Generalized Statistical Methods for Mixed Exponential Families, Part I: Theoretical Foundations

Cécile Levasseur, Kenneth Kreutz-Delgado, and Uwe F. Mayer

Abstract—This work considers the problem of learning the underlying statistical structure of multidimensional data of mixed probability distribution types (continuous and discrete) for the purpose of fitting a generative model and making decisions in a data-driven manner. Using properties of exponential family distributions and generalizing classical linear statistics techniques, a unified theoretical model called Generalized Linear Statistics (GLS) is established. The methodology exploits the split between data space and natural parameter space for exponential family distributions and solves a nonlinear problem by using classical linear statistical tools applied to data that have been mapped into the parameter space. The framework is equivalent to a computationally tractable, mixed data-type hierarchical Bayes graphical model assumption with latent variables constrained to a low-dimensional parameter subspace. We demonstrate that exponential family Principal Component Analysis, Semi-Parametric exponential family Principal Component Analysis, and Bregman soft clustering are not separate unrelated algorithms, but different manifestations of model assumptions and parameter choices taken within this common GLS framework. We readily extend these algorithms to deal with the important mixed data-type case. We study in detail the extreme case corresponding to exponential family Principal Component Analysis and solve problems related to fitting the generative model.

Index Terms—Generalized Linear Models, latent variables, exponential families, graphical models, dimensionality reduction.

1 Introduction

This paper proposes a new methodology for fitting generative models, for both continuous and discrete data, and in both the supervised and the non-supervised setting. The approach proposed and utilized here is a generalization and amalgamation of techniques from classical linear statistics, logistic regression, Principal Component Analysis (PCA), and Generalized Linear Models (GLMs) into a framework referred to, analogously to GLMs theory, as *Generalized Linear Statistics* (GLS). Generalized Linear Statistics includes techniques drawn from latent variable analysis [1], [2] as well as from the theory of Generalized Linear Models (GLMs) and Generalized Linear Mixed Models (GLMMs) [3], [4], [5], [6]. It is based on the use of exponential family distributions to model the various mixed types (continuous or discrete) of measured object properties. Despite the name, this is a *nonlinear* methodology which exploits the distinction in exponential family distributions between the *data space* (also known as the *expected value space*) and the *natural parameter space* as soon as one leaves the domain of purely Gaussian random variables. The point is that although the problem at hand may be nonlinear, it can be attacked using classical linear and other standard statistical tools applied to data that have been *mapped into the parameter space*, which is assumed to have a flat Euclidean space structure. For example, in the parameter space one can perform regression (resulting in the technique of logistic regression and other GLMs methods [3], [7], [5], [6], [4], [8]), PCA (resulting in a variety of "generalized PCA" methods [9], [10], [11], [12], [13], [2]), or clustering [14], [15], [16]. This approach provides an effective way to exploit tractably parameterized latent-variable exponential-family probability models to address the problem of data-driven learning of model parameters and features useful for the development of effective classification and regression algorithms. In addition to providing a better understanding of the data, learning the GLS model provides a generative model of the data, making it possible to generate synthetic data with the same (or at least similar, depending on the goodness of fit) statistical structure as the original data.

Building on a better understanding of previous work that first introduced Generalized Linear Statistics (GLS) [17], [18], [19], we now present a streamlined GLS framework and focus on important and new developments.

The paper is organized as follows. Section 2 presents the proposed Generalized Linear Statistics modeling approach in a mixed data-type hierarchical Bayes graphical model framework. We demonstrate the existence of approximately sufficient statistics in the extreme case of the GLS model corresponding to the exponential family Principal Component Analysis technique proposed in [10]. Section 3 describes the convex optimization prob-

- C. Levasseur and K. Kreutz-Delgado are with the Department of Electrical and Computer Engineering, University of California, San Diego (UCSD), La Jolla, CA, 92093.
E-mail: {clevasseur, kreutz}@ucsd.edu
- Uwe F. Mayer is with the Department of Mathematics, University of Utah, Salt Lake City, UT, 84112.
E-mail: mayer@math.utah.edu

lem related to fitting the above mentioned extreme case of the GLS model to a set of data. In light of the significant numerical difficulties associated with the cyclic-coordinate descent-like algorithm based on Bregman distance properties proposed in [10], especially in the mixed data-type case, this paper focuses on an algorithm based on Iterative Reweighted Least Squares (IRLS), an approach commonly used in the GLMs literature [4], [20], [21]. Using an IRLS-based learning algorithm makes it possible to tractably attack the more general problem of model fitting in a mixed data-type environment. Additionally, because the optimal model parameter values in the optimization problem may be non-finite [10], a penalty function is introduced that defines and places a set of constraints onto the loss function via a penalty parameter in a way so that any divergence to infinity is avoided. A key assumption of the GLS framework is that the parameters are restricted to a low-dimensional subspace. However, an orthonormality constraint is utilized for the matrix that defines this low-dimensional parameter subspace. It can be shown that otherwise the matrix is not unique and that other equivalent representations can be derived by orthogonal transformations of it [22]. The imposed orthonormality constraint reduces the impact of this identifiability problem. Section 4 presents a general point of view that relates the exponential family Principal Component Analysis (exponential PCA) technique of [10] to the Semi-Parametric exponential family Principal Component Analysis (SP-PCA) technique of [15] and to the Bregman soft clustering method presented in [16] and extends these algorithms to deal with the important mixed data-type case. A slight modification to the SP-PCA algorithm to handle the mixed data type is also introduced.

2 Generalized Linear Statistics (GLS)

2.1 Theoretical framework

The problem is abstractly stated as follows. A particular "object" of interest can be associated with a variety of descriptor random variables. Practitioners choose measurable descriptor variables that they believe are likely to be informative about interesting properties "attached to the object". These descriptors can be viewed as comprising the components of a random vector $\mathbf{x} = [x_1, \ldots, x_d] \in \mathbb{R}^d$, where the dimension d is equal to the number of descriptors. Thus the vector $\mathbf{x}$ is a point in a d-dimensional descriptor space.

Following the probabilistic Generalized Latent Variable (GLV) formalism described in [1], [2], it is assumed that training descriptor space points can be drawn from populations having factorable class-conditional probability density functions of the form:

$$p(\mathbf{x}|\boldsymbol{\theta}) = p_1(x_1|\boldsymbol{\theta}) \cdot \ldots \cdot p_d(x_d|\boldsymbol{\theta}) = p_1(x_1|\theta_1) \cdot \ldots \cdot p_d(x_d|\theta_d). \tag{1}$$

This is referred to as the *latent variable assumption* throughout this paper. Delta-functions are admitted so that densities are well-defined for discrete, continuous, and mixed random variables. Note the critical assumption that the components of $\mathbf{x}$ are independent, when conditioned on the parameter vector $\boldsymbol{\theta} \in \mathbb{R}^d$. It is further assumed that $\boldsymbol{\theta}$ can be written as

$$\boldsymbol{\theta} = \mathbf{a}\mathbf{V} + \mathbf{b} \tag{2}$$

with $\mathbf{V} \in \mathbb{R}^{q \times d}$ and $\mathbf{b} \in \mathbb{R}^d$ deterministic, and $\mathbf{a} \in \mathbb{R}^q$. While one generally assumes $q < d$ for dimensionality reduction (and ideally $q << d$), this is strictly speaking not required. This work both considers a Bayesian approach for which $\mathbf{a}$ is treated as a random vector and a classical approach where the vector $\mathbf{a}$ is deterministic. We first assume that $\mathbf{a}$ is a random vector. The randomness of $\mathbf{a}$ causes $\mathbf{a}$ to be called the *random effect*. The notation used here is motivated by the discussions in [10] and [23]. The matrix $\mathbf{V}$ is assumed to have full row-rank so that the relationship between $\mathbf{a}$ and $\boldsymbol{\theta}$ is one-to-one. Then, conditioning on the random vector $\boldsymbol{\theta}$ is equivalent to conditioning on the low-dimensional random vector $\mathbf{a}$, so that

$$p(\mathbf{x}|\boldsymbol{\theta}) = p(\mathbf{x}|\mathbf{a}) = p_1(x_1|\mathbf{a}) \cdot \ldots \cdot p_d(x_d|\mathbf{a}). \tag{3}$$

This is precisely the condition under which $\mathbf{a}$ is a *complete* latent variable [1]. In a probabilistic sense, all of the information that is *mutually* contained in the data vector $\mathbf{x}$ must be contained in the latent variable $\mathbf{a}$. As noted in [1], [24], [2], equations (1) and (2) generalize the classical factor analysis model (as described, e.g., in [25] and [1]) to the case when the marginal densities $p_i(x_i|\theta_i)$ are non-Gaussian. Indeed, the subscript "i" on $p_i(\cdot|\cdot)$ serves to indicate that the marginal densities can all be different, allowing for the possibility of $\mathbf{x}$ containing categorical, discrete, and continuous valued components. As described below, it is further assumed that the marginal densities are each one-parameter exponential family densities, allowing the use of a rich and powerful theory of such densities to be fruitfully exploited, and it is commonly the case that θ_i is taken to be the so-called *natural parameter* (or some bijective function of the natural parameter) of the exponential family density $p_i(\cdot|\cdot)$. A distribution is said to be a member of the exponential family if it has a density function of the form

$$p_i\big(x_i|\theta_i\big) = \exp\big(x_i\theta_i - G_i(\theta_i)\big),$$

where the function $G_i(\cdot)$ is the cumulant generating function, defined as

$$G_i(\theta_i) = \log \int_{\mathcal{X}_i} \mathrm{e}^{\theta_i x_i} \nu_i(\mathrm{d}x_i),$$

and where $\nu_i(\cdot)$ is a σ-finite measure that generates the exponential family and $\mathcal{X}_i$ defines the space of the data component x_i. The gradient of the cumulant generating function is denoted by $g_i(\cdot)$ and is referred to as the link function [26], [27], [28].

Because both GLMs and the Generalized Latent Variable (GLV) methodologies exploit the linear structure (2),

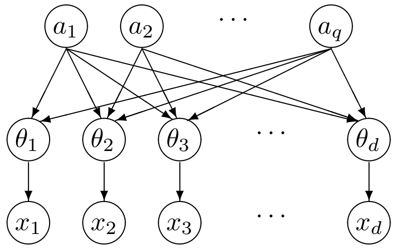

Fig. 1. Graphical model for the GLS approach.

they can be viewed as special cases of a GLS approach to data analysis. In the GLMs theory, $\mathbf{V}$ and $\mathbf{b}$ are known and $\mathbf{a}$ is deterministic and unknown [3], [4]. In both the Generalized Latent Variable theory described in [1], [24], [2] and the random- and Mixed-effects Generalized Linear Models (MGLMs) literature [29], [30], [23], [5], [4], [31], [20], [8], $\mathbf{V}$ and $\mathbf{b}$ are deterministic while $\mathbf{a}$ (and hence $\boldsymbol{\theta}$) is treated as a random vector. The difference between GLV and MGLMs is that in GLV, *all* of the quantities $\mathbf{V}$, $\mathbf{b}$, and $\mathbf{a}$ are unknown, and hence need to be identified, resulting in a so-called "blind" estimation problem, whereas in MGLMs, $\mathbf{V}$ is a known matrix of regressor variables and only the deterministic vector $\mathbf{b}$ and the unknown realizations of the *random effect* vector $\mathbf{a}$ (also known as *latent variable*) must be estimated. In both GLV and MGLMs, it is assumed that the linear relationship (2) holds in parameter space, and that the tools of linear and statistical inverse theory are applicable or insightful, at least conceptually. The MGLMs theory is a generalization of the classical theory of linear regression, while the GLV theory is a generalization of the classical theory of statistical factor analysis and PCA. In both cases, the generalization is based on a move from the data/description space containing the measurement vector $\mathbf{x}$ to the parameter space containing $\boldsymbol{\theta}$ via a generally nonlinear transformation known as a link function [3], [5], [4]. It is in the latter space that the linear relationship (2) is assumed to hold.

Graphical models, also referred to as Bayesian Networks when their graph is directed, are a powerful tool to encode and exploit the underlying statistical structure of complex data sets [32]. The GLS framework represents a subclass of graphical model techniques and its corresponding graphical model is presented in Fig. 1. It is equivalent to a computationally tractable mixed exponential families data-type hierarchical Bayes graphical model with latent variables constrained to a low-dimensional parameter subspace.

Since $\mathbf{a}$ (and hence $\boldsymbol{\theta}$) is treated as a random vector (Bayesian approach), the (non-conditional) probability density function $p(\mathbf{x})$ requires a generally intractable integration over the parameters,

$$p(\mathbf{x}) = \int p(\mathbf{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} = \int \prod_{i=1}^{d} p_i(x_i|\theta_i)\pi(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}, \tag{4}$$

where $\pi(\boldsymbol{\theta})$ is the probability density function of the parameter vector $\boldsymbol{\theta} = \mathbf{aV} + \mathbf{b}$. Given the observation matrix denoted by

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}[1] \\ \mathbf{x}[2] \\ \vdots \\ \mathbf{x}[n] \end{pmatrix} = \begin{pmatrix} x_1[1] & \ldots & x_d[1] \\ x_1[2] & \ldots & x_d[2] \\ \vdots & \ddots & \vdots \\ x_1[n] & \ldots & x_d[n] \end{pmatrix} \in \mathbb{R}^{n \times d} \tag{5}$$

composed of n independent and identically distributed statistical data samples, each assumed to be stochastically equivalent to the random row vector $\mathbf{x}$, $\mathbf{x}[k] = \big[x_1[k], \ldots, x_d[k]\big] \sim \mathbf{x}$, the data likelihood function is defined as

$$p(\mathbf{X}) = \prod_{k=1}^{n} p\big(\mathbf{x}[k]\big) = \prod_{k=1}^{n} \int p\big(\mathbf{x}[k]|\boldsymbol{\theta}\big)\pi(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \tag{6}$$

using the *latent variable assumption*, with $\boldsymbol{\theta} = \mathbf{aV} + \mathbf{b}$. For specified exponential family densities $p_i(\cdot|\cdot), i = 1, \ldots, d$, maximum likelihood identification of the model (4) corresponds to identifying $\pi(\boldsymbol{\theta})$, which, under the linear condition $\boldsymbol{\theta} = \mathbf{aV} + \mathbf{b}$, corresponds to identifying the matrix $\mathbf{V}$, the vector $\mathbf{b}$, and a density function, $\mu(\mathbf{a})$, on the random effect $\mathbf{a}$ via a maximization of the likelihood function $p(\mathbf{X})$ with respect to $\mathbf{V}$, $\mathbf{b}$, and $\mu(\mathbf{a})$. This is generally a quite difficult problem [5], [4], [20] and it is usually attacked using approximation methods which correspond to replacing the integrals in (4) and (6) by sums [23]:

$$p(\mathbf{x}) = \sum_{l=1}^{m} p\big(\mathbf{x}|\underline{\boldsymbol{\theta}}[l]\big)\pi_l = \sum_{l=1}^{m}\prod_{i=1}^{d} p_i\big(x_i|\underline{\theta}_i[l]\big)\pi_l, \tag{7}$$

$$p(\mathbf{X}) = \prod_{k=1}^{n}\sum_{l=1}^{m} p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[l]\big)\pi_{l,k} \tag{8}$$

over a finite number of discrete support points ("atoms") $\underline{\boldsymbol{\theta}}[l]$ (equivalently, $\underline{\mathbf{a}}[l]$) for $l = 1, \ldots, m$, $1 \leq m \leq n$, with point-mass probabilities

$$\pi_l \triangleq \pi\big(\boldsymbol{\theta} = \underline{\boldsymbol{\theta}}[l]\big) = \pi\big(\mathbf{a} = \underline{\mathbf{a}}[l]\big),$$
$$\pi_{l,k} \triangleq \pi\big(\boldsymbol{\theta}[k] = \underline{\boldsymbol{\theta}}[l]\big) = \pi\big(\mathbf{a}[k] = \underline{\mathbf{a}}[l]\big) = \pi_l,$$

the last equality resulting from the *independent and identically distributed statistical samples assumption*. Note that $\boldsymbol{\theta}$ and $\mathbf{a}$ are (discrete) *random variables* while $\underline{\boldsymbol{\theta}}[l]$ and $\underline{\mathbf{a}}[l], l = 1, \ldots, m$, are the m *nonrandom* support point *values* (i.e., the values of the random variables having nonzero probabilities). These m support points are shared by all the data points $\mathbf{x}[k]$, $k = 1, \ldots, n$. Also recall that taking $\pi\big(\underline{\boldsymbol{\theta}}[l]\big) = \pi\big(\underline{\mathbf{a}}[l]\big)$ for $\underline{\boldsymbol{\theta}}[l] = \underline{\mathbf{a}}[l]\mathbf{V} + \mathbf{b}$ with the matrix $\mathbf{V}$ having full row-rank results in the assumption that the relationship between the discrete values $\underline{\boldsymbol{\theta}}[l]$ and $\underline{\mathbf{a}}[l]$ is one-to-one. As clearly described in [23], this approximation is justified either as a Gaussian quadrature approximation to the integral in (6) in the case of a Gaussian assumption for the probability density function $\pi(\boldsymbol{\theta})$ [5], [4], [20], or by appealing to the fact that the Non-Parametric Maximum Likelihood (NPML) estimate [33], [34], [24] of the mixture density $\pi(\boldsymbol{\theta})$ yields a solution which takes a finite number of points of support [35], [33], [36], [37], [38], [39], [34].

