such bias term and assume that the pattern with no active features, e.g., pattern (000), accounts for this term.

Finally, at every iteration of the inference algorithm, after the burn-in period, we obtain a sample of the joint posterior distribution for the latent variables $\mathbf{Z}$ and $\mathbf{B}^d$ as described in Section 4. It is not possible to identify a correspondence between latent features across samples in the MCMC algorithm due to the non-identifiability caused by the label switching problem. Fortunately, one may think of each joint sample of the latent features $\mathbf{Z}$ and their weight vectors $\mathbf{B}^d$ as a potential *explanation* for the process generating the data. After running our inference algorithm, one has access to as many explanations for the data as number of collected samples during inference, either by running one or several independent

Markov chains to facilitate the exploration of the posterior distribution. We next discuss several ways to select such samples or the explanations to be analyzed.

One option is to select the sample that maximizes the log-likelihood across all the available samples, this corresponds to finding the most likely explanation of the data. Alternatively, one could pick the sample that maximizes the test log-likelihood on a subset of the data which are not used to infer the model. This solution corresponds to the explanation that better generalizes to unseen data. Alternatively, one might be interested in analyzing the different modes of the posterior distribution over $\mathbf{Z}$ and $\mathbf{B}^d$, since they might lead to qualitatively different explanations of the data. In order to distinguish between modes in the posterior, one could look for jumps in the trace-plot of the likelihood evaluation, since different modes tend to result in different likelihood values. This issue is not an intrinsic problem of the proposed GLFM but a general one in Bayesian unsupervised models, such as: pPCA, Gaussian mixture models and topic modeling, among others. There are several contributions focused on how to select the best explanation (or parameter/latent variable configuration) for the data (Roberts et al., 2016; Masood and Doshi-Velez, 2019; Greene et al., 2008; Ross et al., 2017). In the following sections, we explore the posterior sample of $\mathbf{Z}$ and $\mathbf{B}^d$ that maximizes the likelihood across samples from five different Markov chains, as this sample corresponds to the most likely explanation found given our data.

5.2.2. Drug effect in a clinical trial for prostate cancer

Clinical trials aim to determine the safety and efficacy of a new drug before it can be sold in the market. Concretely, the main goal of clinical trials is to prove the efficacy of a new treatment for a disease while ensuring its safety, i.e., check whether its adverse effects remain low enough for any dosage level of the drug. As an example, the publicly available *Prostate Cancer dataset*[4] collects data of a clinical trial that analyzed the effects of the drug diethylstilbestrol (DES) as a treatment against prostate cancer. The dataset contains information about 502 patients with prostate cancer in stages[5] 3 and 4, who entered a clinical trial during 1967-1969 and were randomly allocated to different levels of treatment with DES. The prostate cancer dataset has been used by several studies (Byar and Green, 1980; Kay, 1986; Lunn and McNeil, 1995) to analyze the survival times of patients in the clinical trial and the causes behind their death. All these studies have pointed out that a large dose of the treatment tends to reduce the risk of cancer death, but it might also result in an increased risk of cardiovascular death. In this section, we apply the proposed GLFM to the Prostate Cancer dataset to directly discover the statistical dependencies in the data, which in this example corresponds to the effect of different levels of treatment with DES in the presence of prostate cancer and cardiovascular diseases.

Experimental Setup. The prostate cancer dataset consists of 502 patients and 16 attributes, from which we select five attributes listed in Table 2. The selection of these five attributes allows us to focus only on capturing the statistical dependencies between the

[4] dataset available at: `http://biostat.mc.vanderbilt.edu/wiki/Main/DataSets`

[5] The stage of a cancer describes the size of a cancer and how far it has grown. Stage 3 means that the cancer is already quite large and may have started to spread into surrounding tissues or local lymph nodes. Stage 4 is more severe, and refers to a cancer that has already spread from where it started to another body organ. This is also called secondary or metastatic cancer. Find more details in `http://www.cancerresearchuk.org/about-cancer/what-is-cancer/stages-of-cancer`

Attribute description	Type of variable
Stage of the cancer	Categorical with 2 categories
DES treatment level	Ordinal with 3 categories
Tumor size in cm^2	Count data
Serum Prostatic Acid Phosphatase (PAP)	Positive real-valued
Prognosis Status (outcome of the disease)	Categorical with 4 categories

Table 2: **List of considered attributes for the Prostate Cancer dataset.**

target attributes, i.e, the relationship between the different levels of treatment with DES and the suffering of prostate cancer and cardiovascular diseases. In our experiments, we sample the variance of the pseudo-observations in each dimension and choose the parameter values as follows: $\alpha = 5$, $\sigma_B^2 = 1$, and $\sigma_\theta^2 = 1$. We also consider the following transformation that maps from the real numbers to the positive real numbers, for the positive real and count data: $f(x) = \log(w \cdot (x - \mu) + 1)$, where $\mu = \min(\mathbf{x}^d)$ and $w = 2/\text{std}(\mathbf{x}^d)$ are data-driven parameters whose objective is to shift and scale the data. In order to obtain more interpretable results, we also activated the bias term, as explained in Section 5.2.1.

Feature	Empirical Prob.	Main implications
F1	0.1952	Favours stage 3, low DES levels and prostatic death
F2	0.2689	Favours stage 3, highest DES levels, and cardiovascular death
F3	0.1594	Favours stage 4, low DES levels, and mid-level prostatic death
F4	0.1155	Favours stage 4, low DES levels, and most severe prostatic cancer

Table 3: **Empirical feature activation probabilities in the Prostate Cancer dataset.** These probabilities are directly computed from the inferred IBP matrix **Z**. Additionally, the table summarizes the main implications of the activation of each latent feature.

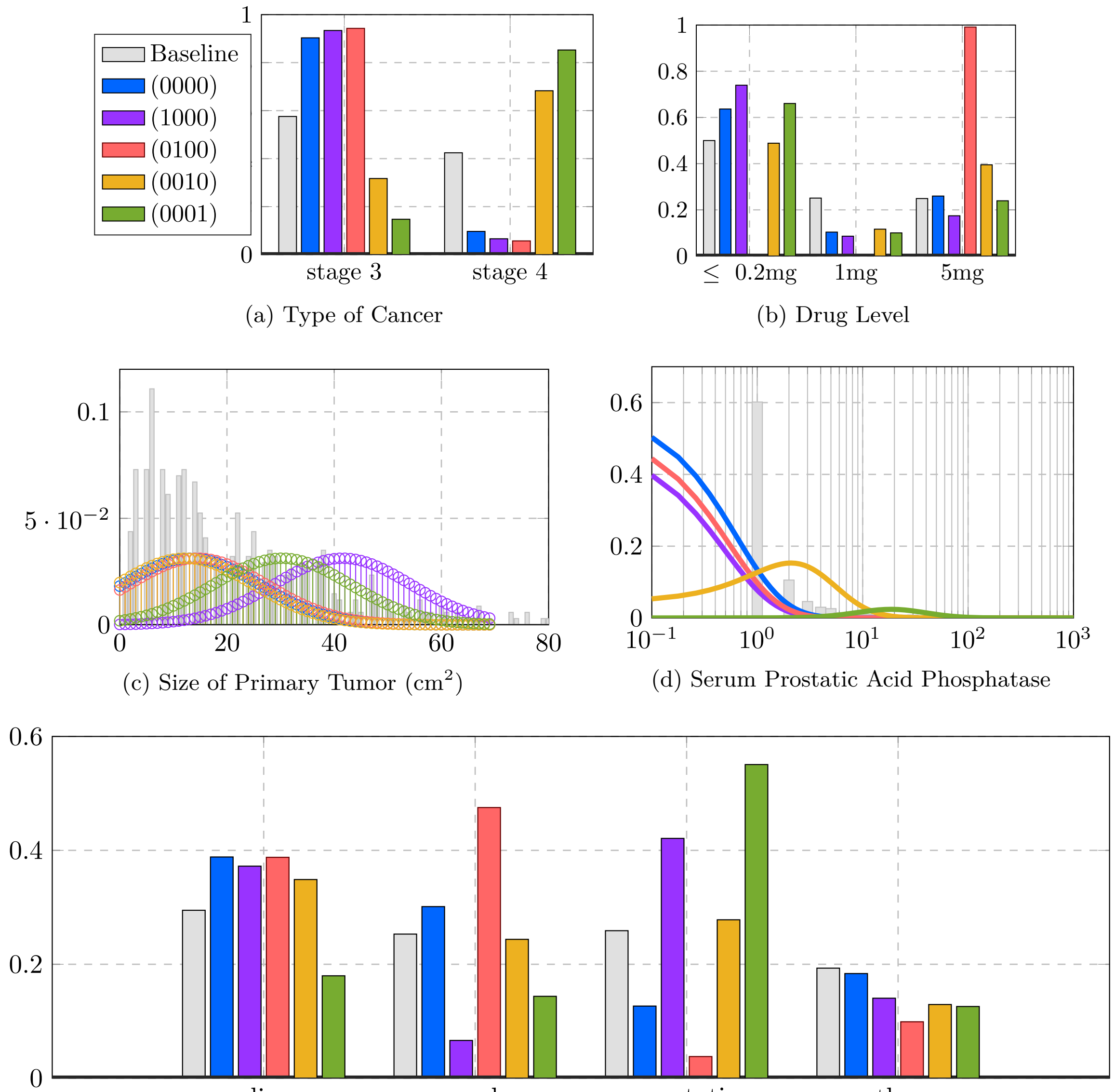
Patterns	(0000)	(0100)	(1000)	(0010)	(0001)	(1100)	(0110)	(0101)	(1010)
Empirical Prob.	0.4641	0.1394	0.0936	0.0757	0.0518	0.0438	0.0359	0.0259	0.0219

Table 4: **Empirical probability of pattern activation for the top-nine most popular patterns.** These probabilities are computed directly from the inferred IBP matrix **Z**.

Results. After running our model, we obtain four latent features, with corresponding empirical activation probabilities and main implications shown in Table 3. Additionally, Table 4 shows the nine most common latent feature vectors, also called *feature patterns*, which capture over 95% of the observations. In order to study the effect of the latent features on each attribute of the dataset, Figure 5 shows the inferred distribution of each attribute for the five most common patterns, which only have one active latent feature plus the bias term.[6]

In Figure 5, we can distinguish two groups of features. The first group, corresponds to patients in stage 3 and includes the bias term and the two first latent features. Within this group, the bias term – depicted as pattern (0000) – and the feature F1 – depicted as pattern (1000) – corresponds to patients in stage 3 with a low average level of treatment with DES, as shown in Figure 5b. However, while the bias term models patients with

[6] In the case of patterns with multiple active latent features, the bias term should be counted only once.



(a) Type of Cancer

(b) Drug Level

(c) Size of Primary Tumor (cm^2)

(d) Serum Prostatic Acid Phosphatase

(e) Prognosis Status

Figure 5: **Data exploration of a prostate cancer clinical trial.** We depict the effect of each latent feature on each attribute. Panels (a)-(d) shows different indicators of the prostate cancer, as well as the dose level of DES. Panel (d) corresponds to Prognosis Status, which indicates whether the patient either is alive or dies from one of the following three causes: vascular disease, prostatic cancer, or other reason. The baseline refers to the empirical distribution of each attribute in the whole dataset. Pattern (0000) corresponds to the bias term described in Section 5.2.1.

low probability ($\sim$ 15%) of prostate cancer death, the first feature accounts for patients with higher probability ($\sim$ 40%) of prostate cancer death, which can be explained by a larger tumor size, as shown in Figure 5c. The feature F2 – or equivalently pattern (0100) – corresponds to patients who exclusively received a high dosage (5 mg) of the drug, as shown in Figure 5b. Patients with an active F2 feature present a small tumor size and the lowest probability of prostatic cancer death, suggesting a positive effect of the drug as a treatment for the cancer. However, they also present a significant increase in the probability of dying from a vascular disease ($\sim 50\%$), indicating a potential adverse-effect of the drug increasing the risk of suffering from cardio-vascular diseases. Such observation is in agreement with previous studies (Byar and Green, 1980; Kay, 1986; Lunn and McNeil, 1995).

