Noname manuscript No.
(will be inserted by the editor)

High-Dimensional Regression with Gaussian Mixtures and Partially-Latent Response Variables

Antoine Deleforge · Florence Forbes · Radu Horaud

the date of receipt and acceptance should be inserted later

Abstract In this work we address the problem of approximating high-dimensional data with a low-dimensional representation. We make the following contributions. We propose an inverse regression method which exchanges the roles of input and response, such that the low-dimensional variable becomes the regressor, and which is tractable. We introduce a mixture of locally-linear probabilistic mapping model that starts with estimating the parameters of inverse regression, and follows with inferring closed-form solutions for the forward parameters of the high-dimensional regression problem of interest. Moreover, we introduce a partially-latent paradigm, such that the vector-valued response variable is composed of both observed and latent entries, thus being able to deal with data contaminated by experimental artifacts that cannot be explained with noise models. The proposed probabilistic formulation could be viewed as a latent-variable augmentation of regression. We devise expectation-maximization (EM) procedures based on a data augmentation strategy which facilitates the maximum-likelihood search over the model parameters. We propose two augmentation schemes and we describe in detail the associated EM inference procedures that may well be viewed as generalizations of a number of EM regression, dimension reduction, and factor analysis algorithms. The proposed framework is validated with both synthetic and real data. We provide experimental evidence that our method outperforms several existing regression techniques.

Keywords Regression · Latent Variable · Mixture Models · Expectation-Maximization · Dimensionality Reduction

A. Deleforge, F. Forbes, R. Horaud
INRIA Grenoble Rhône-Alpes
655 avenue de l'Europe
38330 Montbonnot Saint-Martin, France
Tel.: +33 4 76 61 52 08
E-mail: firstname.lastname@inria.fr

1 Introduction

The task of regression consists of learning a mapping from an input variable onto a response variable, such that the response of a test point could be easily and robustly computed. While this problem has been extensively studied, situations where the input variable is of high dimension, and where the response variable may not be fully observed, still challenge the current state of the art. It is well known that high-dimensional to low-dimensional (or high-to-low) regression is problematic, and usually performed in two separated steps: dimensionality reduction followed by regression. In this paper we propose a novel formulation whose originality is twofold: (i) it overcomes the difficulties of high-to-low regression by exchanging the roles of the input and response variables, and (ii) it incorporates a partially-latent (or partially-observed) response variable model that captures unobserved data.

To bypass the problems associated with high-to-low regression, the roles of the input and response variables are exchanged, such that *the low-dimensional variable becomes the regressor*. We start by estimating the parameters of a low-to-high regression model, or inverse regression (Li, 1991), from which we derive the *forward parameters* characterizing the high-to-low regression problem of interest. We show that, by using mixture models, this inverse-then-forward strategy becomes tractable. Moreover, we allow the low-dimensional variable to be only partially observed (or equivalently, partially latent), *i.e.*, *the vector-valued low-dimensional variable is composed of both observed entries and latent entries*. This is particularly relevant for a variety of applications where the data are too complex to be totally observed.

Starting from standard mixture of linear regressions, we propose a novel mixture of locally-linear regression model that unifies regression and dimensionality reduction into a common framework. The probabilistic formulation that we derive may be seen as a latent-variable augmentation of regression. We devise an associated expectation-maximization procedure based on a data augmentation strategy, thus facilitating the subsequent maximum-likelihood search over the model parameters. We investigate two augmentation schemes and, in practice, we propose two EM algorithms that can be viewed as generalizations of a number of EM algorithms either for regression or for dimension reduction. The proposed method is particularly interesting for solving high-to-low regression problems in the presence of training data corrupted by irrelevant information. It has the potential of dealing with many applications, where the response variable can only be partially observed, either because it cannot be measured with appropriate sensors, or because it cannot be easily annotated. In other terms, the proposed method allows a form of *slack* in the response vector by adding a few latent entries to the vector's observed entries.

The remainder of this paper is organized as follows. Related work, background, and contributions are described in Section 2. Section 3 describes in detail the proposed *Gaussian locally-linear mapping* (GLLiM) model, which solves for *inverse regression*, and derives the formulae for *forward regression*. Next, Section 4 shows how to incorporate a partially-latent variable into GLLiM and discusses the link with a number of existing regression and dimensionality reduction techniques. Section 5 describes the proposed expectation-maximization framework for estimating the parameters of the model. Section 6 describes the experimental validation of our method and compares it with a number of state-of-the-art regression techniques using synthetic data, a dataset of 3D faces, and a dataset of hyper-spectral images of Mars surface. Finally, Section 7 concludes with a discussion and future directions of research. In addition, a companion document referred to as "Supplementary Material" provides details omitted in the manuscript. This document as well as Matlab code for the proposed algorithms and illustrative examples are available online[1].

2 Related Work, Background, and Contributions

2.1 Dealing with High-Dimensional Data

Estimating a function defined over a space of high dimension, say D, is generally hard because standard regression methods have to estimate a large number of parameters, typically of the order of D^2. For this reason, existing methods proceed in two steps: dimension reduction followed by regression. This sequential way of doing presents the risk to map the input onto an intermediate low-dimensional space that does not necessarily contain the information needed to correctly predict the output. To prevent this problem, a number of methods perform the dimension reduction step by taking the output variable into account. The concept of *sufficient reduction* (Cook, 2007) was specifically introduced for solving regression problems of this type. The process of replacing the input with a lower-dimensional representation is called *sufficient dimension reduction* which retains all relevant information about the output. Methods falling into this category are partial least-squares (PLS) (Rosipal and Krämer, 2006), sliced inverse regression (SIR) (Li, 1991), kernel SIR (Wu, 2008), and principal component based methods (Cook, 2007; Adragni and Cook, 2009). SIR methods are not designed specifically for prediction and do not provide a specific predictive method. Once a dimension reduction has been determined, any standard method can then be used to perform predictions, which are likely to be sub-optimal since they are not necessarily consistent with the reduction model. Regarding PLS, its superior performance over standard principal component regression is subject to the relationship between the covariances of input and output variables, and the eigen-structure of the covariance of the input variables (Naik and Tsai, 2000). The principal component methods proposed in (Cook, 2007; Adragni and Cook, 2009) are based on a semi-parametric model of the input given the output and can be used without specifying a model for the joint distribution of input and output variables.

[1] https://team.inria.fr/perception/gllim_toolbox/.

By achieving regression in two steps, these approaches cannot be conveniently expressed in terms of a single optimization problem.

We propose a method that bypasses the difficulty of high-to-low regression by considering the problem the other way around, *i.e.*, low-to-high. We denote the low-dimensional data with $\{\boldsymbol{x}_n\}_{n=1}^N \subset \mathbb{R}^L$, the high-dimensional data with $\{\boldsymbol{y}_n\}_{n=1}^N \subset \mathbb{R}^D$ $(D \gg L)$, and we assume that these data are realizations of two random variables $\boldsymbol{X}$ and $\boldsymbol{Y}$ with joint probability distribution $p(\boldsymbol{Y}, \boldsymbol{X}; \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ denotes the model parameters. At training, the low-dimensional variable $\boldsymbol{X}$ will play the role of the *regressor*, namely $\boldsymbol{Y}$ *is a function of* $\boldsymbol{X}$ possibly corrupted by noise through $p(\boldsymbol{Y}|\boldsymbol{X}; \boldsymbol{\theta})$. Hence, $\boldsymbol{Y}$ *is assumed to lie on a low-dimensional manifold embedded in* $\mathbb{R}^D$ and parameterized by $\boldsymbol{X}$. The low dimension of the regressor $\boldsymbol{X}$ will imply a relatively small number of parameters to be estimated, *i.e.*, approximately linear in $L(D + L)$, thus facilitating the task of estimating the model parameters. Once $\boldsymbol{\theta}$ has been estimated, we show that the computation of the *forward conditional density* $p(\boldsymbol{X}|\boldsymbol{Y}; \boldsymbol{\theta})$ is tractable, and hence is used to predict the low-dimensional response $\boldsymbol{x}$ of a high-dimensional test point $\boldsymbol{y}$. This *inverse-then-forward* regression strategy, thoroughly detailed in Section 3, justifies the unconventional notations: $\boldsymbol{Y}$ for the high-dimensional *input* and $\boldsymbol{X}$ for the low-dimensional *response*.

2.2 Dealing with Non-Linear Data

A classical approach to deal with non-linear data is to use kernel methods. These methods map the data onto high-dimensional, possibly infinite, feature spaces. This is achieved by defining a kernel function over the observation space. Since the kernel function is often not linear, the relations found in this way are accordingly very general. Examples of kernel methods for regression are kernel SIR (Wu, 2008), the relevance vector machine method (Tipping, 2001) or its multivariate extension (Thayananthan et al, 2006). Among kernel methods, Gaussian process latent variable models (GPLVM) form a widely used family of probabilistic models. GPLVM was originally formulated as a dimensionality reduction technique (Lawrence, 2005). It can be viewed as an instance of non-linear probabilistic principal component analysis. GPLVM was then extended to regression (Fusi et al, 2012; Wang and Neal, 2012). One drawback of all kernel methods is that they require a choice for an appropriate kernel function, which is done in an ad-hoc manner and which are highly application- and data-dependent. Moreover, as pointed out in (Lawrence, 2005), the mappings learned with kernel methods cannot be inverted.

Another attractive approach for modeling non-linear data is to use a mixture of *locally linear* models. In the Gaussian case, this boils down to estimating a Gaussian mixture model (GMM) on the joint input-response variable. We will refer to the corresponding family of mappings as *supervised* Gaussian

Locally-Linear Mapping (GLLiM) in the case of regression, *i.e.*, $\boldsymbol{X}$ is fully observed, and *unsupervised* GLLiM in the case of dimensionality reduction, *i.e.*, $\boldsymbol{X}$ is fully unobserved. Supervised GLLiM may be viewed as an affine instance of mixture of experts as formulated in (Xu et al, 1995) or as cluster-weighted modeling (CWM) (Gershenfeld, 1997) except that the response variable is multivariate in GLLiM and scalar in CWM. Interestingly, (Ingrassia et al, 2012) recently proposed an extension of CWM to Student-t distributions. However, they do not address high-dimensional regression and they do not consider a partially-latent variable model. It is worth mentioning that (Ingrassia et al, 2012; Deleforge and Horaud, 2012; **?**) provide similar geometric interpretations of these mixture models. In Section 4 we point out that a number of other regression methods (Quandt and Ramsey, 1978; de Veaux, 1989; Xu et al, 1995; Jedidi et al, 1996; Kain and Macon, 1998; Qiao and Minematsu, 2009; **?**) may be viewed as supervised GLLiM methods, while some dimensionality reduction and factor analysis methods (Tipping and Bishop, 1999a,b; Ghahramani and Hinton, 1996; Wedel and Kamakura, 2001; Bach and Jordan, 2005; Bishop et al, 1998; Kalaitzis and Lawrence, 2012) may be viewed as unsupervised GLLiM methods.

2.3 Dealing with Partially-Observed Response Variables

We propose a generalization of unsupervised and supervised GLLiM referred to as *hybrid GLLiM.* While the high-dimensional variable $\boldsymbol{Y}$ remains fully observed, the low-dimensional variable $\boldsymbol{X}$ is a concatenation of *observed entries*, collectively denoted by $\boldsymbol{T}$, and *latent entries*, collectively denoted by $\boldsymbol{W}$, namely $\boldsymbol{X} = \left[\boldsymbol{T}\,;\boldsymbol{W}\right]$, where $[.;.]$ denotes vertical vector concatenation. The hybrid GLLiM model is particularly interesting for solving regression problems in the presence of data corrupted by irrelevant information for the problem at hand. It has the potential of being well suited in many application scenarios, namely whenever the response variable is only partially observed, because it is neither available, nor observed with appropriate sensors. The idea of the hybrid GLLiM model is to allow some form of *slack* by adding a few latent entries to the response variable.

2.4 Application Scenarios

To further motivate the need for such a model, we consider a few examples. Motion capture methods use regression to infer a map from high-dimensional visual data onto a small number of human-joint angles involved in a particular motion being trained, *e.g.*, (Agarwal and Triggs, 2004, 2006). Nevertheless, the input data contain irrelevant information, such as lighting effects responsible for various artifacts, which aside from the fact that it is not relevant for the task at hand, is almost impossible to be properly modeled, quantified or even annotated. The recovered low-dimensional representation should account for such phenomena that are unobservable.

In the field of planetology, hyper-spectral imaging is used to recover parameters associated with the physical properties of planet surfaces *e.g.*, (Bernard-Michel et al, 2009). To this end, radiative transfer models have been developed, that link the chemical composition, the granularity, or the physical state, to the observed spectrum. They are generally used to simulate huge collections of spectra in order to perform the inversion of hyperspectral images (Douté et al, 2007). As the required computing resources to generate such a database increases exponentially with the number of parameters, they are generally restricted to a small number of parameters, *e.g.*, abundance and grain size of the main chemical components. Other parameters, such as those related to meteorological variability or the incidence angle of the spectrometer are neither explicitly modeled nor measured, in order to keep both the radiative transfer model and the database tractable.

Finally, in sound-source localization, the acoustic input depends on both the source position, which can be observed (Talmon et al, 2011; Deleforge and Horaud, 2012; **?**), and of reverberations, that are strongly dependent on the experimental conditions, and for which ground-truth data are barely available.