With $\boldsymbol{\theta} = \mathbf{a}\mathbf{V} + \mathbf{b}$, with $\mathbf{V}$, $\mathbf{b}$ fixed and $\mathbf{a}$ random, the single-sample likelihood (7) is equal to

$$p(\mathbf{x}) = \sum_{l=1}^{m} p\big(\mathbf{x}|\underline{\boldsymbol{\theta}}[l]\big)\pi_l = \sum_{l=1}^{m} p\big(\mathbf{x}|\underline{\mathbf{a}}[l]\mathbf{V} + \mathbf{b}\big)\pi_l, \tag{9}$$

and the data likelihood (8) is equal to

$$p(\mathbf{X}) = \prod_{k=1}^{n}\sum_{l=1}^{m} p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[l]\big)\pi_l = \prod_{k=1}^{n}\sum_{l=1}^{m} p\big(\mathbf{x}[k]|\underline{\mathbf{a}}[l]\mathbf{V} + \mathbf{b}\big)\pi_l. \tag{10}$$

The data likelihood is thus (approximately) the likelihood of a finite mixture of exponential family densities with unknown mixture proportions or point-mass probability estimates π_l and unknown point-mass support points $\underline{\mathbf{a}}[l]$, with the linear predictor $\underline{\boldsymbol{\theta}}[l] = \underline{\mathbf{a}}[l]\mathbf{V} + \mathbf{b}$ in the lth mixture component [23]. In the mixture models literature, the point-mass probabilities π_l are called mixing proportions or weights, the densities $p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[l]\big)$ are called the component densities of the mixture and equation (9) is referred to as the m-component finite mixture density [24]. The combined problem of Maximum Likelihood Estimation (MLE) of the parameters $\mathbf{V}$, $\mathbf{b}$, the point-mass support points (atoms) $\underline{\mathbf{a}}[l]$ and the point-mass probability estimates $\pi_l, l = 1, \ldots, m$, (as approximations to the unknown, and possibly continuous density $\mu(\mathbf{a})$) is known as the Semiparametric Maximum Likelihood mixture density Estimation (SMLE) problem [34], [40], [24].

This problem can be attacked by using the Expectation-Maximization (EM) algorithm [41], [33], [36], [37], [42], [38], [23], [1], [43], [24], [44], [15]. Then, the number m of distinct support point values is often strictly smaller than the number of data points n, i.e., $m < n$. Note that, historically, Laird's classic 1978 paper [33] appears to be generally acknowledged as the first paper that proposed the EM algorithm for NPML estimation in the mixture density context; then, Lindsay's classic 1983 papers [36], [37] improved upon the theoretical foundations of the NPML estimation approach and later Mallet's 1986 paper [38] further explored some of the fundamental issues raised by Lindsay. As noted above, this problem (i.e., simultaneously identifying $\mathbf{b}$, $\underline{\mathbf{a}}[l]$, π_l for all l, and $\mathbf{V}$) is the subject matter of GLV analysis [1], [24], [45], [2]. The commonly encountered alternative problem of estimating $\mathbf{b}$, $\underline{\mathbf{a}}[l]$ and $\pi_l, l = 1, \ldots, m$, for *known* $\mathbf{V}$, where the elements of $\mathbf{V}$ are comprised of measured regressor variables, is a generalization of classical linear regression and is the subject matter of the theory of MGLMs [3], [29], [23], [5], [4], [31], [20], [8].

However, a classical approach to the GLS estimation problem can also be considered and the vector $\mathbf{a}$ (and hence $\boldsymbol{\theta}$) is treated as a deterministic vector. Then, to each data point $\mathbf{x}[k]$, $k = 1, \ldots, n$, corresponds a (generally different) parameter point, yielding a total of n points $\underline{\boldsymbol{\theta}}[k]$, $k = 1, \ldots, n$, in parameter space (and hence n points $\underline{\mathbf{a}}[k]$, $k = 1, \ldots, n$, in the low-dimensional parameter subspace) as presented in the exponential family Principal Component Analysis technique [10]. The data likelihood is simply equal to

$$p(\mathbf{X}) = \prod_{k=1}^{n} p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[k]\big) = \prod_{k=1}^{n} p\big(\mathbf{x}[k]|\underline{\mathbf{a}}[k]\mathbf{V} + \mathbf{b}\big). \tag{11}$$

Contrary to the Bayesian approach, no point-mass probabilities have to be estimated. For consistency of vocabulary throughout this paper, the points $\underline{\mathbf{a}}[k]$, $k = 1, \ldots, n$, in the low-dimensional parameter subspace are called latent variables for both Bayesian and classical approaches. Similarly, the parameter points $\underline{\boldsymbol{\theta}}[k]$, $k = 1, \ldots, n$, are called atoms in both approaches. The classical approach can also be seen as an extreme case of the Bayesian approach for which the probability density function $\pi(\boldsymbol{\theta})$ is a delta function (one per data point) and the total number of atoms m equals the number of data points n, i.e., $m = n$. Note that while the $m < n$ parameter points of the Bayesian approach are shared by all the data points, the classical approach assigns one parameter point to each data point (hence $m = n$). This extreme case is the approach followed in Section 3. Section 4 presents a general point of view and considers and compares both approaches ($m < n$ and $m = n$).

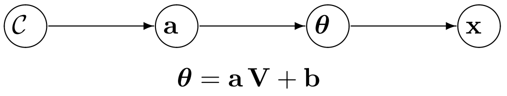

Fig. 2. The GLS model as a Markov chain.

2.2 Approximately sufficient statistics

Interestingly, in this extreme case of delta point-mass probabilities it can be shown that the point-mass support points or latent variables $\underline{\mathbf{a}}$ are approximate sufficient statistics and provide all the information needed to make decisions on future data.

Proof: We consider the problem of classifying data point $\mathbf{x}$. The Maximum A Posteriori (MAP) estimator for the class $\mathcal{C}$ is defined as follows:

$$\widehat{\mathcal{C}}_{MAP} \triangleq \arg\max_{\mathcal{C}} p(\mathcal{C}|\mathbf{x}) = \arg\max_{\mathcal{C}} p(\mathcal{C}, \mathbf{x}). \tag{12}$$

Acknowledging that the GLS graphical model is similar to the Markov chain presented in Figure 2, we have:

$$\begin{aligned} p(\mathcal{C}|\mathbf{x})p(\mathbf{x}) = p(\mathcal{C}, \mathbf{x}) &= \int p(\mathcal{C}, \mathbf{x}, \mathbf{a})\mathrm{d}\mathbf{a} \\ &= \int p(\mathcal{C}|\mathbf{x}, \mathbf{a})p(\mathbf{x}, \mathbf{a})\mathrm{d}\mathbf{a} \end{aligned}$$

and, because of the Markov chain structure,

$$\approx \int p(\mathcal{C}|\mathbf{a})p(\mathbf{a}|\mathbf{x})p(\mathbf{x})\mathrm{d}\mathbf{a}$$

and, because each data point $\mathbf{x}$ exactly corresponds to one support point $\underline{\mathbf{a}} = \underline{\mathbf{a}}(\mathbf{x})$,

$$\begin{aligned} &= \int p(\mathcal{C}|\mathbf{a})\delta(\mathbf{a} - \underline{\mathbf{a}})p(\mathbf{x})\mathrm{d}\mathbf{a} \\ &= p(\mathcal{C}|\underline{\mathbf{a}})p(\mathbf{x}). \end{aligned}$$

Hence, $p(\mathcal{C}|\underline{\mathbf{a}}) \approx p(\mathcal{C}|\mathbf{x})$ and $\underline{\mathbf{a}}$ is an approximate sufficient statistics. □

The goal of the work proposed and analyzed here is to fit an adequately faithful, class-conditional probability model of the form (10) for a Bayesian approach or (11) for a classical approach to labeled (when available) or unlabeled data to develop algorithms for making decisions on new measurements or future data. The family of models provided by (10) and (11), where the component densities are exponential family densities is very flexible and can be used to model both labeled and unlabeled data cases.

3 Learning at one extreme of GLS

This section focuses on the estimation procedure of Generalized Linear Statistics framework components for the extreme case where the number of parameter points equals the number of data points $m = n$. This extreme case is similar to the exponential family Principal Component Analysis (exponential PCA) technique proposed in [10]. We interpret it as a form of PCA performed in parameter space instead of data space as in classical PCA.

3.1 Problem description

The special Generalized Linear Statistics framework case presented in Section 2 where the number of parameter points equals the number of data points, i.e., $m = n$, is now solely considered. Hence, the point-mass probabilities do not need to be estimated and the EM algorithm is unnecessary. To each vector $\mathbf{x}$ corresponds a single vector $\underline{\mathbf{a}}$, i.e., a single vector $\underline{\boldsymbol{\theta}}$, and they all share a common index $k = 1, \ldots, n$.

Let $\mathbf{X}$ be the $(n \times d)$ matrix of observations defined in (5). The dimension of the data space is referred to as d and the number of points in the data set is referred to as n. The k'th row of the matrix $\mathbf{X}$ is the data row vector $\mathbf{x}[k]$. The observations can also be referred to as the data set $\{\mathbf{x}[k]\}_{k=1}^{n}$, where $\mathbf{x}[k] = \big[x_1[k], x_2[k], \ldots, x_d[k]\big]$. The proposed algorithm aims to identify the set of parameters $\{\underline{\boldsymbol{\theta}}[k]\}_{k=1}^{n}$, where each $\underline{\boldsymbol{\theta}}[k]$ is the "projection" of a corresponding $\mathbf{x}[k]$ onto a lower dimensional subspace of the parameter space. The dimension of this lower dimensional subspace is referred to as q, where $q < d$, ideally $q \ll d$, and its basis is defined as $\{\mathbf{v}_j\}_{j=1}^{q}$ where $\mathbf{v}_j = [v_{j1}, v_{j2}, \ldots, v_{jd}]$. Hence, the matrix $\mathbf{V}$ defined by

$$\mathbf{V} = \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_q \end{pmatrix} = \begin{pmatrix} v_{11} & \ldots & v_{1d} \\ v_{21} & \ldots & v_{2d} \\ \vdots & \ddots & \vdots \\ v_{q1} & \ldots & v_{qd} \end{pmatrix}$$

is $(q \times d)$ and identifies the low-dimensional parameter subspace. The latent variable matrix $\underline{\mathbf{A}}$ is $(n \times q)$ and represents the coordinates of each $\underline{\boldsymbol{\theta}}[k]$, $k = 1, \ldots, n$, in this lower dimensional subspace:

$$\underline{\mathbf{A}} = \begin{pmatrix} \underline{\mathbf{a}}[1] \\ \underline{\mathbf{a}}[2] \\ \vdots \\ \underline{\mathbf{a}}[n] \end{pmatrix} = \begin{pmatrix} \underline{a}_1[1] & \ldots & \underline{a}_q[1] \\ \underline{a}_1[2] & \ldots & \underline{a}_q[2] \\ \vdots & \ddots & \vdots \\ \underline{a}_1[n] & \ldots & \underline{a}_q[n] \end{pmatrix}.$$

Therefore, each $\underline{\boldsymbol{\theta}}[k]$, $k = 1, \ldots, n$, can be represented as a linear combination of the basis vectors plus a d-dimensional offset or displacement vector $\mathbf{b}$ as follows:

$$\underline{\boldsymbol{\theta}}[k] = \underline{\mathbf{a}}[k]\mathbf{V} + \mathbf{b} = \sum_{j=1}^{q} \underline{a}_j[k]\mathbf{v}_j + \mathbf{b}, \tag{13}$$

with $\mathbf{b} = [b_1, \ldots, b_d]$. The matrix $\underline{\boldsymbol{\Theta}} = \underline{\mathbf{A}}\mathbf{V} + \mathbf{B}$ is of the same dimensions as the observation matrix, namely $(n \times d)$, where the offset matrix $\mathbf{B}$ is $(n \times d)$ and simply composed of n identical rows $\mathbf{b}$.

Assuming a maximum likelihood estimation path as traditionally used in the GLMs literature [3], finding the $\underline{\boldsymbol{\theta}}[k]$ that is "best" in parameter space for its corresponding $\mathbf{x}[k]$ for all k means minimizing the negative log-likelihood function, or loss function, given by:

$$L(\underline{\mathbf{A}}, \mathbf{V}, \mathbf{b}) = -\log p(\mathbf{X}|\underline{\boldsymbol{\Theta}}), \tag{14}$$

subject to the constraint $\underline{\boldsymbol{\Theta}} = \underline{\mathbf{A}}\mathbf{V} + \mathbf{B}$.

In accordance with the GLS framework, the following assumptions are made:

(i) *the independent and identically distributed statistical samples assumption*: the samples $\mathbf{x}[k]$, $k = 1, \ldots, n$, are drawn independently and identically;

(ii) *the latent variable assumption*: the components $x_i[k]$, $i = 1, \ldots, d$, are independent when conditioned on the random parameter vector $\underline{\boldsymbol{\theta}}[k]$, i.e., $p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[k]\big) = p_1\big(x_1[k]|\underline{\theta}_1[k]\big) \cdot \ldots \cdot p_d\big(x_d[k]|\underline{\theta}_d[k]\big)$ for all k, $k = 1, \ldots, n$;

(iii) *the mixed or hybrid exponential family distributions assumption*: each density function $p_i\big(x_i[k]|\underline{\theta}_i[k]\big)$ is any one-parameter exponential family distribution with $\underline{\theta}_i[k]$ taken to be the natural parameter of the exponential family density or some simple function of it. This assumption allows the rich and powerful theory of exponential family distributions to be fruitfully utilized.

The marginal densities $p_i(\cdot|\cdot)$ can all be different, allowing for the possibility of $\mathbf{x}[k]$ containing continuous and discrete valued components. Consequently, the loss function in equation (14) becomes:

$$L(\underline{\mathbf{A}}, \mathbf{V}, \mathbf{b}) = -\sum_{k=1}^{n}\sum_{i=1}^{d} \log p_i\big(x_i[k]|\underline{\theta}_i[k]\big). \tag{15}$$

By exploiting the previously stated *latent variable assumption*, it can be shown that if $p_i(\cdot|\cdot)$ is the 1-dimensional conditional distribution of the component $x_i[k]$, $i =$

$1,\ldots,d$, of the data point $\mathbf{x}[k]$ given the parameter component $\underline{\theta}_i[k]$, then the vector $\mathbf{x}[k]$ follows a d-dimensional conditional exponential distribution $p(\cdot|\cdot)$ given the vector parameter $\underline{\boldsymbol{\theta}}[k]$.

Proof: Considering the most general case, each component x_i for $i = 1,\ldots,d$ is assumed to be exponentially distributed according to the distribution p_i with parameter $\underline{\theta}_i$, and the components are independent when conditioned on theirs parameters. Following the definition of standard exponential families, a σ-finite measure ν_i is assumed for each distribution, $i = 1,\ldots,d$. Let $\nu = (\nu_1,\ldots,\nu_d)$ with $\nu(\mathrm{d}\mathbf{x}) = \nu_1(\mathrm{d}x_1)\cdot\nu_2(\mathrm{d}x_2)\cdot\ldots\cdot\nu_d(\mathrm{d}x_d)$ be the σ-finite measure in the d-dimensional data space. The 1-dimensional distribution can be written as $p_i(x_i|\underline{\theta}_i) = \exp\big\{\underline{\theta}_i x_i - G_i(\underline{\theta}_i)\big\}$. Based on the *latent variable assumption*, the distribution of the vector $\mathbf{x}$ can be written as follows:

$$p(\mathbf{x}|\underline{\boldsymbol{\theta}}) = \prod_{i=1}^{d} p_i(x_i|\underline{\theta}_i) = \prod_{i=1}^{d} \mathrm{e}^{\underline{\theta}_i x_i - G_i(\underline{\theta}_i)} = \mathrm{e}^{\underline{\boldsymbol{\theta}}\cdot\mathbf{x} - \sum_{i=1}^{d} G_i(\underline{\theta}_i)},$$

where, by definition of an exponential family distribution and using Fubini's theorem [46],

$$\begin{aligned}\sum_{i=1}^{d} G_i(\underline{\theta}_i) &= \sum_{i=1}^{d} \log\int_{\mathcal{X}_i} \mathrm{e}^{\underline{\theta}_i x_i}\nu_i(\mathrm{d}x_i) = \log\prod_{i=1}^{d}\int_{\mathcal{X}_i} \mathrm{e}^{\underline{\theta}_i x_i}\nu_i(\mathrm{d}x_i)\\ &= \log\int_{\mathcal{X}_1}\cdots\int_{\mathcal{X}_d} \mathrm{e}^{\underline{\theta}_1 x_1 + \underline{\theta}_2 x_2 + \cdots + \underline{\theta}_d x_d}\nu_1(\mathrm{d}x_1)\cdot\ldots\cdot\nu_d(\mathrm{d}x_d)\\ &= \log\int_{\mathcal{X}} \mathrm{e}^{\underline{\boldsymbol{\theta}}\mathbf{x}}\nu(\mathrm{d}\mathbf{x}) = G(\underline{\boldsymbol{\theta}}), \end{aligned} \tag{16}$$

where $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_d$ defines the (product) space of the d-dimensional vector $\mathbf{x}$. As a result, $G(\underline{\boldsymbol{\theta}}) = \sum_{i=1}^{d} G_i(\underline{\theta}_i)$ is the cumulant generating function of the multivariate exponential family distribution $p(\mathbf{x}|\underline{\boldsymbol{\theta}}) = \prod_{i=1}^{d} p_i(x_i|\underline{\theta}_i)$. □

3.2 Estimation procedures for a single exponential family

First, the case of a single common exponential family, i.e., $p_i(\cdot|\cdot) = p(\cdot|\cdot)$ for all $i = 1,\ldots,d$, is considered.

