Sketching for Large-Scale Learning of Mixture Models

N. Keriven[⋆§] A. Bourrier[†] R. Gribonval[§] P. Pérez[‡]

[⋆] Université Rennes 1, France
[§] INRIA Rennes-Bretagne Atlantique, France
[†] Gipsa-Lab, St-Martin-d'Hères, France
[‡] Technicolor, Cesson Sévigné, France

École d'été Peyresq, Juillet 2016

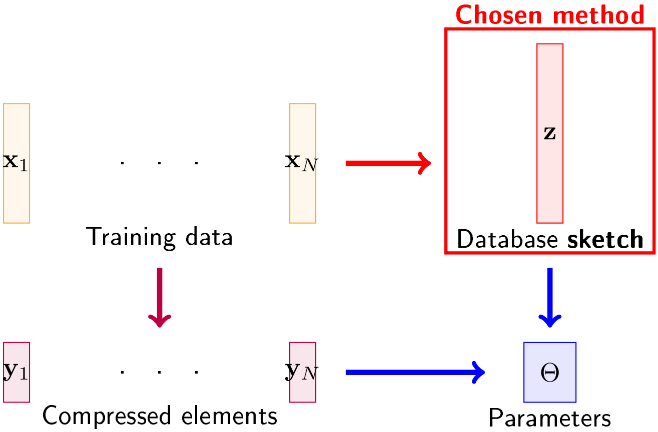
INVENTEURS DU MONDE NUMÉRIQUE

Outline

Paths to Compressive Learning

Objective

Fit density p_Θ on a **large** database $(\mathbf{x}_1, ..., \mathbf{x}_N) \in \mathbb{R}^n$.



Approach : Generalized Compressive Sensing

Traditional Compressive Sensing (CS)

From $\mathbf{y} \approx \mathbf{M}\mathbf{x} \in \mathbb{R}^m$ recover vector $\mathbf{x} \in \mathbb{R}^n$

- Linear $\mathbf{M} \in \mathbb{R}^{m \times n}$ with $m < n$
- Typical assumption: sparse signal $\mathbf{x} = \sum_{k \in \Gamma} x_k \mathbf{e}_k$.

Generalized Compressive Sensing

From $\mathbf{z} \approx \mathcal{A}p \in \mathbb{C}^m$ recover probability distribution $p \in \mathcal{P}$

Must define:

- Linear operator $\mathcal{A} : \mathcal{P} \mapsto \mathbb{C}^m$
- Generalized "sparsity": $p_{\Theta,\boldsymbol{\alpha}} = \sum_{k=1}^{K} \alpha_k p_{\boldsymbol{\theta}_k}$
  - Infinite/continuous dictionary !

Application to Compressive Learning

From theoretical Generalized CS...

$$p \xrightarrow{\mathcal{A}} \mathbf{z} = \mathcal{A}p \xrightarrow{Alg.} p_{\Theta,\boldsymbol{\alpha}}$$

...to practical Compressive Learning:

$$\hat{p} = \frac{1}{N}\sum_i \delta_{\mathbf{x}_i} \xrightarrow{\mathcal{A}} \hat{\mathbf{z}} = \mathcal{A}\hat{p} \xrightarrow{Alg.} p_{\hat{\Theta},\hat{\boldsymbol{\alpha}}}$$

where $(\mathbf{x}_1, ..., \mathbf{x}_N) \overset{i.i.d.}{\sim} p$.

Questions:

- Reconstruction algorithm ?
- Choice of sketching operator $\mathcal{A}$?
- Empirically/theoretically valid ?

Outline

Approach

$$p \xrightarrow{\mathcal{A}} \mathbf{z} = \mathcal{A}p \xrightarrow{\textbf{Alg.}} p_{\Theta,\boldsymbol{\alpha}}$$

Cost function

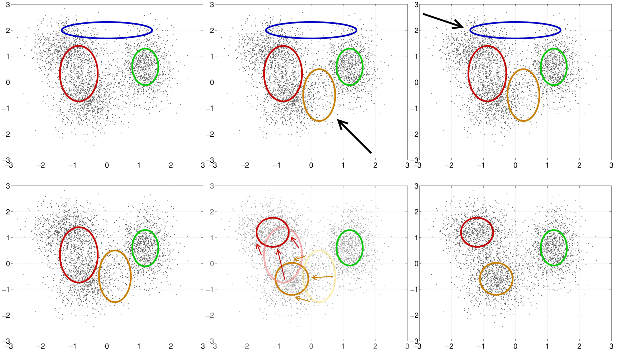
$$\min_{\Theta,\boldsymbol{\alpha}} \|\mathbf{z} - \mathcal{A}p_{\Theta,\boldsymbol{\alpha}}\|_2$$