3 Gaussian Locally-Linear Mapping (GLLiM)

In this section, we describe in detail the GLLiM model which solves for inverse regression, *i.e.*, the roles of input and response variables are exchanged such that the low-dimensional variable $\boldsymbol{X}$ becomes the regressor. GLLiM relies on a piecewise linear model in the following way. Let $\{\boldsymbol{x}_n\}_{n=1}^{n=N} \in \mathbb{R}^L$ and let us assume that any realization $(\boldsymbol{y}, \boldsymbol{x})$ of $(\boldsymbol{Y}, \boldsymbol{X}) \in \mathbb{R}^D \times \mathbb{R}^L$ is such that $\boldsymbol{y}$ is the image of $\boldsymbol{x}$ by an affine transformation τ_k, among K, plus an error term. This is modeled by a missing variable Z such that $Z = k$ if and only if $\boldsymbol{Y}$ is the image of $\boldsymbol{X}$ by τ_k. The following decomposition of the joint probability distribution will be used:

$$p(\boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{X} = \boldsymbol{x}; \boldsymbol{\theta}) = \\ \sum_{k=1}^{K} p(\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{X} = \boldsymbol{x}, Z = k; \boldsymbol{\theta}) p(\boldsymbol{X} = \boldsymbol{x} | Z = k; \boldsymbol{\theta}) p(Z = k; \boldsymbol{\theta}). \tag{1}$$

where $\boldsymbol{\theta}$ denotes the vector of model parameters. The locally affine function that maps $\boldsymbol{X}$ onto $\boldsymbol{Y}$ is

$$\boldsymbol{Y} = \sum_{k=1}^{K} \mathbb{I}(Z = k)(\mathbf{A}_k \boldsymbol{X} + \boldsymbol{b}_k + \boldsymbol{E}_k), \tag{2}$$

where $\mathbb{I}$ is the indicator function, matrix $\mathbf{A}_k \in \mathbb{R}^{D\times L}$ and vector $\boldsymbol{b}_k \in \mathbb{R}^D$ define the transformation τ_k and $\boldsymbol{E}_k \in \mathbb{R}^D$ is an error term capturing both the observation noise in $\mathbb{R}^D$ and the reconstruction error due to the local affine approximation. Under the assumption that $\boldsymbol{E}_k$ is a zero-mean Gaussian variable with covariance matrix $\boldsymbol{\Sigma}_k \in \mathbb{R}^{D\times D}$ that does not depend on $\boldsymbol{X}$, $\boldsymbol{Y}$, and Z, we obtain:

$$p(\boldsymbol{Y}=\boldsymbol{y}|\boldsymbol{X}=\boldsymbol{x}, Z=k;\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{y};\mathbf{A}_k\boldsymbol{x}+\boldsymbol{b}_k,\boldsymbol{\Sigma}_k). \tag{3}$$

To complete the hierarchical definition of (1) and enforce the affine transformations to be local, $\boldsymbol{X}$ is assumed to follow a mixture of K Gaussians defined by

$$\begin{aligned} p(\boldsymbol{X}=\boldsymbol{x}|Z=k;\boldsymbol{\theta}) &= \mathcal{N}(\boldsymbol{x};\boldsymbol{c}_k,\boldsymbol{\Gamma}_k),\\ p(Z=k;\boldsymbol{\theta}) &= \pi_k, \end{aligned} \tag{4}$$

where $\boldsymbol{c}_k \in \mathbb{R}^L$, $\boldsymbol{\Gamma}_k \in \mathbb{R}^{L\times L}$ and $\sum_{k=1}^K \pi_k = 1$. This model induces a partition of $\mathbb{R}^L$ into K regions $\mathcal{R}_k$, where $\mathcal{R}_k$ is the region where the transformation τ_k is the most probable.

It follows that the model parameters are:

$$\boldsymbol{\theta} = \{\boldsymbol{c}_k,\boldsymbol{\Gamma}_k,\pi_k,\mathbf{A}_k,\boldsymbol{b}_k,\boldsymbol{\Sigma}_k\}_{k=1}^K. \tag{5}$$

Once the parameter vector $\boldsymbol{\theta}$ has been estimated, one obtains an inverse regression, from $\mathbb{R}^L$ (low-dimensional space) to $\mathbb{R}^D$ (high-dimensional space), using the following *inverse conditional density*:

$$p(\boldsymbol{Y}=\boldsymbol{y}|\boldsymbol{X}=\boldsymbol{x};\boldsymbol{\theta}) = \sum_{k=1}^K \frac{\pi_k\mathcal{N}(\boldsymbol{x};\boldsymbol{c}_k,\boldsymbol{\Gamma}_k)}{\sum_{j=1}^K \pi_j\mathcal{N}(\boldsymbol{x};\boldsymbol{c}_j,\boldsymbol{\Gamma}_j)}\mathcal{N}(\boldsymbol{y};\mathbf{A}_k\boldsymbol{x}+\boldsymbol{b}_k,\boldsymbol{\Sigma}_k). \tag{6}$$

The forward regression, from $\mathbb{R}^D$ to $\mathbb{R}^L$ is obtained from the *forward conditional density*:

$$p(\boldsymbol{X}=\boldsymbol{x}|\boldsymbol{Y}=\boldsymbol{y};\boldsymbol{\theta}) = \sum_{k=1}^K \frac{\pi_k^*\mathcal{N}(\boldsymbol{y};\boldsymbol{c}_k^*,\boldsymbol{\Gamma}_k^*)}{\sum_{j=1}^K \pi_j^*\mathcal{N}(\boldsymbol{y};\boldsymbol{c}_j^*.\boldsymbol{\Gamma}_j^*)}\mathcal{N}(\boldsymbol{x};\mathbf{A}_k^*\boldsymbol{y}+\boldsymbol{b}_k^*,\boldsymbol{\Sigma}_k^*). \tag{7}$$

Notice that the above density is fully defined by $\boldsymbol{\theta}$. Indeed, the *forward parameter vector*:

$$\boldsymbol{\theta}^* = \{\boldsymbol{c}_k^*,\boldsymbol{\Gamma}_k^*,\pi_k^*,\mathbf{A}_k^*,\boldsymbol{b}_k^*,\boldsymbol{\Sigma}_k^*\}_{k=1}^K, \tag{8}$$

is obtained analytically using the following formulae:

$$\boldsymbol{c}_k^* = \mathbf{A}_k\boldsymbol{c}_k + \boldsymbol{b}_k, \tag{9}$$

$$\boldsymbol{\Gamma}_k^* = \boldsymbol{\Sigma}_k + \mathbf{A}_k\boldsymbol{\Gamma}_k\mathbf{A}_k^\top, \tag{10}$$

$$\pi_k^* = \pi_k, \tag{11}$$

$$\mathbf{A}_k^* = \boldsymbol{\Sigma}_k^*\mathbf{A}_k^\top\boldsymbol{\Sigma}_k^{-1}, \tag{12}$$

$$\boldsymbol{b}_k^* = \boldsymbol{\Sigma}_k^*(\boldsymbol{\Gamma}_k^{-1}\boldsymbol{c}_k - \mathbf{A}_k^\top\boldsymbol{\Sigma}_k^{-1}\boldsymbol{b}_k), \tag{13}$$

$$\boldsymbol{\Sigma}_k^* = (\boldsymbol{\Gamma}_k^{-1} + \mathbf{A}_k^\top\boldsymbol{\Sigma}_k^{-1}\mathbf{A}_k)^{-1}. \tag{14}$$

One interesting feature of the GLLiM model is that both densities (6) and (7) are Gaussian mixtures parameterized by $\boldsymbol{\theta}$. Therefore, one can use the expectation of (6) to obtain a low-to-high *inverse regression function*:

$$\mathbb{E}[\boldsymbol{Y}=\boldsymbol{y}|\boldsymbol{X}=\boldsymbol{x};\boldsymbol{\theta}]=\sum_{k=1}^{K}\frac{\pi_k\mathcal{N}(\boldsymbol{x};\boldsymbol{c}_k,\boldsymbol{\Gamma}_k)}{\sum_{j=1}^{K}\pi_j\mathcal{N}(\boldsymbol{x};\boldsymbol{c}_j,\boldsymbol{\Gamma}_j)}(\mathbf{A}_k\boldsymbol{x}+\boldsymbol{b}_k), \tag{15}$$

or, even more interestingly, the expectation of (7) to obtain a high-to-low *forward regression function*:

$$\mathbb{E}[\boldsymbol{X}=\boldsymbol{x}|\boldsymbol{Y}=\boldsymbol{y};\boldsymbol{\theta}]=\sum_{k=1}^{K}\frac{\pi_k\mathcal{N}(\boldsymbol{y};\boldsymbol{c}_k^*,\boldsymbol{\Gamma}_k^*)}{\sum_{j=1}^{K}\pi_j\mathcal{N}(\boldsymbol{y};\boldsymbol{c}_j^*,\boldsymbol{\Gamma}_j^*)}(\mathbf{A}_k^*\boldsymbol{y}+\boldsymbol{b}_k^*). \tag{16}$$

3.1 Computational Tractability

Let us analyze the cost of computing a low-to-high (inverse) regression. This computation relies on the estimation of the parameter vector $\boldsymbol{\theta}$. Under the constraint that the K transformations τ_k are affine, it is natural to assume that the error vectors $\boldsymbol{E}_k$ are modeled with equal isotropic Gaussian noise, and hence we have $\{\boldsymbol{\Sigma}_k\}_{k=1}^{k=K}=\sigma^2\mathbf{I}_D$. The number of parameters to be estimated, *i.e.*, the size of $\boldsymbol{\theta}$, is $K(1+L+DL+L(L+1)/2+D)$, for example it is equal to $30,060$ for $K=10$, $L=2$, and $D=1000$.

If, instead, a high-to-low regression is directly estimated, the size of the parameter vector becomes $K(1+D+LD+D(D+1)/2+L)$, which is equal to $5,035,030$ in our example. In practice this is computationally intractable because it requires huge amounts of training data. Nevertheless, one may argue that the number of parameters could be drastically reduced by choosing covariance matrices $\{\boldsymbol{\Gamma}_k\}_{k=1}^{k=K}$ to be isotropic. However, this implies that an isotropic Gaussian mixture model is fitted to the high-dimensional data, which either would very poorly model the complexity of the data, or would require a large number of Gaussian components, leading to data over-fitting.

In Appendix A we show that if $\boldsymbol{\theta}$ is totally unconstrained, the joint distribution (1) is that of an unconstrained Gaussian mixture model (GMM) on the joint variable $[\boldsymbol{X};\boldsymbol{Y}]$, also referred to as *joint GMM* (JGMM). The symmetric roles of $\boldsymbol{X}$ and $\boldsymbol{Y}$ in JGMM implies that low-to-high parameter estimation is strictly equivalent to high-to-low parameter estimation. However, JGMM requires the inversion of K non-diagonal covariance matrices of size $(D+L)\times(D+L)$, which, again, becomes intractable for high-dimensional data.

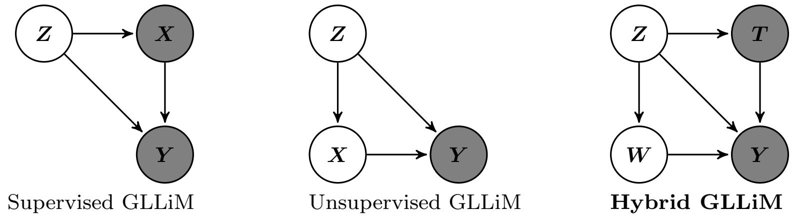

Fig. 1 Graphical representation of the GLLiM models. White-filled circles correspond to unobserved variables while grey-filled circles correspond to observed variables.

4 The Hybrid GLLiM Model

The model just described can be learned with standard EM inference methods if $\boldsymbol{X}$ and $\boldsymbol{Y}$ are both observed. The key idea in this paper is to treat $\boldsymbol{X}$ as a *partially-latent* variable, namely

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{T} \\ \boldsymbol{W} \end{bmatrix},$$

where $\boldsymbol{T} \in \mathbb{R}^{L_\mathrm{t}}$ is observed and $\boldsymbol{W} \in \mathbb{R}^{L_\mathrm{w}}$ is latent ($L = L_\mathrm{t} + L_\mathrm{w}$). Graphical representations of supervised GLLiM, unsupervised GLLiM, and hybrid GLLiM models are illustrated in Figure 1. In hybrid GLLiM, the estimation of model parameters uses observed pairs $\{\boldsymbol{y}_n, \boldsymbol{t}_n\}_{n=1}^N$ while it must also be constrained by the presence of the latent variable $\boldsymbol{W}$. This can be seen as a *latent-variable augmentation* of classical regression, where the observed realizations of $\boldsymbol{Y}$ are affected by the unobserved variable $\boldsymbol{W}$. It can also be viewed as a variant of dimensionality reduction since the unobserved low-dimensional variable $\boldsymbol{W}$ must be recovered from $\{(\boldsymbol{y}_n, \boldsymbol{t}_n)\}_{n=1}^N$. The decomposition of $\boldsymbol{X}$ into observed and latent parts implies that some of the model parameters must be decomposed as well, namely $\boldsymbol{c}_k$, $\boldsymbol{\Gamma}_k$ and $\mathbf{A}_k$. Assuming the independence of $\boldsymbol{T}$ and $\boldsymbol{W}$ given Z we write:

$$\boldsymbol{c}_k = \begin{bmatrix} \boldsymbol{c}_k^\mathrm{t} \\ \boldsymbol{c}_k^\mathrm{w} \end{bmatrix}, \ \boldsymbol{\Gamma}_k = \begin{bmatrix} \boldsymbol{\Gamma}_k^\mathrm{t} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Gamma}_k^\mathrm{w} \end{bmatrix}, \ \mathbf{A}_k = \begin{bmatrix} \mathbf{A}_k^\mathrm{t} \ \mathbf{A}_k^\mathrm{w} \end{bmatrix}. \tag{17}$$

It follows that (2) rewrites as

$$\boldsymbol{Y} = \sum_{k=1}^K \mathbb{I}(Z = k)(\mathbf{A}_k^\mathrm{t}\boldsymbol{T} + \mathbf{A}_k^\mathrm{w}\boldsymbol{W} + \boldsymbol{b}_k + \boldsymbol{E}_k) \tag{18}$$

or equivalently

$$\boldsymbol{Y} = \sum_{k=1}^K \mathbb{I}(Z = k)(\mathbf{A}_k^\mathrm{t}\boldsymbol{T} + \boldsymbol{b}_k + \mathbf{A}_k^\mathrm{w}\boldsymbol{c}_k^\mathrm{w} + \boldsymbol{E}'_k), \tag{19}$$

where the error vector $\boldsymbol{E}'_k$ is modeled by a zero-centered Gaussian variable with a $D \times D$ covariance matrix given by

$$\boldsymbol{\Sigma}'_k = \boldsymbol{\Sigma}_k + \mathbf{A}^{\mathrm{w}}_k \boldsymbol{\Gamma}^{\mathrm{w}}_k \mathbf{A}^{\mathrm{w}\top}_k. \tag{20}$$

Considering realizations of variables $\boldsymbol{T}$ and $\boldsymbol{Y}$, one may thus view hybrid GLLiM as a supervised GLLiM model in which the noise covariance has an unconventional structure, namely (20), where $\mathbf{A}^{\mathrm{w}}_k \boldsymbol{\Gamma}^{\mathrm{w}}_k \mathbf{A}^{\mathrm{w}\top}_k$ is at most a rank-L_{w} matrix. When $\boldsymbol{\Sigma}_k$ is diagonal, this structure is that of factor analysis with at most L_{w} factors, and represents a flexible compromise between a full covariance with $O(D^2)$ parameters on one side, and a diagonal covariance with $O(D)$ parameters on the other side. Let us consider the isotropic case, *i.e.*, $\boldsymbol{\Sigma}_k = \sigma_k^2 \mathbf{I}_D$, for all $k = 1 : K$. We obtain the following three cases for the proposed model:

- $L_{\mathrm{w}} = 0$. This is the fully supervised case, $\boldsymbol{\Sigma}'_k = \boldsymbol{\Sigma}_k$, and is equivalent to the mixture of local linear experts (MLE) model (Xu et al, 1995).
- $L_{\mathrm{w}} = D$. $\boldsymbol{\Sigma}'_k$ takes the form of a general covariance matrix and we obtain the JGMM model (Kain and Macon, 1998; Qiao and Minematsu, 2009) (see Appendix A for a proof). This is the most general GLLiM model, which requires the estimation of K full covariance matrices of size $(D + L) \times (D + L)$. This model becomes over-parameterized and intractable in high dimensions.
- $0 < L_{\mathrm{w}} < D$. This corresponds to the hybrid GLLiM model, and yields a wide variety of novel regression models *in between* MLE and JGMM.