3.2.1 *Loss function and convexity*

The loss function is given by equation (15). Using the definition of an exponential family distribution, the loss function can be written as:

$$\begin{aligned} L(\underline{\mathbf{A}},\mathbf{V},\mathbf{b}) &= \sum_{k=1}^{n}\sum_{i=1}^{d}\Big\{G\big(\underline{\theta}_i[k]\big) - x_i[k]\underline{\theta}_i[k]\Big\}\\ &= \sum_{k=1}^{n}\Big\{G\big(\underline{\mathbf{a}}[k]\mathbf{V}+\mathbf{b}\big) - \big(\underline{\mathbf{a}}[k]\mathbf{V}+\mathbf{b}\big)\mathbf{x}[k]^T\Big\}. \end{aligned} \tag{17}$$

Alternatively, it can be shown that the negative log-likelihood of the density of an exponential family distribution $p(x_i[k]|\underline{\theta}_i[k])$ can be expressed through a Bregman distance $B_F(\cdot\|\cdot)$:

$$-\log p(x_i[k]|\underline{\theta}_i[k]) = B_F\big(x_i[k]\big\|g(\underline{\theta}_i[k])\big) - F\big(x_i[k]\big),$$

where $F(\cdot)$ is the Fenchel conjugate of the cumulant generating function $G(\cdot)$ [28]. Then, the loss function in (15) becomes:

$$\begin{aligned} L(\underline{\mathbf{A}},\mathbf{V},\mathbf{b}) &= \sum_{k=1}^{n}\sum_{i=1}^{d}\Big\{B_F\big(x_i[k]\big\|g(\underline{\theta}_i[k])\big) - F\big(x_i[k]\big)\Big\}\\ &= \sum_{k=1}^{n}\sum_{i=1}^{d}\Big\{B_F\big(x_i[k]\big\|g(\underline{\theta}_i[k])\big)\Big\} - \underline{\sum_{k=1}^{n}\sum_{i=1}^{d} F\big(x_i[k]\big),} \end{aligned}$$

where $\underline{\theta}_i[k] = \sum_{j=1}^{q}\underline{a}_j[k]v_{ji} + b_i$. The underlined term in the above equation does not depend on either $\underline{\mathbf{A}}$, $\mathbf{V}$ or $\mathbf{b}$, resulting in the following minimization problem:

$$\begin{aligned} \operatorname*{arg\,min}_{\underline{\mathbf{A}},\mathbf{V},\mathbf{b}} L(\underline{\mathbf{A}},\mathbf{V},\mathbf{b}) &= \operatorname*{arg\,min}_{\underline{\mathbf{A}},\mathbf{V},\mathbf{b}} \sum_{k=1}^{n}\sum_{i=1}^{d} B_F\big(x_i[k]\big\|g(\underline{\theta}_i[k])\big)\\ &= \operatorname*{arg\,min}_{\underline{\mathbf{A}},\mathbf{V},\mathbf{b}} \sum_{k=1}^{n}\sum_{i=1}^{d} B_F\left(x_i[k]\Big\|g\Big(\sum_{j=1}^{q}\underline{a}_j[k]v_{ji}+b_i\Big)\right). \end{aligned} \tag{18}$$

It can be shown that the loss function is convex in either of its arguments with the others fixed. Indeed, the dual divergences property of the Bregman distance implies that, if $G\big(\underline{\theta}_i[k]\big)$ is strictly convex, then

$$\begin{aligned} B_F\big(x_i[k]\big\|g(\underline{\theta}_i[k])\big) &= B_G\Big(f\big(g(\underline{\theta}_i[k])\big)\Big\|f\big(x_i[k]\big)\Big)\\ &= B_G\big(\underline{\theta}_i[k]\big\|f(x_i[k])\big), \end{aligned} \tag{19}$$

since the function $f(\cdot)$ is the inverse of the link function $g(\cdot)$. The fact that $f(\cdot)$ and $g(\cdot)$ are inverse functions of each other is easily shown. Using equation (19) in equation (18), the minimization problem becomes, if $G\big(\underline{\theta}_i[k]\big)$ is strictly convex:

$$\begin{aligned} \operatorname*{arg\,min}_{\underline{\mathbf{A}},\mathbf{V},\mathbf{b}} L(\underline{\mathbf{A}},\mathbf{V},\mathbf{b}) &= \operatorname*{arg\,min}_{\underline{\mathbf{A}},\mathbf{V},\mathbf{b}} \sum_{k=1}^{n}\sum_{i=1}^{d} B_G\big(\underline{\theta}_i[k]\big\|f(x_i[k])\big)\\ &= \operatorname*{arg\,min}_{\underline{\mathbf{A}},\mathbf{V},\mathbf{b}} \sum_{k=1}^{n}\sum_{i=1}^{d} B_G\big(\sum_{j=1}^{q}\underline{a}_j[k]v_{ji}+b_i\big\|f(x_i[k])\big). \end{aligned}$$

This is a critical step of GLS because the minimization problem is moved from data space fully into parameter space.

It is well-known that $G\big(\underline{\theta}_i[k]\big)$ is a convex function on $\underline{\theta}_i[k]$ and strictly convex if the exponential family is minimal [27]. The convexity property of Bregman distances states that $B_G(\cdot\|\cdot)$ is always convex in the first argument, resulting in the fact that $B_G\big(\underline{\theta}_i[k]\big\|f(x_i[k])\big)$ is convex in $\underline{\theta}_i[k]$ for all $k = 1,\ldots,n$ and $i = 1,\ldots,d$. Then, since $\underline{\theta}_i[k] = \sum_{j=1}^{q}\underline{a}_j[k]v_{ji} + b_i$ is a convex relationship in either $a_j[k], j = 1,\ldots,q$, $v_{ji}, j = 1,\ldots,q$, or b_i with the others fixed, for all $k = 1,\ldots,n$ and $i = 1,\ldots,d$, the Bregman distance $B_G\big(\sum_{j=1}^{q}\underline{a}_j[k]v_{ji}+b_i\big\|f(x_i[k])\big)$ is convex in any of the three arguments when the other two are fixed. Therefore, as a sum of convex functions, the loss function is convex in either of its arguments with the others fixed, i.e., the loss function is convex in $\underline{\boldsymbol{\Theta}} = \underline{\mathbf{A}}\mathbf{V} + \mathbf{B}$.

Since the loss function is convex in either of its arguments with the others fixed, its minimization can be attacked by using an iterative approach. Then, the first step, given a fixed matrix $\mathbf{V}$ and a fixed vector $\mathbf{b}$, is to obtain the matrix $\underline{\mathbf{A}}$ or the set of vectors $\underline{\mathbf{a}}[k]$ for $k = 1, \dots, n$, that minimizes the loss function given by equation (17). The second step, given a fixed matrix $\underline{\mathbf{A}}$ and a fixed vector $\mathbf{b}$, is to obtain the matrix $\mathbf{V}$ or the set of vectors $\mathbf{v}_j$ for $j = 1, \dots, q$, that minimizes the loss function. The last step, given a fixed matrix $\underline{\mathbf{A}}$ and a fixed matrix $\mathbf{V}$, is to obtain the vector $\mathbf{b}$ that minimizes the loss function.

3.2.2 Iterative minimization of the loss function

The classical Newton-Raphson method is used for the iterative minimization of the loss function (17).

The first step in the $(t+1)^{\text{st}}$ iteration consists of the update $\underline{\mathbf{A}}^{(t+1)} = \arg\min_{\underline{\mathbf{A}}} L(\underline{\mathbf{A}}, \mathbf{V}^{(t)}, \mathbf{b}^{(t)})$, with $\underline{\mathbf{A}}^{(t)}$, $\mathbf{V}^{(t)}$ and $\mathbf{b}^{(t)}$ being the updates obtained at the end of the t^{th} iteration. It then requires the computation of the gradient vector $\nabla_{\underline{\mathbf{a}}} l\big(\underline{\mathbf{a}}[k]\big)$ and the Hessian matrix $\nabla^2_{\underline{\mathbf{a}}} l\big(\underline{\mathbf{a}}[k]\big)$ of the loss function $l\big(\underline{\mathbf{a}}[k]\big)$ with respect to the vector $\underline{\mathbf{a}}[k]$, for all $k = 1, \dots, n$, where $l\big(\underline{\mathbf{a}}[k]\big) = l\big(\underline{\mathbf{a}}[k], \mathbf{V}^{(t)}, \mathbf{b}^{(t)}\big)$ collects the elements of the loss function $L(\underline{\mathbf{A}}, \mathbf{V}^{(t)}, \mathbf{b}^{(t)})$ that depend only on the vector $\underline{\mathbf{a}}[k]$:

$$\begin{aligned} l\big(\underline{\mathbf{a}}[k]\big) &= G\big(\underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) - \big(\underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big)\mathbf{x}[k]^T \quad (20)\\ &= \sum_{i=1}^{d} \left\{ G\left(\sum_{j=1}^{q} \underline{a}_j[k] v_{ji}^{(t)} + b_i^{(t)} \right) - x_i[k] \left(\sum_{j=1}^{q} \underline{a}_j[k] v_{ji}^{(t)} + b_i^{(t)} \right) \right\}. \end{aligned}$$

The gradient vector of the loss function $l\big(\underline{\mathbf{a}}[k]\big)$ with respect to the vector $\underline{\mathbf{a}}[k]$, for $k = 1, \dots, n$, is given by

$$\nabla_{\underline{\mathbf{a}}} l\big(\underline{\mathbf{a}}[k]\big) = \frac{\partial l\big(\underline{\mathbf{a}}[k]\big)}{\partial \underline{\mathbf{a}}[k]} = \mathbf{V}^{(t)} \Big(G'\big(\underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) - \mathbf{x}[k]^T \Big),$$

where, for $\underline{\boldsymbol{\theta}}[k] = \underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}$,

$$G'\big(\underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) = \frac{\partial G\big(\underline{\boldsymbol{\theta}}[k]\big)}{\partial \underline{\boldsymbol{\theta}}[k]} \quad \text{and} \quad \frac{\partial \underline{\boldsymbol{\theta}}[k]}{\partial \underline{\mathbf{a}}[k]} = \mathbf{V}^{(t)}.$$

Here, the following convention for derivatives with respect to a row vector is used: for $\underline{\mathbf{a}}[k]$, a $(1 \times q)$ vector, and $l(\cdot)$, a scalar function of $\underline{\mathbf{a}}[k]$, $\partial l\big(\underline{\mathbf{a}}[k]\big)/\partial \underline{\mathbf{a}}[k] = \big[\partial l\big(\underline{\mathbf{a}}[k]\big)/\partial \underline{a}_1[k], \dots, \partial l\big(\underline{\mathbf{a}}[k]\big)/\partial \underline{a}_q[k]\big]^T$ is a $(q \times 1)$ vector. Similarly, for $\underline{\boldsymbol{\theta}}[k]$, a $(1 \times d)$ vector, and $G(\cdot)$, a scalar function of $\underline{\boldsymbol{\theta}}[k]$, $\partial G\big(\underline{\boldsymbol{\theta}}[k]\big)/\partial \underline{\boldsymbol{\theta}}[k]$ is a $(d \times 1)$ vector as follows: $\partial G\big(\underline{\boldsymbol{\theta}}[k]\big)/\partial \underline{\boldsymbol{\theta}}[k] = \big[\partial G\big(\underline{\boldsymbol{\theta}}[k]\big)/\partial \underline{\theta}_1[k], \dots, \partial G\big(\underline{\boldsymbol{\theta}}[k]\big)/\partial \underline{\theta}_d[k]\big]^T$. Then,

$$\begin{aligned} G'\big(\underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) &= \left[\frac{\partial G\big(\underline{\boldsymbol{\theta}}[k]\big)}{\partial \underline{\theta}_1[k]}, \dots, \frac{\partial G\big(\underline{\boldsymbol{\theta}}[k]\big)}{\partial \underline{\theta}_d[k]} \right] \\ &= \left[\frac{\partial}{\partial \underline{\theta}_1[k]} \sum_{i=1}^{d} G\big(\underline{\theta}_i[k]\big), \dots, \frac{\partial}{\partial \underline{\theta}_d[k]} \sum_{i=1}^{d} G\big(\underline{\theta}_i[k]\big) \right]^T \end{aligned}$$

for $\underline{\boldsymbol{\theta}}[k] = \underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}$ and using equation (16)

$$= \big[g\big(\underline{\theta}_1[k]\big), \dots, g\big(\underline{\theta}_d[k]\big) \big],$$

where $g\big(\underline{\theta}_i[k]\big) = \partial G\big(\underline{\theta}_i[k]\big)/\partial \underline{\theta}_i[k]$. The Hessian matrix of the loss function with respect to the vector $\underline{\mathbf{a}}[k]$ is given by

$$\nabla^2_{\underline{\mathbf{a}}} l\big(\underline{\mathbf{a}}[k]\big) = \frac{\partial^2 l\big(\underline{\mathbf{a}}[k]\big)}{\partial \underline{\mathbf{a}}[k]^2} = \mathbf{V}^{(t)} G''\big(\underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) \mathbf{V}^{(t),T},$$

where

$$G''\big(\underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) = \begin{bmatrix} \frac{\partial^2 G(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_1[k]^2} & \cdots & \frac{\partial^2 G(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_d[k] \partial \underline{\theta}_1[k]} \\ \vdots & \vdots & \vdots \\ \frac{\partial^2 G(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_1[k] \partial \underline{\theta}_d[k]} & \cdots & \frac{\partial^2 G(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_d[k]^2} \end{bmatrix}.$$

Furthermore,

$$\frac{\partial^2 G\big(\underline{\boldsymbol{\theta}}[k]\big)}{\partial \underline{\theta}_r[k] \partial \underline{\theta}_s[k]} = \frac{\partial^2}{\partial \underline{\theta}_r[k] \partial \underline{\theta}_s[k]} \sum_{i=1}^{d} G(\underline{\theta}_i[k]) = \frac{\partial}{\partial \underline{\theta}_s[k]} g(\underline{\theta}_r[k]),$$

so that

$$\frac{\partial^2 G\big(\underline{\boldsymbol{\theta}}[k]\big)}{\partial \underline{\theta}_r[k] \partial \underline{\theta}_s[k]} = \begin{cases} 0 & \text{if } r \neq s, \\ \partial g(\underline{\theta}_r[k])/\partial \underline{\theta}_r[k] & \text{if } r = s. \end{cases}$$

As a result,

$$G''\big(\underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) = \begin{bmatrix} \frac{\partial g(\underline{\theta}_1[k])}{\partial \underline{\theta}_1[k]} & \cdots & 0 \\ \vdots & \vdots & \vdots \\ 0 & \cdots & \frac{\partial g(\underline{\theta}_d[k])}{\partial \underline{\theta}_d[k]} \end{bmatrix},$$

i.e., $G''\big(\underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big)$ is a $(d \times d)$ diagonal matrix.

