High dimensional regression with Gaussian mixtures and partially latent response variables

Antoine Deleforge, Florence Forbes, and Radu Horaud

Perception and Mistis teams
INRIA Grenoble Rhone-Alpes
first.last@inria.fr

Statistics and Computing, Springer, 2014
team.inria.fr/perception/research/high-dim-regression/

Dealing with high dimensional data

Find f between $\boldsymbol{y} \in \mathbb{R}^D$ and $\boldsymbol{x} \in \mathbb{R}^L$ with $D >> L$

$$f: \quad \boldsymbol{y} \in \mathbb{R}^D \quad \longrightarrow \quad \boldsymbol{x} \in \mathbb{R}^L$$

from a learning sample $\{(\boldsymbol{y}_n, \boldsymbol{x}_n), n = 1, \dots N\}$

Difficulty : D large $\Longrightarrow$ curse of dimensionality

Solutions : via dimensionality reduction

- Reduce dimension of $\boldsymbol{y}$ before regression: eg. PCA on the $\boldsymbol{y}_n$'s first

  Risk: poor prediction of $\boldsymbol{x}$
- Take $\boldsymbol{x}$ into account: PLS [Rosipal et al 06] , SIR [Li 91] , Kernel SIR [Wu 08], PC based methods [Cook 07, Adragani & Cook 09], etc.

$\Longrightarrow$ **two steps approaches not expressed as a single optimization problem**

Proposed: Inverse regression then forward prediction

Standard regression setting: Fully Observed Input and Output Variables

Learning (regression) Testing (prediction)

$$\begin{array}{cccccccc} \boldsymbol{x} & & \boldsymbol{y} & & \boldsymbol{y} & & \boldsymbol{x} \\ \begin{pmatrix}\bullet\\ \bullet\end{pmatrix} & \Longrightarrow & \begin{pmatrix}\bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\end{pmatrix} & & \begin{pmatrix}\bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\\ \bullet\end{pmatrix} & \Longrightarrow & \begin{pmatrix}?\\ ?\end{pmatrix} \end{array}$$

Proposed Method: An inverse regression strategy

- $\boldsymbol{X} \in \mathcal{X} \subset \mathbb{R}^L$ low-dimensional space,
- $\boldsymbol{Y} \in \mathcal{Y} \subset \mathbb{R}^D$ high-dimensional space,
- $(\boldsymbol{y}, \boldsymbol{x})$ realization of $(\boldsymbol{Y}, \boldsymbol{X}) \sim p(\boldsymbol{Y}, \boldsymbol{X}; \boldsymbol{\theta})$, $\boldsymbol{\theta}$ parameters

- Inverse conditional density: $p(\boldsymbol{Y} \mid \boldsymbol{X}; \boldsymbol{\theta})$
  $\boldsymbol{Y}$ is a noisy function of $\boldsymbol{X}$

  Modeled via mixtures $\rightarrow$ tractable $\boldsymbol{\theta}$ estimation

- Forward conditional density: $p(\boldsymbol{X} \mid \boldsymbol{Y}; \boldsymbol{\theta}^*)$, with $\boldsymbol{\theta}^* = g(\boldsymbol{\theta})$

  $\rightarrow$ high-to-low prediction, eg. $\hat{\boldsymbol{x}} = E[\boldsymbol{X} \mid \boldsymbol{Y} = \boldsymbol{y}; \boldsymbol{\theta}^*]$

Gaussian Locally-linear Mapping (GLLiM)

- $\boldsymbol{X} \in \mathcal{X} \subset \mathbb{R}^L$ low-dimensional space,
- $\boldsymbol{Y} \in \mathcal{Y} \subset \mathbb{R}^D$ high-dimensional space,
- A piecewise affine model: Introduce a missing variable Z

$$p(\boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{X} = \boldsymbol{x}; \boldsymbol{\theta}) =$$

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

$Z = k \Leftrightarrow \boldsymbol{Y}$ is the image of $\boldsymbol{X}$ by an affine transformation τ_k

Hierarchical definition

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

$\mathbb{I}$ Indicator function, $\mathbf{A}_k$ $D \times L$ matrix, $\boldsymbol{b}_k$ D-dim vector

$\boldsymbol{E}_k$: observation noise in $\mathbb{R}^D$ and reconstruction error, Gaussian, centered, independent on $\boldsymbol{X}$, $\boldsymbol{Y}$, and Z

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

- Affine transformations are local: mixture of K Gaussians

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

- The set of all model parameters is:

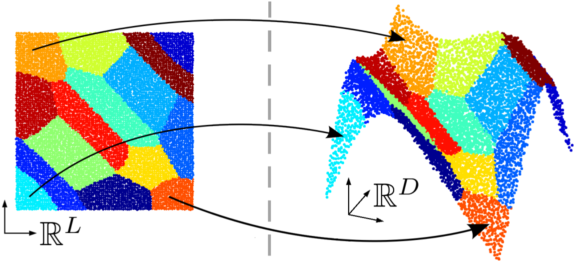
$$\boldsymbol{\theta} = \{\boldsymbol{c}_k, \boldsymbol{\Gamma}_k, \pi_k, \mathbf{A}_k, \boldsymbol{b}_k, \boldsymbol{\Sigma}_k, k = 1 \dots K\}$$

Usually $\{\boldsymbol{\Sigma}_k = \sigma \mathbf{I}_D, k = 1 \dots K\}$ (isotropic reconstruction error)

Geometric Interpretation

This model induces a partition of $\mathbb{R}^L$ into K regions $\mathcal{R}_k$ where the transformation τ_k is the most probable.

If $|\boldsymbol{\Gamma}_1| = \cdots = |\boldsymbol{\Gamma}_K|$: $\{\mathcal{R}_k, k = 1 \dots K\}$ define a Voronoi diagram of centroids $\{\boldsymbol{c}_k, k = 1 \dots K\}$ (Mahalanobis distance $||.||_{\boldsymbol{\Gamma}}$).



$L = 2, D = 3, K = 15.$

Low-to-high (Inverse) Regression

If $\boldsymbol{X}$ and $\boldsymbol{Y}$ are both observed

- The parameter vector, $\boldsymbol{\theta}$, can be estimated in closed-form using an EM inference procedure
- This yields the *inverse conditional density* which is a Gaussian mixture:

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

High-to-low (Forward) Regression

- The forward parameter vector, $\boldsymbol{\theta}^*$, has an analytic expression as a function of $\boldsymbol{\theta}$
- This yields the *forward conditional density* which is a Gaussian mixture as well:

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

The forward parameter vector $\boldsymbol{\theta}^*$ from $\boldsymbol{\theta}$

$$
\begin{aligned}
\boldsymbol{c}_k^* &= \mathbf{A}_k \boldsymbol{c}_k + \boldsymbol{b}_k, \\
\boldsymbol{\Gamma}_k^* &= \boldsymbol{\Sigma}_k + \mathbf{A}_k \boldsymbol{\Gamma}_k \mathbf{A}_k^\top, \\
\pi_k^* &= \pi_k, \\
\mathbf{A}_k^* &= \boldsymbol{\Sigma}_k^* \mathbf{A}_k^\top \boldsymbol{\Sigma}_k^{-1}, \\
\boldsymbol{b}_k^* &= \boldsymbol{\Sigma}_k^* (\boldsymbol{\Gamma}_k^{-1} \boldsymbol{c}_k - \mathbf{A}_k^\top \boldsymbol{\Sigma}_k^{-1} \boldsymbol{b}_k), \\
\boldsymbol{\Sigma}_k^* &= (\boldsymbol{\Gamma}_k^{-1} + \mathbf{A}_k^\top \boldsymbol{\Sigma}_k^{-1} \mathbf{A}_k)^{-1}.
\end{aligned}
$$

Regression functions

Both densities are Gaussian mixtures parameterized by $\boldsymbol{\theta}$.
Therefore, to obtain:

- A low-to-high *inverse* regression function:

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

- A high-to-low *forward* regression function:

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

Low-to-High or High-to-Low?

If $\boldsymbol{\theta}$ is unconstrained

GLLiM $\Leftrightarrow$ Joint GMM on $(\boldsymbol{X}, \boldsymbol{Y})$ (JGMM)

- $\boldsymbol{X}$ and $\boldsymbol{Y}$ roles are symmetric
- Low-to-High or High-to-Low estimation are equivalent

Intractable for high D:

- JGMM requires inversion of K matrices of size $(D+L) \times (D+L)$

Low-to-High or High-to-Low?

Error vectors $\boldsymbol{E}_k$ assumed isotropic Gaussians: $\forall k, \boldsymbol{\Sigma}_k = \sigma \mathbf{I}_D$ ($\boldsymbol{\theta}$ is constrained)

Example: $D = 1000$, $L = 2$, $K = 10$

- Low-to-high regression:
  $K(1 + L + DL + L(L + 1)/2 + D) = 30,060$ parameters.
- High-to-low regression:
  $K(1 + D + LD + D(D + 1)/2 + L) =$ **5,035,030** parameters.
  Requires inversion of 1000×1000 covariance matrices.

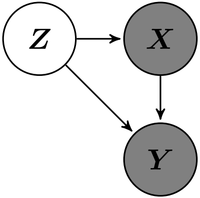
Therefore it is better to perfom a low-dimensional-to-high-dimensional (inverse) regression, and then deduce the forward density.