In Section 6, we experimentally show that in some practical cases it is advantageous to use hybrid GLLiM, *i.e.*, the response variable is only partially observed during training, yielding better results than with MLE, JGMM, or a number of state of the art regressions techniques.

As summarized in Table 1, a number of existing methods can be seen as particular instances of hybrid GLLiM where either L_{t} or L_{w} is equal to 0. Several regression models ($L_{\mathrm{w}} = 0$) are instances of hybrid GLLiM, *i.e.*, supervised GLLiM. This is the case for the mixture of local linear experts (MLE) (Xu et al, 1995) where the noise covariances $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$ are isotropic. Probabilistic piecewise affine mapping (PPAM) (**?**) may be viewed as a variant of MLE where $\{\boldsymbol{\Gamma}_k\}_{k=1}^K$ have equal determinants. As already mentioned, it is shown in Appendix A that JGMM (Kain and Macon, 1998; Qiao and Minematsu, 2009) corresponds to the case of unconstrained parameters. The mixture of linear regressors (MLR) (Quandt and Ramsey, 1978; de Veaux, 1989; Jedidi et al, 1996) may also be viewed, as a supervised GLLiM model where covariances $\{\boldsymbol{\Gamma}_k\}_{k=1}^K$ are set to $\eta \mathbf{I}_L$ with $\eta \to \infty$, i.e., there is no prior on $\boldsymbol{X}$. Similarly, several dimensionality reduction models ($L_{\mathrm{t}} = 0$) are instances of hybrid GLLiM, *i.e.*, unsupervised GLLiM. This is the case for probabilistic principal component analysis (PPCA) (Tipping and Bishop, 1999b) and its mixture version (MPPCA) (Tipping and Bishop, 1999a) where the noise covariances $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$ are

Method	$\boldsymbol{c}_k$	$\boldsymbol{\Gamma}_k$	π_k	$\mathbf{A}_k$	$\boldsymbol{b}_k$	$\boldsymbol{\Sigma}_k$	L_t	L_w	K
MLE (Xu et al, 1995)	-	-	-	-	-	diag	-	0	-
MLR (Jedidi et al, 1996)	$\mathbf{0}_L$	$\infty\mathbf{I}_L$	-	-	-	iso+eq	-	0	-
JGMM (Qiao and Minematsu, 2009)	-	-	-	-	-	-	-	0	-
PPAM (Deleforge and Horaud, 2012)	-	|eq|	eq	-	-	diag+eq	-	0	-
GTM (Bishop et al, 1998)	fixed	$\mathbf{0}_L$	eq.	eq.	$\mathbf{0}_D$	iso+eq	0	-	-
PPCA (Tipping and Bishop, 1999b)	$\mathbf{0}_L$	$\mathbf{I}_L$	-	-	-	iso	0	-	1
MPPCA (Tipping and Bishop, 1999a)	$\mathbf{0}_L$	$\mathbf{I}_L$	-	-	-	iso	0	-	-
MFA (Ghahramani and Hinton, 1996)	$\mathbf{0}_L$	$\mathbf{I}_L$	-	-	-	diag	0	-	-
PCCA (Bach and Jordan, 2005)	$\mathbf{0}_L$	$\mathbf{I}_L$	-	-	-	block	0	-	1
RCA (Kalaitzis and Lawrence, 2012)	$\mathbf{0}_L$	$\mathbf{I}_L$	-	-	-	fixed	0	-	1

Table 1 This table summarizes the link between the proposed model and several existing methods. The first three rows corresponds to supervised GLLiM methods ($L_\mathrm{w} = 0$, Fig. 1(a)) while the last six rows correspond to unsupervised GLLiM methods ($L_\mathrm{t} = 0$, Fig. 1(b)). The following symbols are used: "diag" (diagonal covariance matrices), "eq" (equal covariance matrices), "|eq|" (equal determinants), "fixed" (not estimated), "block" (block-diagonal covariance matrices), "-" (unconstrained).

isotropic. Mixture of factor analyzers (MFA) (Ghahramani and Hinton, 1996) corresponds to diagonal noise covariances, probabilistic canonical correlation analysis (PCCA) (Bach and Jordan, 2005) corresponds to block-diagonal noise covariances, and residual component analysis (RCA) (Kalaitzis and Lawrence, 2012) corresponds to fixed (not estimated) noise covariances. The generative topographic mapping (GTM) (Bishop et al, 1998) may also be viewed as an unsupervised GLLiM model where covariances $\{\boldsymbol{\Gamma}_k\}_{k=1}^K$ are set to $\epsilon\mathbf{I}_L$ with $\epsilon \to 0$, *i.e.*, the prior on $\boldsymbol{X}$ is a mixture of Dirac functions. While providing a unifying perspective over these methods, hybrid GLLiM enables a wide range of generalizations corresponding to $L_\mathrm{t} > 0$, $L_\mathrm{w} > 0$.

Finally, it is worth to be noticed that an appropriate choice of the kernel function in the Gaussian process latent variable model (GPLVM) (Lawrence, 2005) allows to account for a partially observed input variable. This was notably studied in (Fusi et al, 2012). However, as explained in (Lawrence, 2005), the mapping yielded by GPLVM cannot be "inverted", due to the non-linear nature of the kernels used in practice. Hence, GPLVM allows regression with partially-latent *input*, and not with partially-latent *response*. The existence of a closed-form expression for the forward regression function, *i.e.*, (16), is therefore a crucial ingredient of the proposed model that fully justifies the usefulness of GLLiM when the task is to regress high-dimensional data onto a partially-observed response.

5 Expectation-Maximization for Hybrid-GLLiM

In this section we devise an EM algorithm to estimate the parameters of the proposed model. The principle of the suggested algorithm is based on a data augmentation strategy that consists of augmenting the observed variables with the unobserved ones, in order to facilitate the subsequent maximum-likelihood search over the parameters.

5.1 Data Augmentation Schemes

There are two sets of missing variables, $Z_{1:N} = \{Z_n\}_{n=1}^N$ and $\boldsymbol{W}_{1:N} = \{\boldsymbol{W}_n\}_{n=1}^N$, associated with the training data set $(\boldsymbol{y}, \boldsymbol{t})_{1:N} = \{\boldsymbol{y}_n, \boldsymbol{t}_n\}_{n=1}^N$, given the number K of linear components and the latent dimension L_w. Two augmentation schemes arise naturally. The first scheme is referred to as general hybrid GLLiM-EM, or general-hGLLiM, and consists of augmenting the observed data with both variables $(Z, \boldsymbol{W})_{1:N}$ while the second scheme, referred to as marginal-hGLLiM, consists of integrating out the continuous variables $\boldsymbol{W}_{1:N}$ previous to data augmentation with the discrete variables $Z_{1:N}$. The difference between these two schemes is in the amount of missing information and this may be of interest considering the well-known fact that the convergence rates of EM procedures are determined by the portion of missing information in the complete data. To accelerate standard EM algorithms it is natural to decrease the amount of missing data, but the practical computational gain is effective only on the premise that the corresponding M-step can be solved efficiently. Another strategy, as a suitable tradeoff between simplicity (or efficiency) and convergence, is based on an extension of the Expectation Conditional Maximization (ECM) algorithm (Meng and Rubin, 1993), referred to as the Alternating ECM (AECM) algorithm (Meng and Van Dyk, 1997). In AECM, the amount of missing data is allowed to be different in each conditional maximization (CM) step. An application of AECM to mixture of factor analysers (MFA) with all its CM-steps in closed-form is given in (McLachlan et al, 2003) and can be compared to the standard EM for MFA described in (Ghahramani and Hinton, 1996). In the case of the proposed hybrid GLLiM model, as it is the case for MFA, using an AECM algorithm typically affects the estimations of the Gaussian means, namely the $\boldsymbol{b}_k$'s in (3). For the latter estimations, the expected empirical weighted mean of the observations is not recovered with standard EM while it is with AECM (see details in Section 5.4).

5.2 Generalization of Other Algorithms

The general hybrid GLLiM-EM algorithm, described in detail below, leads to closed-form expressions for a wide range of constraints onto the covariance matrices $\{\boldsymbol{\Gamma}_k\}_{k=1}^K$ and $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$. Moreover, the algorithm can be applied to both supervised ($L_\mathrm{w} = 0$) and unsupervised ($L_\mathrm{t} = 0$) GLLiM models. Hence, it can be viewed as a generalization of a number of EM inference techniques for regression, *e.g.*, MLR, MLE, JGMM, GTM, or for dimensionality reduction, *e.g.*, MPPCA, MFA, PPCA, and RCA. The marginal hybrid GLLiM-EM algorithm, which is described in detail in Appendix B, is less general. Nevertheless, it is of interest because it provides both an algorithmic insight into the hybrid GLLiM model as well as a natural initialization strategy for the general algorithm. Note that, as mentioned in Appendix B, the marginal hybrid GLLiM-EM also admits an ECM variant. A comprehensive Matlab toolbox including all necessary functions for GLLiM as well as illustrative examples is available online[2].

5.3 Non-Identifiability Issues

Notice that the means $\{\boldsymbol{c}_k^{\mathrm{w}}\}_{k=1}^K$ and covariance matrices $\{\boldsymbol{\Gamma}_k^{\mathrm{w}}\}_{k=1}^K$ must be fixed to avoid non-identifiability issues. Indeed, changing their values respectively corresponds to shifting and scaling the unobserved variables $\boldsymbol{W}_{1:N} \in \mathbb{R}^{L_{\mathrm{w}}}$, which can be compensated by changes in the parameters of the affine transformations $\{\mathbf{A}_k^{\mathrm{w}}\}_{k=1}^K$ and $\{\boldsymbol{b}_k\}_{k=1}^K$. The same issue is observed in all latent variable models used for dimensionality reduction and is always solved by fixing these parameters. In GTM (Bishop et al, 1998) the means are spread on a regular grid and the covariance matrices are set to $\mathbf{0}$ (Dirac functions), while in MPPCA (Tipping and Bishop, 1999a) and MFA (Ghahramani and Hinton, 1996) all means and covariance matrices are respectively set to zero and to identity matrices. The latter option will also be used in our experiments (sections 6.2, 6.3 and 6.4), but for the sake of generality, the following EM algorithm is derived for means and covariance matrices that are arbitrarily fixed.

5.4 The General Hybrid GLLiM-EM Algorithm

Considering the complete data, with $(\boldsymbol{Y},\boldsymbol{T})_{1:N}$ being the observed variables and $(Z,\boldsymbol{W})_{1:N}$ being the missing ones, the corresponding EM algorithm consists of estimating the parameter vector $\boldsymbol{\theta}^{(i+1)}$ that maximizes the expected complete-data log-likelihood, given the current parameter vector $\boldsymbol{\theta}^{(i)}$ and the observed data:

$$\boldsymbol{\theta}^{(i+1)} = \arg\max_{\boldsymbol{\theta}} \mathbb{E}[\log p((\boldsymbol{y},\boldsymbol{t},\boldsymbol{W},Z)_{1:N};\boldsymbol{\theta})|(\boldsymbol{y},\boldsymbol{t})_{1:N};\boldsymbol{\theta}^{(i)}]. \tag{21}$$

Using that $\boldsymbol{W}_{1:N}$ and $\boldsymbol{T}_{1:N}$ are independent conditionally on $Z_{1:N}$ and that $\{\boldsymbol{c}_k^{\mathrm{w}}\}_{k=1}^K$ and $\{\boldsymbol{\Gamma}_k^{\mathrm{w}}\}_{k=1}^K$ are fixed, maximizing (21) is then equivalent to maximizing the following expression:

$$\mathbb{E}_{r_Z^{(i+1)}}[\mathbb{E}_{r_{W|Z}^{(i+1)}}[\log p(\boldsymbol{y}_{1:N} \mid (\boldsymbol{t},\mathbf{W},Z)_{1:N};\boldsymbol{\theta})] + \log p((\boldsymbol{t},Z)_{1:N};\boldsymbol{\theta})], \tag{22}$$

[2] https://team.inria.fr/perception/gllim_toolbox/

where $r_Z^{(i+1)}$ and $r_{W|Z}^{(i+1)}$ denote the posterior distributions

$$r_Z^{(i+1)} = p(\mathbf{Z}_{1:N} | (\boldsymbol{y}, \boldsymbol{t})_{1:N}; \boldsymbol{\theta}^{(i)}) \tag{23}$$

$$r_{W|Z}^{(i+1)} = p(\mathbf{W}_{1:N} | (\boldsymbol{y}, \boldsymbol{t}, Z)_{1:N}; \boldsymbol{\theta}^{(i)}). \tag{24}$$

It follows that the E-step splits into an **E-W** step and an **E-Z** step in the following way. For the sake of readability, the current iteration superscript $(i+1)$ is replaced with a tilde. Hence, $\boldsymbol{\theta}^{(i+1)} = \widetilde{\boldsymbol{\theta}}$ (the model parameter vector).

E-W-step: The posterior probability $\widetilde{r}_{W|Z}$, given parameter estimates, is fully defined by computing the distributions $p(\boldsymbol{w}_n | Z_n = k, \boldsymbol{t}_n, \boldsymbol{y}_n; \boldsymbol{\theta}^{(i)})$, for all n and all k, which can be shown to be Gaussian, with mean $\widetilde{\boldsymbol{\mu}}^{\mathrm{w}}_{nk}$ and covariance matrix $\widetilde{\mathbf{S}}^{\mathrm{w}}_k$ given by:

$$\widetilde{\boldsymbol{\mu}}^{\mathrm{w}}_{nk} = \widetilde{\mathbf{S}}^{\mathrm{w}}_k \big((\mathbf{A}^{\mathrm{w}(i)}_k)^\top (\boldsymbol{\Sigma}^{(i)}_k)^{-1} (\boldsymbol{y}_n - \mathbf{A}^{\mathrm{t}(i)}_k \boldsymbol{t}_n - \boldsymbol{b}^{(i)}_k) + (\boldsymbol{\Gamma}^{\mathrm{w}(i)}_k)^{-1} \boldsymbol{c}^{\mathrm{w}(i)}_k \big) \tag{25}$$

$$\widetilde{\mathbf{S}}^{\mathrm{w}}_k = \big((\boldsymbol{\Gamma}^{\mathrm{w}(i)}_k)^{-1} + (\mathbf{A}^{\mathrm{w}(i)}_k)^\top (\boldsymbol{\Sigma}^{(i)}_k)^{-1} \mathbf{A}^{\mathrm{w}(i)}_k \big)^{-1}. \tag{26}$$

Conditionally to $Z_n = k$, equation (19) shows that this step amounts to a factor analysis step. Indeed, we recover standard formula for the posterior over latent factors where the observations are replaced by the *current residuals*, namely $\boldsymbol{y}_n - \mathbf{A}^{\mathrm{t}(i)}_k \boldsymbol{t}_n - \boldsymbol{b}^{(i)}_k$. Details on the **E-W-step** are given in Section 1.1 of the Supplementary Materials.