The Newton-Raphson technique simply solves the minimization problem $\arg\min_{\underline{\mathbf{a}}} l(\underline{\mathbf{a}}[k], \mathbf{V}^{(t)}, \mathbf{b}^{(t)})$ at iteration $(t+1)$ by using the update

$$\begin{aligned} &\underline{\mathbf{a}}^{(t+1)}[k] = \underline{\mathbf{a}}^{(t)}[k] \\ &- \alpha_{\underline{\mathbf{a}}}^{(t+1)} \Big(\nabla^2_{\underline{\mathbf{a}}} l\big(\underline{\mathbf{a}}^{(t)}[k], \mathbf{V}^{(t)}, \mathbf{b}^{(t)}\big) \Big)^{-1} \nabla_{\underline{\mathbf{a}}} l\big(\underline{\mathbf{a}}^{(t)}[k], \mathbf{V}^{(t)}, \mathbf{b}^{(t)}\big), \end{aligned}$$

where $\nabla l(\cdot)$ is the gradient of the function $l(\cdot)$, $\nabla^2 l(\cdot)$ its Hessian matrix and $\alpha_{\underline{\mathbf{a}}}^{(t+1)}$ the so-called step size. It yields the following update equation for the set of vectors $\underline{\mathbf{a}}^{(t+1)}[k]$ at iteration $(t+1)$ for $k = 1, \dots, n$:

$$\begin{aligned} \underline{\mathbf{a}}^{(t+1)}[k]^T &= \underline{\mathbf{a}}^{(t)}[k]^T \\ &\quad - \alpha_{\underline{\mathbf{a}}}^{(t+1)} \Big(\mathbf{V}^{(t)} G''\big(\underline{\mathbf{a}}^{(t)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) \mathbf{V}^{(t),T} \Big)^{-1} \\ &\quad \cdot \Big(\mathbf{V}^{(t)} \big(G'(\underline{\mathbf{a}}^{(t)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}) - \mathbf{x}[k]^T \big) \Big). \end{aligned} \quad (21)$$

The second step in the iterative minimization method consists of the update $\mathbf{V}^{(t+1)} = \arg\min_{\mathbf{V}} L(\underline{\mathbf{A}}^{(t+1)}, \mathbf{V}, \mathbf{b}^{(t)})$. It requires the computation of the gradient vector $\nabla_{\mathbf{v}} l\big(\mathbf{v}_j\big)$ and the Hessian matrix $\nabla^2_{\mathbf{v}} l\big(\mathbf{v}_j\big)$ of the loss function $l\big(\mathbf{v}_j\big)$ with respect to the vector $\mathbf{v}_j$, for all $j = 1, \dots, q$, where $l\big(\mathbf{v}_j\big) = l\big(\underline{\mathbf{A}}^{(t+1)}, \{\mathbf{v}_j\}_{j=1}^q, \mathbf{b}^{(t)}\big)$ collects the elements of the loss function $L(\underline{\mathbf{A}}^{(t+1)}, \mathbf{V}, \mathbf{b}^{(t)})$ that depend only on the vector $\mathbf{v}_j$.

$$
\begin{aligned}
l(\mathbf{v}_j) &= \sum_{k=1}^{n} \Bigg\{ G\left(\sum_{r=1}^{q} \underline{a}_r^{(t+1)}[k]\mathbf{v}_r + \mathbf{b}^{(t)} \right) \\
&\qquad - \left(\sum_{r=1}^{q} \underline{a}_r^{(t+1)}[k]\mathbf{v}_r + \mathbf{b}^{(t)} \right) \mathbf{x}[k]^T \Bigg\}, \\
\nabla_{\mathbf{v}} l(\mathbf{v}_j) &= \frac{\partial l(\mathbf{v}_j)}{\partial \mathbf{v}_j} \\
&= \sum_{k=1}^{n} \underline{a}_j^{(t+1)}[k] \big\{ G'\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V} + \mathbf{b}^{(t)}\big) - \mathbf{x}[k]^T \big\}, \\
\nabla_{\mathbf{v}}^2 l(\mathbf{v}_j) &= \frac{\partial^2 l(\mathbf{v}_j)}{\partial \mathbf{v}_j^2} = \sum_{k=1}^{n} \underline{a}_j^{(t+1)}[k]^2 G''\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V} + \mathbf{b}^{(t)}\big).
\end{aligned}
$$

Then, the update equation is given as follows for $j = 1, \ldots, q$:

$$
\begin{aligned}
\mathbf{v}_j^{(t+1),T} &= \mathbf{v}_j^{(t),T} \\
&- \alpha_{\mathbf{v}}^{(t+1)} \left(\sum_{k=1}^{n} \underline{a}_j^{(t+1)}[k]^2 G''\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) \right)^{-1} \\
&\cdot \left(\sum_{k=1}^{n} \underline{a}_j^{(t+1)}[k] \big\{ G'\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) - \mathbf{x}[k]^T \big\} \right),
\end{aligned}
\tag{22}
$$

where

$$
\sum_{k=1}^{n} \underline{a}_j^{(t+1)}[k]^2 G''(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}) =
$$
$$
\begin{bmatrix}
\sum_{k=1}^{n} \underline{a}_j^{(t+1)}[k]^2 \frac{\partial g(\underline{\theta}_1[k])}{\partial \underline{\theta}_1[k]} & & 0 \\
\vdots & \ddots & \vdots \\
0 & & \sum_{k=1}^{n} \underline{a}_j^{(m+1)}[k]^2 \frac{\partial g(\underline{\theta}_d[k])}{\partial \underline{\theta}_d[k]}
\end{bmatrix}
$$

for $\underline{\boldsymbol{\theta}}[k] = \underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}$.

The last step in the iterative minimization method consists of the update $\mathbf{b}^{(t+1)} = \arg\min_{\mathbf{b}} L(\underline{\mathbf{A}}^{(t+1)}, \mathbf{V}^{(t+1)}, \mathbf{b})$. It requires the computation of the gradient vector $\nabla_{\mathbf{b}} l\big(\mathbf{b}\big)$ and the Hessian matrix $\nabla_{\mathbf{b}}^2 l\big(\mathbf{b}\big)$ of the loss function $l\big(\mathbf{b}\big)$ with respect to the offset vector $\mathbf{b}$, where $l\big(\mathbf{b}\big) = l\big(\underline{\mathbf{A}}^{(t+1)}, \mathbf{V}^{(t+1)}, \mathbf{b}\big)$ collects the elements of the loss function $L(\underline{\mathbf{A}}^{(t+1)}, \mathbf{V}^{(t+1)}, \mathbf{b})$ that depend only on the vector $\mathbf{b}$.

$$
\begin{aligned}
l(\mathbf{b}) &= \sum_{k=1}^{n} \Big\{ G\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}\big) \\
&\qquad - \big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}\big)\mathbf{x}[k]^T \Big\}, \\
\nabla_{\mathbf{b}} l(\mathbf{b}) &= \frac{\partial l(\mathbf{b})}{\partial \mathbf{b}} = \sum_{k=1}^{n} \Big\{ G'\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}\big) - \mathbf{x}[k]^T \Big\}, \\
\nabla_{\mathbf{b}}^2 l(\mathbf{b}) &= \frac{\partial^2 l(\mathbf{b})}{\partial \mathbf{b}^2} = \sum_{k=1}^{n} G''\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}\big).
\end{aligned}
$$

Then, the update equation is given as follows:

$$
\begin{aligned}
\mathbf{b}^{(t+1),T} = \mathbf{b}^{(t),T} - \alpha_{\mathbf{b}}^{(t+1)} &\left(\sum_{k=1}^{n} G''\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}^{(t)}\big) \right)^{-1} \\
&\cdot \left(\sum_{k=1}^{n} \big\{ G'\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}^{(t)}\big) - \mathbf{x}[k]^T \big\} \right),
\end{aligned}
\tag{23}
$$

where

$$
G''(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}^{(t)}) =
\begin{bmatrix}
\frac{\partial g(\underline{\theta}_1[k])}{\partial \underline{\theta}_1[k]} & & 0 \\
\vdots & \ddots & \vdots \\
0 & & \frac{\partial g(\underline{\theta}_d[k])}{\partial \underline{\theta}_d[k]}
\end{bmatrix}
$$

for $\underline{\boldsymbol{\theta}}[k] = \underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}^{(t)}$.

Note that only the cumulant generating function $G(\cdot)$ needs to be changed in order to get an algorithm for a loss function involving a new exponential family. This is pertinent since the cumulant generating function uniquely defines the exponential family [27].

3.2.3 Penalty function approach

As noted in [10], it is possible for the atoms obtained with the extreme GLS case corresponding to exponential PCA to diverge since the optimum may be at infinity. To avoid such behavior, we introduce a penalty function that defines and places a set of constraints into the loss function via a penalty parameter in a way that penalizes any divergence to infinity.

The penalty function approach is used to convert the nonlinear programming problem with equality and inequality constraints into an unconstrained problem, or into a problem with simple constraints [47], [48], [49]. This transformation is accomplished by defining an appropriate auxiliary function in terms of the problem functions to define a new objective or loss function.

The penalty function is defined as follows for $\underline{\boldsymbol{\theta}} = [\underline{\theta}_1, \ldots, \underline{\theta}_d]$:

$$
\begin{aligned}
\psi(\underline{\boldsymbol{\theta}}) = \sum_{i=1}^{d} \Big\{ &\exp\big(- \beta_{min}(\underline{\theta}_i - \theta_{min})\big) \\
&+ \exp\big(\beta_{max}(\underline{\theta}_i - \theta_{max})\big) \Big\},
\end{aligned}
\tag{24}
$$

and was designed so that $\psi(\underline{\boldsymbol{\theta}})$ is close to zero for $\theta_{min} \leq \underline{\theta}_i \leq \theta_{max}$, $i = 1, \ldots, d$, and reaches infinity otherwise. Figure 3 shows possible shapes for the penalty function depending on the parameters β_{min} and β_{max} values.

The loss function becomes:

$$
\bar{L}(\underline{\mathbf{A}}, \mathbf{V}, \mathbf{b}) = \sum_{k=1}^{n} \Big\{ B_F\big(\mathbf{x}[k] \big\| g(\underline{\mathbf{a}}[k]\mathbf{V} + \mathbf{b})\big) + c \cdot \psi\big(\underline{\mathbf{a}}[k]\mathbf{V} + \mathbf{b}\big) \Big\}
\tag{25}
$$

instead of

$$
L(\underline{\mathbf{A}}, \mathbf{V}, \mathbf{b}) = \sum_{k=1}^{n} B_F\big(\mathbf{x}[k] \big\| g(\underline{\mathbf{a}}[k]\mathbf{V} + \mathbf{b})\big),
$$

where the scalar c is called the *penalty parameter*.

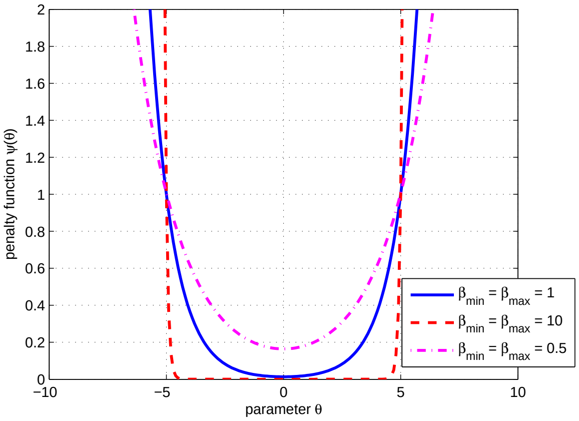

Fig. 3. Sketches for a possible penalty function ($\theta_{min} = -5, \theta_{max} = 5$): solid line for penalty function parameters $\beta_{min} = \beta_{max} = 1$, dashed line for parameters $\beta_{min} = \beta_{max} = 10$ and dashdot line for $\beta_{min} = \beta_{max} = 0.5$.

The previously developed iterative minimization algorithm is then used on $\bar{L}(\underline{\mathbf{A}}, \mathbf{V}, \mathbf{b})$. For the first step of the iterative minimization problem, the update equation becomes for $k = 1, \ldots, n$:

$$\underline{\mathbf{a}}^{(t+1)}[k]^T = \underline{\mathbf{a}}^{(t)}[k]^T - \alpha_{\underline{\mathbf{a}}}^{(t+1)} \cdot \Big\{\mathbf{V}^{(t)}\{G''(\underline{\mathbf{a}}^{(t)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}) + c \cdot \psi''(\underline{\mathbf{a}}^{(t)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)})\}\mathbf{V}^{(t),T}\Big\}^{-1} \cdot \mathbf{V}^{(t)} \cdot \Big\{G'(\underline{\mathbf{a}}^{(t)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}) - \mathbf{x}[k]^T + c \cdot \psi'(\underline{\mathbf{a}}^{(t)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)})\Big\}.$$

Note that, for $\underline{\boldsymbol{\theta}}[k] = \underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}$, the gradient of the penalty function is given by:

$$\psi'(\underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}) = \frac{\partial \psi(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\boldsymbol{\theta}}[k]} = \left[\frac{\partial \psi(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_1[k]}, \ldots, \frac{\partial \psi(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_d[k]}\right]$$

where, for $i = 1, \ldots, d$ and $k = 1, \ldots, n$,

$$\begin{aligned}\frac{\partial \psi(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_i[k]} &= \frac{\partial}{\partial \underline{\theta}_i[k]} \sum_{r=1}^{d} \exp\big(-\beta_{min}(\underline{\theta}_r[k] - \theta_{min})\big) \\ &\quad + \frac{\partial}{\partial \underline{\theta}_i[k]} \sum_{r=1}^{d} \exp\big(\beta_{max}(\underline{\theta}_r[k] - \theta_{max})\big) \\ &= -\beta_{min} \exp\big(-\beta_{min}(\underline{\theta}_i[k] - \theta_{min})\big) \\ &\quad + \beta_{max} \exp\big(\beta_{max}(\underline{\theta}_i[k] - \theta_{max})\big).\end{aligned}$$

The Hessian matrix is given by:

$$\psi''(\underline{\mathbf{a}}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}) = \begin{bmatrix} \frac{\partial^2 \psi(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_1[k]^2} & \cdots & \frac{\partial^2 \psi(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_d[k] \partial \underline{\theta}_1[k]} \\ \vdots & \vdots & \vdots \\ \frac{\partial^2 \psi(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_1[k] \partial \underline{\theta}_d[k]} & \cdots & \frac{\partial^2 \psi(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_d[k]^2} \end{bmatrix},$$

with

$$\begin{aligned}\frac{\partial^2 \psi(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_r[k] \partial \underline{\theta}_s[k]} &= \beta_{min}^2 \exp\big(-\beta_{min}(\underline{\theta}_r[k] - \theta_{min})\big) \\ &\quad + \beta_{max}^2 \exp\big(\beta_{max}(\underline{\theta}_r[k] - \theta_{max})\big)\end{aligned}$$

for $r = s$, and $\frac{\partial^2 \psi(\underline{\boldsymbol{\theta}}[k])}{\partial \underline{\theta}_r[k] \partial \underline{\theta}_s[k]} = 0$ otherwise.

Similarly, the second step of the iterative minimization yields the following update equation for $j = 1, \ldots, q$:

$$\begin{aligned}\mathbf{v}_j^{(t+1),T} &= \mathbf{v}_j^{(t),T} - \alpha_{\mathbf{v}}^{(t+1)} \\ &\cdot \Bigg(\sum_{k=1}^{n} \underline{a}_j^{(t+1)}[k]^2 \Big\{G''\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) \\ &+ c \cdot \psi''\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big)\Big\}\Bigg)^{-1} \cdot \\ &\Bigg(\sum_{k=1}^{n} \underline{a}_j^{(t+1)}[k]\big\{G'\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) - \mathbf{x}[k]^T \\ &+ c \cdot \psi'\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big)\big\}\Bigg).\end{aligned}$$

Finally, the last step of the minimization problem yields the following update equation:

$$\begin{aligned}\mathbf{b}^{(t+1),T} &= \mathbf{b}^{(t),T} - \alpha_{\mathbf{b}}^{(t+1)} \cdot \\ &\Bigg(\sum_{k=1}^{n} \Big\{G''\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}^{(t)}\big) \\ &+ c \cdot \psi''\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}^{(t)}\big)\Bigg)^{-1} \cdot \\ &\Bigg(\sum_{k=1}^{n} \Big\{G'\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}^{(t)}\big) - \mathbf{x}[k]^T \\ &+ c \cdot \psi'\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}^{(t)}\big)\Big\}\Bigg).\end{aligned}$$

3.3 Uniqueness and identifiability

The matrix $\mathbf{V} \in \mathbb{R}^{q \times d}$ defines the low-dimensional parameter subspace. It can be shown that the matrix $\mathbf{V}$, when $q > 1$, is not unique and that other equivalent representations can be derived by orthogonal transformations of it [22]. In cases where $q > 1$, there are an infinity of choices for $\mathbf{V}$. The constraint $\underline{\boldsymbol{\theta}}[k] = \underline{\mathbf{a}}[k]\mathbf{V} + \mathbf{b}$ for $k = 1, \ldots, n$, is still satisfied if $\underline{\mathbf{a}}[k]$ is replaced by $\underline{\mathbf{a}}[k]\mathbf{M}$ and $\mathbf{V}$ by $\mathbf{M}^T\mathbf{V}$, where $\mathbf{M}$ is any orthogonal matrix of dimension $(q \times q)$.