The second group of features, which includes features F3 and F4 – depicted as the activation patterns (0010) and (0001) – corresponds to patients in stage 4 with mild and severe conditions, respectively. In particular, the F3 feature corresponds to patients with a small tumor size but with intermediate values for the PAP biomarker, suggesting a certain spread in the degree of the tumor compared to the features in the first group, but not as severe as for patients with pattern (0001). Indeed, the pattern (0001) models those patients in stage 4 with relatively high tumor size and highest PAP values, which is related to metastasis – it is thus not surprising that those patients present the highest probability (above 50%) of prostatic death.

5.2.3. Impact of Social Background on Mental Disorders

In this section, we extended the analysis in (Ruiz et al., 2013) to take into account the influence of certain features that reflect the social background of subjects such as age, gender, etc. in the probability of a subject suffering from a comorbid disorder. To this end, in addition to the diagnoses of the 20 most common psychiatric disorders detailed below, we also use of the information provided by the *Nesarc dataset*[7], which includes information both on the mental condition and on the social background of participants.

Several studies have analyzed the impact of a subject's social background in the development of mental disorders. These studies usually focus on the relationship between a mental disorder and a specific aspect of the social background of the subjects. Some examples in this area study the relationship between depression and gender (Weissman et al., 1993; Kessler et al., 1993), or the link between common mental disorders and poverty or social class (Weich and Lewis, 1998; Dohrenwend, 1975; Hollingshead and Redlich, 1953). Other studies (Blanco et al., 2013; Ruiz et al., 2013) have focused on finding and analyzing the co-occurring (comorbidity) pattern among the 20 most common psychiatric illnesses. These studies found that the 20 most common disorders can be divided into three meta-groups of disorders: i) externalizing disorders, which include substance use disorders (alcohol abuse and dependence, drug abuse and dependence, and nicotine dependence); ii) internalizing disorders, which include mood and anxiety disorders (major depressive disorder (MDD), bipolar disorder and dysthymia, panic disorder, social anxiety disorder (SAD), specific phobia and generalized anxiety disorder (GAD), as well as pathological gambling (PG)); and iii) personality disorders (avoidant, dependent, obsessive-compulsive (OC), paranoid, schizoid, histrionic and antisocial personality disorders (PDs)). Additionally, such studies also found that comorbid or co-occurring disorders tend to belong to the same group of

[7] dataset available at: `http://aspe.hhs.gov/hsp/06/catalog-ai-an-na/Nesarc.htm`

Attribute description	Type of variable
Gender	Categorical with 2 categories
Age	Count data
Census region	Categorical with 4 categories
Race/ethnicity	Categorical with 5 categories
Marital status	Categorical with 6 categories
Highest grade or years of school completed	Ordinal with 14 categories

Table 5: **List of considered social background attributes.** We look for correlations between each of these attributes with the twenty most common psychiatric disorders among the subjects in the Nesarc dataset.

disorders (Valera et al., 2016). To the best of our knowledge, there are currently no studies about the impact of social background in the suffering of comorbid disorders.

Experimental Setup. The Nesarc dataset contains the responses of a representative sample of the U.S. population to a survey with questions related to the social background of participants, alcohol and other drug consumption, and behaviors related to mental disorders. The first wave of Nesarc sampled the adult U.S. population with over 43,000 respondents who answered almost 3,000 questions. The dataset also includes the diagnoses for each of the participants of the survey. In this experiment, in addition to the diagnoses of the 20 most common psychiatric disorders described above, we included one by one each of the social background questions as input data to the proposed model. Table 5 summarizes the considered questions and the considered data types when we introduce them into our model as input variables. Note that the diagnoses of the 20 psychiatric disorders correspond to categorical variables with two possible categories, e.g., a patient suffering or not from a disorder. Each attribute related to the social background of the participants is introduced independently to ensure that the model captures the dependencies between latent disorders and social background, instead of the correlations among the different aspects of the social background. This helps to focus on the correlations between each aspect of the social background of the participants and the probability of suffering from each disorder, which are in the order of 10^{-2}, see Figure 6(a).

The following experimental results are reported: we ran the inference algorithm in Section **??** for each question independently with parameter values given by: $\alpha = 5$, $\sigma_B^2 = 1$, $\sigma_y^2 = 1$, $\sigma_\theta^2 = 1$. We consider the following transformation that maps from the real numbers to the positive real numbers: $f(x) = x^2$ for the positive real and count data. We chose a different mapping function to show that the proposed model works with any differentiable and invertible function. Similarly to Ruiz et al. (2013), we activated the bias term mentioned in Section 5.2.1, i.e., an additional latent feature which is active to all the subjects in the data set, so that we do not sample the rows of $\mathbf{Z}$ corresponding to those subjects who do not suffer from any of the 20 disorders, but instead fix their latent features to zero. The idea is that the bias term captures the population that does not suffer from any disorder, while the rest of the active features in matrix $\mathbf{Z}$ characterize the disorders. As mentioned previously, the bias term is useful for the interpretability of the inferred latent features.

Results. After running our model, we find that the census region, race, ethnicity, marital status and educational level, which corresponds to the highest grade or years of school completed, do not appear to have any influence in the comorbidity patterns of the 20 most common psychiatric disorders. In contrast, as detailed below, gender and age of the participants influence the probability of suffering from a set of comorbid or co-occurring disorders.

Gender. We model the gender information of the participants in the Nesarc as a categorical variable with two categories: {'male', 'female'}. The percentage of males in the Nesarc dataset is approximately 43%. In this case, the GLFM found three latent features, with corresponding empirical probabilities reported in Table 6. Furthermore, Table 7 shows the empirical probability of all the feature pattern found in the dataset. Here, we observe that the three latent features activate mostly in isolation, since the probability of jointly activating two or more features is below 1%. Figure 6a shows the probability of meeting each diagnostics criteria for the latent feature vectors $\mathbf{z}_n$ listed in the legend and in the dataset (baseline). Note that the obtained latent features are similar to the ones in (Ruiz et al., 2013), i.e., feature F1 – pattern (100) – mainly models the seven personality disorders (PDs), feature F2, which corresponds to pattern (010), models the alcohol and drug abuse disorders and the antisocial PD, while feature F3 – pattern (001) – models the anxiety and mood disorders. Additionally, in Figure 6b, we show the probability of being male and female for the latent feature vectors $\mathbf{z}_n$ shown in the legend and the empirical probability of being male and female in the dataset (baseline).

In Figure 6b, we observe that having no active features (pattern (000), which corresponds to people that do not suffer from any disorder, is more common in male subjects, suggesting that that females tend to suffer more from psychiatric disorders. Moreover, F1 feature active – pattern (100) – suggests a positive correlation between being a women and suffering from mood and anxiety disorders; while feature F3 – pattern (001) – indicates that PDs are more common in men.

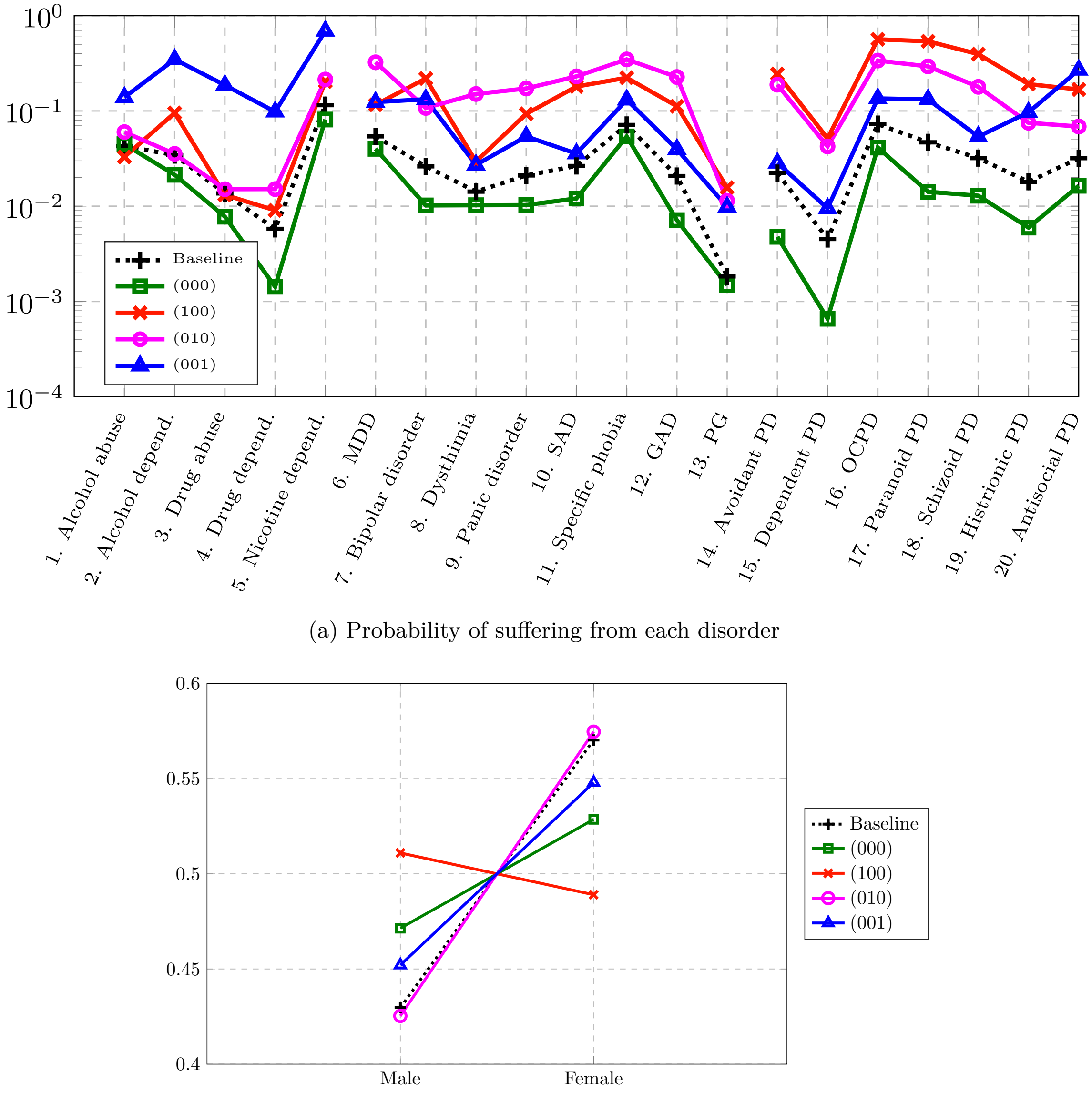
Feature	Empirical Prob.	Main implications
Feature F1	0.0341	Correlates personality disorders and male gender
Feature F2	0.0470	Model subjects with alcohol and drug abuse disorders
Feature F3	0.0460	Correlates anxiety and mood disorders and female gender

Table 6: **Gender: empirical probabilities of possessing at least one latent feature.** These probabilities are directly computed from the inferred IBP matrix **Z**.

Patterns	(000)	(010)	(001)	(100)	(111)	(011)	(110)
Empirical Prob.	0.8615	0.0427	0.0414	0.0298	0.0023	0.0022	0.0020

Table 7: **Gender: Empirical probability of feature pattern activations.** These probabilities are computed directly from the inferred IBP matrix **Z**.