E-Z-step: The posterior probability $\widetilde{r}_Z$ is defined by:

$$\widetilde{r}_{nk} = p(Z_n = k | \boldsymbol{t}_n, \boldsymbol{y}_n; \boldsymbol{\theta}^{(i)}) = \frac{\pi^{(i)}_k p(\boldsymbol{y}_n, \boldsymbol{t}_n | Z_n = k; \boldsymbol{\theta}^{(i)})}{\sum_{j=1}^K \pi^{(i)}_j p(\boldsymbol{y}_n, \boldsymbol{t}_n | Z_n = j; \boldsymbol{\theta}^{(i)})} \tag{27}$$

for all n and all k, where

$$p(\boldsymbol{y}_n, \boldsymbol{t}_n | Z_n = k; \boldsymbol{\theta}^{(i)}) = p(\boldsymbol{y}_n | \boldsymbol{t}_n, Z_n = k; \boldsymbol{\theta}^{(i)}) \; p(\boldsymbol{t}_n | Z_n = k; \boldsymbol{\theta}^{(i)}).$$

The second term is equal to $\mathcal{N}(\boldsymbol{t}_n; \boldsymbol{c}^{\mathrm{t}}_k, \boldsymbol{\Gamma}^{\mathrm{t}}_k)$ by virtue of (4) and (17) while it is clear from (19) that

$$p(\boldsymbol{y}_n | \boldsymbol{t}_n, Z_n = k; \boldsymbol{\theta}^{(i)}) = \mathcal{N}(\boldsymbol{y}_n; \mathbf{A}^{(i)}_k [\boldsymbol{t}_n; \boldsymbol{c}^{\mathrm{w}}_k] + \boldsymbol{b}^{(i)}_k, \mathbf{A}^{\mathrm{w}(i)}_k \boldsymbol{\Gamma}^{\mathrm{w}}_k \mathbf{A}^{\mathrm{w}(i)\top}_k + \boldsymbol{\Sigma}^{(i)}_k).$$

The maximization of (21) can then be performed using the posterior probabilities $\widetilde{r}_{nk}$ and the sufficient statistics $\widetilde{\boldsymbol{\mu}}^{\mathrm{w}}_{nk}$ and $\widetilde{\mathbf{S}}^{\mathrm{w}}_k$. We use the following notations: $\widetilde{r}_k = \sum_{n=1}^N \widetilde{r}_{nk}$ and $\widetilde{\boldsymbol{x}}_{nk} = [\boldsymbol{t}_n; \widetilde{\boldsymbol{\mu}}^{\mathrm{w}}_{nk}] \in \mathbb{R}^L$. It can be easily seen from the decomposition (22) of (21), that the M-step can be divided into two separate steps.

First, the updates of parameters $\widetilde{\pi}_k, \widetilde{\boldsymbol{c}}^{\mathrm{t}}_k$ and $\widetilde{\boldsymbol{\Gamma}}^{\mathrm{t}}_k$ correspond to those of a standard Gaussian mixture model on $\boldsymbol{T}_{1:N}$, so that we get straightforwardly:

M-GMM-step:

$$\widetilde{\pi}_k = \frac{\widetilde{r}_k}{N}, \tag{28}$$

$$\widetilde{\boldsymbol{c}}_k^t = \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}}{\widetilde{r}_k} \boldsymbol{t}_n, \tag{29}$$

$$\widetilde{\boldsymbol{\Gamma}}_k^t = \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}}{\widetilde{r}_k} (\boldsymbol{t}_n - \widetilde{\boldsymbol{c}}_k^t)(\boldsymbol{t}_n - \widetilde{\boldsymbol{c}}_k^t)^\top. \tag{30}$$

Second, the updating of the mapping parameters $\{\mathbf{A}_k, \boldsymbol{b}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$ is also in closed-form. Details on the **M-mapping-step** below are provided in Section 1.2 of the Supplementary Materials.

M-mapping-step:

$$\widetilde{\mathbf{A}}_k = \widetilde{\mathbf{Y}}_k \widetilde{\mathbf{X}}_k^\top (\widetilde{\mathbf{S}}_k^{\mathrm{x}} + \widetilde{\mathbf{X}}_k \widetilde{\mathbf{X}}_k^\top)^{-1} \tag{31}$$

where:

$$\widetilde{\mathbf{S}}_k^{\mathrm{x}} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \widetilde{\mathbf{S}}_k^{\mathrm{w}} \end{bmatrix}, \tag{32}$$

$$\widetilde{\mathbf{X}}_k = \frac{1}{\sqrt{\widetilde{r}_k}} \left[\sqrt{\widetilde{r}_{1k}}(\widetilde{\boldsymbol{x}}_{1k} - \widetilde{\boldsymbol{x}}_k), \ldots, \sqrt{\widetilde{r}_{Nk}}(\widetilde{\boldsymbol{x}}_{Nk} - \widetilde{\boldsymbol{x}}_k) \right], \tag{33}$$

$$\widetilde{\mathbf{Y}}_k = \frac{1}{\sqrt{\widetilde{r}_k}} \left[\sqrt{\widetilde{r}_{1k}}(\boldsymbol{y}_1 - \widetilde{\boldsymbol{y}}_k), \ldots, \sqrt{\widetilde{r}_{Nk}}(\boldsymbol{y}_N - \widetilde{\boldsymbol{y}}_k) \right], \tag{34}$$

$$\widetilde{\boldsymbol{x}}_k = \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}}{\widetilde{r}_k} \widetilde{\boldsymbol{x}}_{nk}, \tag{35}$$

$$\widetilde{\boldsymbol{y}}_k = \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}}{\widetilde{r}_k} \boldsymbol{y}_n. \tag{36}$$

Note that in (**??**) and (**??**), $[.,.]$ denotes horizontal concatenation. When $L_{\mathrm{w}} = 0$ then $\widetilde{\mathbf{S}}_k^{\mathrm{x}} = 0$ and the expression (31) of $\widetilde{\mathbf{A}}_k$ is that of standard linear regression from $\{\boldsymbol{t}_n\}_{n=1}^N$ to $\{\boldsymbol{y}_n\}_{n=1}^N$ weighted by $\{\widetilde{r}_{nk}\}_{n=1}^N$. When $L_{\mathrm{t}} = 0$ then $\widetilde{\mathbf{S}}_k^{\mathrm{x}} = \widetilde{\mathbf{S}}_k^{\mathrm{w}}$ and we obtain the principal components update of the EM algorithm for PPCA (Tipping and Bishop, 1999b). The intercept parameter is updated with:

$$\widetilde{\boldsymbol{b}}_k = \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}}{\widetilde{r}_k} (\boldsymbol{y}_n - \widetilde{\mathbf{A}}_k \widetilde{\boldsymbol{x}}_{nk}), \tag{37}$$

or equivalently:

$$\widetilde{\boldsymbol{b}}_k = \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}}{\widetilde{r}_k} (\boldsymbol{y}_n - \widetilde{\mathbf{A}}_k^{\mathrm{t}} \boldsymbol{t}_n) - \widetilde{\mathbf{A}}_k^{\mathrm{w}} \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}}{\widetilde{r}_k} \widetilde{\boldsymbol{\mu}}_{nk}^{\mathrm{w}}.$$

The second term in this expression is the one that would disappear in an AECM algorithm. Finally, we obtain the following expression for $\widetilde{\boldsymbol{\Sigma}}_k$:

$$\widetilde{\boldsymbol{\Sigma}}_k = \widetilde{\mathbf{A}}^{\mathrm{w}}_k \widetilde{\mathbf{S}}^{\mathrm{w}}_k \widetilde{\mathbf{A}}^{\mathrm{w}\top}_k + \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}}{\widetilde{r}_k}(\boldsymbol{y}_n - \widetilde{\mathbf{A}}_k \widetilde{\boldsymbol{x}}_{nk} - \widetilde{\boldsymbol{b}}_k)(\boldsymbol{y}_n - \widetilde{\mathbf{A}}_k \widetilde{\boldsymbol{x}}_{nk} - \widetilde{\boldsymbol{b}}_k)^\top. \quad (38)$$

Note that the previous formulas can be seen as standard ones after *imputation* of the missing variables $\boldsymbol{w}_n$ by their mean values $\widetilde{\boldsymbol{\mu}}^{\mathrm{w}}_{nk}$ via the definition of $\widetilde{\boldsymbol{x}}_{nk}$. As such a direct imputation by the mean necessarily underestimates the variance, the above formula also contains an additional term typically involving the variance $\widetilde{\boldsymbol{S}}^{\mathrm{w}}_k$ of the missing data.

Formulas are given for unconstrained parameters, but can be straightforwardly adapted to different constraints. For instance, if $\{\mathbf{M}_k\}_{k=1}^K \subset \mathbb{R}^{P \times P}$ are solutions for unconstrained covariance matrices $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$ or $\{\boldsymbol{\Gamma}_k\}_{k=1}^K$, then solutions with diagonal (diag), isotropic (iso) and/or equal (eq) for all k constraints are respectively given by $\mathbf{M}_k^{\mathrm{diag}} = \mathrm{diag}(\mathbf{M}_k)$, $\mathbf{M}_k^{\mathrm{iso}} = \mathrm{tr}(\mathbf{M}_k)\mathbf{I}_P/P$ and $\mathbf{M}^{\mathrm{eq}} = \sum_{k=1}^K \widetilde{\pi}_k \mathbf{M}_k$.

5.5 Algorithm Initialization

In general, EM algorithms are known to be sensitive to initialization and likely to converge to local maxima of the likelihood, if not appropriately initialized. Initialization could be achieved either by choosing a set of parameter values and proceeding with the E-step, or by choosing a set of posterior probabilities and proceeding with the M-step. The general hybrid GLLiM-EM algorithm however, is such that there is no straightforward way of choosing a complete set of initial posteriors (namely $r_{nk}^{(0)}, \boldsymbol{\mu}_{nk}^{\mathrm{w}(0)}$ and $\mathbf{S}_k^{\mathrm{w}(0)}$ for all n, k) or a complete set of initial parameters $\boldsymbol{\theta}^{(0)}$ including all the affine transformations. This issue is addressed by deriving the *marginal* hybrid GLLiM-EM algorithm, a variant of thegeneral hybrid GLLiM-EM, in which latent variables $\boldsymbol{W}_{1:N}$ are integrated out, leaving only the estimation of posteriors r_Z in the E-step. Full details on this variant are given in Appendix B. As explained there, this variant is much easier to initialize but it has closed-form steps only if the covariance matrices $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$ are isotropic and distinct. In practice, we start with one iteration of the marginal hybrid GLLiM-EM to obtain a set of initial parameters $\theta^{(0)}$ and continue with the general hybrid GLLiM-EM until convergence.

5.6 Latent Dimension Estimation Using BIC

Once a set of parameters $\widetilde{\boldsymbol{\theta}}$ has been learned with hGLLiM-EM, the *Bayesian information criterion* (BIC) can be computed as follows:

$$BIC(\widetilde{\boldsymbol{\theta}}, N) = -2\mathcal{L}(\widetilde{\boldsymbol{\theta}}) + \mathcal{D}(\widetilde{\boldsymbol{\theta}}) \log N, \quad (39)$$

where $\mathcal{L}$ denotes the observed-data log-likelihood and $\mathcal{D}(\widetilde{\boldsymbol{\theta}})$ denotes the dimension of the complete parameter vector $\widetilde{\boldsymbol{\theta}}$. Assuming, *e.g.*, isotropic and equal noise covariance matrices $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$, we have:

$$\mathcal{D}(\widetilde{\boldsymbol{\theta}}) = K(D(L_{\mathrm{w}} + L_{\mathrm{t}} + 1) + L_{\mathrm{t}}(L_{\mathrm{t}} + 3)/2 + 1) \text{ and} \tag{40}$$

$$\mathcal{L}(\widetilde{\boldsymbol{\theta}}) = \sum_{n=1}^{N} \log p(\boldsymbol{y}_n, \boldsymbol{t}_n; \widetilde{\boldsymbol{\theta}}), \tag{41}$$

where $p(\boldsymbol{y}_n, \boldsymbol{t}_n; \widetilde{\boldsymbol{\theta}})$ is given by the denominator of (27). A natural way to choose a value for L_{w} is, for a given value of K, to train hGLLiM-EM with different values of L_{w}, and select the value minimizing BIC. We will refer to the corresponding method as hGLLiM-BIC. It has the advantage of not requiring the parameter L_{w}, but it is more computationally demanding because it requires to run hGLLiM-EM for all tested values of L_{w}. However, efficient implementations could parallelize these runs.

6 Experiments and Results

6.1 Evaluation methodology

In this Section, we evaluate the performance of the general hybrid GLLiM-EM algorithm (hGLLiM) proposed in Section 5.4 on 3 different datasets. In Section 6.2 we inverse high-dimensional functions using synthetic data. In Section 6.3 we retrieve pose or light information from face images. In Section 6.4 we recover some physical properties of the Mars surface from hyperspectral images. For each of these 3 datasets we consider situations where the target low-dimensional variable is only partially observed during training. hGLLiM-BIC and other hGLLiM models corresponding to a fixed value of L_{w} are tested. The latter are denoted hGLLiM-L_{w}. As mentioned in Table 1, hGLLiM-0 is actually equivalent to a mixture of local linear experts model (Xu et al, 1995) and will thus be referred to as MLE in this Section. In practice, the MLE parameters are estimated using the proposed general hGLLiM-EM algorithm, by setting L_{w} to 0. In all tasks considered, N observed training couples $\{(\boldsymbol{t}_n, \boldsymbol{y}_n)\}_{n=1}^N$ are used to obtain a set of parameters. Then, we use the forward mapping function (16) to compute an estimate $\hat{\boldsymbol{t}}'$ given a test observation $\boldsymbol{y}'$ (please refer to Section 3). This is repeated for N' test observations $\{\boldsymbol{y}'_n\}_{n=1}^{N'}$. The training and the test sets are disjoints in all experiments. Note that MLE was not developed in the context of inverse-then-forward regression in its original paper, and hence, was not suited for high-to-low dimensional regression. Recently, (Deleforge and Horaud, 2012; **?**) combined a variant of MLE with an inverse-then-forward strategy. This variant is called PPAM and includes additional constraints on $\{\boldsymbol{\Gamma}_k\}_{k=1}^{k=K}$ (see Table 1).

Hybrid GLLiM and MLE models are also compared to three other regression techniques, namely joint GMM (JGMM) (Qiao and Minematsu, 2009) which is equivalent to hGLLiM with $L_{\mathrm{w}} \geq D$ (see Section 4 and Appendix A), *sliced inverse regression* (SIR) (Li, 1991) and *multivariate relevance vector machine* (RVM) (Thayananthan et al, 2006). SIR is used with one (SIR-1) or two (SIR-2) principal axes for dimensionality reduction, 20 slices (the number of slices is known to have very little influence on the results), and polynomial regression of order three (higher orders did not show significant improvements in our experiments). SIR quantizes the low-dimensional data $\boldsymbol{X}$ into *slices* or clusters which in turn induces a quantization of the $\boldsymbol{Y}$-space. Each $\boldsymbol{Y}$-slice (all points $\boldsymbol{y}_n$ that map to the same $\boldsymbol{X}$-slice) is then replaced with its mean and PCA is carried out on these means. The resulting dimensionality reduction is then informed by $\boldsymbol{X}$ values through the preliminary slicing. RVM (Thayananthan et al, 2006) may be view as a multivariate probabilistic formulation of *support vector regression* (Smola and Schölkopf, 2004). As all kernel methods, it critically depends on the choice of a kernel function. Using the authors' freely available code[3], we ran preliminary tests to determine an optimal kernel choice for each dataset considered. We tested 14 kernel types with 10 different scales ranging from 1 to 30, hence, 140 kernels for each dataset in total.