Extension: the Hybrid GLLiM model

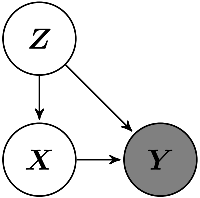
Incorporate a latent component into the low-dimensional variable:

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

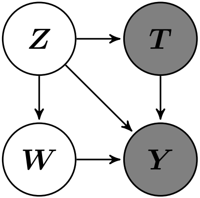
where $\boldsymbol{T} \in \mathbb{R}^{L_{\mathrm{t}}}$ is observed and $\boldsymbol{W} \in \mathbb{R}^{L_{\mathrm{w}}}$ is latent ($L = L_{\mathrm{t}} + L_{\mathrm{w}}$)



Supervised GLLiM (regression)



Unsupervised GLLiM (dim reduction)



Hybrid GLLiM

Fully-latent Output Variable: Dimensionality reduction, eg. PPCA

Learning

$$\boldsymbol{x} \qquad\qquad \boldsymbol{y}$$

$$\begin{pmatrix}\circ\\\circ\\\circ\end{pmatrix} \Longrightarrow \begin{pmatrix}\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\end{pmatrix}$$

Testing

$$\boldsymbol{y} \qquad\qquad \boldsymbol{x}$$

$$\begin{pmatrix}\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\\\bullet\end{pmatrix} \Longrightarrow \begin{pmatrix}?\\?\\?\end{pmatrix}$$

Partially-latent Output Variable : Hybrid GLLiM

Learning | Testing

$$\begin{array}{ccc} \boldsymbol{x} & & \boldsymbol{y} \\ \begin{pmatrix} \bullet \\ \bullet \\ \circ \end{pmatrix} & \Longrightarrow & \begin{pmatrix} \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \end{pmatrix} \end{array} \qquad \begin{array}{ccc} \boldsymbol{y} & & \boldsymbol{x} \\ \begin{pmatrix} \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \end{pmatrix} & \Longrightarrow & \begin{pmatrix} ? \\ ? \\ ? \end{pmatrix} \end{array}$$

The hybrid GLLiM model

Hybrid between regression and dimensionality reduction:

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

- Observed pairs $\{(\boldsymbol{y}_n, \boldsymbol{t}_n), n = 1 \dots N\}$ ($\boldsymbol{T} \in \mathbb{R}^{L_\mathrm{t}}$)
- Additional latent variable $\boldsymbol{W}$ ($\boldsymbol{W} \in \mathbb{R}^{L_\mathrm{w}}$)
- Assuming the independence of $\boldsymbol{T}$ and $\boldsymbol{W}$ given Z :

$$p(\boldsymbol{X} = (\boldsymbol{t}, \boldsymbol{w})^\top \mid Z = k) = \mathcal{N}_L((\boldsymbol{t}, \boldsymbol{w})^\top; \boldsymbol{c}_k, \boldsymbol{\Gamma}_k)$$

with $\quad \boldsymbol{c}_k = \begin{bmatrix} \boldsymbol{c}_k^\mathrm{t} \\ \boldsymbol{c}_k^\mathrm{w} \end{bmatrix}, \; \boldsymbol{\Gamma}_k = \begin{bmatrix} \boldsymbol{\Gamma}_k^\mathrm{t} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Gamma}_k^\mathrm{w} \end{bmatrix}$

The hybrid GLLiM model

With $\mathbf{A}_k = [\ \mathbf{A}_k^{\mathrm{t}} \quad \mathbf{A}_k^{\mathrm{w}}\ ]$,

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

with $\boldsymbol{E}_k \sim \mathcal{N}_D(0, \boldsymbol{\Sigma}_k)$

rewrites

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

with $\boldsymbol{E}'_k \sim \mathcal{N}_D(0, \boldsymbol{\Sigma}_k + \mathbf{A}_k^{\mathrm{w}}\boldsymbol{\Gamma}_k^{\mathrm{w}}\mathbf{A}_k^{\mathrm{w}\top})$

- Supervised GLLiM with *unconventional* covariance structure
- Diagonal $\boldsymbol{\Sigma}_k \longrightarrow$ Factor analysis with L_{w} factors (at most)
- A compromise between full $O(D^2)$ and diagonal $O(D)$ covariances

Link to other models

Assuming $\mathbf{\Sigma}_k = \sigma_k^2 \mathbf{I}_D$, $\quad (\mathbf{\Sigma}'_k = \mathbf{\Sigma}_k + \mathbf{A}_k^{\mathrm{w}} \mathbf{\Gamma}_k^{\mathrm{w}} \mathbf{A}_k^{\mathrm{w}\top})$

- $L_{\mathrm{w}} = 0$, Supervised case, $\mathbf{\Sigma}'_k = \mathbf{\Sigma}_k$:
  Mixture of local linear experts (MLE) [Xu et al 95]
- $L_{\mathrm{w}} = D$, $\mathbf{\Sigma}'_k$ general covariance matrix:
  JGMM model [Qiao et al 09], the most general GLLiM model
  Over-parameterized, intractable ($(D+L) \times (D+L)$ matrices)
- $0 < L_{\mathrm{w}} < D$: a wide variety of models *between* MLE and JGMM.