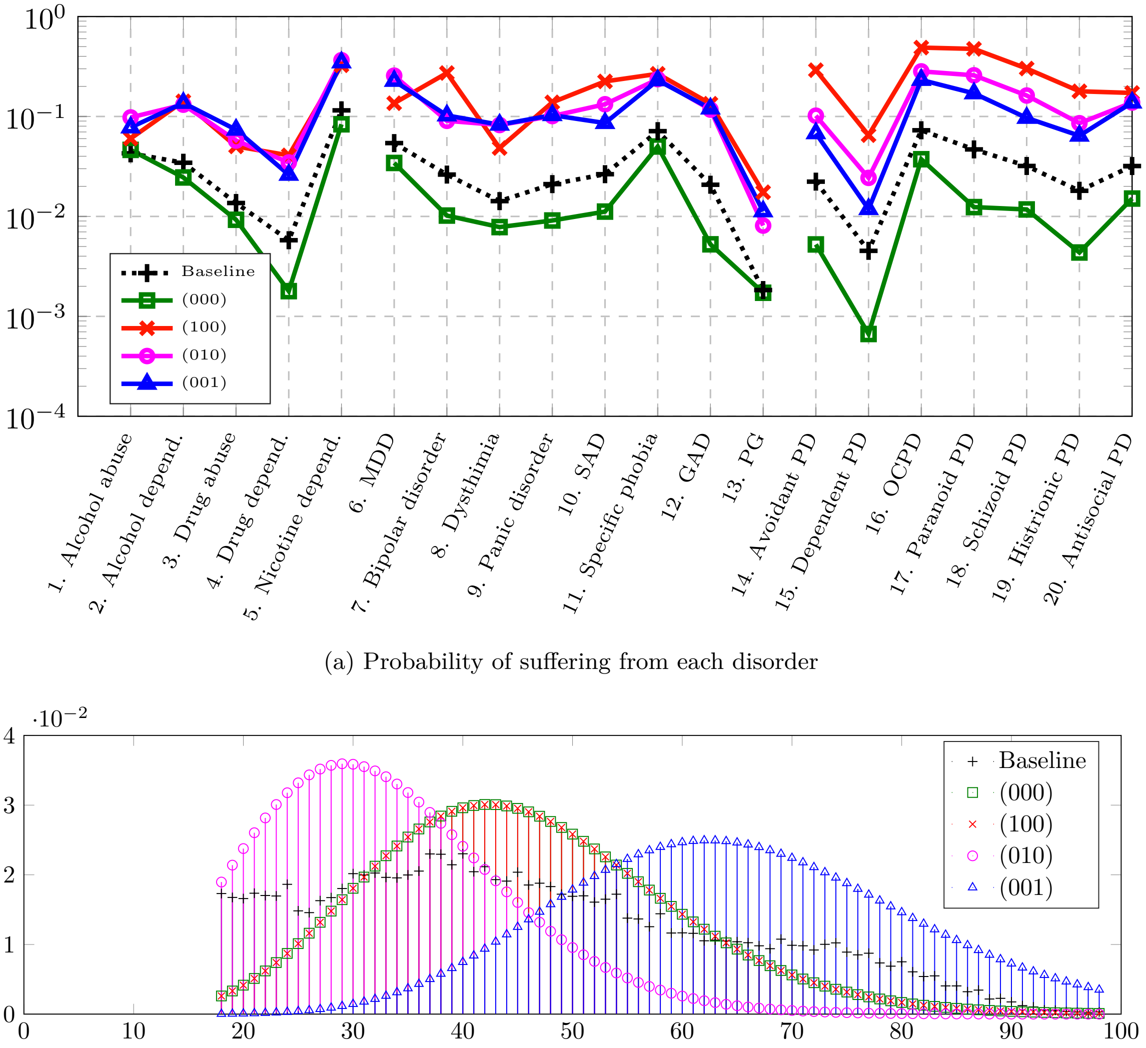
Age. We focus on the age of the participants, which we model as count data. After running our inference algorithm with the diagnoses of the 20 disorders and the subjects age as input data, we again obtain three latent features that activate mostly in isolation (combination of two features below 3%), with corresponding empirical probabilities listed in Table 8. Furthermore, Table 9 shows the empirical probability of each feature pattern in the dataset.



(a) Probability of suffering from each disorder

(b) Gender

Figure 6: **Feature effects including gender in the analysis.** (a) Probabilities of suffering from the 20 considered disorders and (b) probability of being male and female for the latent feature vectors $\mathbf{z}_n$ shown in the legend and for the baseline.



(a) Probability of suffering from each disorder

(b) Age

Figure 7: **Feature effects including age in the analysis.** (a) Probabilities of suffering from the 20 considered disorders and (b) the age distribution for the latent feature vectors $\mathbf{z}_n$ shown in the legend and baseline probability distribution.

Figure 7a shows the probability of meeting each diagnostic criteria for the latent feature vectors $\mathbf{z}_n$ listed in the legend and in the dataset (baseline). In addition to the baseline probability distribution, in Figure 7b we plot the inferred probability distributions over the age when none or only one of the latent variables is active. This corresponds to the most common feature patterns. In Figure 7b, the empirical probability distribution over the age based on the data is shown, denoted by 'baseline'. Here, we observe that introducing the age of the participants as an input variable has changed the inferred latent features (with respect to the features in (Ruiz et al., 2013) depicted in Figure 6a). In particular, we observe that the obtained latent features mainly differ in the probability of suffering from personality disorders (i.e., disorders from 14 to 20), and the probability of suffering from disorders 1 to 13 is similar for the three plotted latent feature patterns. In this figure, we observe that the vector $\mathbf{z}_n$ with no active latent features, e.g., pattern (000), captures the average age in the dataset (which coincides with middle-aged subjects, i.e., $30 - 50$ years old). Furthermore, we observe that the subjects with the highest probability of suffering from personality disorders – pattern (100) – are likely to be middle-aged, followed in a decreasing order by young adults – pattern (010) – and elderly people – pattern (001). Additionally, if we focus on the differences among the three features in disorders from 1 to 13, we also observe that, while young and elderly people tend to suffer from depression, middle-aged people tend to suffer from bipolar disorder. Based on Figure 7, we conclude that the bipolar disorder and the seven personality disorders tend to show up mostly in the mature age, while young and elderly people tend to suffer from depression more often.

Feature	Empirical Prob.	Main implications
Feature F1	0.0332	Captures severe personality disorders and middle-age subjects
Feature F2	0.0550	Captures mid-severe personality disorders and young subjects
Feature F3	0.0569	Captures mid-severe personality disorders and older subjects

Table 8: **Age: empirical probabilities of possessing at least one latent feature.** These probabilities are directly computed from the inferred IBP matrix $\mathbf{Z}$.

Patterns	(000)	(010)	(001)	(100)	(011)	(110)	(101)
Empirical Prob.	0.8615	0.0522	0.0503	0.0294	0.0028	0.0019	0.0019

Table 9: **Age: Empirical probability of feature pattern activations.** These probabilities are computed directly from the inferred IBP matrix $\mathbf{Z}$.

5.2.4. Voters profile in presidential election

Finally, we apply the proposed model to understand the correlations between demographic profiles and political vote tendencies. In particular, we focus on the United States presidential election of 1992, in which three major candidates ran for the race: the incumbent Republican president George H. W. Bush, the Democratic Arkansas governor Bill Clinton, and the independent Texas businessman Ross Perot. In 1992, the public's concern about the federal budget deficit and fears of professional politicians allowed the independent candidacy of billionaire Texan Ross Perot to appear on the scene dramatically (Alvarez and Nagler, 1995), to the point of even leading against the major party candidates in the polls during the electoral race[8]. The race ended up with the victory of Bill Clinton by a wide margin in the Electoral College, receiving 43% of the popular vote against Bush's 37.5% and Perot's 18.9% (Lacy and Burden, 1999). The election results are known to be the highest vote share of a third-party candidate since 1912, even if Perot did not obtain any electoral votes (Lacy and Burden, 1999).

Our primary objective in this sections to find and analyze the different types of voters' profiles, as well as which candidate each profile tends to favor. We used the publicly available *Counties dataset* which contains diverse information about voting results, demographics

[8] New York Times: http://www.nytimes.com/1992/06/11/us/the-1992-campaign-on-the-trail-poll-gives-perot-a-clear-lead.html

and sociological factors per counties[9]. This dataset contains information for 3,141 counties. Table 10 lists the per-county attributes that we used as input for our model.

Attribute description	Type of data
State in which the county is located	Categorical with 51 categories
Population density in 1992 per squared miles	Positive real data
% of white population in 1990	Positive real data
% of people with age above 65 in 1990	Positive real data
% of people above 25 years old with bachelor's degree or higher	Positive real data
Median family income in 1989 (in dollars)	Count data
% of farm population in 1990	Positive real data
% of votes cast for Democratic president	Positive real data
% of votes cast for Republican president	Positive real data
% of votes cast for Ross Perot	Positive real data

Table 10: **List of considered attributes regarding the United States presidential election of 1992.** Attributes 1 to 7 include demographic information and sociological factors, while the last three attributes summarize the percentage voting outcome in each county.

Feature	Empirical Prob.	Main implications
Feature F1	0.4874	Favours Perot, increases the probability of white population, and decreases average income.
Feature F2	0.2703	Favours the Democrat candidate, increases population density, and decreases family income, percentages of white population, farming and college degrees.
Feature F3	0.2700	Favours the Republican candidate and Perot, increases the percentage of farming, and decreases population density.
Feature F4 Feature F5	0.0411 0.0372	Capture the tails of the distributions of different attributes.

Table 11: **Empirical feature activation probabilities for the Counties dataset.** We show the empirical probability of possessing at least one latent feature, as well as the main implications of the activation of each feature. These are directly computed from the inferred IBP matrix $\mathbf{Z}$.

Patterns	(000)	(100)	(101)	(010)	(110)
Empirical Prob.	0.2636	0.2407	0.1063	0.1060	0.0748

Table 12: **Empirical probability of pattern activation for the top-five most popular patterns.** These probabilities are computed directly from the inferred IBP matrix $\mathbf{Z}$. Features F4 and F5 are always switched off, and are thus omitted from the labels.

Experimental Setup. We ran our inference algorithm with $\alpha = 5$, $\sigma_B^2 = 1$, $\sigma_\theta^2 = 1$ and used the following mapping transformation from the real numbers to the positive real numbers: $f(x) = \log(w\cdot(x-\mu)+1)$, with $\mu = \min(\mathbf{x}^d)$ and $w = 2/\text{std}(\mathbf{x}^d)$. We activated the

[9] dataset available at: `http://biostat.mc.vanderbilt.edu/wiki/pub/Main/DataSets`

bias term and sampled the variance of the pseudo-observations for each dimension/attribute. A challenging aspect of this dataset is that the distributions of some of its attributes are heavy-tailed, leading to a large number of latent features as output of the GLFM, whose purpose is to capture the tails of the distributions. This is not an issue for estimation and imputation of missing data, but it renders data exploration more tedious. To solve this limitation, we performed an additional data preprocessing step by applying a logarithmic transformation to heavy-tailed attributes.[10] In more detail, we applied the function $g_1(x) = \log(x+1)$ for population density, median family income, and percentage of farm population. For the percentage of white population, we used the function $g_2(x) = \log((100 - x) + 1)$ since the distribution has most of its density close to 100%. In the Appendix B we discuss further details about the data preprocessing step we used before applying GLFM as well as the implementation of the GLFM software package.