6.2 High-dimensional Function Inversion

In this Section, we evaluate the ability of the different regression methods to learn a low-to-high dimensional function $\boldsymbol{f}$ from noisy training examples in order to inverse it. We consider a situation where some components $\boldsymbol{w}$ of the function's support are hidden during training, and where given a new image $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{t}, \boldsymbol{w})$, $\boldsymbol{t}$ is to be recovered. In this Section, MLE and hGLLiM were constrained with equal and isotropic covariance matrices $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$ as it showed to yield the best results with the synthetic functions considered. For RVM, the kernel leading to the best results out of 140 tested kernels was the *linear spline kernel* (Vapnik et al, 1997) with a scale parameter of 8, which is thus used in this Section.

The three vector-valued function families used for testing are of the form $\boldsymbol{f} : \mathbb{R}^2 \to \mathbb{R}^D$, $\boldsymbol{g} : \mathbb{R}^2 \to \mathbb{R}^D$ and $\boldsymbol{h} : \mathbb{R}^3 \to \mathbb{R}^D$. The three functions depend on $\boldsymbol{x} = [t; \boldsymbol{w}]$ which has an observed 1-dimensional part t, and an unobserved 1 or 2-dimensional part $\boldsymbol{w}$. Using the decomposition $\boldsymbol{f} = (f_1 \dots f_d \dots f_D)^\top$ for each function, each component is defined by:

$$\begin{aligned} f_d(t, w_1) &= \alpha_d \cos(\eta_d \, t/10 + \phi_d) + \gamma_d w_1^3 \\ g_d(t, w_1) &= \alpha_d \cos(\eta_d \, t/10 + \beta_d w_1 + \phi_d) \\ h_d(t, w_1, w_2) &= \alpha_d \cos(\eta_d \, t/10 + \beta_d w_1 + \phi_d) + \gamma_d w_2^3 \end{aligned}$$

[3] http://www.mvrvm.com/Multivariate_Relevance_Vector

Table 2 50-D synthetic data: Average (Avg), standard deviation (Std) and percentage of extreme values (Ex) of the absolute error obtained with different methods.

	$\boldsymbol{f}$			$\boldsymbol{g}$			$\boldsymbol{h}$		
Method	Avg	Std	Ex	Avg	Std	Ex	Avg	Std	Ex
JGMM	1.78	2.21	19.5	2.45	2.76	28.4	2.26	2.87	22.4
SIR-1	1.28	1.07	5.92	1.73	1.39	14.9	1.64	1.31	13.0
SIR-2	0.60	0.69	1.02	1.02	1.02	4.20	1.03	1.06	4.91
RVM	0.59	0.53	0.30	0.86	0.68	0.52	0.93	0.75	1.00
MLE	0.36	0.53	0.50	0.36	0.34	0.04	0.61	0.69	0.99
hGLLiM-1	0.20	0.24	0.00	0.25	0.28	0.01	0.46	0.48	0.22
hGLLiM-2	0.23	0.24	0.00	0.25	0.25	0.00	0.36	0.38	0.04
hGLLiM-3	0.24	0.24	0.00	0.26	0.25	0.00	0.34	0.34	0.01
hGLLiM-4	0.23	0.23	0.01	0.28	0.27	0.00	0.35	0.34	0.01
hGLLiM-BIC	**0.18**	**0.21**	**0.00**	**0.24**	**0.26**	**0.00**	**0.33**	**0.35**	**0.06**

where $\boldsymbol{\xi} = \{\alpha_d, \eta_d, \phi_d, \beta_d, \gamma_d\}_{d=1}^{D}$ are scalars in respectively $[0, 2]$, $[0, 4\pi]$, $[0, 2\pi]$, $[0, \pi]$ and $[0, 2]$. This choice allows to generate a wide range of high-dimensional functions with different properties, *e.g.*, monotonicity, periodicity or sharpness. In particular, the generated functions are chosen to be rather challenging for the piecewise affine assumption made in hybrid GLLiM.

One hundred functions of each of these three types were generated, each time using different values for $\boldsymbol{\xi}$ drawn uniformly at random. For each function, a set of N training couples $\{(t_n, \boldsymbol{y}_n)\}_{n=1}^{N}$ and a set of N' test couples $\{(t'_n, \boldsymbol{y}'_n)\}_{n=1}^{N'}$ were synthesized by randomly drawing t and $\boldsymbol{w}$ values and by adding some random isotropic Gaussian noise $\boldsymbol{e}$, *e.g.*, $\boldsymbol{y} = \boldsymbol{f}(t, \boldsymbol{w}) + \boldsymbol{e}$. Values of t were drawn uniformly in $[0, 10]$, while values of $\boldsymbol{w}$ were drawn uniformly in $[-1, 1]$ for $\boldsymbol{f}$ and $\boldsymbol{g}$, and in $[-1, 1]^2$ for $\boldsymbol{h}$. Training couples were used to train the different regression algorithms tested. The task was then to compute an estimate $\hat{t'}_n$ given a test observation $\boldsymbol{y}'_n = \boldsymbol{f}(t'_n, \boldsymbol{w}'_n) + \boldsymbol{e}'_n$.

Table 2 displays the average (Avg), standard deviation (Std) and percentage of *extreme values* (Ex) of the absolute errors $|\hat{t}'_n - t'_n|$ obtained with the different methods. For each generated function, we used an observation dimension $D = 50$, an average signal to noise ratio[4] (SNR) of 6 *decibels* (dB), $N = 200$ training points and $N' = 200$ test points, totaling $20,000$ tests per function type. MLE, JGMM and hGLLiM where used with $K = 5$ mixture components. We define *extreme values* (Ex) as those higher than the average error that would be obtained by an algorithm returning random values of t from the training set. Since training values are uniformly spread in an interval in all considered experiments, this corresponds to one third of the interval's length, *e.g.*, $10/3$ for the synthetic functions. This measure will be repeatedly used throughout the experiments.

As showed in Table 2, all the hGLLiM-L_w models with $L_\mathrm{w} > 0$ significantly outperform the five other regression techniques for the three functions considered, demonstrating the effectiveness of the proposed partially-latent variable

[4] $\mathrm{SNR} = 10 \log (\|\boldsymbol{y}\|^2 \,/\, \|\boldsymbol{e}\|^2)$

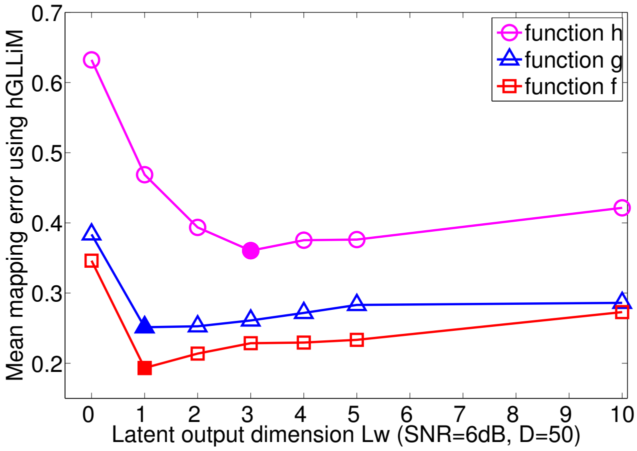

Fig. 2 Influence of the parameter L_{w} of hGLLiM on the mean mapping error of synthetic functions $\boldsymbol{f}$, $\boldsymbol{g}$ and $\boldsymbol{h}$. The minimum of each plot is showed with a filled marker. Each point corresponds to an average error over $10,000$ tests on 50 distinct functions.

model. For each generated training set, the hGLLiM-BIC method minimized BIC for $0 \leq L_{\mathrm{w}} \leq 10$, and used the corresponding model to perform the regression. As showed in 2, hGLLiM-BIC outperformed all the other methods. In practice, hGLLiM-BIC did not use the same L_{w} value for all functions of a given type. Rather, it was able to automatically select a value according to the importance of the latent components in the generated function. The second best method is MLE, *i.e.*, hGLLiM-0. The relative decrease of average error between MLE and hGLLiM-BIC is of respectively 50% for function $\boldsymbol{f}$, 33% for function $\boldsymbol{g}$ and 46% for function $\boldsymbol{h}$. This is a significant improvement, since errors are averaged over $20,000$ tests. Moreover, the addition of latent components in hGLLiM reduced the percentage of extreme errors. Interestingly, BIC selected the *"expected" latent dimension* L_{w}^* for 72% of the 300 generated functions, *i.e.*, $L_{\mathrm{w}}^* = 1$ for $\boldsymbol{f}$ and $\boldsymbol{g}$ and $L_{\mathrm{w}}^* = 2$ for $\boldsymbol{h}$.

Fig. 2 shows the results obtained using hGLLiM-L_{w} and different values of L_{w} for functions $\boldsymbol{f}$, $\boldsymbol{g}$ and $\boldsymbol{h}$. For $\boldsymbol{f}$ and $\boldsymbol{g}$, the lowest average error is obtained using $L_{\mathrm{w}} = 1$. This is expected since $L_{\mathrm{w}}^* = 1$ for $\boldsymbol{f}$ and $\boldsymbol{g}$. However, an interesting observation is made for function $\boldsymbol{h}$. Although $L_{\mathrm{w}}^* = 2$ for $\boldsymbol{h}$, even slightly lower average errors are obtained using $L_{\mathrm{w}} = 3$. While using the expected latent dimension $L_{\mathrm{w}} = L_{\mathrm{w}}^*$ always reduces the mean error with respect to $L_{\mathrm{w}} = 0$ (MLE), the error may be farther reduced by selecting a latent dimension slightly larger that the expected one. This suggests that the actual non linear latent effects on the observations could be modeled more accurately by choosing a latent dimension that is higher than the dimension expected intuitively.

Fig. 3 illustrates how hGLLiM provides a whole range of alternative models *in between* MLE and JGMM, as explained in Section 4. Values of L_{w} in the range $1 \ldots 20$ improve results upon MLE which does not model unobserved

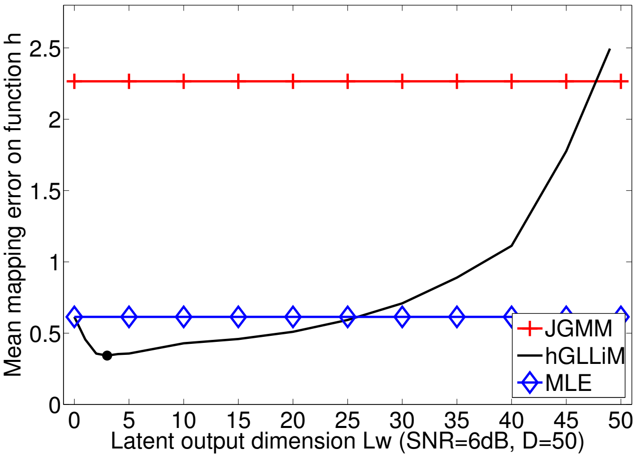

Fig. 3 Influence of the parameter L_{w} of hGLLiM on the mean mapping error of synthetic functions $\boldsymbol{h}$. The minimum is showed with a filled marker. Mean mapping errors obtained with MLE and JGMM on the same data are also showed for comparison. Each point corresponds to an average error over $10,000$ tests on 50 distinct functions.

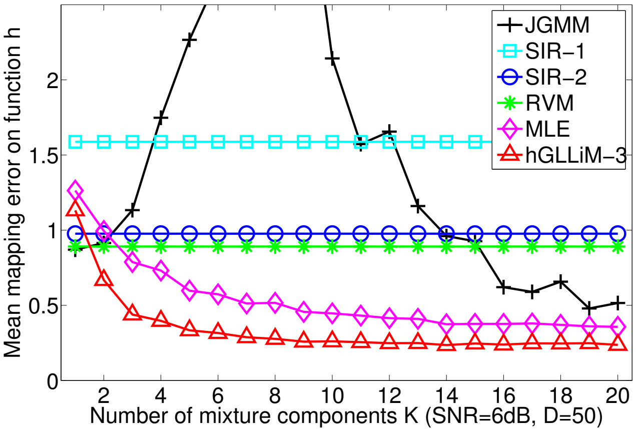

Fig. 4 Influence of the number of initial mixture components K in MLE, JGMM and hGLLiM-3 on the mean mapping error of synthetic function $\boldsymbol{h}$. Errors obtained with SIR-1, SIR-2 and RVM on the same data are also showed for comparison. Each point corresponds to an average error over $10,000$ tests on 50 distinct functions.

variables. As L_{w} increases beyond L_{w}^{*} the number of parameters to estimate becomes larger and larger and the model becomes less and less constrained until becoming equivalent to JGMM with equal unconstrained covariances (see Section 4 and Appendix A).

Extensive experiments showed that obtained errors generally decrease when K increases. However too high values of K lead to degenerate covariance matrices in classes where there are too few samples. Such classes are simply removed along the execution of the algorithms, thus reducing K. This is well illustrated

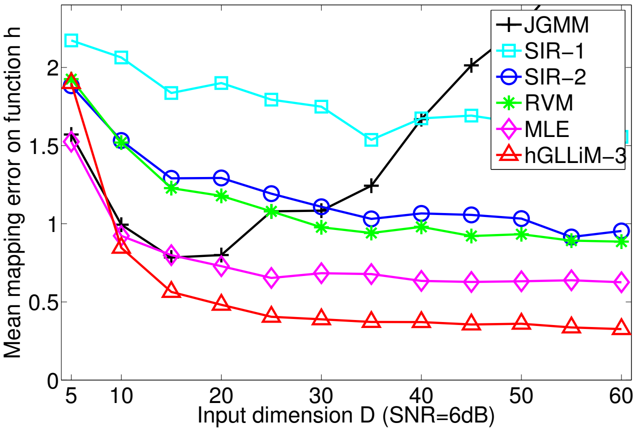

Fig. 5 Influence of D on the mean mapping error of synthetic functions $\boldsymbol{h}$ using different methods. Each point corresponds to an average error over $10,000$ tests on 50 distinct functions.