Gaussian Process Latent Variable Model [Lawrence 05, Fusi & al 12]:
Regression with partially-latent *input*, but not with partially-latent *response*

GPLVM mapping not inversible (non-linear nature of the kernels used in practice)

Particular instances of the hybrid GLLiM model

First three rows: supervised GLLiM methods ($L_\mathrm{w} = 0$)

Last six rows: unsupervised GLLiM methods ($L_\mathrm{t} = 0$)

Model	$\boldsymbol{c}_k$	$\boldsymbol{\Gamma}_k$	π_k	$\mathbf{A}_k$	$\boldsymbol{b}_k$	$\boldsymbol{\Sigma}_k$	L_t	L_w	K
MLE [Xu et al 95]	-	-	-	-	-	diag	-	0	-
MLR [Jedidi et al 96]	$\mathbf{0}_L$	$\infty\mathbf{I}_L$	-	-	-	iso+eq	-	0	-
JGMM [Qiao et al 09]	-	-	-	-	-	-	-	0	-
PPAM [Deleforge et al 12]	-	|eq|	eq	-	-	diag+eq	-	0	-
GTM [Bishop et al 98]	fixed	$\mathbf{0}_L$	eq.	eq.	$\mathbf{0}_D$	iso+eq	0	-	-
PPCA [Tipping et al 99a]	$\mathbf{0}_L$	$\mathbf{I}_L$	-	-	-	iso	0	-	1
MPPCA [Tipping et al 99b]	$\mathbf{0}_L$	$\mathbf{I}_L$	-	-	-	iso	0	-	-
MFA [Ghahramani et al 96]	$\mathbf{0}_L$	$\mathbf{I}_L$	-	-	-	diag	0	-	-
PCCA [Bach et al 05]	$\mathbf{0}_L$	$\mathbf{I}_L$	-	-	-	block	0	-	1
RCA [Kalaitzis et al 11]	$\mathbf{0}_L$	$\mathbf{I}_L$	-	-	-	fixed	0	-	1

Expectation Maximization for Hybrid GLLiM

2 data augmentation schemes: Convergence speed/M-step tractability tradeoff

(other: Alternating ECM [Meng & Rubin 97] eg. for MFA [McLachlan et al 03])

- General hybrid GLLiM-EM: augmenting with both $(Z, \boldsymbol{W})$
  - Closed-form expressions for a wide range of $\{\boldsymbol{\Gamma}_k, \boldsymbol{\Sigma}_k, k = 1 \dots K\}$
- Marginal-hGLLiM: integrating out the $\boldsymbol{W}$
  - Less general, closed form only for distinc isotropic $\{\boldsymbol{\Sigma}_k, k = 1 \dots K\}$
  - Algorithmic insight: alternation of a regression & reduction step
  - Natural initialization strategy

Identifiability issues

As for latent variable models for dimensionality reduction (eg MPPCA, MFA):

$\{\boldsymbol{c}_k^{\mathrm{w}}\}_{k=1}^K$ and $\{\boldsymbol{\Gamma}_k^{\mathrm{w}}\}_{k=1}^K$ must be fixed , eg. $\boldsymbol{c}_k^{\mathrm{w}} = 0$ and $\boldsymbol{\Gamma}_k^{\mathrm{w}} = \mathbf{I}_{L_w}$

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

$$(\boldsymbol{W}|Z=k) \sim \mathcal{N}(\boldsymbol{c}_k^{\mathrm{w}}, \boldsymbol{\Gamma}_k^{\mathrm{w}}) \implies$$
$$(\mathbf{A}_k^{\mathrm{w}}\boldsymbol{W} + \boldsymbol{b}_k|Z=k) \sim \mathcal{N}(\mathbf{A}_k^{\mathrm{w}}\boldsymbol{c}_k^{\mathrm{w}} + \boldsymbol{b}_k, \mathbf{A}_k^{\mathrm{w}}\boldsymbol{\Gamma}_k^{\mathrm{w}}\mathbf{A}_k^{\mathrm{w}\top})$$