Results. After running GLFM on this dataset we found 5 latent features, with corresponding empirical activation probabilities shown in Table 11. We observe that while the first three features are active for at least 27% of the counties, the last two features are active only for around 4% of the counties. Furthermore, we find that the different combinations of the three first latent features represent more than 92% of the counties in USA. In the following, we focus only on the analysis of the three first features and, in particular, on the top-five most popular feature patterns. In Table 12, the empirical probabilities of these five patterns are shown, which represent around 80% of the U.S. counties. Figure 8 shows the distribution of vote percentage per candidate associated to each of these top-five patterns, while Figure 9 shows the corresponding geographic distribution (i.e., the empirical activation probability) across states for each of these patterns. In these figures, we observe that:

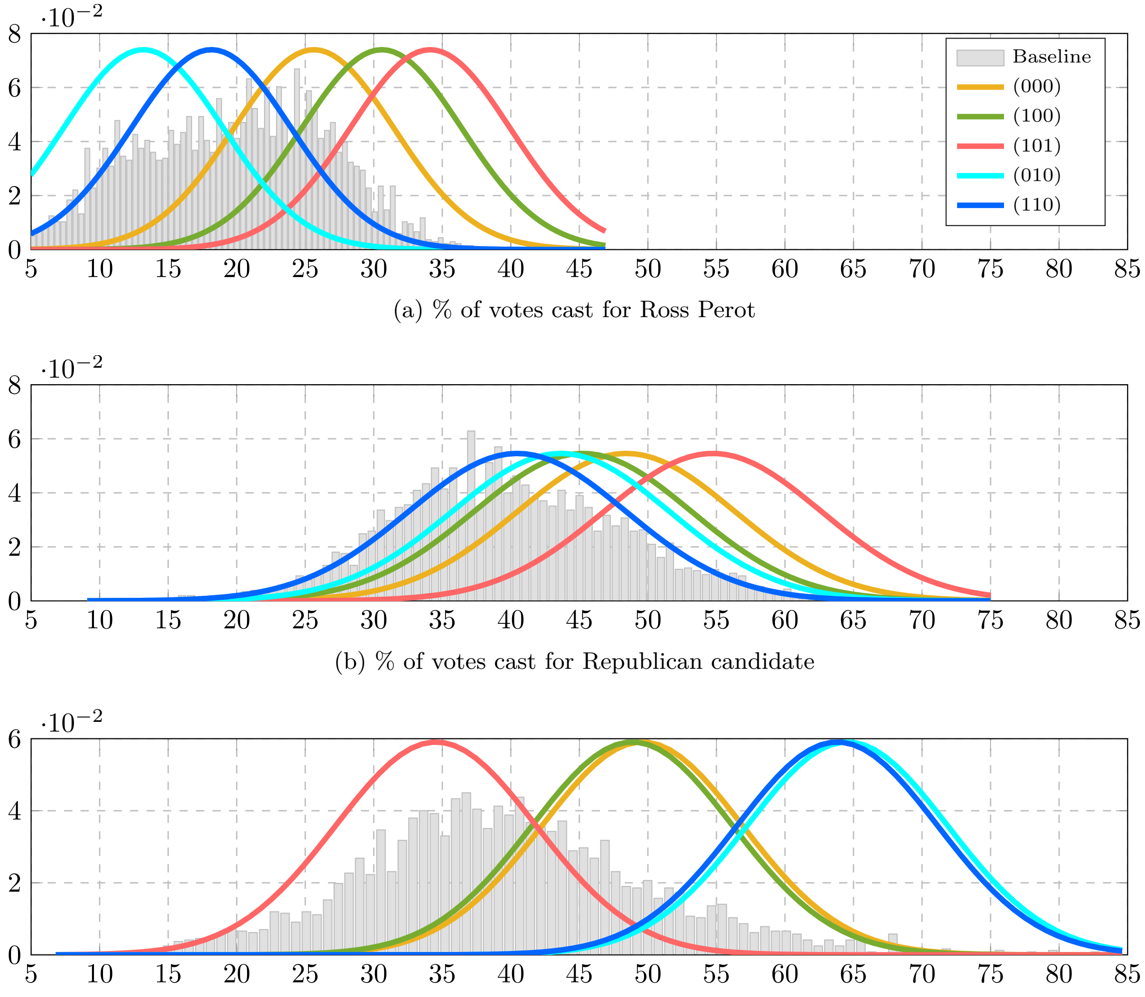
(i) pattern (000), corresponding to the bias term, tends to model middle values for the percentage of votes for the three candidates (with an average percentage of votes of $\sim 50\%$ for the Democrat candidate, $\sim 48\%$ for the Republican candidate and $\sim 27\%$ for Perot), and activates mainly in the east and west coasts of the country, as well as Florida;

(ii) pattern (100) provides similar percentage of votes for the Democrat and Republican candidates as in pattern (000), but it favors the independent candidate Perot (with an average percentage of votes above 30%). This pattern activates mostly in the north central-east region of the country and Maine (the state where Perot's party managed to beat the Republican party);

(iii) pattern (101) activates in the north central-west region of the USA (not including the coast) and represents a profile inclined towards the Republican party (with an average percentage of votes of $\sim 55\%$) while also favoring in a lower extent the independent candidate; and

(iv) patterns (010) and (001) clearly capture Democrat-oriented profiles, and activate mainly in the south east region of the USA, including the state from which Bill Clinton originally comes from, Arkansas.

[10] The functionality of defining external pre-processing transformations for each dimension is supported by our open-source GLFM package.

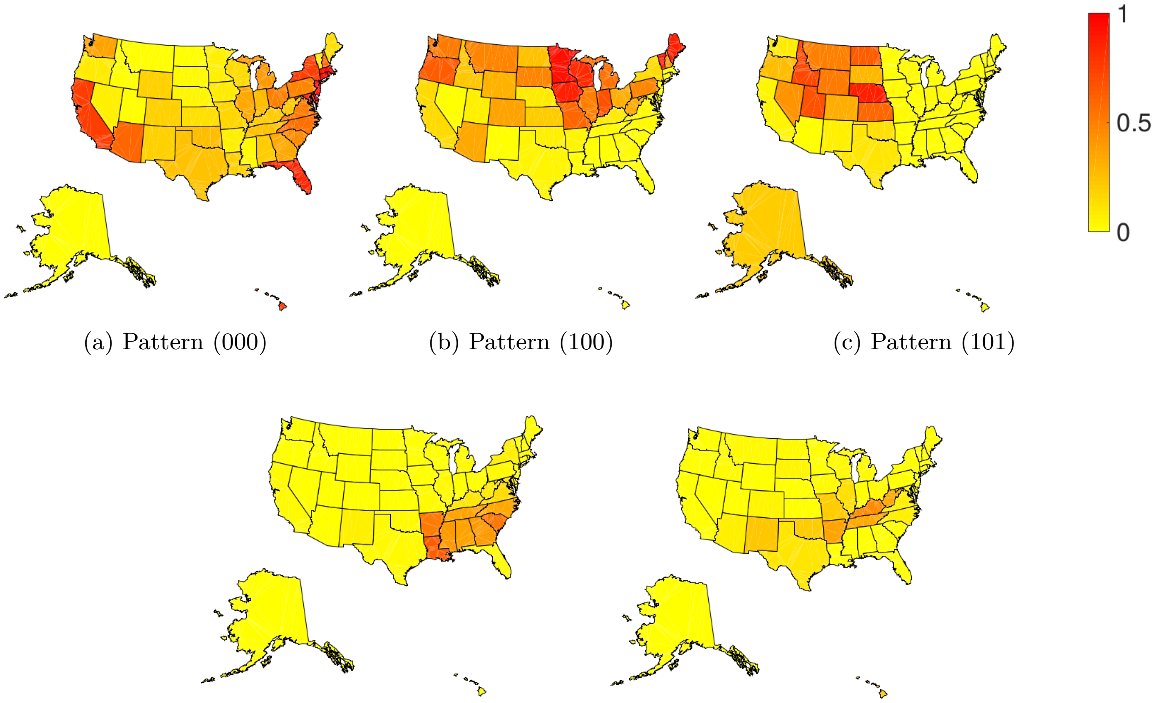


(a) % of votes cast for Ross Perot

(b) % of votes cast for Republican candidate

(c) % of votes cast for Democrat candidate

Figure 8: **Inferred probability distribution for the five most popular patterns.** The patterns are sorted in the legend according to their degree of popularity, as described in Table 12. The baseline refers to the empirical distribution of each attribute in the entire dataset.



(a) Pattern (000) (b) Pattern (100) (c) Pattern (101)

(d) Pattern (010) (e) Pattern (110)

Figure 9: **Empirical probability of pattern activation per state.** We focus on the top-five most popular combinations of features. The label for each pattern indicates whether Features F1, F2, and F3 are active (value '1') or not (value '0'). Features F4 are F5 are always inactive in the five most common patterns, and thus are omitted in the labels.

The demographic results reported above are in agreement with the outcome of the election per counties[11], as shown in Figure 10. Next, we analyzed the demographic information associated to each of the feature patterns above. In Figure 11, the distribution of each attribute/dimension of the data for each of the considered patterns is displayed. First, we observe that pattern (000), which activates mostly in the coasts and Florida, corresponds to the highest population density, average income, and percentage of college degrees, as well as an important race diversity and low farming activity. These observations align with the typical profile characterizing "big-cities". As stated before, this pattern is the most balanced in terms of voting tendency, with an equilibrated support for both Democrat and Republican, as well as intermediate values for the percentage of votes cast for Perot.

Second, patterns (100) and (101) represent the largest share of Perot's votes, both with an average percentage of votes above 30% for Perot. Figure 11 shows that Perot's main supporters, characterized mainly by pattern (101), also correspond to Republican main supporters, who tend to live in low populated areas in the north central part of the country where farming activity is considerable, and the percentages of white population and over-65 years old population are also high. The second voting force backing Perot, captured by

[11] `https://en.wikipedia.org/wiki/United_States_presidential_election,_1992`

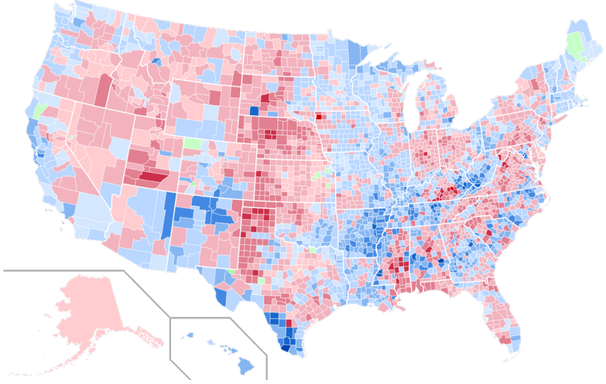


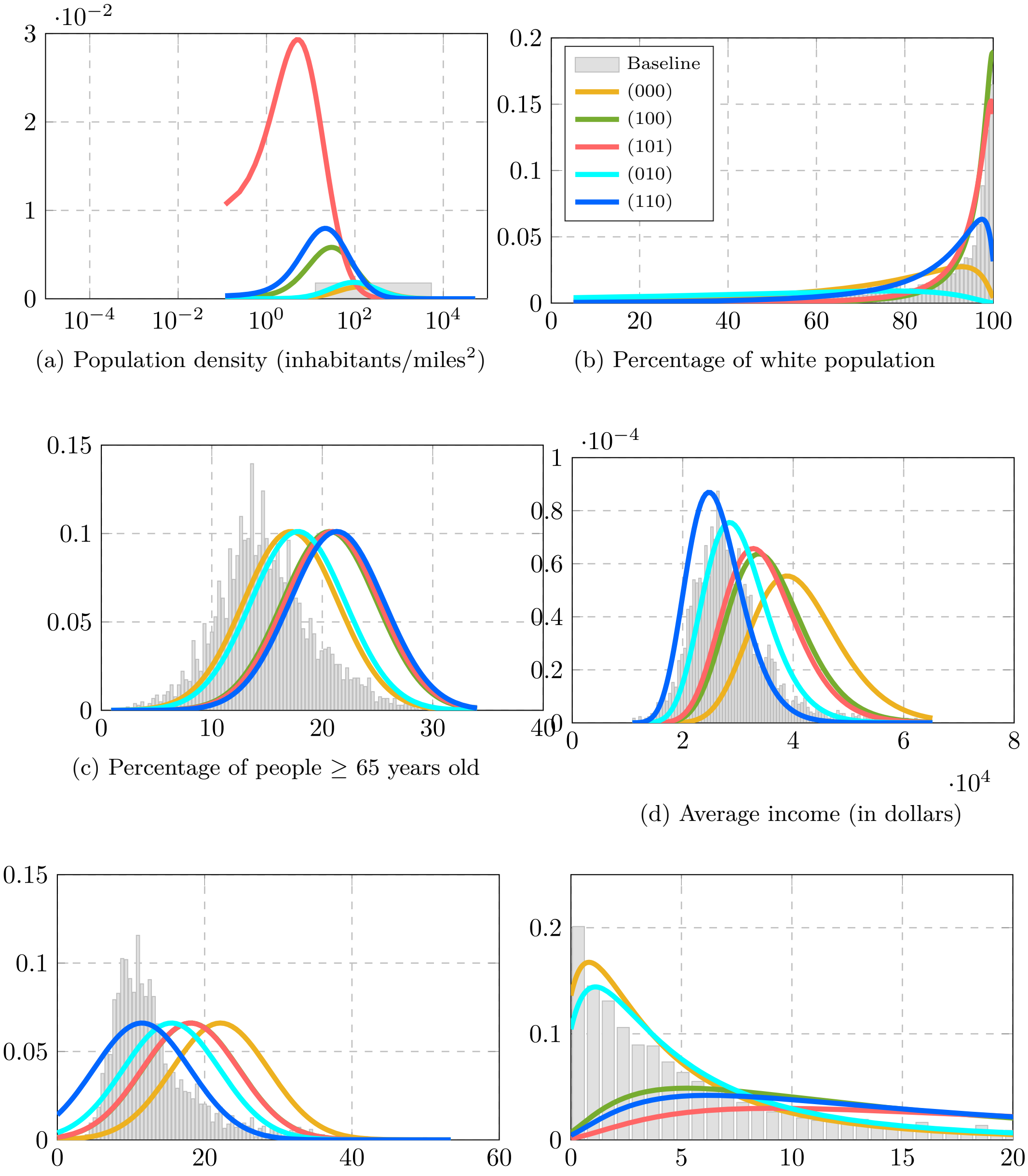
Figure 10: **Outcome of the 1992 presidential election per counties.** Blue color corresponds to a majority of votes for the Democrat party, red corresponds to a victory for the Republican party, green corresponds to a victory of the independent party of Ross Perot.