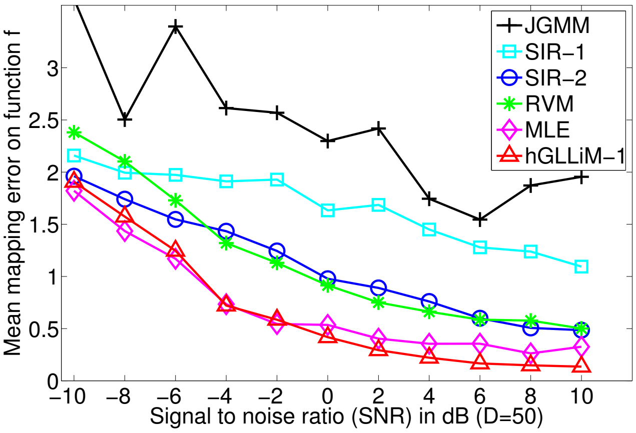

Fig. 6 Influence of the signal-to-noise ration (SNR) on the mean mapping error of synthetic functions $\boldsymbol{f}$ using different methods. Each point corresponds to an average error over $10,000$ tests on 50 distinct functions.

in Fig. 3: results obtained with hGLLiM do not significantly change for initial values of K larger than 9 in that case. Similarly, although K is manually set to a fixed value in the remaining experiments, further tests showed that higher values of K always yielded either similar or better results, at the cost of more computational time. Fig. 3 also shows that the error made by JGMM severely increases with K for $K < 10$, and then decreases to become around 40% larger than MLE. This is in fact an overfitting effect due to the very large numbers of parameters in JGMM. Indeed, the JGMM error with $K = 20$ turned out to increase by more than 100% when decreasing the SNR from 6dB to 3dB, while it increased by less than 30% using all the other methods.

Finally, Figures 5 and 6 show the influence of the observation space dimension D and the SNR on the mean mapping error using various methods. While for low values of D the 6 methods yield similar results, the hybrid GLLiM approach significantly outperforms all of them in higher dimension (Average error 45% lower than with MLE for all $D > 30$). Similarly, apart from JGMM which is very prone to overfitting due to its large number of parameters when D is high, all techniques perform similarly under extreme noise level (SNR $= -10$ dB, where *dB* means *decibels*) while hybrid GLLiM decreases the error up to 60% compared to MLE for positive SNRs.

6.3 Robustly Retrieving Either Pose Or Light From Face Images

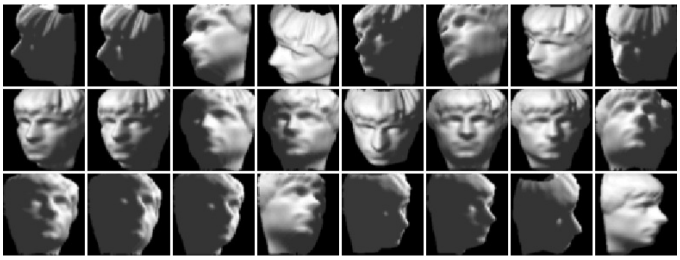

Fig. 7 Example of face images from the Stanford's face dataset.

We now test the different regression methods on the *face dataset*[5] which consists of 697 images (of size 64×64 pixels) of a 3D model of a head whose pose is parameterized by a left-right *pan* angle ranging from $-75°$ to $+75°$ and an up-down *tilt* angle ranging from $-10°$ to $+10°$. Example of such images are given in Figure 7. The image of a face depends on the (pan,tilt) angles as well as on lighting that is absolutely necessary for rendering. The latter is simulated with one parameter taking integer values between 105 and 255. Images were down-sampled[6] to 16×16 and stacked into $D = 256$ dimensional vectors. In the tasks considered, the algorithms were trained using a random subset of $N = 597$ images, and tested with the remaining $N' = 100$ images. We repeated this train-then-test process 50 times for each task ($5,000$ tests per task in total). We used $K = 10$ for MLE, hGLLiM and JGMM (see discussion on K in Section 6.2). Again, hGLLiM and MLE were constrained with equal and isotropic covariance matrices $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$ as it showed to yield the best results. Regarding RVM, as done previously, the best out of 140 kernels was used, *i.e.*, linear spline with scale 20.

We consider two tasks where the target variable is only partially annotated. Firstly, the methods are used to learn the image-to-pose mapping using

[5] http://isomap.stanford.edu/datasets.html

[6] We kept one every 4 pixels horizontally and vertically.

Table 3 Face dataset: Average (Avg), standard deviation (Std) and percentage of extreme values (Ex) of absolute pan and tilt angular errors and light errors obtained with different methods. Superscript * stands for L_w set to its true value L_w^* while † stands for L_w set to the best found dimension in terms of empirical error.

	Pan error (°)			Tilt error (°)		
Method	Avg	Std	Ex	Avg	Std	Ex
JGMM	13.2	26.6	8.2	2.32	3.01	7.0
SIR-1	16.0	11.3	1.4	2.64	2.06	4.9
SIR-2	10.6	9.73	0.4	1.81	1.66	1.9
RVM	14.0	12.2	1.9	2.63	2.13	5.8
MLE	6.01	5.35	0.0	1.84	1.64	1.8
hGLLiM-1*	3.80	4.33	0.0	1.58	1.46	1.0
hGLLiM-13†	**2.65**	**2.39**	**0.0**	**1.19**	**1.11**	**0.2**
hGLLiM-BIC	4.11	4.66	0.0	1.58	1.47	1.0

	Light error		
Method	Avg	Std	Ex
JGMM	18.2	21.0	6.7
SIR-1	15.2	13.2	3.2
SIR-2	13.6	13.2	2.8
RVM	18.7	15.7	4.82
MLE	10.9	8.84	0.2
hGLLiM-2*	10.1	8.84	0.2
hGLLiM-19†	**8.71**	**7.54**	**0.0**
hGLLiM-BIC	10.3	8.66	0.2

pairs of image-pose observations for training while the lighting is unobserved, *i.e.*, *light-invariant face pose estimation.* Secondly, the methods are used to learn the image-to-lighting mapping using pairs of image-light observations for training while the pose is unobserved, *i.e.*, *pose-invariant light-direction estimation.* Table 3 shows results obtained with the different methods. We show results obtained with hGLLiM-L_{w}^*, hGLLiM-$L_{\mathrm{w}}^\dagger$ and hGLLiM-BIC. L_{w}^* denotes the expected latent dimension, and $L_{\mathrm{w}}^\dagger$ is the latent dimension which empirically showed the best results, when varying L_{w} between 0 and 30 (larger values showed to systematically increase the error). For each training set, the hGLLiM-BIC method minimized BIC for $0 \leq L_{\mathrm{w}} \leq 30$, and used the corresponding model to perform the regression. For light-invariant face pose estimation the expected latent dimension is $L_{\mathrm{w}}^* = 1$, and we obtained the best results with $L_{\mathrm{w}}^\dagger = 13$ (values in $[6, 20]$ yielded similar errors). For pose-invariant light-direction estimation the expected latent dimension is $L_{\mathrm{w}}^* = 2$, and we obtained the best results with $L_{\mathrm{w}}^\dagger = 19$ (values in $[11, 20]$ yielded similar errors). As in Section 6.2, we observe that while the expected latent dimension improves upon $L_{\mathrm{w}} = 0$, the error may be farther reduced by selecting a latent dimension larger that the true one. Overall, hGLLiM-$L_{\mathrm{w}}^\dagger$ achieved a 20% to 60% improvement with respect to MLE on this standard dataset. This time, hGLLiM-BIC performed worse than hGLLiM-L_{w}^* and hGLLiM-$L_{\mathrm{w}}^\dagger$ and performed only slightly better than MLE. The expected latent dimension 1 was estimated in 70% of the case for the face pose estimation task, but BIC found a latent dimension of 0 or 1 instead of 2 for the light-direction estimation task.

Another experiment was run to verify whether the latent variable values recovered with our method were meaningful. Once a set of model parameters $\widetilde{\boldsymbol{\theta}}$ were estimated using hGLLiM-1 and with a training set of 597 pose-to-image associations, a different test image $\boldsymbol{y}'$ was selected at random and was used to recover $\hat{\boldsymbol{t}'} \in \mathbb{R}^2$ and $\hat{w'} \in \mathbb{R}$ based on the forward regression function (16), *i.e.*, $\hat{\boldsymbol{x}'} = [\hat{\boldsymbol{t}'}; \hat{w'}] = \mathbb{E}[\boldsymbol{X}|\boldsymbol{y}'; \widetilde{\boldsymbol{\theta}}]$ (see Section 3). An image was then reconstructed using the inverse regression function (15), *i.e.*, $\hat{\boldsymbol{y}'} = \mathbb{E}[\boldsymbol{Y}|[\hat{\boldsymbol{t}'}; w]; \widetilde{\boldsymbol{\theta}}]$, while varying the value of w in order to visually observe its influence on the

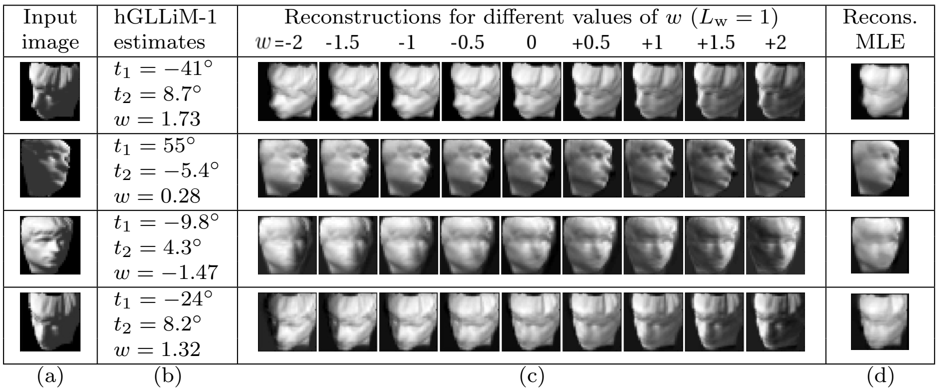

Fig. 8 Recovering the pose of a face (t_1=pan angle, t_2=tilt angle) with lighting being modeled by the latent variable W. (a) The input image. (b) The pose and lighting estimates using hybrid GLLiM. (c) Reconstructed images using the estimated pose parameters and different values for w. (d) Reconstructed images using the pose parameters estimated using MLE.

reconstructed image. Results obtained for different test images are displayed in Fig. 8. These results show that the latent variable $\boldsymbol{W}$ of hybrid GLLiM does capture lighting effects, whereas an explicit lighting parameterization was not present in the training set. For comparison, we show images obtained after projection and reconstruction when MLE (or $L_\mathrm{w} = 0$) is used instead. As it may be observed, the image reconstructed with MLE looks like a blurred average over all possible lightings, while hybrid GLLiM allows a much more accurate image reconstruction process. This is because hybrid GLLiM encodes images with 3 rather than 2 variables, one of which being latent and estimated in an unsupervised way.

6.4 Retrieval of Mars surface physical properties from hyperspectral images

Visible and near infrared imaging spectroscopy is a key remote sensing technique used to study and monitor planets. It records the visible and infrared light reflected from the planet in a given wavelength range and produces cubes of data where each observed surface location is associated with a spectrum. Physical properties of the planets' surface, such as chemical composition, granularity, texture, etc, are some of the most important parameters that characterize the morphology of spectra. In the case of Mars, radiative transfer models have been developed to numerically evaluate the link between these parameters and observable spectra. Such models allow to simulate spectra from a given set of parameter values, *e.g.*, (Douté et al, 2007). In practice, the goal is to scan the Mars ground from an orbit in order to observe gas and dust in the atmosphere and look for signs of specific materials such as silicates, carbonates and ice at the surface. We are thus interested in solving the associate inverse problem which is to deduce physical parameter values from the observed spectra. Since this inverse problem cannot generally be solved analytically, the use of optimization or statistical methods has been investigated, *e.g.* (Bernard-Michel et al, 2009). In particular, training approaches have been considered with the advantage that, once a relationship between parameters and spectra has been established trough training, the learn relationship can be used for very large datasets and for all new images having the same physical model.

Within this category of methods, we investigate the potential of the proposed hybrid GLLiM model using a dataset of hyperspectral images collected from the imaging spectrometer OMEGA instrument (Bibring et al, 2004) onboard of the Mars express spacecraft. To this end a database of synthetic spectra with their associated parameter values were generated using a radiative transfer model. This database is composed of 15,407 spectra associated with five real parameter values, namely, proportion of water ice, proportion of CO_2 ice, proportion of dust, grain size of water ice, and grain size of CO_2 ice. Each spectrum is made of 184 wavelenghts. The hybrid GLLiM method can be used, first to learn as *inverse* regression between parameters and spectra from the database, and second to estimate the corresponding parameters for each new spectrum using the learned relationship. Since no ground truth is available for Mars, the synthetic database will also serve as a first test set to evaluate the accuracy of the predicted parameter values. In order to fully illustrate the potential of hybrid GLLiM, we deliberately ignore two of the parameters in the database and consider them as latent variables. We chose to ignore the proportion of water ice and the grain size of CO_2 ice. These two parameters appear in some previous study (Bernard-Michel et al, 2009) to be sensitive to the same wavelengths than the proportion of dust and are suspected to mix with the other parameters in the synthetic transfer model so that they are harder to estimate. We observed that using them in the inversion tend to degrade the estimation of the other three parameters, which are of particular interest, namely proportion of CO_2 ice, proportion of dust and grain size of water ice. Therefore, we excluded the proportion of water ice and the grain size of CO_2 ice, treated them as latent variables, and did the regression with the three remaining parameters.

Hybrid GLLiM was then compared to JGMM, SIR-1, SIR-2, RVM and MLE. An objective evaluation was done by cross validation. We selected 10,000 training couples at random from the training set, tested on the 5,407 remaining spectra, and repeated this 20 times. For all algorithms, training data were normalized to have 0 mean and unit variance using scaling and translating factors. These factors were then used on test data and estimated output to obtain final estimates. This technique showed to noticeably improve results of all methods. We used $K = 50$ for MLE, HGLLiM and JGMM. MLE and JGMM were constrained with equal, diagonal covariance matrices as it showed to yield the best results. For each training set, the hGLLiM-BIC method minimized BIC for $0 \leq L_{\mathrm{w}} \leq 20$, and used the corresponding model to perform the regression. As regards RVM, the best out of 140 kernels was used. A third

Table 4 Normalized root mean squared error (NRMSE) for Mars surface physical properties recovered from hyperspectral images, using synthetic data and different methods.