The general Hybrid GLLiM EM algorithm

Observed $\{(\boldsymbol{Y}_n, \boldsymbol{T}_n), n = 1:N\}$ and Missing variables $\{(Z_n, \boldsymbol{W}_n), n = 1:N\}$

At iteration (i), update:

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

E-step: compute posterior distributions: $\forall n, \forall k$

$$p(Z_n = k | \boldsymbol{t}_n, \boldsymbol{y}_n; \boldsymbol{\theta}^{(i)}) = \frac{\pi_k^{(i)} p(\boldsymbol{y}_n, \boldsymbol{t}_n | Z_n = k; \boldsymbol{\theta}^{(i)})}{\sum_{j=1}^{K} \pi_j^{(i)} p(\boldsymbol{y}_n, \boldsymbol{t}_n | Z_n = j; \boldsymbol{\theta}^{(i)})}$$

$$p(\boldsymbol{w}_n | Z_n = k, \boldsymbol{t}_n, \boldsymbol{y}_n; \boldsymbol{\theta}^{(i)}) \sim \mathcal{N}(\widetilde{\boldsymbol{\mu}}_{nk}^{\mathrm{w}}, \widetilde{\mathbf{S}}_k^{\mathrm{w}}) \quad \text{(Factor Analysis like)}$$

The general Hybrid GLLiM EM algorithm

With $\widetilde{r}_{nk} = p(Z_n = k | \boldsymbol{t}_n, \boldsymbol{y}_n; \boldsymbol{\theta}^{(i)})$

M-step: divides in two

- Updating $\pi_k, \boldsymbol{c}_k^{\mathrm{t}}, \boldsymbol{\Gamma}_k^{\mathrm{t}}$: standard Gaussian mixture step on $\{\boldsymbol{t}_n, n = 1 \dots N\}$
- Updating the mapping parameters $\mathbf{A}_k, \boldsymbol{b}_k, \boldsymbol{\Sigma}_k$
  - $L_{\mathrm{w}} = 0$: $\mathbf{A}_k$ is that of standard linear regression from $\{\boldsymbol{t}_n, n = 1 \dots N\}$ to $\{\boldsymbol{y}_n, n = 1 \dots N\}$ weighted by $\{\widetilde{r}_{nk}, n = 1 \dots N\}$
  - $L_{\mathrm{t}} = 0$: principal components update of PPCA

M-GMM step

With $\widetilde{r}_k = \sum_{n=1}^{N} \widetilde{r}_{nk}$

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

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

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

M-mapping step

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

where:

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

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

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

$$\widetilde{\boldsymbol{x}}_{nk} = [\boldsymbol{t}_n; \widetilde{\boldsymbol{\mu}}_{nk}^{\mathrm{w}}] \in \mathbb{R}^L, \quad \widetilde{\boldsymbol{x}}_k = \sum_{n=1}^N \frac{\widetilde{r}_{kn}}{\widetilde{r}_k} \widetilde{\boldsymbol{x}}_{nk}, \quad \widetilde{\boldsymbol{y}}_k = \sum_{n=1}^N \frac{\widetilde{r}_{kn}}{\widetilde{r}_k} \boldsymbol{y}_n.$$

And

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

Practical setting

Algorithm initialization

No straightforward way of choosing $r_{nk}^{(0)}, \boldsymbol{\mu}_{nk}^{\mathrm{w}(0)}$, $\mathbf{S}_k^{\mathrm{w}(0)}$ or a complete set $\boldsymbol{\theta}^{(0)}$ including all affine transformations

$\longrightarrow$ Use one iteration of Marginal hGLLiM EM to get $\boldsymbol{\theta}^{(0)}$

Latent dimension L_w estimation

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

$\mathcal{L}$: observed-data log-likelihood

$\mathcal{D}(\widetilde{\boldsymbol{\theta}})$: dimension of the complete parameter vector

The Marginal Hybrid GLLiM-EM

$\boldsymbol{c}_k^{\mathrm{w}} = 0$ and $\boldsymbol{\Gamma}_k^{\mathrm{w}} = \mathbf{I}_{L_w} \Longrightarrow$

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

with $\boldsymbol{E}'_k \sim \mathcal{N}_D(0, \boldsymbol{\Sigma}_k + \mathbf{A}_k^{\mathrm{w}} \, \mathbf{A}_k^{\mathrm{w}\top})$

- No E-W step (marginalized)
- Same E-Z step ($\widetilde{r}_{nk}$ initialized via eg. K-means)
- Same M-GMM step $(\pi_k, \boldsymbol{c}_k^{\mathrm{t}}, \boldsymbol{\Gamma}_k^{\mathrm{t}})$