pattern (100) and located in the north east-central part of USA, corresponds mostly to white population with an intermediate-high average income and an average percentage of college degrees around 18% (the red curve in Figure 11e overlaps the green line). These results back the analysis in (Lewis et al., 1994), which showed that the majority of Perot's voters (57%) belonged to the middle class, earning between $15,000 and $49,000 annually, with the bulk of the remainder belonging to the upper middle class (29% earning more than $50,000 annually). Perot's campaign ended up taking 18.9% of the votes, finishing second in Maine and Utah, as captured by pattern (100) and (101) respectively.

Finally, Democrat's patterns (010) and (110) are mainly active in the Southeastern United States, and capture a diverse range of voters in terms of their demographic properties. On one hand, pattern (010) captures highly populated counties, with low values of family income, percentage of college degrees, percentage of white population and percentage of farming population. On the other hand, pattern (110) captures low populated counties with a large percentage of population above 65 year old, as well as a larger presence of farming activity and lower average income. These results might be explained by the broad appeal across all socio-ethno-economic demographics that the Democratic party has historically targeted.

6. Conclusions

In this paper, we have developed an efficient general latent feature model, named GLFM. The GLFM model is suitable for modeling tasks with real-world heterogeneous datasets. The proposed model presents attractive properties in terms of flexibility and interpretability. First, its nonparametric nature allows it to automatically infer the appropriate model complexity (i.e., number of latent features) from data. Second, since the latent features are binary-valued, it is easier to identify and interpret meaningful patterns in the data exploration process. Third, we derived an augmented model that inherits the properties of conjugate models, which allow us to extend an efficient inference scheme that scales lin-



(a) Population density (inhabitants/miles^2)

(b) Percentage of white population

(c) Percentage of people $\geq$ 65 years old

(d) Average income (in dollars)

(e) Percentage of college degrees

(f) Percentage of farming population

Figure 11: **Inferred probability distribution for the most occuring patterns.** The baseline refers to the empirical distribution of each attribute in the whole dataset.

early with the number of observations (objects) and dimensions (attributes) in the data per MCMC iteration.

We showed the flexibility and applicability of the proposed GLFM, and its available software implementation detailed in Appendix B, by solving both prediction and data exploration tasks in several real-world datasets. In particular, we used the proposed model to estimate and replace missing data in heterogeneous datasets. Also, we used the proposed GLFM for data exploratory analysis of real-world datasets related to diverse application domains: clinical trials, psychiatry, sociology and politics.

As future research lines, it would be interesting to incorporate automatic detection of the type of data before training the model, to fully automatize the whole procedure for table completion or data exploration (Valera and Ghahramani, 2017). Other promising directions include replacing the latent feature model by more complex models, such as for example considering non-linear latent variable models for the pseudo-observations by parameterizing the model using deep neural networks, leading to variational autoencoder (Kingma and Welling, 2013) architectures suitable for heterogeneous types of data. Finally, the usage of GLFM for data exploration would increase if we manage to incorporate prediction-constraints to the generative model (Hughes et al., 2017). This would result in latent representations that not only allow us to explain the data, but that can also be used for an end-task prediction problem, bringing close together the generative and discriminative perspectives of probabilistic models.

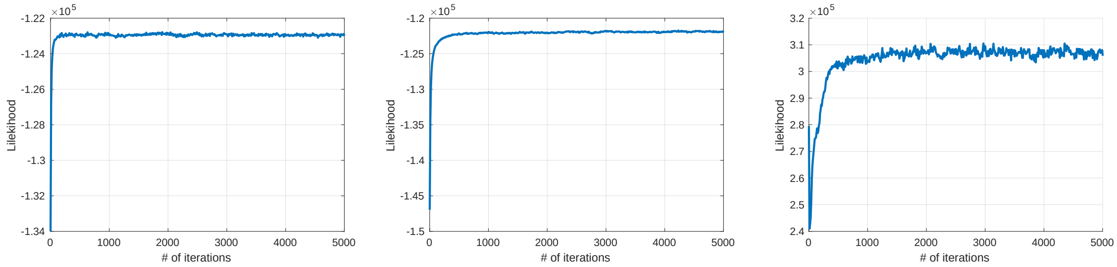
Acknowledgments

Isabel Valera acknowledges her Humboldt Research Fellowship for Postdoctoral Researchers that funder her during the period that this research was carried out. Zoubin Ghahramani and Maria Lomeli acknowledge support from the Alan Turing Institute (EPSRC Grant EP/N510129/1) and EPSRC Grant EP/N014162/1, and donations from Google and Microsoft Research. Melanie F. Pradier is grateful to the Harvard Data Science Initiative, the Center for Research on Computation and Society at Harvard, and the European Union 7th Framework Programme through the Marie Curie Initial Training Network “Machine Learning for Personalized Medicine” MLPM2012, Grant No. 316861 (http://mlpm.eu). This work has been partly supported by MINECO/FEDER (’ADVENTURE’, id. TEC2015-69868-C2-1-R), and Comunidad de Madrid (project ’CASI-CAM-CM’, id. S2013/ICE-2845).

Appendix A. Additional results

A.1. Log-likelihood Trace Plots

Before entering in the details of the missing data estimation task, we first evaluate the convergence of the proposed MCMC sampler. To this end, we track the evolution of the log-likelihood with respect to the number of iterations of the sampler. As an example, Figure 12 shows three examples of log-likelihood trace plots for three of the considered datasets in Section 5.1. We can observe here that the burn-in period of the sampler consists only of a few hundred samples for the three datasets. Then, for the rest of our experiments we decide to run 5000 iterations of the sampler, where the first 1000 iterations are considered as burn-in period.



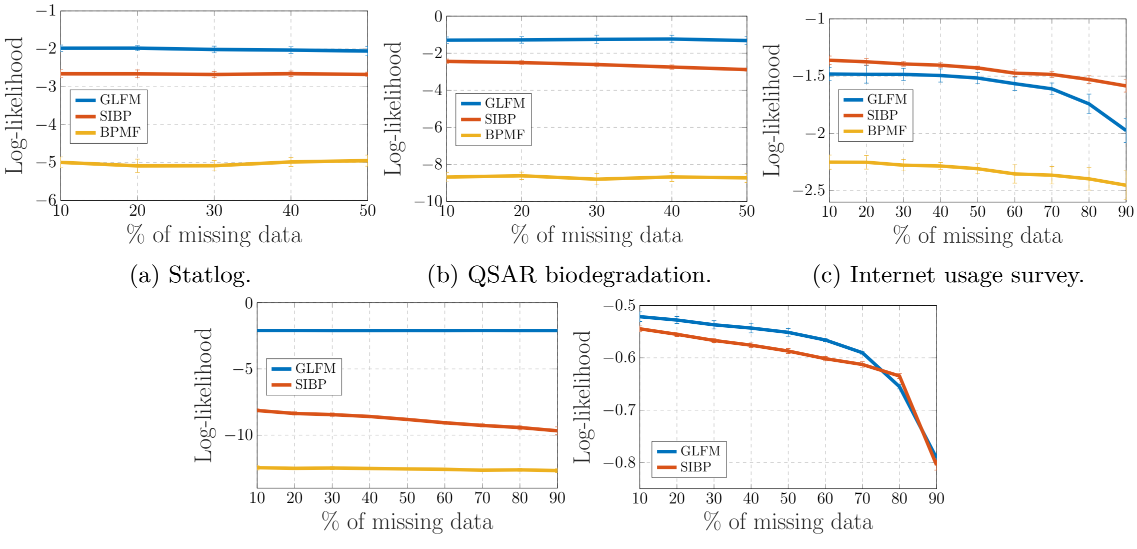
(a) Statlog. (b) Internet usage survey. (c) Wine quality.

Figure 12: Trace plot of the log-likelihood per iteration of the sampler.

A.2. Missing Data Extimation

In this section, we provide additional results that compare the proposed GLFM with the baselines in terms of test log-likelihood. To evaluate the test log-likelihood for the SIBP and the BPMF models for discrete data, we compute the integral of the Gaussian likelihood function in the interval $(x_n^d - 0.5, x_n^d + 0.5]$ (or $(-\infty, x + 0.5]$ or $(x - 0.5, +\infty)$, respectively for the first and last discrete observed values in the data). Here, $\mathbf{x}_n^d$ corresponds to the true value of the variable, and therefore this integral computes the probability of imputing the true value for $\mathbf{x}_n^d$ under each of these models. This way of computing the test likelihood ensures therefore a fair comparison between GLFM and the baselines. Note that test log-likelihood results are provided for the baseline MDFA, since the exact computation of the likelihood is not tractable and (Khan et al., 2010) only provide a bound on the likelihood.

Results. The plots in Figure 13 show the average predictive log-likelihood per missing value as a function of the percentage of missing data. Each value in Figure 13 was obtained by averaging the results across 20 independently split sets where the missing values were randomly chosen. In Figures 13b and 13c, we cut the plot at a missing percentage of 50% because, in these two datasets, the discrete attributes present a mode value that appears for more than 80% of the instances. As a consequence, the SIBP and the BPMF algorithms assign probability close to one to the mode, which results in an artificial increase in the average test log-likelihood when larger percentages of missing data are present. For the BPMF model, we used different numbers of latent features: 10, 20 and 50, respectively. We



(a) Statlog.

(b) QSAR biodegradation.

(c) Internet usage survey.