Method	Proportion of CO2 ice	Proportion of dust	Grain size of water ice
JGMM	0.83 ± 1.61	0.62 ± 1.00	0.79 ± 1.09
SIR-1	1.27 ± 2.09	1.03 ± 1.71	0.70 ± 0.94
SIR-2	0.96 ± 1.72	0.87 ± 1.45	0.63 ± 0.88
RVM	0.52 ± 0.99	0.40 ± 0.64	0.48 ± 0.64
MLE	0.54 ± 1.00	0.42 ± 0.70	0.61 ± 0.92
hGLLiM-1	0.36 ± 0.70	0.28 ± 0.49	0.45 ± 0.75
hGLLiM-2*†	$\mathbf{0.34 \pm 0.63}$	$\mathbf{0.25 \pm 0.44}$	$\mathbf{0.39 \pm 0.71}$
hGLLiM-3	0.35 ± 0.66	0.25 ± 0.44	0.39 ± 0.66
hGLLiM-4	0.38 ± 0.71	0.28 ± 0.49	0.38 ± 0.65
hGLLiM-5	0.43 ± 0.81	0.32 ± 0.56	0.41 ± 0.67
hGLLiM-20	0.51 ± 0.94	0.38 ± 0.65	0.47 ± 0.71
hGLLiM-BIC	$\mathbf{0.34 \pm 0.63}$	$\mathbf{0.25 \pm 0.44}$	$\mathbf{0.39 \pm 0.71}$

degree polynomial kernel with scale 6 showed the best results using cross-validation on a subset of the database. As a quality measure of the estimated parameters, we computed normalized root mean squared errors (NRMSE[7]). The NRMSE quantifies the difference between the estimated and real parameter values. This measure is normalized enabling direct comparison between the parameters which are of very different range. The closer NRMSE is to zero the more accurate are the predicted values. Table 4 shows obtained NRMSE for the three parameters considered. The expected latent variable dimension is $L_{\mathrm{w}}^{*} = 2$, and accordingly, the empirically best dimension for hGLLiM was $L_{\mathrm{w}}^{\dagger} = 2$. hGLLiM-2 outperformed all the other methods on that task, with an error 36% lower than the second best method RVM, closely followed by MLE. No significant difference was observed between hGLLiM-2 and hGLLiM-3. Note that due to the computation of the $D \times D$ kernel matrix, the computational and memory costs of RVM for training were about 10 times higher than those of hGLLiM, using Matlab implementations. Interestingly, BIC performed very well on these large training sets ($N = 10,000$) as it correctly selected $L_{\mathrm{w}} = 2$ for the 20 considered training sets, yielding identical results as the best method hGLLiM-2.

Finally, we used an adequately selected subset of the synthetic database, *e.g.*, (Bernard-Michel et al, 2009) to train the algorithms, and test them on real data made of observed spectra. In particular, we focus on a dataset of Mars South polar cap. Since no ground truth is currently available for the physical properties of Mars polar regions, we propose a qualitative evaluation using hGLLiM-2 and the three best performing methods, among the tested ones, namely RVM, MLE and JGMM. This evaluation is detailed in Section 2 of the Supplementary Materials. hGLLiM-2 appears to matches satisfyingly expected results from planetology.

[7] NMRSE $= \sqrt{\frac{\sum_{m=1}^{M}(\hat{t}_m - t_m)^2}{\sum_{m=1}^{M}(t_m - \bar{t})^2}}$ with $\bar{t} = M^{-1}\sum_{m=1}^{M} t_m$.

7 Conclusion

The main idea of the method proposed in this paper is to introduce the concept of partially-latent response augmentation in regression. Starting with the mixture of linear regressors family of techniques, we introduced the hybrid GLLiM model. The methodological implementation of the proposed model is investigated. We devised and described in detail an expectation-maximization inference procedure that can be viewed as a generalization of a number of existing probabilistic mapping techniques that span both regression and dimensionality reduction. The method is particularly well suited for estimating the parameters of high-dimensional to low-dimensional mapping problems, all in the presence of training data that contain both pertinent and irrelevant information for the problem at hand. The practical advantages of adding a latent component to the observed outputs is thoroughly tested with both simulated and real data and compared with a large number of probabilistic and deterministic regression methods. In the light of these experiments one may conclude that the proposed algorithm outperforms several existing techniques. This paves the road towards a deeper understanding of a wide range of applications for which training data, that capture the full complexity of natural phenomena, are merely available. The introduction of a latent component allows to capture data behaviors that cannot be easily modeled; in the same time it introduces some form of slack in the parameter inference procedure.

As regards the automatic estimation of the latent component dimension, the generative nature of our probabilistic model allows to treat this issue as a model selection problem and to consider standard information criteria, such as the Bayesian information criterion. This criterion showed very interesting results and good performance especially for large training data sets. However, it imposes to run a number of different models to select the best one and may therefore be computationally costly.

Further research could then include the investigation of adaptive ways to select the latent dimension or other criteria as mentioned by (Bouveyron et al, 2011) for estimating the intrinsic dimension in high dimensional data. Another useful extension would be to take into account more complex dependencies between variables especially when data correspond to images with some spatial structure. Also, similarly to (Ingrassia et al, 2012), more complex noise models could be investigated via Student distributions, *e.g.*, (McLachlan and Peel, 1998) to allow for outliers accommodation and more robust estimation. Finally, it would be interesting to assess the behavior of our method in the presence of irrelevant regressors especially by comparison to other standard methods which are not designed to handle such regressors.

A Link between joint GMM and GLLiM

Proposition 1 *A GLLiM model on $\boldsymbol{X}, \boldsymbol{Y}$ with unconstrained parameters $\boldsymbol{\theta} = \{\boldsymbol{c}_k, \boldsymbol{\Gamma}_k, \pi_k, \mathbf{A}_k, \boldsymbol{b}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$ is equivalent to a Gaussian mixture model on the joint variable $[\boldsymbol{X}; \boldsymbol{Y}]$ with unconstrained parameters $\boldsymbol{\psi} = \{\boldsymbol{m}_k, \mathbf{V}_k, \rho_k\}_{k=1}^K$, i.e.,*

$$p(\boldsymbol{X} = \boldsymbol{x}, \boldsymbol{Y} = \boldsymbol{y}; \boldsymbol{\theta}) = \sum_{k=1}^K \rho_k \mathcal{N}([\boldsymbol{x}; \boldsymbol{y}]; \boldsymbol{m}_k, \mathbf{V}_k). \tag{42}$$

The parameter vector $\boldsymbol{\theta}$ can be expressed as a function of $\boldsymbol{\psi}$ by:

$$\begin{aligned}
\pi_k &= \rho_k, \\
\boldsymbol{c}_k &= \boldsymbol{m}_k^x, \\
\boldsymbol{\Gamma}_k &= \mathbf{V}_k^{xx}, \\
\mathbf{A}_k &= \mathbf{V}_k^{xy\top}(\mathbf{V}_k^{xx})^{-1}, \\
\boldsymbol{b}_k &= \boldsymbol{m}_k^y - (\mathbf{V}_k^{xy})^\top (\mathbf{V}_k^{xx})^{-1} \boldsymbol{m}_k^x, \\
\boldsymbol{\Sigma}_k &= \mathbf{V}_k^{yy} - (\mathbf{V}_k^{xy})^\top (\mathbf{V}_k^{xx})^{-1} \mathbf{V}_k^{xy}, \\
\textit{where } \boldsymbol{m}_k &= \begin{bmatrix} \boldsymbol{m}_k^x \\ \boldsymbol{m}_k^y \end{bmatrix} \\
\textit{and } \mathbf{V}_k &= \begin{bmatrix} \mathbf{V}_k^{xx} & \mathbf{V}_k^{xy} \\ \mathbf{V}_k^{xy\top} & \mathbf{V}_k^{yy} \end{bmatrix}.
\end{aligned} \tag{43}$$

The parameter $\boldsymbol{\psi}$ can be expressed as a function of $\boldsymbol{\theta}$ by:

$$\begin{aligned}
\rho_k &= \pi_k, \\
\boldsymbol{m}_k &= \begin{bmatrix} \boldsymbol{c}_k \\ \mathbf{A}_k \boldsymbol{c}_k + \boldsymbol{b}_k \end{bmatrix}, \\
\mathbf{V}_k &= \begin{bmatrix} \boldsymbol{\Gamma}_k & \boldsymbol{\Gamma}_k \mathbf{A}_k^\top \\ \mathbf{A}_k \boldsymbol{\Gamma}_k & \boldsymbol{\Sigma}_k + \mathbf{A}_k \boldsymbol{\Gamma}_k \mathbf{A}_k^\top \end{bmatrix}.
\end{aligned} \tag{44}$$

Note that this proposition was proved for $D = 1$ in (Ingrassia et al, 2012), but not in the general case as proposed here.

Proof. (43) is obtained using (44) and formulas for conditional multivariate Gaussian variables. (44) is obtained from standard algebra by identifying the joint distribution $p(\boldsymbol{X}, \boldsymbol{Y} | \boldsymbol{Z}; \boldsymbol{\theta})$ defined by (3) and (4) with a multivariate Gaussian distribution. To complete the proof, one need to prove the following two statements:

(i) For any $\rho_k \in \mathbb{R}, \boldsymbol{m}_k \in \mathbb{R}^{D+L}$ and $\mathbf{V}_k \in \mathcal{S}_+^{L+D}$, there is a set of parameters $\boldsymbol{c}_k \in \mathbb{R}^L, \boldsymbol{\Gamma}_k \in \mathcal{S}_+^L, \pi_k \in \mathbb{R}, \mathbf{A}_k \in \mathbb{R}^{D \times L}, \boldsymbol{b}_k \in \mathbb{R}^D$ and $\boldsymbol{\Sigma}_k \in \mathcal{S}_+^D$ such that (43) holds.

(ii) Reciprocally, for any $\boldsymbol{c}_k \in \mathbb{R}^L, \boldsymbol{\Gamma}_k \in \mathcal{S}_+^L, \pi_k \in \mathbb{R}, \mathbf{A}_k \in \mathbb{R}^{D \times L}, \boldsymbol{b}_k \in \mathbb{R}^D, \boldsymbol{\Sigma}_k \in \mathcal{S}_+^D$ there is a set of parameters $\rho_k \in \mathbb{R}, \boldsymbol{m}_k \in \mathbb{R}^{L+D}$ and $\mathbf{V}_k \in \mathcal{S}_+^{D+L}$ such that (44) holds, where $\mathcal{S}_+^M$ denotes the set of $M \times M$ symmetric positive definite matrices. We introduce the following lemma:

Lemma 1 *If*

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}^{\mathrm{xx}} & \mathbf{V}^{\mathrm{xy}} \\ \mathbf{V}^{\mathrm{xy}\top} & \mathbf{V}^{\mathrm{yy}} \end{bmatrix} \in \mathcal{S}_+^{L+D},$$

then $\boldsymbol{\Sigma} = \mathbf{V}^{\mathrm{yy}} - \mathbf{V}^{\mathrm{xy}\top}\mathbf{V}^{\mathrm{xx}-1}\mathbf{V}^{\mathrm{xy}} \in \mathcal{S}_+^D$.

Proof. Since $\mathbf{V} \in \mathcal{S}_+^{L+D}$ we have $\boldsymbol{u}^\top\mathbf{V}\boldsymbol{u} > 0$ for all non null $\boldsymbol{u} \in \mathbb{R}^{L+D*}$. Using the decomposition $\boldsymbol{u} = [\boldsymbol{u}^{\mathrm{x}}; \boldsymbol{u}^{\mathrm{y}}]$ we obtain

$$\boldsymbol{u}^{\mathrm{x}\top}\mathbf{V}^{\mathrm{xx}}\boldsymbol{u}^{\mathrm{x}} + 2\boldsymbol{u}^{\mathrm{x}\top}\mathbf{V}^{\mathrm{xy}}\boldsymbol{u}^{\mathrm{y}} + \boldsymbol{u}^{\mathrm{y}\top}\mathbf{V}^{\mathrm{yy}}\boldsymbol{u}^{\mathrm{y}} > 0 \quad \forall\ \boldsymbol{u}^{\mathrm{x}} \in \mathbb{R}^{L*},\ \ \forall\ \boldsymbol{u}^{\mathrm{y}} \in \mathbb{R}^{D*}.$$

In particular, for $\boldsymbol{u}^{\mathrm{x}} = -\mathbf{V}^{\mathrm{xx}-1}\boldsymbol{u}^{\mathrm{y}}\mathbf{V}^{\mathrm{xy}}$ we obtain

$$\boldsymbol{u}^{\mathrm{y}\top}(\mathbf{V}^{\mathrm{yy}} - \mathbf{V}^{\mathrm{xy}\top}\mathbf{V}^{\mathrm{xx}-1}\mathbf{V}^{\mathrm{xy}})\boldsymbol{u}^{\mathrm{y}} > 0 \ \Leftrightarrow\ \boldsymbol{u}^{\mathrm{y}\top}\boldsymbol{\Sigma}\boldsymbol{u}^{\mathrm{y}} > 0 \quad \forall\ \boldsymbol{u}^{\mathrm{y}} \in \mathbb{R}^{D*}$$

and hence $\boldsymbol{\Sigma} \in \mathcal{S}_+^D$. ■

Lemma 2 *If* $\mathbf{A} \in \mathbb{R}^{D\times L}, \boldsymbol{\Gamma} \in \mathcal{S}_+^L, \boldsymbol{\Sigma} \in \mathcal{S}_+^D$, *then*

$$\mathbf{V} = \begin{bmatrix} \boldsymbol{\Gamma} & \boldsymbol{\Gamma}\mathbf{A}^\top \\ \mathbf{A}\boldsymbol{\Gamma} & \boldsymbol{\Sigma} + \mathbf{A}\boldsymbol{\Gamma}\mathbf{A}^\top \end{bmatrix} \in \mathcal{S}_+^{L+D}.$$

Proof. Since $\boldsymbol{\Gamma} \in \mathcal{S}_+^L$ there is a unique symmetric positive definite matrix $\boldsymbol{\Lambda} \in \mathcal{S}_+^L$ such that $\boldsymbol{\Gamma} = \boldsymbol{\Lambda}^2$. Using standard algebra, we obtain that for all non null $\boldsymbol{u} = [\boldsymbol{u}^{\mathrm{x}}; \boldsymbol{u}^{\mathrm{y}}] \in \mathbb{R}^{L+D*}$,

$$\boldsymbol{u}^\top\mathbf{V}\boldsymbol{u} = ||\boldsymbol{\Lambda}\boldsymbol{u}^{\mathrm{x}} + \boldsymbol{\Lambda}\mathbf{A}^\top\boldsymbol{u}^{\mathrm{y}}||^2 + \boldsymbol{u}^{\mathrm{y}\top}\boldsymbol{\Sigma}\boldsymbol{u}^{\mathrm{y}}$$

where $||.||$ denotes the standard Euclidean distance. The first term of the sum is positive for all $[\boldsymbol{u}^{\mathrm{x}}; \boldsymbol{u}^{\mathrm{y}}] \in \mathbb{R}^{L+D*}$ and the second term strictly positive for all $\boldsymbol{u}^{\mathrm{y}} \in \mathbb{R}^{D*}$ because $\boldsymbol{\Sigma} \in \mathcal{S}_+^D$ by hypothesis. Therefore, $\mathbf{V} \in \mathcal{S}_+^{L+D}$. ■

Lemma 1 and the correspondence formulae (43) prove **(i)**, Lemma 2 and the correspondence formulae (44) prove **(ii)**, hence completing the proof. ■

B The Marginal Hybrid GLLiM-EM

By marginalizing out the hidden variables $\boldsymbol{W}_{1:N}$, we obtain a different EM algorithm than the one presented in section 5, with hidden variables $\boldsymbol{Z}_{1:N}$ only. For a clearer connection with standard procedures we assume here, as already specified, that $\boldsymbol{c}_k^{\mathrm{w}} = \mathbf{0}_{L_{\mathrm{w}}}$ and $\boldsymbol{\Gamma}_k^{\mathrm{w}} = \mathbf{I}_{L_{\mathrm{w}}}$. The **E-W-step** disappears while the **E-Z-step** and the following updating of π_k, $\boldsymbol{c}_k^{\mathrm{t}}$ and $\boldsymbol{\Gamma}_k^{\mathrm{t}}$ in the **M-GMM-step** are exactly the same as in section 5.4. However, the marginalization of $\boldsymbol{W}_{1:N}$

leads to a clearer separation between the regression parameters $\mathbf{A}_k^{\mathrm{t}}$ and $\boldsymbol{b}_k$ (**M-regression-step**) and the other parameters $\mathbf{A}_k^{\mathrm{w}}$ and $\boldsymbol{\Sigma}_k$ (**M-residual-step**). This can be seen straightforwardly from equation (19) which shows that after marginalizing $\boldsymbol{W}$, the model parameters separate into a standard regression part $\boldsymbol{A}_k^{\mathrm{t}}\boldsymbol{t}_n + \boldsymbol{b}_k$ for which standard estimators do not involve the noise variance and a PPCA-like part on the regression residuals $\boldsymbol{y}_n - \tilde{\boldsymbol{A}}_k^{\mathrm{t}}\boldsymbol{t}_n - \tilde{\boldsymbol{b}}_k$, in which the non standard noise covariance $\boldsymbol{\Sigma}_k + \mathbf{A}_k^{\mathrm{w}}(\mathbf{A}_k^{\mathrm{w}})^\top$ is typically dealt with by adding a latent variable $\boldsymbol{W}$.