- M-regression step $(\mathbf{A}_k^{\mathrm{t}}, \boldsymbol{b}_k)$: standard, does not involve noise variance
- M-residual step $(\mathbf{A}_k^{\mathrm{w}}, \boldsymbol{\Sigma}_k)$: PPCA like on residuals $\boldsymbol{y}_n - \mathbf{A}_k^{\mathrm{t}} \boldsymbol{t}_n - \boldsymbol{b}_k$ (time consumming)

The Marginal Hybrid GLLiM M-step

M-regression-step: Weighted affine regression from $\{\boldsymbol{t}_n, n = 1:N\}$ to $\{\boldsymbol{y}_n, n = 1:N\}$ with weights $\widetilde{r}_{nk}$,

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

with

$$\widetilde{\mathbf{T}}_k = \left[\sqrt{\widetilde{r}_{1k}}(\boldsymbol{t}_1 - \widetilde{\boldsymbol{t}}_k) \dots \sqrt{\widetilde{r}_{Nk}}(\boldsymbol{t}_N - \widetilde{\boldsymbol{t}}_k) \right] \sqrt{\widetilde{r}_k}$$

and

$$\widetilde{\boldsymbol{t}}_k = \sum_{n=1}^{N} (\widetilde{r}_{kn} / \widetilde{r}_k) \boldsymbol{t}_n$$

M-residual-step: Minimization of the following criterion:

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

where $\boldsymbol{u}_{kn} = \sqrt{\widetilde{r}_{nk} / \widetilde{r}_k} (\boldsymbol{y}_n - \widetilde{\mathbf{A}}_k^{\mathrm{t}} \boldsymbol{t}_n - \widetilde{\boldsymbol{b}}_k)$.

Experiments and results

High dimensional function regression

$$\boldsymbol{\phi} = (\phi_1 \dots \phi_d \dots \phi_D)^\top$$

$$\boldsymbol{\phi} = \boldsymbol{f}, \boldsymbol{g}, \boldsymbol{h}$$

$\boldsymbol{f} : \mathbb{R}^2 \to \mathbb{R}^D$ with $f_d(t, w_1) = \alpha_d \cos(\eta_d \, t/10 + \phi_d) + \gamma_d w_1^3$

$\boldsymbol{g} : \mathbb{R}^2 \to \mathbb{R}^D$ with $g_d(t, w_1) = \alpha_d \cos(\eta_d \, t/10 + \beta_d w_1 + \phi_d)$

$\boldsymbol{h} : \mathbb{R}^3 \to \mathbb{R}^D$ with
$h_d(t, w_1, w_2) = \alpha_d \cos(\eta_d \, t/10 + \beta_d w_1 + \phi_d) + \gamma_d w_2^3$

$\boldsymbol{\xi} = \{\alpha_d, \eta_d, \phi_d, \beta_d, \gamma_d\}_{d=1}^D$ in $[0, 2]$, $[0, 4\pi]$, $[0, 2\pi]$, $[0, \pi]$, $[0, 2]$

High dimensional function regression

100 $\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{h}$ functions generated using different random values for $\boldsymbol{\xi}$

N training couples $\{(t_n, \boldsymbol{y}_n)\}_{n=1}^{N}$
N' test couples $\{(t'_n, \boldsymbol{y}'_n)\}_{n=1}^{N'}$

by randomly drawing $t \in [0, 10]$ and $\boldsymbol{w} \in [-1, 1]$ $(\boldsymbol{f}, \boldsymbol{g})$ or $\in [-1, 1]^2$ $(\boldsymbol{h})$

and adding some random isotropic Gaussian noise $\boldsymbol{y} = \boldsymbol{\phi}(t, \boldsymbol{w}) + \boldsymbol{e}$.

Training couples: train the different regression algorithms tested (h-GLLiM, SIR, RVM, MLE,JGMM)

Task: Estimate $\hat{t'}_n$ given a test observation $\boldsymbol{y}'_n = \boldsymbol{\phi}(t'_n, \boldsymbol{w}'_n) + \boldsymbol{e}'_n$

High dimensional function regression $D = 50$

Average, standard deviation and % of extreme values of the absolute error $|\hat{t}'_n - t'_n|$. $N = 200, N' = 200$, $K = 5$ (MLE, JGMM,hGLLiM)

MLE: $L_w = 0$, JGMM : $L_w > D$

	$\boldsymbol{f}$			$\boldsymbol{g}$			$\boldsymbol{h}$		
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**