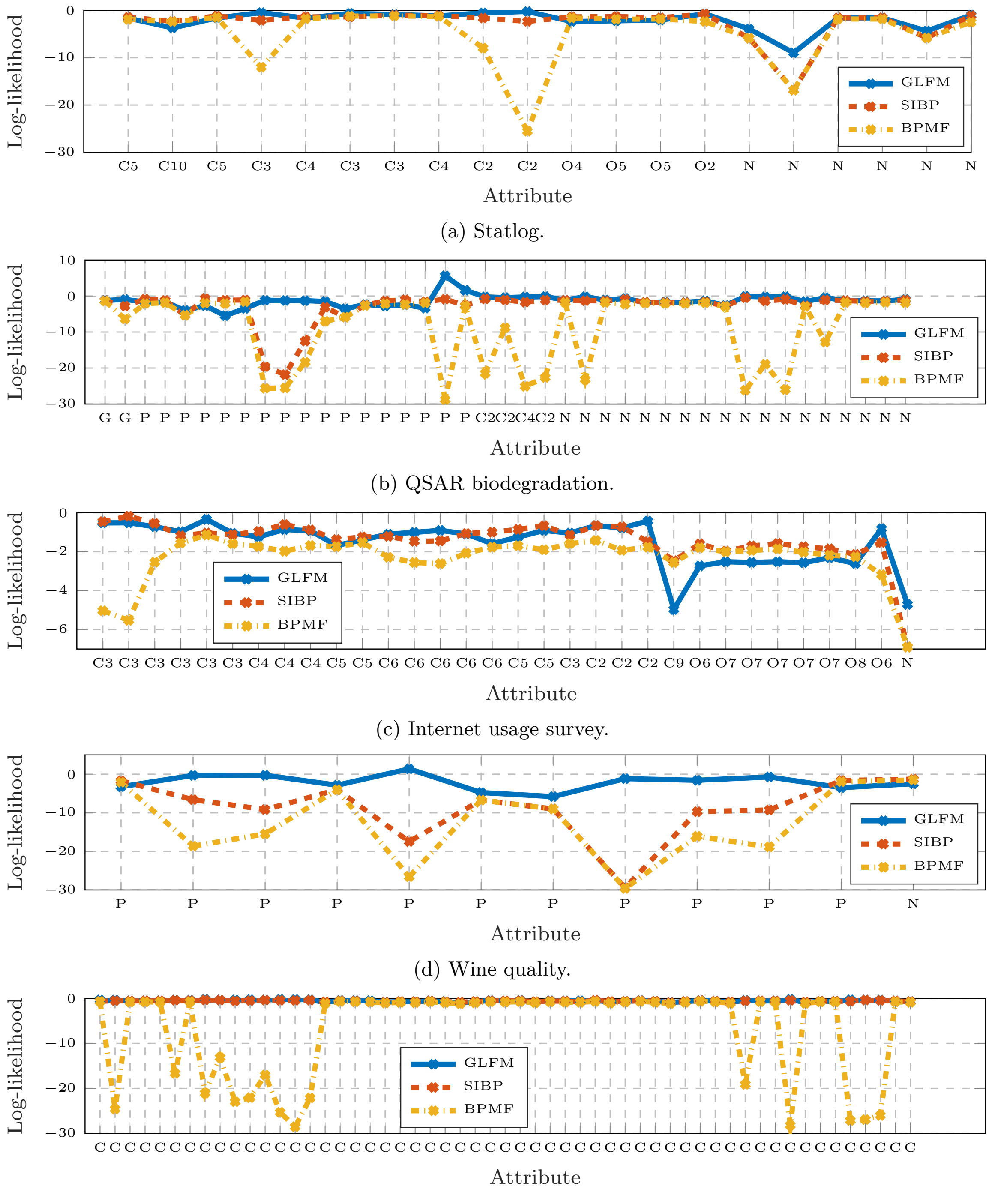
(d) Wine quality.

(e) Nesarc dataset.

Figure 13: **Average test log-likelihood per missing datum versus percentage of missing data.** The 'whiskers' show one standard deviation away from the average test log-likelihood.

only show the best results for each dataset, specifically, $K = 10$ for the Nesarc and the wine datasets, and $K = 50$ for the remainder. Both GLFM and SIBP have not learnt a number of binary latent features above 25 in any case. In Figure 13e, we only plot the test log-likelihood for GLFM and SIBP because BPMF provides much lower values. As expected, we observe in Figure 13 that the average test log-likelihood decreases for the three models as the number of missing values increases. The curves shown here have a flat shape due to the logarithmic scale of the y-axis. In this figure, we also observe that the proposed GLFM outperforms SIBP and BPMF for four of the datasets, being SIBP slightly better for the Internet dataset. BPMF model presents the worst test log-likelihood in all datasets.

Successively, we analyzed the performance of the three models for each kind of discrete and continuous variables. Figure 14 shows the average predictive likelihood per missing value for each attribute in the table, which corresponds to each dimension in $\mathbf{X}$. In this figure, we have grouped the dimensions according to the kind of data that they contain, the x-axis shows the number of considered categories for the case of categorical and ordinal data. In the case of the Nesarc dataset, all the attributes are binary. The figure shows that the GLFM presents similar performance for all the attributes in the five datasets, while for the SIBP and the BPMF models, the test log-likelihood falls drastically for some of the attributes. This low-likelihood effect is more dramatic in the case of BPMF as can be seen in Figure 13. This effect is more evident in Figures 13b and 13d, respectively. In Figures 13 and 14, we observe that both IBP-based approaches (GLFM and SIBP) outperform BPMF, with our proposed GLFM being the one that best performs across all datasets. We can conclude that, unlike BPMF and SIBP, the GLFM model provides better estimates for the missing data regardless of their discrete or continuous nature.



(a) Statlog.

(b) QSAR biodegradation.

(c) Internet usage survey.

(d) Wine quality.

(e) Nesarc dataset.

Figure 14: **Average test log-likelihood per missing datum in each dimension.** Here, we consider 50% of missing data. In the x-axis 'R' stands for real-valued variables, 'P' for positive real-valued variables, 'C' for categorical variables, 'O' for ordinal variables and 'N' for count variables. The number that accompanies 'C' or 'O' corresponds to the number of categories. In the Nesarc dataset, all the variables are binary, i.e., 'C2'.

Appendix B. GLFM Software package

This appendix provides the details about the implementation and usage of the GLFM software package, which allows to perform latent feature modeling in heterogeneous datasets, where the attributes describing each object can be either discrete, continuous or mixed variables. To the best of our knowledge, this library provides the first available software for latent feature modeling in heterogeneous data, and includes functions for the two main applications of GLFM, i.e., missing data estimation (or table completion) and exploratory data analysis.

B.1. Implementation

The GLFM package contains an efficient C++ implementation, together with user interfaces in Python, Matlab and R, of the collapsed Gibbs sampling algorithm described in Algorithm 1. The main function of the package, `hidden = GLFM.infer(data)`,[12] runs the inference algorithm given the input structure `data` and returns the learned latent variables in the output structure `hidden`. This function receives as input an observation matrix $\mathbf{X}$ and a vector indicating the type of data for each dimension. Optionally, model hyperparameters and simulation settings can be customized by the user. The latent variables are learned by using the mapping transformations listed in Table 13 to account for the continuous and discrete data types mentioned above. The parameters μ and w are used to shift and scale the raw input data, and are set to the empirical mean and the standard deviation, respectively in the case of real-valued attributes. These parameters are set to the minimum value and to the empirical standard deviation, in the case of positive real-valued and count attributes. This guarantees that the prior distributions on the latent variables is equally good for all the attributes in the dataset, regardless their support. The output structure `hidden` contains the latest MCMC sample of the latent feature vectors $\mathbf{z}_n$ for $n = 1, \dots, N$, the weighting vectors $\mathbf{B}^d$, as well as the auxiliary variables, which include the pseudo-observation variances σ_d^2 and the thresholds θ_r, necessary for the corresponding transformation $f_d(\cdot)$, for each dimension $d = 1, \dots, D$.

Furthermore, since the Bayesian nonparametric nature of the GLFM allows the model complexity (i.e., the length of the vectors $\mathbf{z}^n$ and $\mathbf{B}^d$) to grow with the number of observations, we sometimes need to put a bound on the model complexity, which is an additional input that can be set in the GLFM package. This bound allows us not only to keep the model complexity, and the running time under control, but also to efficiently manage the memory allocation. Finally, our implementation of the GLFM makes use of the GNU Scientific Library (GSL),[13] to efficiently perform a large variety of mathematical routines such as random number generation, and matrix or vector operations.

[12] This call corresponds to a python call. The equivalent call in Matlab is hidden = GLFM_infer(data) and in R output → GLFM_infer(data).

[13] https://www.gnu.org/software/gsl/

Type	Domain	Transformation $x = f_d(y)$	Hyperparam.
Real-valued	$x \in \Re$	$x = w(y + u) + \mu$	$\mu = mean(\mathbf{x}^d)$ $w = 2/std(\mathbf{x}^d)$
Positive	$x \in \Re^+$	$x = \log(\exp(w(y + u) + \mu) + 1)$	$\mu = min(\mathbf{x}^d)$ $w = 2/std(\mathbf{x}^d)$
Categorical	$x \in \{1, 2, \dots, R\}$ (unordered set)	$x = \arg\max_{r \in \{1,\dots,R\}} y_r$	
Ordinal	$x \in \{1, 2, \dots, R\}$ (ordered set)	$x = \begin{cases} 1 & \text{if} \quad y \leq \theta_1 \\ 2 & \text{if} \quad \theta_1 < y \leq \theta_2 \\ \vdots & \\ R & \text{if} \quad \theta_{R-1} < y \end{cases}$	
Count	$x \in \{1, 2, 3, \dots\}$	$x = \lfloor \log(\exp(w(y + u) + \mu) + 1) \rfloor$	$\mu = min(\mathbf{x}^d)$ $w = 2/std(\mathbf{x}^d)$

Table 13: Mapping functions implemented in the toolbox.

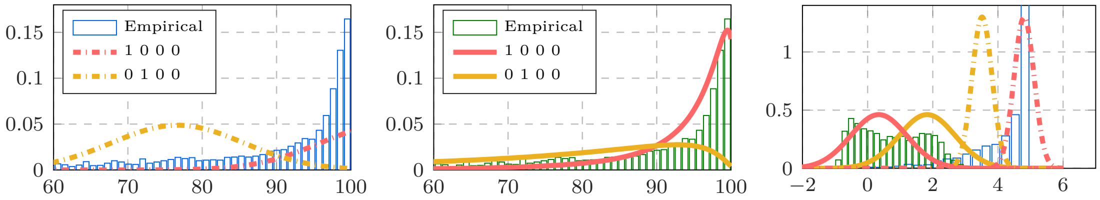
B.2. Usage

B.2.1. Data preprocessing and initialization

A convenient property of the GLFM package is that it can be used blindly on raw data without requiring any preprocessing step on the dataset, or special tuning of the hyperparameters. The only requirement for the user is to format the data as a numerical matrix of size $N \times D$ and build an additional vector for the type of data for each of the D attributes.

The set of hyperparameters of the GLFM can be divided into two groups, the parameters related to the prior distribution for the latent variables $\mathbf{Z}$ and $\mathbf{B}$, and the hyperparameters related to the link functions. The key hyperparameters are the ones related to the link functions which allow us to map an heterogeneous observation to the corresponding pseudo-observation, since the pseudo-observations could in principle take any value in $\Re$. As mentioned above, the parameters of the transformations in Table 13 are internally fixed such that the output of the inverse link function $f_d^{-1}(\cdot)$ per dimension is normalized with comparable mean and variance across different dimensions, which facilitate that the pseudo-observations fall in a ball centered around the zero vector in $\Re^D$. This choice was made to ensure that prior distribution of the weighting vectors $\mathbf{B}^d \sim \mathcal{N}(\mathbf{0}, \sigma_B^2)$ is independent to the data type of each dimension, such that the user does not need to specify a different prior suitable for each attribute (dimension) but instead a common prior has a similar effect across all the dimensions (and data types) in the data.

Since the GLFM model assumes that the the pseudo-observations are distributed as a mixture of Gaussian distributions with a potentially infinite number of components—of the form $\sum_{p=1,\dots,2^{K_+}} \pi_p \mathcal{N}(y^d | \mathbf{z}_p \mathbf{B}^d, \sigma_{y^d}^2)$, where K_+ is the total number of active features, p indicates the binary feature vector (a.k.a pattern), and π_p is the empirical probability of pattern p (such that $\sum_{p=1,\dots,2^{K_+}} \pi_p = 1$)—, it is able to fit any pseudo-observation distribution to within arbitrary error. However, for data exploratory tasks, if the pseudo-observation distribution is highly non-Gaussian, we may infer a large number of additional features to capture the non-Gaussianity, leading to less interpretable results. To alleviate this issue, we incorporate an additional functionality that allows the user to specify external preprocessing (external transformation) to the data in order to favor Gaussianity, and thus to further



(a) $\mathbf{X}^d$ without preprocessing (b) $\mathbf{X}^d$ with log preprocessing (c) Pseudo-observations $\mathbf{y}^d$

Figure 15: Illustration of optional data preprocessing. Panel (a) and (b) show the histograms of, respectively, a heavy-tailed attribute and the attribute after a logarithmic transformation, as well the distribution of the inferred latent feature patterns. Panel (c) shows the histogram of the pseudo-observations inferred for the original and the preprocessed attributes, as well the distribution of the inferred latent feature patterns. Here, we observe that the distribution of the attribute is better captured by the latent model when a preprocessing step is performed to correct/minimize the non-Gaussian behaviour of the attribute.

improve the performance of the algorithm. For instance, in cases in which the distribution of an attribute presents a clearly non-Gaussian behavior, e.g., it is concentrated around a single value or has a heavy-tailed distribution, it might be suitable to preprocess this variable by applying a logarithmic transformation, as shown in Figure 15. This functionality is exploited in our data exploration examples in Section 5.2. As an example, Figure 16 shows the empirical distribution of the pseudo-observations and the fitting provided by GLFM for all numerical variables in the dataset of the United States presidential election of 1992. The first two rows correspond to the dimensions depicted in Figure 11 in the main text, whereas the last row corresponds to the dimensions illustrated in Figure 8 in the main text.