The algorithm is therefore made of the **E-Z-step** and **M-GMM-step** detailed in 5.4, and the following M-steps:

M-regression-step: The $\mathbf{A}_k^{\mathrm{t}}$ and $\boldsymbol{b}_k$ parameters are obtained using standard weighted affine regression from $\{\boldsymbol{t}_n\}_{n=1}^N$ to $\{\boldsymbol{y}_n\}_{n=1}^N$ with weights $\widetilde{r}_{nk}$, *i.e.*,

$$\widetilde{\mathbf{A}}_k^{\mathrm{t}} = \widetilde{\mathbf{Y}}_k\widetilde{\mathbf{T}}_k^\top(\widetilde{\mathbf{T}}_k\widetilde{\mathbf{T}}_k^\top)^{-1}, \quad \widetilde{\boldsymbol{b}}_k = \sum_{n=1}^N \frac{\widetilde{r}_{kn}}{\widetilde{r}_k}(\boldsymbol{y}_n - \widetilde{\mathbf{A}}_k^{t}\boldsymbol{t}_n), \tag{45}$$

with $\widetilde{\mathbf{T}}_k = \left[\sqrt{\widetilde{r}_{1k}}(\boldsymbol{t}_1 - \widetilde{\boldsymbol{t}}_k)\dots\sqrt{\widetilde{r}_{Nk}}(\boldsymbol{t}_N - \widetilde{\boldsymbol{t}}_k)\right]/\sqrt{\widetilde{r}_k}, \widetilde{\boldsymbol{t}}_k = \sum_{n=1}^N(\widetilde{r}_{kn}/\widetilde{r}_k)\boldsymbol{t}_n$.

M-residual-step: Optimal values for $\mathbf{A}_k^{\mathrm{w}}$ and $\boldsymbol{\Sigma}_k$ are obtained by minimization of the following criterion:

$$Q_k(\boldsymbol{\Sigma}_k, \mathbf{A}_k^{\mathrm{w}}) = -\frac{1}{2}\left(\log|\boldsymbol{\Sigma}_k + \mathbf{A}_k^{\mathrm{w}}\mathbf{A}_k^{\mathrm{w}\top}| + \sum_{n=1}^N \boldsymbol{u}_{kn}^\top(\boldsymbol{\Sigma}_k + \mathbf{A}_k^{\mathrm{w}}\mathbf{A}_k^{\mathrm{w}\top})^{-1}\boldsymbol{u}_{kn}\right), \tag{46}$$

where $\boldsymbol{u}_{kn} = \sqrt{\widetilde{r}_{nk}/\widetilde{r}_k}(\boldsymbol{y}_n - \widetilde{\mathbf{A}}_k^{\mathrm{t}}\boldsymbol{t}_n - \widetilde{\boldsymbol{b}}_k)$. Vectors $\{\boldsymbol{u}_{kn}\}_{n=1}^N$ can be seen as the *residuals* of the k-th local affine transformation. No closed-form solution exists in the general case. A first option is to make use of an inner loop such as a gradient descent technique, or to consider Q_k as the new target observed-data likelihood and use an inner EM corresponding to the general EM described in previous section with $L_{\mathrm{t}} = 0$ and $K = 1$. Another option is to use the Expectation Conditional Maximization (ECM) algorithm (Meng and Rubin, 1993) proposed by (Zhao and Yu, 2008). The ECM algorithm replaces the M-step of the EM algorithm with a sequence of conditional maximization (CM) steps. Such CM steps lead, in the general case, to a conditional (to $\boldsymbol{\Sigma}_k$) update of $\mathbf{A}_k^{\mathrm{w}}$ which is similar to PPCA (Tipping and Bishop, 1999b) with an isotropic noise variance provided and equal to 1. It follows very convenient closed-form expressions (Zhao and Yu, 2008) as is detailed below. (Zhao and Yu, 2008) shows that such an ECM algorithm is computationally more efficient than EM in the case of large sample size relative to the data dimension and that the reverse may as well be true in other situations.

However, in the particular case $\boldsymbol{\Sigma}_k = \sigma_k^2\mathbf{I}_D$, we can afford a standard EM as it connects to PPCA. Indeed, one may notice that Q_k has then exactly the same form as the observed-data log-likelihood in PPCA, with parameters

$(\sigma_k^2, \mathbf{A}_k^{\mathrm{w}})$ and observations $\{\boldsymbol{u}_{kn}\}_{n=1}^N$. Denoting with $\mathbf{C}_k = \sum_{n=1}^N \boldsymbol{u}_{kn}\boldsymbol{u}_{kn}^T/N$ the $D \times D$ sample *residual covariance matrix* and with $\lambda_{1k} > \cdots > \lambda_{Dk}$ its eigenvalues in decreasing order, we can therefore use the key result of (Tipping and Bishop, 1999b) to see that a global maximum of Q_k is obtained for

$$\widetilde{\mathbf{A}}_k^{\mathrm{w}} = \mathbf{U}_k(\boldsymbol{\Lambda}_k - \sigma_k^2\mathbf{I}_{L_{\mathrm{w}}})^{1/2}, \tag{47}$$

$$\widetilde{\sigma}_k^2 = \frac{\sum_{d=L_{\mathrm{w}}+1}^D \lambda_{dk}}{D - L_{\mathrm{w}}}, \tag{48}$$

where $\mathbf{U}_k$ denotes the $D \times L_{\mathrm{w}}$ matrix whose column vectors are the first eigenvectors of $\mathbf{C}_k$ and $\boldsymbol{\Lambda}_k$ is a $L_{\mathrm{w}} \times L_{\mathrm{w}}$ diagonal matrix containing the corresponding first eigenvalues.

The hybrid nature of hGLLiM (at the crossroads of regression and dimensionality reduction) is striking in this variant, as it alternates between a mixture-of-Gaussians step, a local-linear-regression step and a local-linear-dimensionality-reduction step on residuals. This variant is also much easier to initialize as a set of initial posterior values $\{r_{nk}^{(0)}\}_{n=1,k=1}^{N,K}$ can be obtained using the K-means algorithm or the standard GMM-EM algorithm on $\boldsymbol{t}_{1:N}$ or on the joint data $[\boldsymbol{y};\boldsymbol{t}]_{1:N}$ as done in (Qiao and Minematsu, 2009) before proceeding to the M-step. On the other hand, due to the time-consuming eigenvalue decomposition needed at each iteration, the marginal hGLLiM-EM turns out to be slower that the general hGLLiM-EM algorithm described in section 5. We thus use the marginal algorithm as an initialization procedure for the general hGLLiM-EM algorithm.

References

1. Adragni KP, Cook RD (2009) Sufficient dimension reduction and prediction in regression. Philosophical Transactions of the Royal Society A 367(1906):4385–4405
2. Agarwal A, Triggs B (2004) Learning to track 3D human motion from silhouettes. In: International Conference on Machine Learning, Banff, Canada, pp 9–16
3. Agarwal A, Triggs B (2006) Recovering 3D human pose from monocular images. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(1):44–58
4. Bach FR, Jordan MI (2005) A probabilistic interpretation of canonical correlation analysis. Tech. Rep. 688, Department of Statistics, University of California, Berkeley
5. Bernard-Michel C, Douté S, Fauvel M, Gardes L, Girard S (2009) Retrieval of Mars surface physical properties from OMEGA hyperspectral images using Regularized Sliced Inverse Regression. Journal of Geophysical Research: Planets 114(E6)

6. Bibring JP, Soufflot A, Berthé M, Langevin Y, Gondet B, Drossart P, Bouyé M, Combes M, Puget P, Semery A, et al (2004) Omega: Observatoire pour la minéralogie, l'eau, les glaces et l'activité. In: Mars Express: The Scientific Payload, vol 1240, pp 37–49
7. Bishop CM, Svensén M, Williams CKI (1998) GTM: The generative topographic mapping. Neural computation 10(1):215–234
8. Bouveyron C, Celeux G, Girard S (2011) Intrinsic dimension estimation by maximum likelihood in isotropic probabilistic PCA. Pattern Recognition Letters 32:1706–1713
9. Cook RD (2007) Fisher lecture: Dimension reduction in regression. Statistical Science 22(1):1–26
10. Deleforge A, Horaud R (2012) 2D sound-source localization on the binaural manifold. In: IEEE Workshop on Machine Learning for Signal Processing, Santander, Spain
11. Douté S, Schmitt B, Bibring JP, Langevin Y, Altieri F, Bellucci G, Gondet B, the MEX OMEGA team (2005) Nature and composition of the icy terrains of the south pole of Mars from MEX OMEGA observations. In: The 36th Lunar and Planetary Science Conference, (Lunar and Planetary Science XXXVI)
12. Douté S, Deforas E, Schmidt F, Oliva R, Schmitt B (2007) A comprehensive numerical package for the modeling of Mars hyperspectral images. In: The 38th Lunar and Planetary Science Conference, (Lunar and Planetary Science XXXVIII)
13. Fusi N, Stegle O, Lawrence N (2012) Joint modelling of confounding factors and prominent genetic regulators provides increased accuracy in genetical genomics studies. PLoS Computational Biology 8(1):e1002,330
14. Gershenfeld N (1997) Nonlinear inference and cluster-weighted modeling. Annals of the New York Academy of Sciences 808(1):18–24
15. Ghahramani Z, Hinton GE (1996) The EM algorithm for mixtures of factor analyzers. Tech. Rep. CRG-TR-96-1, University of Toronto
16. Ingrassia S, Minotti SC, Vittadini G (2012) Local statistical modeling via a cluster-weighted approach with elliptical distributions. Journal of classification 29(3):363–401
17. Jedidi K, Ramaswamy V, DeSarbo WS, Wedel M (1996) On estimating finite mixtures of multivariate regression and simultaneous equation models. Structural Equation Modeling: A Multidisciplinary Journal 3(3):266–289
18. Kain A, Macon MW (1998) Spectral voice conversion for text-to-speech synthesis. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, Seattle, WA, USA, vol 1, pp 285–288
19. Kalaitzis A, Lawrence N (2012) Residual component analysis: Generalising pca for more flexible inference in linear-gaussian models. In: International Conference on Machine Learning, Edinburgh, Scotland, UK
20. Lawrence N (2005) Probabilistic non-linear principal component analysis with gaussian process latent variable models. The Journal of Machine Learning Research 6:1783–1816

21. Li KC (1991) Sliced inverse regression for dimension reduction. Journal of the American Statistical Association 86(414):316–327

22. McLachlan GJ, Peel D (1998) Robust cluster analysis via mixtures of multivariate t-distributions. In: Lecture Notes in Computer Science, Springer-Verlag, pp 658–666

23. McLachlan GJ, Peel D, Bean R (2003) Modelling high-dimensional data by mixtures of factor analyzers. Computational Statistics and Data Analysis 41(3-4):379–388

24. Meng XL, Rubin DB (1993) Maximum likelihood estimation via the ECM algorithm: A general framework. Biometrika 80(2):267–278

25. Meng XL, Van Dyk D (1997) The EM algorithm: an old folk-song sung to a fast new tune. Journal of the Royal Statistical Society B 59(3):511–567

26. Naik P, Tsai CL (2000) Partial least squares estimator for single-index models. Journal of the Royal Statistical Society B 62(4):763–771

27. Qiao Y, Minematsu N (2009) Mixture of probabilistic linear regressions: A unified view of GMM-based mapping techiques. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, pp 3913–3916

28. Quandt RE, Ramsey JB (1978) Estimating mixtures of normal distributions and switching regressions. Journal of the American Statistical Association 73(364):730–738

29. Rosipal R, Krämer N (2006) Overview and recent advances in partial least squares. In: Saunders C, Grobelnik M, Gunn S, Shawe-Taylor J (eds) Subspace, Latent Structure and Feature Selection, Lecture Notes in Computer Science, vol 3940, Springer, pp 34–51

30. Smola AJ, Schölkopf B (2004) A tutorial on support vector regression. Statistics and computing 14(3):199–222

31. Talmon R, Cohen I, Gannot S (2011) Supervised source localization using diffusion kernels. In: Workshop on Applications of Signal Processing to Audio and Acoustics, pp 245 –248

32. Thayananthan A, Navaratnam R, Stenger B, Torr P, Cipolla R (2006) Multivariate relevance vector machines for tracking. In: European Conference on Computer Vision, Springer, pp 124–138

33. Tipping M (2001) Sparse Bayesian learning and the relevance vector machine. The Journal of Machine Learning Research 1:211–244

34. Tipping ME, Bishop CM (1999a) Mixtures of probabilistic principal component analyzers. Neural Computation 11(2):443–482

35. Tipping ME, Bishop CM (1999b) Probabilistic principal component analysis. Journal of the Royal Statistical Society B 61(3):611–622

36. Vapnik V, Golowich S, Smola A (1997) Support vector method for function approximation, regression estimation, and signal processing. In: Advances in Neural Information Processing, MIT Press, pp 281–287

37. de Veaux RD (1989) Mixtures of linear regressions. Computational Statistics and Data Analalysis 8(3):227–245

38. Wang C, Neal RM (2012) Gaussian process regression with heteroscedastic or non-gaussian residuals. Computing Research Repository

39. Wedel M, Kamakura WA (2001) Factor analysis with (mixed) observed and latent variables in the exponential family. Psychometrika 66(4):515–530
40. Wu H (2008) Kernel sliced inverse regression with applications to classification. Journal of Computational and Graphical Statistics 17(3):590–610
41. Xu L, Jordan MI, Hinton GE (1995) An alternative model for mixtures of experts. In: Advances in Neural Information Processing Systems, MIT Press, Cambridge, MA, USA, pp 633–640
42. Zhao JH, Yu PL (2008) Fast ML estimation for the mixture of factor analyzers via an ECM algorithm. IEEE Transactions on Neural Networks 19(11):1956–1961