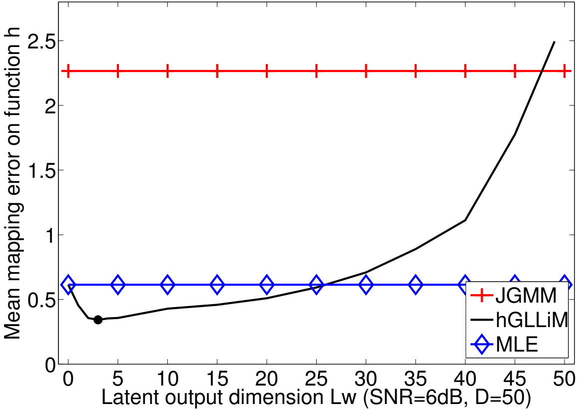
hGLLiM-BIC minimizes BIC for $0 < L_w < 10$: expected latent dimension L_w ($L_w = 2$ or $L_w = 1$) selected 72 times over 100 (non-linear effects could be modeled by higher L_w)

Influence of L_w

Influence of the parameter L_w of hGLLiM on the mean mapping error of $\boldsymbol{h}$.

Each point corresponds to an average error over $10,000$ tests on 50 distinct functions

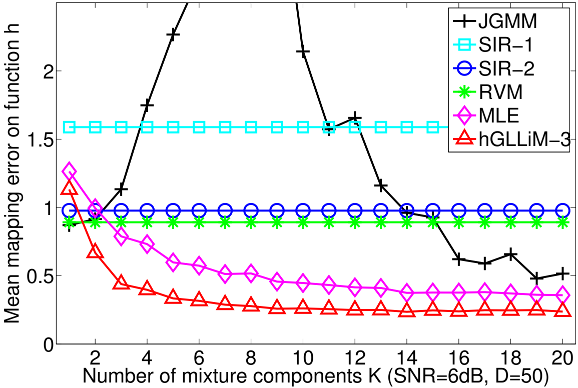
MLE: $L_w = 0$, JGMM : $L_w > D$



Influence of K

Influence of K in MLE, JGMM and hGLLiM-3 on the mean mapping error of synthetic function $\boldsymbol{h}$.

Each point corresponds to an average error over $10,000$ tests on 50 distinct functions

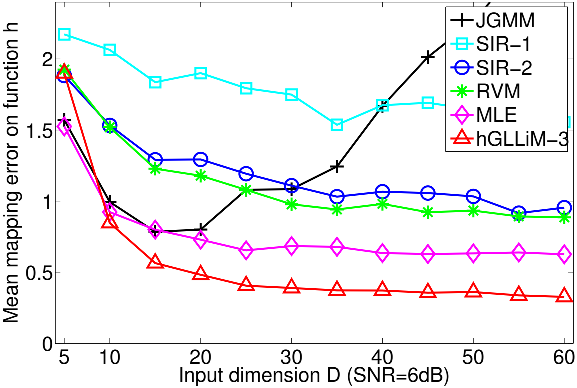


Errors generally decrease with K. Overfitting for $K > 10$ for JGMM

Influence of D

Influence of D on the mean mapping error of synthetic functions $\boldsymbol{h}$

Each point corresponds to an average error over $10,000$ tests on 50 distinct functions

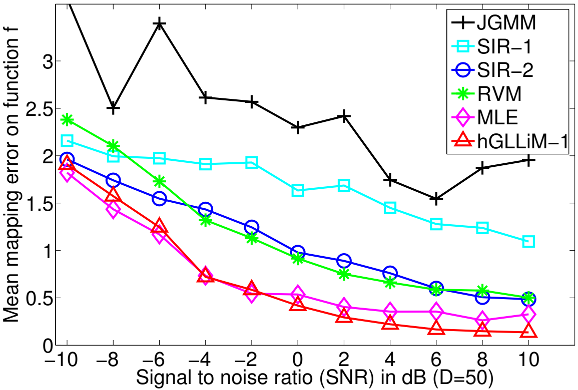


h-GLLiM performs better in high-dimension

Influence of the SNR

Influence of the signal-to-noise ration (SNR) on the mean mapping error of synthetic functions $\boldsymbol{f}$

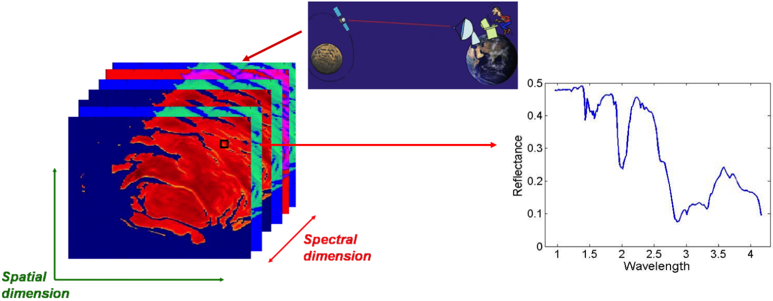
Each point corresponds to an average error over $10,000$ tests on 50 distinct functions