Next, we also discuss the sensitivity of the inferred latent model with respect to the selection of the concentration parameter of the IBP, which we set as the prior for the latent feature matrix $\mathbf{Z}$. We point out that this is particularly important in the case of data exploration tasks, such as the ones performed in Section 5.2, since the insights obtained from the data exploration are desired to be robust with respect to the hyperparameters of the model. The concentration parameter α is directly related with the expected number of features for a given number of datapoints N (Griffiths and Ghahramani, 2011). Thus, one may expect that different values of this parameter may result in a different number of feature, and thus, to different explanations of the data. In order to show that the results are consistent independently of the prior parameters, we depict in Figure 17 normalize histogram obtained using three independent runs and and 500 samples for each run of the Gibbs sampler for the for the Counties dataset. Here we observe, that while the distribution of the inferred number of features becomes more heavy-tailed as we increase the concentration parameter α, the mode of the distribution barely changes. In fact, if we remove those feature that are active in less than 5% of the data, the resulting number of features does not change among samples or runs of the Gibbs sampler.

B.2.2. Missing Data Estimation

The GLFM toolbox can be used for estimation and imputation of missing data in heterogeneous datasets, where the missing values can be encoded with any (numerical) value that the user specifies. The Bayesian nature of the GLFM allows to efficiently infer the latent feature representation of the data using the available information (i.e., the non-missing val-

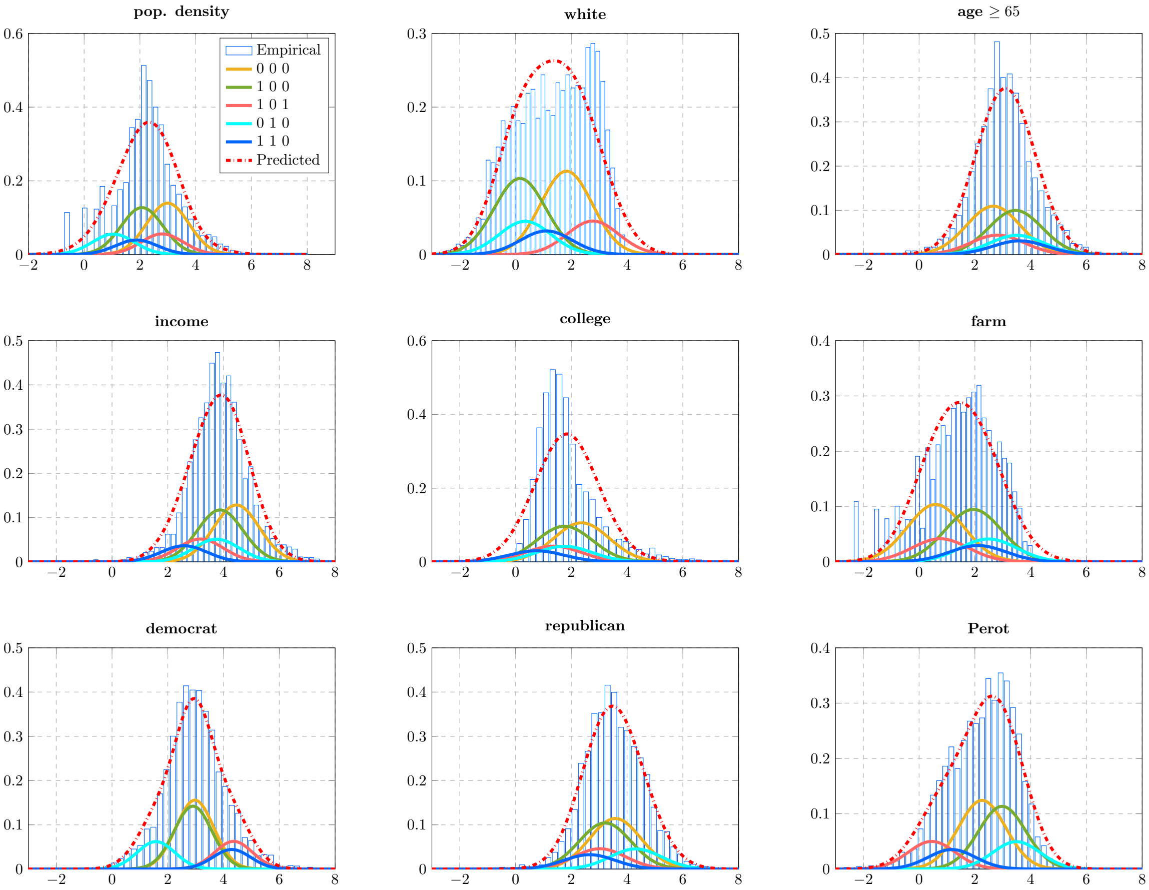


Figure 16: Distribution of the pseudo-observations for the dataset of the United States presidential election of 1992. Here, variables 'pop. density', 'income', 'farm', and 'white' present an additional pre-processing.

ues), and using it to compute the posterior distribution of each missing value in the data. Note that given the posterior distribution of each missing value, one might opt for different approaches to impute missing values, e.g., one might opt for imputing a sample of the posterior distribution or simply the maximum a posteriori (MAP) value. The GLFM package provides the function `[Xmap, hidden] = GLFM.complete(data)` which infers the latent feature representation, given the (incomplete) observation matrix, and returns a complete matrix where the missing values have been imputed to their MAP value; and the hidden structure containing all the inferred latent variables. This function runs the C++ inference engine `GLFM.infer()`, as well as the function `GLFM.computeMAP()`, which computes the MAP of a single missing element x_n^d given $\mathbf{z}^n$ and $\mathbf{B}^d$.

B.2.3. Data Exploration Analysis

The GLFM toolbox can also be used as a tool for data exploratory analysis, since it is able to find the latent structure in the data and capture the statistical dependencies among the objects and their attributes in the data. The GLFM toolbox provides weighted binary

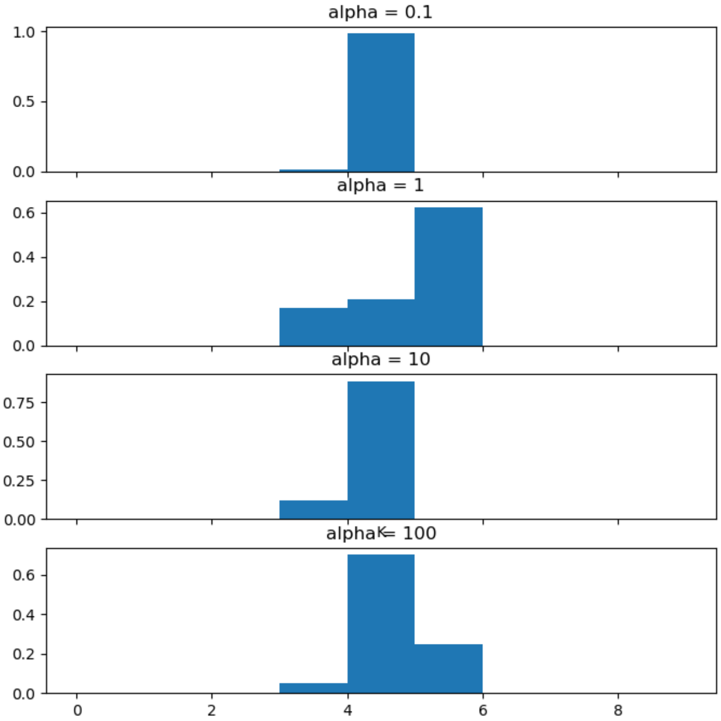


Figure 17: Sensitivity of the results to the concentration parameter α. We show the histogram of the number of inferred features for 500 samples of 3 independent run of the Gibbs sampler in the Counties dataset.

latent features, easing their interpretation and making it possible to cluster the objects according to their activation patterns of latent features. Furthermore, it also allows to activate a latent feature that is active for all the objects (a bias term), which is useful to capture the mode of the distribution of each attribute in the dataset. In order to help with data exploration, GLFM provides the function `GLFM.plotPatterns()`, which plots the posterior distribution of each attribute under the given latent feature patterns. This function allows us to find patterns and dependencies across both objects and attributes. This function, in turn, makes use of the function `GLFM.computePDF()`, which evaluates the posterior distribution of an attribute under a given latent feature vector. More generally, the function `GLFM.computeLogLikelihood()` computes the log likelihood of each entry in the provided matrix of observations given the different data type of each dimension.

B.2.4. Examples

The package manual contains simple examples demonstrating the package usage. Additionally, we provide the following demonstrations (with scripts in Python, Matlab and R):

- demo toy example: Simple illustration of GLFM pipeline, replicating the example of the IBP linear-Gaussian model in (Griffiths and Ghahramani, 2011).
- demo completion: Illustration of missing data estimation on the MNIST image dataset.
- demo data exploration (counties & prostate): Replication of results on data exploration in Section 6. This demo requires data download, which is instructed.

B.3. Availability and Documentation

GLFM code is publicly available in `https://github.com/ivaleraM/GLFM`, where we provide a technical document introducing the model and a user manual describing the usage details of the toolkit, including software requirements. The Python and Matlab implementations are under MIT license. The R implementation extends the *RcppGSLExample*[14], and therefore, is under GPL (>= 2) license.

References

O. Aguilar and M. West. Bayesian dynamic factor models and portfolio allocation. *Journal of Business & Economic Statistics*, 18(3):338–357, 2000.

R. M. Alvarez and J. Nagler. Economics, issues and the perot candidacy: voter choice in the 1992 presidential election. *American Journal of Political Science*, pages 714–744, 1995.

T. W. Anderson and Herman Rubin. Statistical inference in factor analysis. In *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability, Volume 5: Contributions to Econometrics, Industrial Research, and Psychometry*, pages 111–150, Berkeley, Calif., 1956. University of California Press. URL `https://projecteuclid.org/euclid.bsmsp/1200511860`.

C. Blanco, R. F. Krueger, D. S. Hasin, S. M. Liu, S. Wang, B. T. Kerridge, T. Saha, and M. Olfson. Mapping common psychiatric disorders: Structure and predictive validity in the National Epidemiologic Survey on Alcohol and Related Conditions. *Journal of the American Medical Association Psychiatry*, 70(2):199–208, 2013.

T. Broderick, J. Pitman, and M. I. Jordan. Feature allocations, probability functions, and paintboxes. *Bayesian Analysis*, 2013.

W. Buntine and A. Jakulin. Discrete Component Analysis. *arXiv:math/0604410*, 3940: 1–33, 2006.

D. P. Byar and S. B. Green. The choice of treatment for cancer patients based on covariate information: application to prostate cancer. *Bulletin du Cancer*, 67:477–490, 1980.

Pew Research Centre. 25th anniversary of the web. *Available on: http://www.pewinternet.org/datasets/january-2014-25th-anniversary-of-the-web-omnibus/*, 2014.

W. Chu and Z. Ghahramani. Gaussian processes for ordinal regression. *J. Mach. Learn. Res.*, 6:1019–1041, December 2005. ISSN 1532-4435.