In order to reduce the identifiability problem of the matrix $\underline{\boldsymbol{\Theta}} = \underline{\mathbf{A}}\mathbf{V} + \mathbf{B}$, an orthonormality constraint is used, i.e., the condition

$$\mathbf{V}\mathbf{V}^T = \mathbf{I}_{q \times q} \tag{26}$$

is enforced. Consider the matrix space $\mathcal{M} = \mathbb{R}^{q \times d}$, then $\mathbf{V} \in \mathcal{M}$. As the iterative minimization process proposed earlier goes on, the successive updates of the matrix $\mathbf{V}$ evolve, giving rise to a curve $\mathbf{V}(t)$ in $\mathcal{M}$, where t describes time. The constraint $\mathbf{V}\mathbf{V}^T = \mathbf{I}_{q \times q}$ corresponds to a hyperplane in $\mathcal{M}$ and an easy way to comply with it would be to impose that the curve $\mathbf{V}(t)$ remains tangential to the $\mathbf{V}\mathbf{V}^T = \mathbf{I}_{q \times q}$ hyperplane. In other words, the progression along the curve $\mathbf{V}(t)$ should remain on the tangent of the $\mathbf{V}\mathbf{V}^T = \mathbf{I}_{q \times q}$ hyperplane.

Considering the notation $\dot{\mathbf{V}} = \mathrm{d}\mathbf{V}/\mathrm{d}t$, the tangent to the $\mathbf{V}\mathbf{V}^T = \mathbf{I}_{q\times q}$ hyperplane can be defined by the equation

$$\frac{\dot{\mathbf{V}}\mathbf{V}^T + \mathbf{V}\dot{\mathbf{V}}^T}{2} = \mathbf{0}, \tag{27}$$

where the denominator is used for later convenience. Let $\mathbf{M} = \dot{\mathbf{V}} \in \mathcal{M}$ and the following operator is defined:

$$\mathcal{A}(\mathbf{M}) \triangleq \frac{\mathbf{M}\mathbf{V}^T + \mathbf{V}\mathbf{M}^T}{2}. \tag{28}$$

It is easily shown that the operator $\mathcal{A} : \mathcal{M} \to \mathcal{S}$, where $\mathcal{S} \subset \mathbb{R}^{q\times q}$, is linear. It is as well onto, i.e., $\mathcal{R}(\mathcal{A}) = \mathcal{S}$ the range of $\mathcal{A}$ or $\mathcal{N}(\mathcal{A}^*) = \{\mathbf{0}\}$ the null space of its adjoint operator, as shown below. Note that $\mathcal{A}(\mathbf{M})^T = \mathcal{A}(\mathbf{M})$, so that $\mathcal{S}$ only contains symmetric matrices.

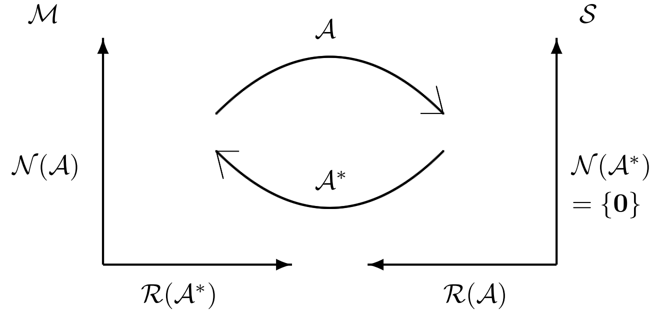
$\mathcal{M}$
$\mathcal{A}$
$\mathcal{S}$
$\mathcal{N}(\mathcal{A})$
$\mathcal{A}^*$
$\mathcal{N}(\mathcal{A}^*)$
$= \{\mathbf{0}\}$
$\mathcal{R}(\mathcal{A}^*)$
$\mathcal{R}(\mathcal{A})$

Fig. 4. The operator $\mathcal{A}$, its range $\mathcal{R}(\mathcal{A})$ and null space $\mathcal{N}(\mathcal{A})$ in relation with its adjoint operator $\mathcal{A}^*$, its range $\mathcal{R}(\mathcal{A}^*)$ and null space $\mathcal{N}(\mathcal{A}^*) = \{\mathbf{0}\}$, with $\mathcal{M} = \mathcal{N}(\mathcal{A}) \cup \mathcal{R}(\mathcal{A}^*)$ and $\mathcal{S} = \mathcal{N}(\mathcal{A}^*) \cup \mathcal{R}(\mathcal{A})$.

Proof: For any $\mathbf{M} \in \mathcal{M}$ and any $\mathbf{W} \in \mathcal{S}$, the adjoint operator $\mathcal{A}^*$ is defined by

$$\langle \mathbf{W}, \mathcal{A}(\mathbf{M})\rangle = \langle \mathcal{A}^*(\mathbf{W}), \mathbf{M}\rangle, \tag{29}$$

where, using the trace operator tr,

$$\begin{aligned}\langle \mathbf{W}, \mathcal{A}(\mathbf{M})\rangle &= \operatorname{tr}\mathbf{W}^T\mathcal{A}(\mathbf{M})\\ &= \frac{1}{2}\operatorname{tr}\big(\mathbf{W}^T\mathbf{M}\mathbf{V}^T + \mathbf{W}^T\mathbf{V}\mathbf{M}^T\big)\\ &= \frac{1}{2}\operatorname{tr}\big(\mathbf{W}^T\mathbf{M}\mathbf{V}^T + \mathbf{M}\mathbf{V}^T\mathbf{W}\big)\\ &= \frac{1}{2}\operatorname{tr}\big(\mathbf{W}^T\mathbf{M}\mathbf{V}^T + \mathbf{W}\mathbf{M}\mathbf{V}^T\big)\end{aligned}$$

using properties of the trace operator,

$$= \operatorname{tr}\left(\frac{\mathbf{W}^T + \mathbf{W}}{2}\right)\mathbf{M}\mathbf{V}^T = \operatorname{tr}\mathbf{W}\mathbf{M}\mathbf{V}^T$$

since $\mathbf{W} \in \mathcal{S}$ is symmetric,

$$\begin{aligned}&= \operatorname{tr}\mathbf{V}^T\mathbf{W}\mathbf{M} = \operatorname{tr}\mathbf{V}^T\mathbf{W}^T\mathbf{M}\\ &= \operatorname{tr}(\mathbf{W}\mathbf{V})^T\mathbf{M} = \langle \mathbf{W}\mathbf{V}, \mathbf{M}\rangle.\end{aligned} \tag{30}$$

Now, combining equations (29) and (30) results in

$$\langle \mathcal{A}^*(\mathbf{W}), \mathbf{M}\rangle = \langle \mathbf{W}\mathbf{V}, \mathbf{M}\rangle.$$

Consequently,

$$\mathcal{A}^*(\mathbf{W}) = \mathbf{W}\mathbf{V}. \tag{31}$$

If $\mathcal{A}^*(\mathbf{W}) = \mathbf{0}$, then $\mathbf{W}\mathbf{V} = \mathbf{0}$, meaning $\mathbf{W} = \mathbf{0}$ since $\mathbf{V}$ cannot be the $\mathbf{0}$ matrix. Therefore, $\mathcal{N}(\mathcal{A}^*) = \{\mathbf{0}\}$ and $\mathcal{A}$ is onto.

Figure 4 shows the relationship between the range and null space of $\mathcal{A}$ and the range and null space of its adjoint operator $\mathcal{A}^*$. □

Imposing that the curve $\mathbf{V}(t)$ remains tangential to the $\mathbf{V}\mathbf{V}^T = \mathbf{I}_{q\times q}$ hyperplane is equivalent to imposing $\mathcal{A}(\dot{\mathbf{V}}) = \mathbf{0}$, i.e., $\dot{\mathbf{V}}(= \mathrm{d}\mathbf{V}/\mathrm{d}t \simeq \delta\mathbf{V})$ or the increment $\delta\mathbf{V}$ in the $\mathbf{V}$ update equation $\big(\mathbf{V}^{(m+1)} = \mathbf{V}^{(m)} - \alpha_{\mathbf{V}}^{(m+1)}\delta\mathbf{V}$, cf. equation (22)$\big)$ needs to be projected onto the null space of $\mathcal{A}$. The projection operator onto the null space of $\mathcal{A}$ is defined as $\mathrm{P}_{\mathcal{N}(\mathcal{A})} = \mathrm{I} - \mathrm{P}_{\mathcal{R}(\mathcal{A}^*)} = \mathrm{I} - \mathcal{A}^+\mathcal{A}$, where the subscript $^+$ denotes a pseudo-inverse. The goal now is to compute $\mathcal{A}^+$. Since $\mathcal{A}$ is onto, $\mathcal{R}(\mathcal{A}) = \mathcal{S}$, for any $\mathbf{M} \in \mathcal{M}$ there exists a matrix $\mathbf{W} \in \mathcal{S}$ such that

$$\mathcal{A}(\mathbf{M}) = \mathbf{W}. \tag{32}$$

Similarly, for any $\mathbf{M} \in \mathcal{M}$ there exists a matrix $\mathbf{\Lambda} \in \mathcal{S}$ such that

$$\mathbf{M} = \mathcal{A}^*(\mathbf{\Lambda}). \tag{33}$$

Then, combining equations (31) and (33) gives

$$\mathbf{M} = \mathcal{A}^*(\mathbf{\Lambda}) = \mathbf{\Lambda}\mathbf{V}. \tag{34}$$

Now, using both equations (32) and (34) gives

$$\begin{aligned}\mathcal{A}(\mathbf{M}) &= \mathcal{A}(\mathbf{\Lambda}\mathbf{V}) = \mathbf{W}\\ &= \frac{\mathbf{\Lambda}\mathbf{V}\mathbf{V}^T + \mathbf{V}\mathbf{V}^T\mathbf{\Lambda}^T}{2} = \frac{\mathbf{\Lambda} + \mathbf{\Lambda}^T}{2} = \mathbf{\Lambda},\end{aligned}$$

since $\mathbf{V}\mathbf{V}^T = \mathbf{I}_{q\times q}$ and $\mathbf{\Lambda} \in \mathcal{S}$ is symmetric. As a result, $\mathbf{\Lambda} = \mathbf{W}$. Consequently, $\mathbf{M} = \mathbf{W}\mathbf{V} = \mathcal{A}^+(\mathbf{W})$. The projection operator onto the range of $\mathcal{A}^*$ is defined by

$$\mathrm{P}_{\mathcal{R}(\mathcal{A}^*)}(\mathbf{M}) = \mathcal{A}^+\mathcal{A}(\mathbf{M}) = \frac{\mathbf{M}\mathbf{V}^T + \mathbf{V}\mathbf{M}^T}{2}\mathbf{V}.$$

It can be shown that $\mathrm{P}_{\mathcal{R}(\mathcal{A}^*)}$ is correctly defined as a projection operator since it is idempotent.

Proof:

$$\begin{aligned}\mathrm{P}^2_{\mathcal{R}(\mathcal{A}^*)}(\mathbf{M}) &= \mathrm{P}_{\mathcal{R}(\mathcal{A}^*)}\left(\frac{\mathbf{M}\mathbf{V}^T + \mathbf{V}\mathbf{M}^T}{2}\mathbf{V}\right)\\ &= \frac{\left(\frac{\mathbf{M}\mathbf{V}^T + \mathbf{V}\mathbf{M}^T}{2}\mathbf{V}\right)\mathbf{V}^T + \mathbf{V}\left(\frac{\mathbf{M}\mathbf{V}^T + \mathbf{V}\mathbf{M}^T}{2}\mathbf{V}\right)^T}{2}\mathbf{V}\\ &= \frac{1}{2}\left\{\left(\frac{\mathbf{M}\mathbf{V}^T + \mathbf{V}\mathbf{M}^T}{2}\right)\mathbf{V} + \left(\frac{\mathbf{M}\mathbf{V}^T + \mathbf{V}\mathbf{M}^T}{2}\right)\mathbf{V}\right\}\\ &= \left(\frac{\mathbf{M}\mathbf{V}^T + \mathbf{V}\mathbf{M}^T}{2}\right)\mathbf{V} = \mathrm{P}_{\mathcal{R}(\mathcal{A}^*)}(\mathbf{M}),\end{aligned}$$

since $\mathbf{V}\mathbf{V}^T = \mathbf{I}_{q\times q}$ and $\mathcal{A}(\mathbf{M})$ is symmetric. □

Then, the projection operator onto the null space of $\mathcal{A}$ is defined by

$$\mathrm{P}_{\mathcal{N}(\mathcal{A})}(\mathbf{M}) = \big(\mathrm{I} - \mathrm{P}_{\mathcal{R}(\mathcal{A}^*)}\big)(\mathbf{M}) = \mathbf{M} - \left(\frac{\mathbf{M}\mathbf{V}^T + \mathbf{V}\mathbf{M}^T}{2}\right)\mathbf{V}.$$

It is indeed a projector onto the null space of $\mathcal{A}$ since

$$\begin{aligned}\mathcal{A}\left(\mathrm{P}_{\mathcal{N}(\mathcal{A})}(\mathbf{M})\right) &= \frac{1}{2}\left(\mathbf{M}-\left(\frac{\mathbf{M}\mathbf{V}^T+\mathbf{V}\mathbf{M}^T}{2}\right)\mathbf{V}\right)\mathbf{V}^T \\ &\quad + \frac{1}{2}\mathbf{V}\left(\mathbf{M}-\left(\frac{\mathbf{M}\mathbf{V}^T+\mathbf{V}\mathbf{M}^T}{2}\right)\mathbf{V}\right)^T \\ &= \frac{1}{2}\left(\frac{\mathbf{M}\mathbf{V}^T-\mathbf{V}\mathbf{M}^T}{2}\right)-\frac{1}{2}\left(\frac{\mathbf{M}\mathbf{V}^T-\mathbf{V}\mathbf{M}^T}{2}\right)=\mathbf{0}.\end{aligned}$$

Finally, the update equation for $j=1,\dots,q$ becomes:

$$\begin{aligned}\mathbf{v}_j^{(t+1),T} &= \mathbf{v}_j^{(t),T} - \alpha_{\mathbf{v}}^{(t+1)} \\ &\cdot \mathrm{P}_{\mathcal{N}(\mathcal{A})}\left(\sum_{k=1}^{n}\underline{a}_j^{(t+1)}[k]^2 G''\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)}+\mathbf{b}^{(t)}\big)\right)^{-1} \\ &\cdot \left(\sum_{k=1}^{n}\underline{a}_j^{(t+1)}[k]\big\{G'\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)}+\mathbf{b}^{(t)}\big)-\mathbf{x}[k]^T\big\}\right).\end{aligned}$$

3.4 Mixed data types

The above approach was proposed assuming that the data attributes have the same distribution. It can be extended to the problem for which different types of distributions can be used for different attributes. This situation is referred to as the mixed data-type case and is exposed below. Only two types of exponential family distributions are considered here (for example, the Bernoulli distribution and the Gaussian distribution). Of course, this approach generalizes to any number of exponential family distributions.