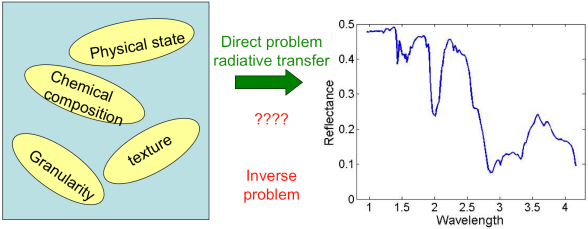
All methods perform similarly under extreme noise (SNR=-10dB) (except for JGMM)

Retrieval of Mars physical properties

Hyperspectral images



Radiative transfer model



Synthetic data

15,407 spectra ($D = 184$ wavelengths) and $L = 5$ real parameters

(proportion of water ice, of CO_2 ice, of dust, grain size of water ice, of CO_2 ice)

Proportion of water ice & grain size of CO_2 ice ignored from training

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	$\mathbf{0.34 \pm 0.63}$	$\mathbf{0.25 \pm 0.44}$	$\mathbf{0.39 \pm 0.71}$

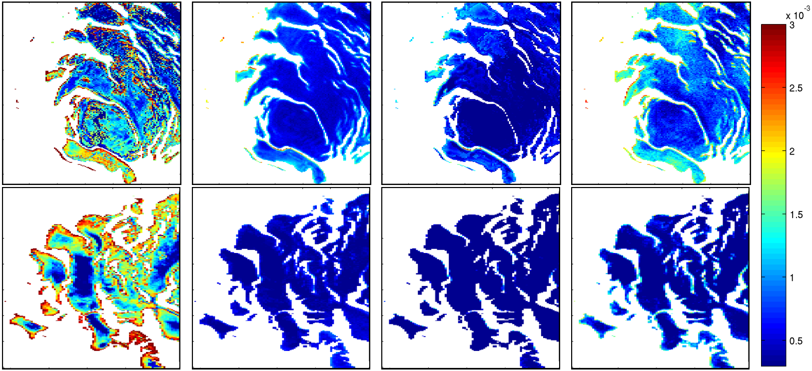
NRMSE for Mars surface physical properties recovered from synthetic spectra: cross validation with 10,000 training pairs at random and 5,407 test pairs ($\times 20$)

$K = 50$ for MLE, LGMM, hGLLiM

Hyperspectral images of South polar cap of Mars

Omega instrument, Mars Express

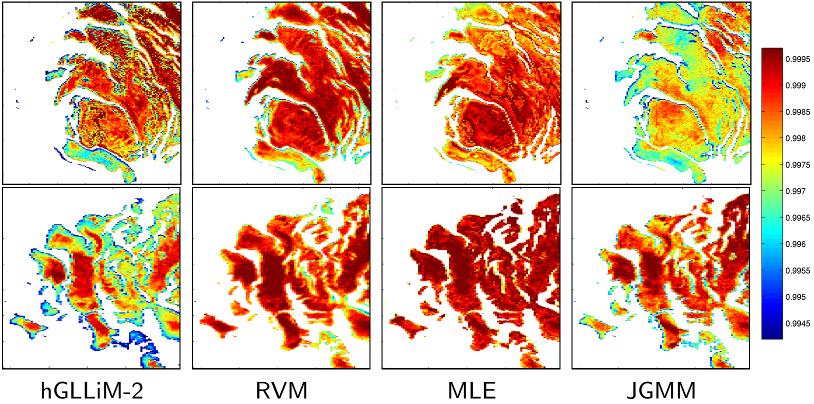
Proportions of dust for the South polar cap of Mars: orbits 41 and 61



hGLLiM-2 RVM MLE JGMM

Hyperspectral images of South polar cap of Mars

Proportions of CO_2 ice for the South polar cap of Mars: orbits 41 and orbit 61



Conclusion/ Perspectives

- We propose a novel inverse approach to high-dimensional regression based on mixture- and latent-variable models.
- Latent component allows to capture behaviors that cannot be easily modeled

- Adaptive latent dimension L_w selection
- More complex dependencies between variables (eg. $(Z_1 \ldots Z_N)$ is a MRF)
- More complex noise models, eg, Student for outliers accommodation and robustness

Matlab code available at: https://team.inria.fr/perception/gllim_toolbox/

A. Deleforge, F. Forbes and R. Horaud, High-Dimensional Regression with Gaussian Mixtures and Partially-Latent Response Variables. Statistics & Computing.

A. Deleforge, F. Forbes and R. Horaud, Hyper-spectral Image Analysis with Partially-Latent Regression. EUSIPCO, Lisbon, Portugal, September 2014.

MRF modelling