[14] `https://github.com/eddelbuettel/rcppgsl/tree/master/inst/examples/RcppGSLExample`

M. Collins, S. Dasgupta, and R. E. Schapire. A generalization of principal components analysis to the exponential family. In *Advances in neural information processing systems*, pages 617–624, 2002.

G. Conti, S. Frühwirth-Schnatter, J.J. Heckman, and R. Piatek. Bayesian exploratory factor analysis. *Journal of Econometrics*, 2014.

P. Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems. Dataset available on: http://archive.ics.uci.edu/ml/datasets.html*, 47(4):547–553, 2009.

B. P. Dohrenwend. Sociocultural and social-psychological factors in the genesis of mental disorders. *Journal of Health and Social Behavior*, 16(4):365–392, 1975.

F. Doshi-Velez and Z. Ghahramani. Accelerated sampling for the indian buffet process. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 273–280, 2009.

F. Doshi-Velez, K. T. Miller, J. Van Gael, and Y. W. Teh. Variational inference for the indian buffet process. In *AISTATS*, 2009.

J. Eggermont, J. N. Kok, and W. A Kosters. Genetic programming for data classification: Partitioning the search space. In *Proceedings of the 2004 ACM symposium on Applied computing*, pages 1001–1005, 2004.

S. Frühwirth-Schnatter and H. F. Lopes. Parsimonious bayesian factor analysis when the number of factors is unknown. *Unpublished Working Paper, Booth Business*, 2010.

J. Geweke and G. Zhou. Measuring the Pricing Error of the Arbitrage Pricing Theory. CEMA Working Papers 276, China Economics and Management Academy, Central University of Finance and Economics, 1996.

Z. Ghahramani and G. Hinton. The EM algorithm for mixture of factor analysers. Technical report, University of Toronto, 1996.

M. Girolami and S. Rogers. Variational Bayesian multinomial probit regression with Gaussian process priors. *Neural Computation*, 18:2006, 2005.

P. Gopalan, F. J. R. Ruiz, R. Ranganath, and D. M. Blei. Bayesian Nonparametric Poisson Factorization for Recommendation Systems. *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2014.

D. Görür and C. E. Rasmussen. Nonparametric mixtures of factor analyzers. In *Signal Processing and Communications Applications Conference, 2009. SIU 2009. IEEE 17th*, pages 708–711. IEEE, 2009.

D. Greene, G. Cagney, N. Krogan, and P. Cunningham. Ensemble non-negative matrix factorization methods for clustering proteinâĂŞprotein interactions. *Bioinformatics*, 24 (15):1722–1728, August 2008. ISSN 1367-4803. doi: 10.1093/bioinformatics/btn286. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3493126/`.

T. L. Griffiths and Z. Ghahramani. The Indian buffet process: an introduction and review. *Journal of Machine Learning Research*, 12:1185–1224, 2011.

S. Gunasekar, P. Ravikumar, and J. Ghosh. Exponential family matrix completion under structural constraints. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1917–1925, Bejing, China, 22–24 Jun 2014. PMLR.

L. A. Hannah, D. M. Blei, and W. B. Powell. Dirichlet process mixtures of generalized linear models. *Journal of Machine Learning Research*, 12(Jun):1923–1953, 2011.

M. D. Hoffman, D. M. Blei, and P. R. Cook. Bayesian Nonparametric Matrix Factorization for Recorded Music. In *ICML*, pages 439–446, 2010.

A. B. Hollingshead and F. C. Redlich. Social stratification and psychiatric disorders. *American Sociological Review*, 18(2):163–169, 1953.

M. C Hughes, G. Hope, L. Weiner, T. H McCoy, R. H Perlis, E. B Sudderth, and F. Doshi-Velez. Prediction-constrained topic models for antidepressant recommendation. *arXiv preprint arXiv:1712.00499*, 2017.

A. Hyvärinen. Independent componen analysis by minimisation of mutual information. Technical report, University of technology, laboratory of computer and information science., 1997.

R. Kay. Treatment effects in competing-risks analysis of prostate cancer data. *Biometrics*, 42(1):203–211, 1986.

R. C. Kessler, K. A. McGonagle, M. Swartz, D. G. Blazer, and C. B. Nelson. Sex and depression in the national comorbidity survey i: Lifetime prevalence, chronicity and recurrence. *Journal of Affective Disorders*, 29(2):85–96, 1993.

M. E. Khan, A. Aravkin, M. Friedlander, and M. Seeger. Fast dual variational inference for non-conjugate latent gaussian models. In *International Conference on Machine Learning*, pages 951–959, 2013.

Mohammad E Khan, Guillaume Bouchard, Kevin P Murphy, and Benjamin M Marlin. Variational bounds for mixed-data factor analysis. In *Advances in Neural Information Processing Systems*, pages 1108–1116, 2010.

D. P Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

A. Klami, S. Virtanen, and S. Kaski. Bayesian exponential family projections for coupled data sources. *arXiv preprint arXiv:1203.3489*, 2012.

D. Lacy and B. C Burden. The vote-stealing and turnout effects of ross perot in the 1992 us presidential election. *American Journal of Political Science*, pages 233–255, 1999.

J. Lee, P. M´uller, K. Gulukota, and Ji Y. A Bayesian feature allocation model for tumor heterogeneity. *Annals of Applied Statistics*, 2015.

P. Lewis, C. McCracken, and R. Hunt. Politics: Who cares. *American Demographics*, pages 16–23, 1994.

X.-B. Li. A Bayesian approach for estimating and replacing missing categorical data. *J. Data and Information Quality*, 1(1):3:1–3:11, June 2009. ISSN 1936-1955.

H. Lopes and M. West. Bayesian model assessment in factor analysis. *Statistica Sinica*, 14 (1):41–67, 2004.

M. Lunn and D. McNeil. Applying cox regression to competing risks. *Biometrics*, 51(2): 524–532, 1995.

K. Mansouri, T. Ringsted, D. Ballabio, R. Todeschini, and V. Consonni. Quantitative structure activity relationship models for ready biodegradability of chemicals. *Journal of Chemical Information and Modeling. Dataset available on: http://archive.ics.uci.edu/ml/datasets.html*, 2013.

M. Arjumand Masood and Finale Doshi-Velez. A particle-based variational approach to Bayesian Non-negative Matrix Factorization. *Journal of Machine Learning Research*, 20 (90):1–56, 2019.

R. Mazumder, T. Hastie, and R. Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. *Journal of machine learning research*, 11(Aug):2287–2322, 2010.

K. Miller, M. I. Jordan, and T. L. Griffiths. Nonparametric Latent Feature Models for Link Prediction. In Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 1276–1284. Curran Associates, Inc., 2009. URL `http://papers.nips.cc/paper/3846-nonparametric-latent-feature-models-for-link-prediction.pdf`.

S. Mohamed, Z. Ghahramani, and K. A. Heller. Bayesian exponential family PCA. In *Advances in neural information processing systems*, pages 1089–1096, 2009.

J. Paisley and L. Carin. Nonparametric Factor Analysis with Beta Process Priors. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML'09, pages 777–784, New York, NY, USA, 2009. ACM.

K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 1901.

M. F Pradier, V. Stojkoski, Z. Utkovski, L. Kocarev, and F. Perez-Cruz. Sparse three-parameter restricted indian buffet process for understanding international trade. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2476–2480. IEEE, 2018.

M. F. Pradier, B. Reis, Lori Jukofsky, F. Milletti, T. Ohtomo, F. Perez-Cruz, and O. Puig. Case-control indian buffet process identifies biomarkers of response to codrituzumab. *BMC cancer*, 19(1):278, 2019.

C. Reed and Z. Ghahramani. Scaling the indian buffet process via submodular maximization. In *International Conference on Machine Learning*, pages 1013–1021, 2013.

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

M. E. Roberts, B. M. Stewart, and D. Tingley. Navigating the local modes of big data. *Computational Social Science*, 51, 2016.

A. S. Ross, M. C. Hughes, and F. Doshi-Velez. Right for the Right Reasons: Training Differentiable Models by Constraining their Explanations. *arXiv preprint arXiv:1703.03717*, 2017.

S. Roweis. EM algorithms for PCA and SPCA. In *Neural Information Processing Systems*, 1997.

F. J. R. Ruiz, I. Valera, C. Blanco, and F. Perez-Cruz. Bayesian nonparametric modeling of suicide attempts. *Advances in Neural Information Processing Systems*, 25:1862–1870, 2012.

F. J. R. Ruiz, I. Valera, C. Blanco, and F. Perez-Cruz. Bayesian nonparametric comorbidity analysis of psychiatric disorders. *Journal of Machine Learning Research.*, 2013.

F. J. R. Ruiz, I. Valera, C. Blanco, and F. Perez-Cruz. Bayesian Nonparametric Comorbidity Analysis of Psychiatric Disorders. *Journal of Machine Learning Research*, 15(1):1215–1247, January 2014.

R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, 2007.

R. Salakhutdinov and A. Mnih. Bayesian probabilistic matrix factorization using Markov Chain Monte Carlo. *Proceedings of the 25th International Conference on Machine Learning (ICML'08)*, pages 880–887, 2008.

J. L. Schafer and J. W Graham. Missing data: our view of the state of the art. *Psychological methods*, 7(2):147, 2002.

S. Seaman, J. Galati, D. Jackson, and J. Carlin. What is meant by "Missing at Random? *Statistical Science*, 2013.

A. P. Singh and G. J. Gordon. A Unified View of Matrix Factorization Models. In Walter Daelemans, Bart Goethals, and Katharina Morik, editors, *Machine Learning and Knowledge Discovery in Databases*, volume 5212, pages 358–373. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

M. Tanner and W. Wong. The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association*, 82:528–550, 1987.

Y. W. Teh and D. Görür. Indian buffet processes with power-law behavior. In *Advances in Neural Information Processing Systems*, volume 22, pages 1838–1846, 2009.

L. L. Thurstone. Multiple factor analysis. The University of Chicago Press, 1931.

M. Tipping and C. Bishop. Probabilistic principal component analysis. Technical report NCRG/97/010, Asto University, Department of computer science and applied mathematics, 1997.

M. Titsias. The infinite gamma-Poisson feature model. *Advances in Neural Information Processing Systems*, 19, 2007.

A. Todeschini, F. Caron, and M. Chavent. Probabilistic low-rank matrix completion with adaptive spectral regularization algorithms. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 845–853. Curran Associates, Inc., Dec. 2013.

Z. Utkovski, M. F. Pradier, V. Stojkoski, F. Perez-Cruz, and L. Kocarev. Economic complexity unfolded: Interpretable model for the productive structure of economies. *PloS one*, 13(8), 2018.

I. Valera and Z. Ghahramani. General table completion using a bayesian nonparametric model. *Advances in Neural Information Processing Systems 27 (NIPS 2014)*, 27:981–989, 2014.

I. Valera and Z. Ghahramani. Automatic discovery of the statistical types of variables in a dataset. In *International Conference on Machine Learning*, pages 3521–3529, 2017.

I. Valera, F. J. R. Ruiz, P. M. Olmos, C. Blanco, and F. Perez-Cruz. Infinite continuous feature model for psychiatric comorbidity analysis. *Neural computation*, 2016.

C. Wang and D. M Blei. Variational inference in nonconjugate models. *Journal of Machine Learning Research*, 14(Apr):1005–1031, 2013.

S. Weich and G. Lewis. Poverty, unemployment, and common mental disorders: population based cohort study. *BMJ*, 317(7151):115–119, 1998.

M. M. Weissman, Bland R., P. R. Joyce, S. Newman, J.E. Wells, and H.-U. Wittchen. Sex differences in rates of depression: cross-national perspectives. *Journal of Affective Disorders*, 29:77 – 84, 1993. Special Issue Toward a New Psychobiology of Depression in Women.

S. Williamson, C. Wang, K. Heller, and D. Blei. The IBP compound Dirichlet process and its application to focused topic modeling. *Proceedings of the 27th Annual International Conference on Machine Learning*, 2010.