For simplicity of presentation, we consider that the f first attributes are distributed according to the exponential family distribution $p^{(\mathbf{1})}$ and the $(d-f)$ last attributes are distributed according to the exponential family distribution $p^{(\mathbf{2})}$. Following the previously stated example, the bold superscript $^{(\mathbf{1})}$ would correspond to Bernoulli distributed attributes and $^{(\mathbf{2})}$ to Gaussian distributed attributes. Then,

$$\mathbf{X}=\left(\begin{array}{ccc|ccc} x_1[1] & \dots & x_f[1] & x_{f+1}[1] & \dots & x_d[1] \\ x_1[2] & \dots & x_f[2] & x_{f+1}[2] & \dots & x_d[2] \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_1[n] & \dots & x_f[n] & x_{f+1}[n] & \dots & x_d[n] \end{array}\right)=\left(\mathbf{X}^{(\mathbf{1})}\,\middle|\,\mathbf{X}^{(\mathbf{2})}\right).$$

The loss function is expressed as follows:

$$L(\underline{\mathbf{A}},\mathbf{V},\mathbf{b})=-\log p(\mathbf{X}|\underline{\mathbf{A}},\mathbf{V},\mathbf{b})=-\sum_{k=1}^{n}\log p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[k]\big),$$

using the *iid statistical samples assumption*. Then, using the *latent variable assumption*,

$$p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[k]\big)=p^{(\mathbf{1})}\big(\mathbf{x}^{(\mathbf{1})}[k]|\underline{\boldsymbol{\theta}}^{(\mathbf{1})}[k]\big)\cdot p^{(\mathbf{2})}\big(\mathbf{x}^{(\mathbf{2})}[k]|\underline{\boldsymbol{\theta}}^{(\mathbf{2})}[k]\big), \tag{35}$$

where

$$\underline{\boldsymbol{\Theta}}=\left(\begin{array}{ccc|ccc} \underline{\theta}_1[1] & \dots & \underline{\theta}_f[1] & \underline{\theta}_{f+1}[1] & \dots & \underline{\theta}_d[1] \\ \underline{\theta}_1[2] & \dots & \underline{\theta}_f[2] & \underline{\theta}_{f+1}[2] & \dots & \underline{\theta}_d[2] \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \underline{\theta}_1[n] & \dots & \underline{\theta}_f[n] & \underline{\theta}_{f+1}[n] & \dots & \underline{\theta}_d[n] \end{array}\right)=\left(\underline{\boldsymbol{\Theta}}^{(\mathbf{1})}\,\middle|\,\underline{\boldsymbol{\Theta}}^{(\mathbf{2})}\right).$$

The matrix of parameters $\underline{\boldsymbol{\Theta}}=\underline{\mathbf{A}}\mathbf{V}+\mathbf{B}$ results in the following decompositions:

$$\mathbf{V}=\left(\begin{array}{ccc|ccc} v_{11} & \dots & v_{1f} & v_{1(f+1)} & \dots & v_{1d} \\ v_{21} & \dots & v_{2f} & v_{2(f+1)} & \dots & v_{2d} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ v_{q1} & \dots & v_{qf} & v_{q(f+1)} & \dots & v_{qd} \end{array}\right)=\left(\mathbf{V}^{(\mathbf{1})}\,\middle|\,\mathbf{V}^{(\mathbf{2})}\right),$$

and $\mathbf{B}=\left(\mathbf{B}^{(\mathbf{1})}\,\middle|\,\mathbf{B}^{(\mathbf{2})}\right)$, $\mathbf{B}^{(\mathbf{1})}=[\mathbf{b}^{(\mathbf{1})},\dots,\mathbf{b}^{(\mathbf{1})}]^T$ with $\mathbf{b}^{(\mathbf{1})}=[b_1,\dots,b_f]$, and $\mathbf{B}^{(\mathbf{2})}=[\mathbf{b}^{(\mathbf{2})},\dots,\mathbf{b}^{(\mathbf{2})}]^T$ with $\mathbf{b}^{(\mathbf{2})}=[b_{f+1},\dots,b_d]$. Then,

$$\underline{\boldsymbol{\Theta}}=\underline{\mathbf{A}}\mathbf{V}+\mathbf{B}=\left(\underline{\mathbf{A}}\mathbf{V}^{(\mathbf{1})}+\mathbf{B}^{(\mathbf{1})}\,\middle|\,\underline{\mathbf{A}}\mathbf{V}^{(\mathbf{2})}+\mathbf{B}^{(\mathbf{2})}\right).$$

Now, notice that

$$\underline{\mathbf{A}}\mathbf{V}^{(\mathbf{1})}+\mathbf{B}^{(\mathbf{1})}=\underline{\begin{pmatrix} \underline{a}_1[1] & \dots & \underline{a}_q[1] \\ \underline{a}_1[2] & \dots & \underline{a}_q[2] \\ \vdots & \ddots & \vdots \\ \underline{a}_1[n] & \dots & \underline{a}_q[n] \end{pmatrix}\begin{pmatrix} v_{11} & \dots & v_{1f} \\ v_{21} & \dots & v_{2f} \\ \vdots & \ddots & \vdots \\ v_{q1} & \dots & v_{qf} \end{pmatrix}}.$$

The underlined term is a $(n\times f)$ matrix whose elements are $\sum_{j=1}^{q}\underline{a}_j[k]v_{ji}$ for all $k=1,\dots,n$ and $i=1,\dots,f$. Consequently, the elements of the $(n\times f)$ matrix $\underline{\mathbf{A}}\mathbf{V}^{(\mathbf{1})}+\mathbf{B}^{(\mathbf{1})}$ are of the form $\sum_{j=1}^{q}\underline{a}_j[k]v_{ji}+b_i$ for all $k=1,\dots,n$ and $i=1,\dots,f$. The matrix $\underline{\boldsymbol{\Theta}}^{(\mathbf{1})}$ is also $(n\times f)$ and its elements take the following form: $\underline{\theta}_i[k]=\sum_{j=1}^{q}\underline{a}_j[k]v_{ji}+b_i$ for all $k=1,\dots,n$ and $i=1,\dots,f$. Besides, knowing that $\underline{\theta}_i[k]=\sum_{j=1}^{q}\underline{a}_j[k]v_{ji}+b_i$ for all $k=1,\dots,n$ and $i=1,\dots,d$, it becomes clear that $\underline{\boldsymbol{\Theta}}^{(\mathbf{1})}$ equals $\underline{\mathbf{A}}\mathbf{V}^{(\mathbf{1})}+\mathbf{B}^{(\mathbf{1})}$, and $\underline{\boldsymbol{\Theta}}^{(\mathbf{2})}$ is $\underline{\mathbf{A}}\mathbf{V}^{(\mathbf{2})}+\mathbf{B}^{(\mathbf{2})}$. Note that, even though we are able to separate the matrix $\underline{\boldsymbol{\Theta}}$ into two blocks, the matrix $\underline{\mathbf{A}}$ is common to both $\underline{\boldsymbol{\Theta}}^{(\mathbf{1})}$ and $\underline{\boldsymbol{\Theta}}^{(\mathbf{2})}$. Therefore, the loss function takes the following form:

$$\begin{aligned}L(\underline{\mathbf{A}},\mathbf{V},\mathbf{b}) &= -\sum_{k=1}^{n}\log p^{(\mathbf{1})}\big(\mathbf{x}^{(\mathbf{1})}[k]|\underline{\mathbf{a}}[k],\mathbf{V}^{(\mathbf{1})},\mathbf{b}^{(\mathbf{1})}\big) \\ &\quad -\sum_{k=1}^{n}\log p^{(\mathbf{2})}\big(\mathbf{x}^{(\mathbf{2})}[k]|\underline{\mathbf{a}}[k],\mathbf{V}^{(\mathbf{2})},\mathbf{b}^{(\mathbf{2})}\big).\end{aligned}$$

Since the linear combination of convex functions with nonnegative coefficients is always convex [47], the loss function remains convex in either of its arguments with the others fixed. Therefore, the iterative minimization technique proposed for the single exponential family can be applied in the mixture of exponential families case.

The first step in the Newton-Raphson minimization technique, given a fixed matrix $\mathbf{V}$ and fixed vector $\mathbf{b}$, is to obtain the matrix $\underline{\mathbf{A}}$, or the set of vectors $\underline{\mathbf{a}}[k]$ for $k=1,\dots,n$, that minimizes the loss function. The second step, given a fixed matrix $\underline{\mathbf{A}}$ and fixed vector $\mathbf{b}$, is to obtain the matrix $\mathbf{V}$ that minimizes the loss function. The last step, given a fixed matrix $\underline{\mathbf{A}}$ and a fixed matrix $\mathbf{V}$, is to obtain the vector $\mathbf{b}$. The updates are derived in a way similar to the one used in Section 3.2. As previously, the superscript $^{(t)}$ means an estimate obtained at the end of the t^{th} iteration of the iterative minimization process. Note that, in order to avoid confusion, the step superscript is not bold whereas the mixture superscripts ${}^{(\mathbf{1})}$ and ${}^{(\mathbf{2})}$ are.

$$\begin{aligned} l\big(\underline{\mathbf{a}}[k]\big) = G^{(\mathbf{1})}\big(&\underline{\mathbf{a}}[k]\mathbf{V}^{(\mathbf{1})(t)} + \mathbf{b}^{(\mathbf{1})(t)}\big) \\ &- \big(\underline{\mathbf{a}}[k]\mathbf{V}^{(\mathbf{1})(t)} + \mathbf{b}^{(\mathbf{1})(t)}\big)\mathbf{x}^{(\mathbf{1})}[k]^T \\ &+ G^{(\mathbf{2})}\big(\underline{\mathbf{a}}[k]\mathbf{V}^{(\mathbf{2})(t)} + \mathbf{b}^{(\mathbf{2})(t)}\big) \\ &- \big(\underline{\mathbf{a}}[k]\mathbf{V}^{(\mathbf{2})(t)} + \mathbf{b}^{(\mathbf{2})(t)}\big)\mathbf{x}^{(\mathbf{2})}[k]^T. \end{aligned}$$

The update equation for the set of vectors $\underline{\mathbf{a}}[k]$ for $k = 1, \cdots, n$ is:

$$\begin{aligned} &\underline{\mathbf{a}}^{(t+1)}[k]^T = \underline{\mathbf{a}}^{(t)}[k]^T \\ &\quad - \alpha_{\underline{\mathbf{a}}}^{(t+1)}\Big\{\mathbf{V}^{(\mathbf{1})(t)}G^{(\mathbf{1})''}\big(\underline{\mathbf{a}}^{(t)}[k]\mathbf{V}^{(\mathbf{1})(t)} + \mathbf{b}^{(\mathbf{1})(t)}\big)\mathbf{V}^{(\mathbf{1})(t),T} \\ &\quad + \mathbf{V}^{(\mathbf{2})(t)}G^{(\mathbf{2})''}\big(\underline{\mathbf{a}}^{(t)}[k]\mathbf{V}^{(\mathbf{2})(t)} + \mathbf{b}^{(\mathbf{2})(t)}\big)\mathbf{V}^{(\mathbf{2})(t),T}\Big\}^{-1} \\ &\quad \cdot \Big\{\mathbf{V}^{(\mathbf{1})(t)}\Big(G^{(\mathbf{1})'}\big(\underline{\mathbf{a}}^{(t)}[k]\mathbf{V}^{(\mathbf{1})(t)} + \mathbf{b}^{(\mathbf{1})(t)}\big) - \mathbf{x}[k]^{(\mathbf{1}),T}\Big) \\ &\quad + \mathbf{V}^{(\mathbf{2})(t)}\Big(G^{(\mathbf{2})'}\big(\underline{\mathbf{a}}^{(t)}[k]\mathbf{V}^{(\mathbf{2})(t)} + \mathbf{b}^{(\mathbf{2})(t)}\big) - \mathbf{x}[k]^{(\mathbf{2}),T}\Big)\Big\}. \end{aligned} \tag{36}$$

For the second step, the two sets of row vectors $\{\mathbf{v}_j^{(\mathbf{1})}\}_{j=1}^q$ and $\{\mathbf{v}_j^{(\mathbf{2})}\}_{j=1}^q$ are updated separately. For the sake of simplicity, the update equation is written for the set $\{\mathbf{v}_j\}_{j=1}^q$ indistinct of the mixture superscript and is given as follows for $j = 1, \ldots, q$:

$$\begin{aligned} &\mathbf{v}_j^{(t+1),T} = \mathbf{v}_j^{(t),T} \\ &\quad - \alpha_{\mathbf{v}}^{(t+1)}\left(\sum_{k=1}^n \underline{a}_j^{(t+1)}[k]^2 G''\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big)\right)^{-1} \\ &\quad \cdot \left(\sum_{k=1}^n \underline{a}_j^{(t+1)}[k]\big\{G'\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big) - \mathbf{x}[k]^T\big\}\right). \end{aligned} \tag{37}$$

For the last step, as for $\{\mathbf{v}_j^{(\mathbf{1})}\}_{j=1}^q$ and $\{\mathbf{v}_j^{(\mathbf{2})}\}_{j=1}^q$, the derivations are made for the vector $\mathbf{b}$ indistinct of the mixture superscript:

$$\begin{aligned} \mathbf{b}^{(t+1),T} = \mathbf{b}^{(t),T} - \alpha_{\mathbf{b}}^{(t+1)}&\left(\sum_{k=1}^n G''\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}\big)\right)^{-1} \\ &\cdot \left(\sum_{k=1}^n \big\{G'\big(\underline{\mathbf{a}}^{(t+1)}[k]\mathbf{V}^{(t+1)} + \mathbf{b}\big) - \mathbf{x}[k]^T\big\}\right). \end{aligned} \tag{38}$$

Equations (37) can be used for $\{\mathbf{v}_j^{(\mathbf{1})}\}_{j=1}^q$ and $\{\mathbf{v}_j^{(\mathbf{2})}\}_{j=1}^q$, and equation (38) can be used for $\mathbf{b}^{(\mathbf{1})}$ and $\mathbf{b}^{(\mathbf{2})}$ by changing $\mathbf{v}_j$ to $\mathbf{v}_j^{(\mathbf{1})}$, respectively to $\mathbf{v}_j^{(\mathbf{2})}$, $\mathbf{V}$ to $\mathbf{V}^{(\mathbf{1})}$, respectively to $\mathbf{V}^{(\mathbf{2})}$, $\mathbf{b}$ to $\mathbf{b}^{(\mathbf{1})}$, respectively to $\mathbf{b}^{(\mathbf{2})}$, $G(\cdot), G'(\cdot)$, and $G''(\cdot)$ to $G^{(\mathbf{1})}(\cdot), G^{(\mathbf{1})'}(\cdot)$, and $G^{(\mathbf{1})''}(\cdot)$, respectively to $G^{(\mathbf{2})}(\cdot), G^{(\mathbf{2})'}(\cdot)$, and $G^{(\mathbf{2})''}(\cdot)$.

Table 1 summarizes exponential PCA algorithm.

```
Algorithm: Exponential PCA [10]

Input: a set of observations $\{\mathbf{x}[k]\}_{k=1}^n \subseteq \mathbb{R}^d$, two exponential family distribution $p^{(\mathbf{1})}, p^{(\mathbf{2})}$ defined by their cumulant generating functions $G^{(\mathbf{1})}, G^{(\mathbf{2})}$, a number of atoms $n$, $q \ll d$ the dimension of the latent variable lower dimensional subspace.

Output: the ML estimator $\{\widehat{\underline{\boldsymbol{\theta}}}[k]\}_{k=1}^n$ that minimizes the loss function $L(\underline{\mathbf{A}}, \mathbf{V}, \mathbf{b})$ in (17): $\widehat{\underline{\boldsymbol{\theta}}}[k] = \widehat{\underline{\mathbf{a}}}[k]\widehat{\mathbf{V}} + \widehat{\mathbf{b}}$ for all $k$, $\{\widehat{\underline{\mathbf{a}}}[k]\}_{k=1}^n \in \mathbb{R}^q$, $\widehat{\mathbf{V}} \in \mathbb{R}^{q\times d}$ and $\widehat{\mathbf{b}} \in \mathbb{R}^d$.

Method:
  Initialize $\mathbf{V}$, $\mathbf{b}$ and $\{\underline{\mathbf{a}}[k]\}_{k=1}^n$; $\underline{\boldsymbol{\theta}}[k] = \underline{\mathbf{a}}[k]\mathbf{V} + \mathbf{b} \in \boldsymbol{\Theta}$ for all $k$; $p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[k]\big)$ as defined in (35) for all $k$;
  repeat
    {The Newton-Raphson iterative algorithm}
    for $k = 1$ to $n$ do
      $\underline{\mathbf{a}}[l] \longleftarrow$ penalty-modified update equation (36)
    end for
    for $j = 1$ to $q$ do
      $\mathbf{v}_j \longleftarrow$ penalty-modified update equation (37)
    end for
    $\mathbf{b} \longleftarrow$ penalty-modified update equation (38)
  until convergence;
return $\{\widehat{\underline{\boldsymbol{\theta}}}[k] = \widehat{\underline{\mathbf{a}}}[k]\widehat{\mathbf{V}} + \widehat{\mathbf{b}}\}_{k=1}^n$.
```

TABLE 1
Exponential PCA algorithm.

4 Two other GLS special cases and extensions to mixed data-type cases

In addition to the exponential family Principal Component Analysis (exponential PCA) technique, the two other GLS special cases considered here are the Semi-Parametric exponential family Principal Component Analysis (SP-PCA) and the Bregman soft clustering techniques. They all utilize Bregman distances and can all be explained within a single hierarchical Bayes graphical model framework shown in Figure 1. They are not separate unrelated algorithms but different manifestations of model assumptions and parameter choices taken within a common framework. Because of this insight, these algorithms are readily extended to deal with the important mixed data-type case.

Figure 5 considers the number of atoms as a common characteristic for comparison purposes. The exponential PCA technique corresponds to a classical approach to the GLS estimation problem. The classical approach can be seen as an extreme case of the Bayesian approach for which the probability density function $\pi(\boldsymbol{\theta})$ is a delta function (one per data point) and the total number of distinct natural parameter values m equals the number of data points n, i.e., $m = n$. While the $m < n$ parameters of the Bayesian approach consistent with SP-PCA and the Bregman soft clustering techniques are shared by all the data points, the classical approach assigns one parameter point to each data point (hence $m = n$). The Bregman soft clustering approach considers an even smaller number of natural parameters or atoms than SP-PCA. Since its primary goal is clustering, the atoms play the role of cluster centers in parameter space and their total number is generally small. Furthermore, both exponential PCA and SP-PCA impose a low-dimensional (unknown) latent variable subspace in their structure. However, Bregman soft clustering does not impose this lower dimensional constraint and hence can be seen as a degenerate case.

It becomes clear while looking at Table 1, Table 2 and Table 3 that both SP-PCA and Bregman soft clustering utilize the EM algorithm for estimation purposes whereas exponential PCA does not. Indeed, because exponential PCA assumes a classical approach, no point-mass probabilities need to be estimated.

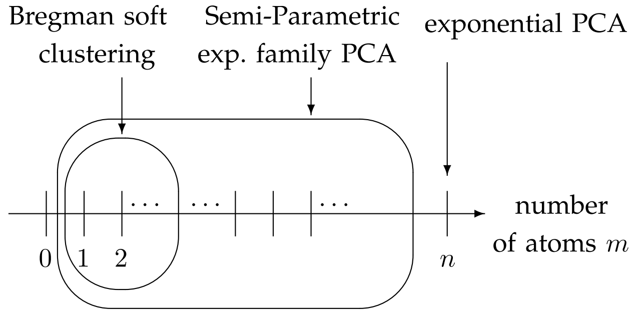

Fig. 5. General point of view on SP-PCA, exponential PCA and Bregman soft clustering based on the number of NPML atoms.

4.1 Semi-parametric exponential family PCA

The Semi-Parametric exponential family Principal Component Analysis (SP-PCA) approach presented in [15] attacks the Semiparametric Maximum Likelihood mixture density Estimation (SMLE) problem exposed in Section 2 by using the Expectation-Maximization (EM) algorithm [41]. We directly present an SP-PCA modified approach for mixed data types and use the mixed data-type notations exposed in the previous section for exponential PCA. Using equation (10), the log-likelihood function is

$$L(\mathcal{Q}) = \log \prod_{k=1}^{n} \sum_{l=1}^{m} p\big(\mathbf{x}[k] | \underline{\mathbf{a}}[l]\mathbf{V} + \mathbf{b}\big)\pi_l, \tag{39}$$

with $\mathcal{Q} = \big\{\underline{\boldsymbol{\theta}}[l] = \underline{\mathbf{a}}[l]\mathbf{V} + \mathbf{b}, \pi_l\big\}_{l=1}^{m}$ the mixing distribution. The EM approach introduces a *missing* (unobserved) variable $\mathbf{z}_k = [z_{k1}, \ldots, z_{km}]$, for $k = 1, \ldots, n$. This variable is an m-dimensional binary vector whose lth component equals 1 if the observed variable $\mathbf{x}[k]$ was drawn from the lth mixture component and 0 otherwise; its value is estimated during the E-step. Using this information, a *complete* log-likelihood function is defined as follows:

$$L^{(c)}\big(\mathcal{Q}, \{\mathbf{z}_k\}_{k=1}^{n}\big) = \log \prod_{k=1}^{n} \prod_{l=1}^{m} p\big(\mathbf{x}[k] | \underline{\boldsymbol{\theta}}[l]\big)^{z_{kl}} \pi_l^{z_{kl}}, \tag{40}$$

with $\underline{\boldsymbol{\theta}}[l] = \underline{\mathbf{a}}[l]\mathbf{V} + \mathbf{b}$. Because z_{kl} equals 1 exactly for one l if k is fixed, reflecting the assumption that each $\mathbf{x}[k]$ is drawn from exactly one mixture component, the inner sum in equation (39) has in fact for each k exactly one non-zero term. In equation (40) it is exactly that non-zero term which is present in the product, all others have an exponent of $z_{kl} = 0$, and hence do not contribute to the product. The maximization of the complete log-likelihood function (the M-step) yields parameters $\underline{\mathbf{A}}$, $\mathbf{V}$ and $\mathbf{b}$ estimates. Then,

$$L^{(c)}\big(\mathcal{Q}, \{\mathbf{z}_k\}_{k=1}^{n}\big) = \sum_{k=1}^{n} \sum_{l=1}^{m} z_{kl} \log \pi_l$$
$$+ \sum_{k=1}^{n} \sum_{l=1}^{m} z_{kl} \log p^{(\mathbf{1})}\big(\mathbf{x}^{(\mathbf{1})}[k] | \underline{\boldsymbol{\theta}}^{(\mathbf{1})}[l]\big) \cdot p^{(\mathbf{2})}\big(\mathbf{x}^{(\mathbf{2})}[k] | \underline{\boldsymbol{\theta}}^{(\mathbf{2})}[l]\big). \tag{41}$$

The E-step yields for $k = 1, \ldots, n$ and $l = 1, \ldots, m$:

$$\widehat{z}_{kl} = \mathrm{E}\left\{z_{kl} | \mathbf{x}[k], \pi_1, \ldots, \pi_m\right\}$$
$$= \frac{p^{(\mathbf{1})}\big(\mathbf{x}^{(\mathbf{1})}[k] | \underline{\boldsymbol{\theta}}^{(\mathbf{1})}[l]\big) \cdot p^{(\mathbf{2})}\big(\mathbf{x}^{(\mathbf{2})}[k] | \underline{\boldsymbol{\theta}}^{(\mathbf{2})}[l]\big)\big)\pi_l}{\sum_{r=1}^{m} p^{(\mathbf{1})}\big(\mathbf{x}^{(\mathbf{1})}[k] | \underline{\boldsymbol{\theta}}^{(\mathbf{1})}[r]\big) \cdot p^{(\mathbf{2})}\big(\mathbf{x}^{(\mathbf{2})}[k] | \underline{\boldsymbol{\theta}}^{(\mathbf{2})}[r]\big)\pi_r}.$$

For all l and all k, each data point $\mathbf{x}[k]$ has an estimated probability $\widehat{z}_{kl}$ of belonging to the lth mixture component.

The M-step first yields the estimates for the point-mass probabilities:

$$\pi_l^{(new)} = \frac{\sum_{k=1}^{n} \widehat{z}_{kl}}{n},$$

corresponding to the number of samples $\mathbf{x}[k]$ drawn from the lth mixture, divided by the number of samples overall. The second part of the M-step, i.e., the estimation of the parameters $\mathbf{V}$, $\mathbf{b}$, and the latent variables $\underline{\mathbf{A}} = \big[\underline{\mathbf{a}}[1]^T, \ldots, \underline{\mathbf{a}}[m]^T\big]^T \in \mathbb{R}^{m,q}$, is affected by the mixed data type assumption. It consists of maximizing the complete log-likelihood function (41) with respect to these parameters:

$$\arg\max_{\underline{\mathbf{A}},\mathbf{V},\mathbf{b}} L^{(c)}\big(\{\underline{\boldsymbol{\theta}}[l], \pi_l^{(new)}\}_{l=1}^{m}, \{\widehat{\mathbf{z}}_k\}_{k=1}^{n}\big)$$
$$= \arg\max_{\underline{\mathbf{A}},\mathbf{V},\mathbf{b}}$$
$$\sum_{k=1}^{n} \sum_{l=1}^{m} \widehat{z}_{kl} \Big\{G^{(\mathbf{1})}\big(\underline{\mathbf{a}}[l]\mathbf{V}^{(\mathbf{1})} + \mathbf{b}^{(\mathbf{1})}\big) - \big(\underline{\mathbf{a}}[l]\mathbf{V}^{(\mathbf{1})} + \mathbf{b}^{(\mathbf{1})}\big)\mathbf{x}[k]^T\Big\}$$
$$+ \sum_{k=1}^{n} \sum_{l=1}^{m} \widehat{z}_{kl} \Big\{G^{(\mathbf{2})}\big(\underline{\mathbf{a}}[l]\mathbf{V}^{(\mathbf{2})} + \mathbf{b}^{(\mathbf{2})}\big) - \big(\underline{\mathbf{a}}[l]\mathbf{V}^{(\mathbf{2})} + \mathbf{b}^{(\mathbf{2})}\big)\mathbf{x}[k]^T\Big\}.$$

We set, for $l = 1, \ldots, m$, $\widetilde{\mathbf{x}}[l] = \sum_{k=1}^{n} \widehat{z}_{kl}\mathbf{x}[k] / \sum_{k=1}^{n} \widehat{z}_{kl}$, the lth mixture component center. It can be shown, using exponential family properties, that the loss function is:

$$L(\underline{\mathbf{A}}, \mathbf{V}, \mathbf{b})$$
$$= \sum_{l=1}^{m} \pi_l^{(new)} \Big\{G^{(\mathbf{1})}\big(\underline{\mathbf{a}}[l]\mathbf{V}^{(\mathbf{1})} + \mathbf{b}^{(\mathbf{1})}\big) - \big(\underline{\mathbf{a}}[l]\mathbf{V}^{(\mathbf{1})} + \mathbf{b}^{(\mathbf{1})}\big)\widetilde{\mathbf{x}}[l]^T\Big\}$$
$$+ \sum_{l=1}^{m} \pi_l^{(new)} \Big\{G^{(\mathbf{2})}\big(\underline{\mathbf{a}}[l]\mathbf{V}^{(\mathbf{2})} + \mathbf{b}^{(\mathbf{2})}\big) - \big(\underline{\mathbf{a}}[l]\mathbf{V}^{(\mathbf{2})} + \mathbf{b}^{(\mathbf{2})}\big)\widetilde{\mathbf{x}}[l]^T\Big\}, \tag{42}$$

since $(1/n)\sum_{k=1}^{n} \widehat{z}_{kl} = \pi_l^{(new)}$. Note that these coefficients $\pi_l^{(new)}$, $l = 1, \ldots, m$, are not present in the algorithm proposed in [15].

The Newton-Raphson method is used for the iterative minimization of the loss function (42) and the resulting update equations are as follows. First at iteration $(t+1)$, for $l = 1, \dots, m$,

$$\begin{aligned}\underline{\mathbf{a}}^{(t+1)}[l]^T &= \underline{\mathbf{a}}^{(t)}[l]^T - \alpha_{\underline{\mathbf{a}}}^{(t+1)} \\ &\cdot \Big\{\mathbf{V}^{(\mathbf{1})(t)} G^{(\mathbf{1})''}\big(\underline{\mathbf{a}}^{(t)}[l]\mathbf{V}^{(\mathbf{1})(t)}+\mathbf{b}^{(\mathbf{1})(t)}\big)\mathbf{V}^{(\mathbf{1})(t),T} \\ &+ \mathbf{V}^{(\mathbf{2})(t)} G^{(\mathbf{2})''}\big(\underline{\mathbf{a}}^{(t)}[l]\mathbf{V}^{(\mathbf{2})(t)}+\mathbf{b}^{(\mathbf{2})(t)}\big)\mathbf{V}^{(\mathbf{2})(t),T}\Big\}^{-1} \\ &\cdot \Big\{\mathbf{V}^{(\mathbf{1})(t)}\big(G^{(\mathbf{1})'}\big(\underline{\mathbf{a}}^{(t)}[l]\mathbf{V}^{(\mathbf{1})(t)}+\mathbf{b}^{(\mathbf{1})(t)}\big) - \widetilde{\mathbf{x}}[l]^T\big) \\ &+ \mathbf{V}^{(\mathbf{2})(t)}\big(G^{(\mathbf{2})'}\big(\underline{\mathbf{a}}^{(t)}[l]\mathbf{V}^{(\mathbf{2})(t)}+\mathbf{b}^{(\mathbf{2})(t)}\big) - \widetilde{\mathbf{x}}[l]^T\big)\Big\}.\end{aligned} \tag{43}$$

For the second step, the two sets of row vectors $\{\mathbf{v}_j^{(\mathbf{1})}\}_{j=1}^q$ and $\{\mathbf{v}_j^{(\mathbf{2})}\}_{j=1}^q$ are updated separately. For $j = 1, \dots, q$:

$$\begin{aligned}\mathbf{v}_j^{(t+1),T} &= \mathbf{v}_j^{(t),T} \\ &- \alpha_{\mathbf{v}}^{(t+1)}\left(\sum_{l=1}^{m} \pi_l^{(new)} \underline{a}_j^{(t+1)}[l]^2 G''\big(\underline{\mathbf{a}}^{(t+1)}[l]\mathbf{V}^{(t)}+\mathbf{b}^{(t)}\big)\right)^{-1} \\ &\cdot\left(\sum_{l=1}^{m} \pi_l^{(new)} \underline{a}_j^{(t+1)}[l]\big\{G'\big(\underline{\mathbf{a}}^{(t+1)}[l]\mathbf{V}^{(t)} + \mathbf{b}^{(t)}\big)-\widetilde{\mathbf{x}}[l]^T\big\}\right).\end{aligned} \tag{44}$$

And finally for the last step, the update equation is:

$$\begin{aligned}\mathbf{b}^{(t+1),T} &= \mathbf{b}^{(t),T} \\ &= \mathbf{b}^{(t),T} - \alpha_{\mathbf{b}}^{(t+1)}\left(\sum_{l=1}^{m} \pi_l^{(new)} G''\big(\underline{\mathbf{a}}^{(t+1)}[l]\mathbf{V}^{(t+1)} + \mathbf{b}\big)\right)^{-1} \\ &\cdot\left(\sum_{l=1}^{m} \pi_l^{(new)}\big\{G'\big(\underline{\mathbf{a}}^{(t+1)}[l]\mathbf{V}^{(t+1)} + \mathbf{b}\big) - \widetilde{\mathbf{x}}[l]^T\big\}\right).\end{aligned} \tag{45}$$

Equations (44) can be used for $\{\mathbf{v}_j^{(\mathbf{1})}\}_{j=1}^q$ and $\{\mathbf{v}_j^{(\mathbf{2})}\}_{j=1}^q$, and equation (45) can be used for $\mathbf{b}^{(\mathbf{1})}$ and $\mathbf{b}^{(\mathbf{2})}$ by changing $\mathbf{v}_j$ to $\mathbf{v}_j^{(\mathbf{1})}$, respectively to $\mathbf{v}_j^{(\mathbf{2})}$, $\mathbf{V}$ to $\mathbf{V}^{(\mathbf{1})}$, respectively to $\mathbf{V}^{(\mathbf{2})}$, $\mathbf{b}$ to $\mathbf{b}^{(\mathbf{1})}$, respectively to $\mathbf{b}^{(\mathbf{2})}$, $G(\cdot), G'(\cdot)$, and $G''(\cdot)$ to $G^{(\mathbf{1})}(\cdot), G^{(\mathbf{1})'}(\cdot)$, and $G^{(\mathbf{1})''}(\cdot)$, respectively to $G^{(\mathbf{2})}(\cdot), G^{(\mathbf{2})'}(\cdot)$, and $G^{(\mathbf{2})''}(\cdot)$.

Table 2 summarizes the SP-PCA algorithm.

Algorithm: Semi-Parametric exp. family PCA [15]

Input: a set of observations $\{\mathbf{x}[k]\}_{k=1}^n \subseteq \mathbb{R}^d$, two exponential family distribution $p^{(\mathbf{1})}, p^{(\mathbf{2})}$ defined by their cumulant generating functions $G^{(\mathbf{1})}, G^{(\mathbf{2})}$, a number of atoms m, $q \ll d$ the dimension of the latent variable lower dimensional subspace.

Output: the NPML estimator that maximizes the complete log-likelihood function $L^{(c)}\big(\mathcal{Q}, \{\mathbf{z}_k\}_{k=1}^n\big)$: $\widehat{\mathcal{Q}} = \{\widehat{\underline{\boldsymbol{\theta}}}[l], \widehat{\pi}_l\}_{l=1}^m$ with $\widehat{\underline{\boldsymbol{\theta}}}[l] = \widehat{\underline{\mathbf{a}}}[l]\widehat{\mathbf{V}}+\widehat{\mathbf{b}}$ for all l, $\{\widehat{\underline{\mathbf{a}}}[l]\}_{l=1}^m \in \mathbb{R}^q$, $\widehat{\mathbf{V}} \in \mathbb{R}^{q\times d}$ and $\widehat{\mathbf{b}} \in \mathbb{R}^d$.

Method:

```
    Initialize $\mathbf{V}$, $\mathbf{b}$ and $\{\underline{\mathbf{a}}[l], \pi_l\}_{l=1}^m$ with $\pi_l \geq 0$ for all $l$ and $\sum_{l=1}^m \pi_l = 1$; $\underline{\boldsymbol{\theta}}[l] = \underline{\mathbf{a}}[l]\mathbf{V} + \mathbf{b} \in \boldsymbol{\Theta}$ for all $l$; $p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[l]\big)$ as defined in (35) for all $k$ and $l$;
  repeat
    {The Expectation Step}
    for $k = 1$ to $n$ do
      for $l = 1$ to $m$ do
        $\widehat{z}_{kl} \longleftarrow p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[l]\big)\pi_l / \sum_{r=1}^m p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[r]\big)\pi_r$
      end for
    end for
    {The Maximization Step}
      for $l = 1$ to $m$ do
        $\pi_l \longleftarrow (1/n)\sum_{k=1}^n \widehat{z}_{kl}$
      end for
      {The Newton-Raphson iterative algorithm}
      for $l = 1$ to $m$ do
        $\underline{\mathbf{a}}[l] \longleftarrow$ update equation (43)
      end for
      for $j = 1$ to $q$ do
        $\mathbf{v}_j \longleftarrow$ update equation (44)
      end for
      $\mathbf{b} \longleftarrow$ update equation (45)
  until convergence;
return $\widehat{\mathcal{Q}} = \{\widehat{\underline{\boldsymbol{\theta}}}[l] = \widehat{\underline{\mathbf{a}}}[l]\widehat{\mathbf{V}} + \widehat{\mathbf{b}}, \widehat{\pi}_l\}_{l=1}^m$.
```

TABLE 2
Semi-Parametric exponential family PCA algorithm.

Algorithm: Bregman Soft Clustering [16]

Input: a set of observations $\{\mathbf{x}[k]\}_{k=1}^n \subseteq \mathbb{R}^d$, two exponential family distribution $p^{(\mathbf{1})}, p^{(\mathbf{2})}$ defined by their cumulant generating functions $G^{(\mathbf{1})}, G^{(\mathbf{2})}$, a number of atoms m.

Output: the NPML estimator that maximizes the complete log-likelihood function $L^{(c)}\big(\mathcal{Q}, \{\mathbf{z}_k\}_{k=1}^n\big)$: $\widehat{\mathcal{Q}} = \{\widehat{\underline{\boldsymbol{\theta}}}[l], \widehat{\pi}_l\}_{l=1}^m$.

Method:

```
    Initialize $\{\underline{\boldsymbol{\theta}}[l], \pi_l\}_{l=1}^m$ with $\pi_l \geq 0$ for all $l$ and $\sum_{l=1}^m \pi_l = 1$; $p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[l]\big)$ as defined in (35) for all $k$ and $l$; $\underline{\boldsymbol{\theta}}[l] \in \boldsymbol{\Theta}$ for all $l$;
  repeat
    {The Expectation Step}
    for $k = 1$ to $n$ do
      for $l = 1$ to $m$ do
        $\widehat{z}_{kl} \longleftarrow p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[l]\big)\pi_l / \sum_{r=1}^m p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[r]\big)\pi_r$
      end for
    end for
    {The Maximization Step}
    for $l = 1$ to $m$ do
      $\pi_l \longleftarrow (1/n)\sum_{k=1}^n \widehat{z}_{kl}$
      $\underline{\boldsymbol{\theta}}[l] \longleftarrow$ solve for $\underline{\boldsymbol{\theta}}[l]$:
          $G'(\underline{\boldsymbol{\theta}}[l]) = \sum_{k=1}^n \widehat{z}_{kl}\mathbf{x}[k] / \sum_{k=1}^n \widehat{z}_{kl}$
    end for
  until convergence;
return $\widehat{\mathcal{Q}} = \{\widehat{\underline{\boldsymbol{\theta}}}[l], \widehat{\pi}_l\}_{l=1}^m$.
```

TABLE 3
Bregman soft clustering algorithm.

4.2 Bregman soft clustering

The Bregman soft clustering approach presented in [16] utilizes an alternative interpretation of the EM algorithm for learning models involving mixtures of exponential family distributions. It is a simple soft clustering algorithm for all Bregman divergences, i.e., for all exponential family distributions. We choose here to present this technique without referring to the Bregman divergence as in [16] but by using its corresponding exponential family probability distribution for the sake of comparison with SP-PCA and exponential PCA.

Given a data set of observations $\{\mathbf{x}[k]\}_{k=1}^n$, Bregman soft clustering aims at modeling the statistical structure of the data as a mixture of m densities of the same exponential family. The clusters correspond to the components of the mixture model and the soft membership of a data point in each cluster is proportional to the probability of the data point being generated by the corresponding density function. The Bregman soft clustering problem is based on a maximum likelihood estimation of the cluster parameters $\{\underline{\boldsymbol{\theta}}[l], \pi_l\}_{l=1}^{m}$ satisfying the following mixture structure:

$$p(\mathbf{x}) = \sum_{l=1}^{m} p\big(\mathbf{x}|\underline{\boldsymbol{\theta}}[l]\big)\pi_l,$$

where $p(\mathbf{x}|\cdot)$ is an exponential family distribution. The data likelihood function takes the following form:

$$p(\mathbf{X}) = \prod_{k=1}^{n}\sum_{l=1}^{m} p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[l]\big)\pi_l. \tag{46}$$

The data likelihood function in (46) is similar to the data likelihood function in (8) without the linear constraint $\underline{\boldsymbol{\theta}}[l] = \underline{\mathbf{a}}[l]\mathbf{V} + \mathbf{b}$ for $l = 1, \ldots, m$. Hence, the Bregman soft clustering problem is similar to the SP-PCA problem without the lower dimensional subspace constraint and a simple EM algorithm is used to estimate the cluster parameters. We consider again the mixed data-type case. The E-step and the first part of the M-step yield the same results as for SP-PCA. In the second part of the M-step, the component parameters $\underline{\boldsymbol{\theta}}[l], l = 1, \ldots, m$, are estimated in the following way:

$$\begin{aligned}\underline{\boldsymbol{\theta}}[l]^{(new)} &= \arg\max_{\underline{\boldsymbol{\theta}}[l]} \sum_{k=1}^{n}\sum_{r=1}^{m} \widehat{z}_{kr} \log p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[r]\big)\\ &= \arg\max_{\underline{\boldsymbol{\theta}}[l]} \Big\{ \sum_{k=1}^{n}\sum_{r=1}^{m} \widehat{z}_{kr} \log p^{(\mathbf{1})}\big(\mathbf{x}^{(\mathbf{1})}[k]|\underline{\boldsymbol{\theta}}^{(\mathbf{1})}[r]\big)\\ &\qquad + \sum_{k=1}^{n}\sum_{r=1}^{m} \widehat{z}_{kr} \log p^{(\mathbf{2})}\big(\mathbf{x}^{(\mathbf{2})}[k]|\underline{\boldsymbol{\theta}}^{(\mathbf{2})}[r]\big)\Big\},\end{aligned}$$

with $\log p\big(\mathbf{x}[k]|\underline{\boldsymbol{\theta}}[r]\big) = \underline{\boldsymbol{\theta}}[r]\mathbf{x}[k]^T - G\big(\underline{\boldsymbol{\theta}}[r]\big)$. Using the convexity properties of $G(\cdot)$, it is easily shown that:

$$G'\big(\underline{\boldsymbol{\theta}}[l]^{(new)}\big) = \Big(\sum_{k=1}^{n} \widehat{z}_{kl}\mathbf{x}[k]\Big) / \Big(\sum_{k=1}^{n} \widehat{z}_{kl}\Big)$$

can be solved for $\underline{\boldsymbol{\theta}}[l]^{(new),(\mathbf{1})}$ and $\underline{\boldsymbol{\theta}}[l]^{(new),(\mathbf{2})}$ by changing $\mathbf{x}$ to $\mathbf{x}^{(\mathbf{1})}$, respectively to $\mathbf{x}^{(\mathbf{2})}$, $G'(\cdot)$ to $G^{(\mathbf{1})'}(\cdot)$, respectively to $G^{(\mathbf{2})'}(\cdot)$.

Table 3 summarizes the Bregman soft clustering algorithm.

5 Conclusion

This paper considers the problem of learning the underlying statistical structure of data of mixed types for fitting generative models. A unified generative model, the *Generalized Linear Statistics* (GLS) model, was established using exponential family properties. Specifically, this work considered mixed data-type records which have both continuous (e.g., Exponential and Gaussian) and discrete (e.g., count and binary) components. The GLS approach allows for the data components to have different parametric forms by using the large range of exponential family distributions. The specific GLS framework developed here is equivalent to a computationally tractable exponential families mixed data-type hierarchical Bayes graphical model with latent variables constrained to a low-dimensional parameter subspace. The exponential family Principal Component Analysis (exponential PCA) technique of [10], the Semi-Parametric exponential family Principal Component Analysis (SP-PCA) technique of [15] and the Bregman soft clustering method presented in [14] were demonstrated not to be separate unrelated algorithms, but rather different manifestations of model assumptions and parameter choices within the GLS framework. Because of this insight, the three algorithms can be extended to readily derive novel extensions that dealt with the important mixed data-type case. As an example, the convex optimization problem related to fitting to a set of data the extreme GLS case corresponding to exponential family Principal Component Analysis is described in detail.

Learning the GLS model provides a generative model of the data, making it possible to both generate synthetic data and perform effective detection or prediction on data of mixed types in parameter space. This data-driven decision making aspect of GLS is presented in a forthcoming Part II paper [50].

References

[1] D. J. Bartholomew and M. Knott, *Latent Variable Models and Factor Analysis*. Kendall's Library of Statistics, Oxford University Press, New York, 2nd edition, 1999, vol. 7.

[2] A. Skrondal and S. Rabe-Hesketh, *Generalized Latent Variable Modeling: Multilevel, Longitudinal, and Structural Equation Models*. Interdisciplinary Statistics, Chapman and Hall/CRC, 2004.

[3] P. McCullagh and J. A. Nelder, *Generalized Linear Models*. Monographs on Statistics and Applied Probability 37, Chapman and Hall/CRC, London, 2nd edition, 1989.

[4] C. E. McCulloch and S. R. Searle, *Generalized, Linear and Mixed Models*. Wiley Series in Probability and Statistics, Wiley-Interscience, New York, 2001.

[5] L. Fahrmeir and G. Tutz, *Multivariate Statistical Modelling Based on Generalized Linear Models*. Springer Series in Statistics, Springer-Verlag, New York, 2nd edition, 2001.

[6] J. Gill, *Generalized Linear Models: A Unified Approach*. Quantitative Applications in the Social Sciences, Sage Publications, Thousand Oaks, California, 2001.

[7] C. R. Rao and H. Toutenburg, *Linear Models: Least Squares and Alternatives*. Springer Series in Statistics, Springer-Verlag, New York, 2nd edition, 1999.

[8] E. W. Frees, *Longitudinal and Panel Data: Analysis and Applications in the Social Sciences*. Cambridge University Press, New York, 2nd edition, 2004.

[9] H. S. Lynn and C. E. McCulloch, "Using principal component analysis and correspondence analysis for estimation in latent variable models," *Journal of the American Statistical Society*, vol. 95, no. 450, pp. 561–572, 2000.

[10] 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*, 2001, vol. 14.

[11] D. B. Dunson and S. D. Perreault, "Factor analytic models of clustered multivariate data with informative censoring," *Biometrics*, vol. 57, pp. 302–308, 2001.

[12] D. B. Dunson, Z. Chen, and J. Harry, "A Bayesian approach for joint modeling of cluster size and subunit-specific outcomes," *Biometrics*, vol. 59, pp. 521–530, 2003.

[13] A. Skrondal and S. Rabe-Hesketh, "Some applications of generalized linear latent and mixed models in epidemiology," *Norsk Epidenmiologi*, vol. 13, no. 2, pp. 265–278, 2003.

[14] A. Banerjee, S. Merugu, I. Dhillon, and J. Ghosh, "Clustering with Bregman divergences," *SIAM International Conference on Data Mining (SDM)*, 2004.

[15] Sajama and A. Orlitsky, "Semi-parametric exponential family PCA," *Advances in Neural Information Processing Systems*, vol. 17, 2004.

[16] A. Banerjee, S. Merugu, I. Dhillon, and J. Ghosh, "Clustering with Bregman divergences," *Journal of Machine Learning Research*, vol. 6, pp. 1705–1749, 2005.

[17] C. Levasseur, K. Kreutz-Delgado, U. Mayer, and G. Gancarz, "Data-pattern discovery methods for detection in nongaussian high-dimensional data sets," *Conference Record of the 39th Asilomar Conference on Signals, Systems and Computers*, pp. 545–549, 2005.

[18] C. Levasseur, U. F. Mayer, B. Burdge, and K. Kreutz-Delgado, "Generalized statistical methods for unsupervised minority class detection in mixed data sets," *Proceedings of the First IAPR Workshop on Cognitive Information Processing*, pp. 126–131, 2008.

[19] C. Levasseur, B. Burdge, K. Kreutz-Delgado, and U. F. Mayer, "A unifying viewpoint of some clustering techniques using Bregman divergences and extensions to mixed data sets," *Proceedings of the First IEEE Int'l Workshop on Data Mining and Artificial Intelligence (DMAI)*, pp. 56–63, 2008.

[20] C. E. McCulloch, *Generalized Linear Mixed Models*. Institute of Mathematical Statistics, Hayward, California, 2003.

[21] N. Roy, G. Gordon, and S. Thrun, "Finding approximate POMDP solutions through belief compression," *Journal of Artificial Intelligence Research*, vol. 23, pp. 1–40, 2005.

[22] D. N. Lawley and A. E. Maxwell, *Factor Analysis as a Statistical Method*. New York, American Elsevier Pub. Co., 1971.

[23] M. Aitkin, "A general maximum likelihood analysis of variance components in generalized linear models," *Biometrics*, vol. 55, pp. 117–128, 1999.

[24] G. McLachlan and D. Peel, *Finite Mixture Models*. Wiley Series in Probability and Statistics, Wiley-Interscience, New York, 2000.

[25] I. T. Jolliffe, *Principal Component Analysis*. Springer Series in Statistics, Springer-Verlag, New York, 2nd edition, 2002.

[26] O. E. Barndorff-Nielsen, *Information and Exponential Families in Statistical Theory*. Wiley Series in Probability and Mathematical Statistics, John Wiley and Sons, 1978.

[27] L. D. Brown, *Fundamentals of Statistical Exponential Families: with Applications in Statistical Decision Theory*. Institute of Mathematical Statistics, 1986.

[28] K. S. Azoury and M. K. Warmuth, "Relative loss bounds for online density estimation with the exponential family of distributions," *Machine Learning*, vol. 43, pp. 211–246, 2001.

[29] Y. Lee and J. A. Nelder, "Hierarchical generalized linear models," *Journal of the Royal Statistical Society*, vol. 58, no. 4, pp. 619–678, 1996.

[30] P. J. Green, "Hierarchical generalized linear models," *Journal of the Royal Statistical Society, B*, vol. 58, no. 4, pp. 619–678, 1996.

[31] M. Aitkin and R. Rocci, "A general maximum likelihood analysis of measurement error in generalized linear models," *Statistics and Computing*, vol. 12, pp. 163–174, 2002.

[32] M. I. Jordan and T. J. Sejnowski, *Graphical Models: Foundations of Neural Computation*. Computational Neuroscience, The MIT Press, 1st edition, 2001.

[33] N. Laird, "Nonparametric maximum likelihood estimation of a mixing distribution," *Journal of the American Statistical Society*, vol. 73, 1978.

[34] B. G. Lindsay, *Mixture Models: Theory, Geometry, and Applications*. Institute of Mathematical Statistics, New York, 1995.

[35] J. Kiefer and J. Wolfowitz, "Consistency of the maximum likelihood estimator in the presence of infinitely many incidental parameters," *The Annals of Mathematical Statistics*, vol. 27, pp. 887–906, 1956.

[36] B. G. Lindsay, "The geometry of mixture likelihoods: a general theory," *Annals of Statistics*, vol. 11, no. 1, pp. 86–94, 1983.

[37] ——, "The geometry of mixture likelihoods, part ii: the exponential family," *Annals of Statistics*, vol. 11, no. 3, pp. 783–792, 1983.

[38] A. Mallet, "A maximum likelihood estimation method for random coefficient regression models," *Biometrika*, vol. 73, no. 3, pp. 645–656, 1986.

[39] A. Schumitzky, "Nonparametric EM algorithms for estimating prior distributions," *Applied Mathematics and Computation*, vol. 45, no. 2, pp. 143–157, 1991.

[40] B. G. Lindsay and M. L. Lesperance, "A review of semiparametric mixture models," *Journal of Statistical Planning and Inference*, vol. 47, pp. 29–99, 1995.

[41] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood estimation from incomplete data via the EM algorithm," *Journal of Royal Statistical Society, B*, vol. 39, pp. 1–38, 1977.

[42] R. A. Redner and H. F. Walker, "Mixture densities, maximum likelihood and the EM algorithm," *SIAM Review*, vol. 26, no. 2, pp. 195–239, 1984.

[43] D. Boehning, *Computer-Assited Analysis of Mixtures and Applications: Meta-Analysis, Disease Mapping, and Others*. Monographs on Statistics and Applied Probability 81, Chapman and Hall/CRC, New York, 2000.

[44] R. S. Pilla and B. Lindsay, "Alternative EM methods for nonparametric finite mixture models," *Biometrika*, vol. 88, no. 2, pp. 535–550, 2001.

[45] S. Rabe-Hesketh, A. Pickles, and C. Taylor, "GLLAMM: A general class of multilevel models and a Stata program," *Multilevel Modelling Newsletter*, vol. 13, pp. 17–23, 2001.

[46] E. L. Lehmann and G. Castella, *Theory of Point Estimation*. Springer Texts in Statistics, Springer, 2nd edition, 1998.

[47] D. P. Bertsekas, A. Nedić, and A. E. Ozdaglar, *Convex Analysis and Optimization*. Athena Scientific, Belmont, Massachusetts, 2003.

[48] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambrigde University Press, 2004.

[49] H. Hindi, "A tutorial on convex optimization II: duality and interior point methods," *Proceedings of the 2006 American Control Conference*, pp. 686–696, 2006.

[50] C. Levasseur, U. F. Mayer, and K. Kreutz-Delgado, "Generalized statistical methods for mixed exponential families, part II: applications," *submitted to IEEE Transactions of Pattern Analysis and Machine Intelligence (PAMI), available at http://dsp.ucsd.edu/∼cecile/*, 2009.