- Similar to $\min_{\mathbf{x}:\|\mathbf{x}\|_0 \leq s} \|\mathbf{y} - \mathbf{M}\mathbf{x}\|_2$ in CS.

Need approximate algorithms !

Proposed Algorithm: quick overview

- Greedy : progressively add components $p_{\boldsymbol{\theta}_k}$
- Inspired by OMP, adapted to continuous settings
- Two versions
  - Compressive Learning OMP (CLOMP)
  - CLOMPR (with Replacement): slower but better results



What is left ?

$$p \xrightarrow{\mathcal{A}} \mathbf{z} = \mathcal{A}p \xrightarrow{CLOMP(R)} p_{\Theta,\boldsymbol{\alpha}} = \sum_k \alpha_k p_{\boldsymbol{\theta}_k}$$

To perform CLOMP(R), $\mathcal{A}p_{\boldsymbol{\theta}}$ and $\nabla_{\boldsymbol{\theta}}\mathcal{A}p_{\boldsymbol{\theta}}$ must have a closed-form expression.

- Here:
  - $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\sigma})$ and $p_{\boldsymbol{\theta}}$: GMMs with diagonal covariance
- Soon-to-be-released toolbox:
  - K-means
  - full GMMs
  - GLLiM *[Deleforge 2014]*
  - α-stable (in progress)
  - User-defined ! (black-box implementation)

Sketching operator

$$p \xrightarrow{\mathcal{A}} \mathbf{z} = \mathcal{A}p \xrightarrow{CLOMP(R)} p_{\Theta,\boldsymbol{\alpha}}$$

Random Sampling of the characteristic function *[Bourrier 2013]*

Given $(\boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_m) \in \mathbb{R}^n$,

$$\mathcal{A}p = \left[\mathbb{E}_{\mathbf{x}\sim p}(e^{\mathrm{i}\boldsymbol{\omega}^T\mathbf{x}})\right]_{j=1,...,m}$$

- Closed-form for many models !
- Analog to Random Fourier Sampling: $(\boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_m) \overset{i.i.d.}{\sim} \Lambda$
- $\hat{\mathbf{z}} = \left[\frac{1}{N}\sum_i e^{\mathrm{i}\boldsymbol{\omega}_j^T\mathbf{x}_i}\right]_{j=1,...,m}$ easily computable (distributed, GPU, streaming...)

Designing the frequency distribution

The frequency distribution must "scale" with (the variances of) the GMM.

Approach 1 Optimize the variance of a Gaussian frequency distribution

- Classical choice in kernel methods *[Sutherland 2015]*

Designing the frequency distribution

The frequency distribution must "scale" with (the variances of) the GMM.

Approach 1 Optimize the variance of a Gaussian frequency distribution

Approach 2 Proposed:

- Partial preprocessing to compute the appropriate "scaling"

The proposed distribution

- Yields better precision in the reconstruction
- Is $20\times$ to $100\times$ faster to design

To summarize

$$\hat{p} \xrightarrow{\mathcal{X}\to\Lambda\to\mathcal{A}} \hat{\mathbf{z}} = \mathcal{A}\hat{p} \xrightarrow{CLOMP(R)} p_{\Theta,\boldsymbol{\alpha}}$$

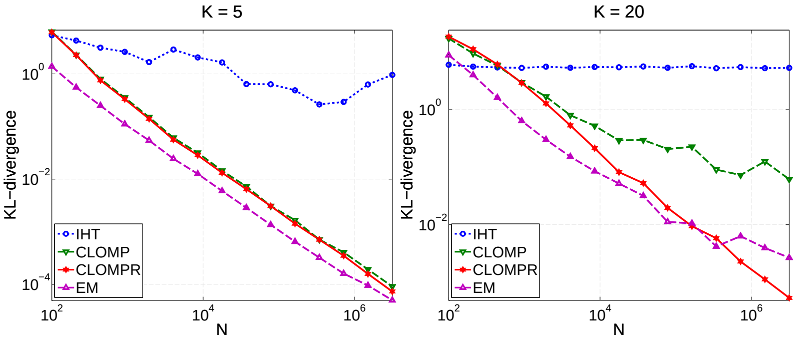
Given a database $\mathcal{X} = (\mathbf{x}_1, ..., \mathbf{x}_N) \in \mathbb{R}^n$, m, K:

- Design $\mathcal{A}$
  - Partial preprocessing to choose the frequency distribution Λ
  - Draw m frequencies $(\boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_m) \in \mathbb{R}^n$
- Compute $\hat{\mathbf{z}} = \frac{1}{\sqrt{m}} \left[\frac{1}{N} \sum_i e^{\mathrm{i}\boldsymbol{\omega}_j^T \mathbf{x}_i} \right]_{j=1,...,m}$
  - GPU, distributed computing, etc.
- Estimate a K-GMM $p_{\Theta,\boldsymbol{\alpha}}$ from $\hat{\mathbf{z}}$ using CLOMP(R).

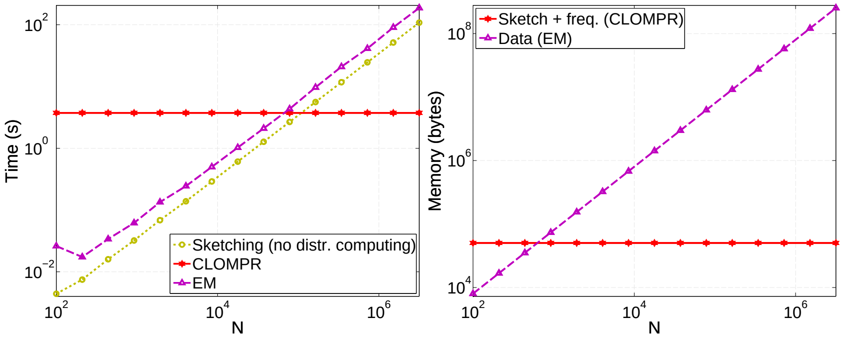
Outline

Reconstruction results

Comparison with EM (VLFeat toolbox) and previous Compressive Learning IHT *[Bourrier 2013]*. KL-div (lower is better), $n = 10$, $m = 5(2n+1)K$.



Memory usage and computation time



- Remember : Sketching easily done on GPU/cluster

Proof of concept : speaker verification

- *NIST2005 database with MFCCs:* $N = 2 \cdot 10^8$

- A large database indeed enhances the results
- Limitations are observed for large K : difficult "sparse approximation" task of a non-sparse distribution

Outline

Information preservation guarantees ?

$$p \xrightarrow{\mathcal{A}} \mathbf{z} = \mathcal{A}p \xrightarrow{CLOMP(R)} p_{\Theta,\boldsymbol{\alpha}}$$

- CLOMP(R) attempts to solve $\min_{\Theta,\boldsymbol{\alpha}} \|\mathbf{z} - \mathcal{A}p_{\Theta,\boldsymbol{\alpha}}\|_2$
  - Difficult to obtain guarantees for CLOMP(R): non-convex, random...

Information preservation guarantees ?

$$p \xrightarrow{\mathcal{A}} \mathbf{z} = \mathcal{A}p \xrightarrow{CLOMP(R)} p_{\Theta,\boldsymbol{\alpha}}$$

- CLOMP(R) attempts to solve $\min_{\Theta,\boldsymbol{\alpha}} \|\mathbf{z} - \mathcal{A}p_{\Theta,\boldsymbol{\alpha}}\|_2$
  - Difficult to obtain guarantees for CLOMP(R): non-convex, random...
- More fundamentally: if we **were** able to **exactly** solve

$$\min_{p\in\Sigma} \|\mathbf{z} - \mathcal{A}p\|_2,$$

with Σ "low-dimensional" set of distribution (e.g. K-sparse GMMs), do we have any guarantee ?

Information preservation guarantees ?

$$p \xrightarrow{\mathcal{A}} \mathbf{z} = \mathcal{A}p \xrightarrow{\text{Best algo. possible}} \bar{p} \in \arg\min_{p\in\Sigma} \|\mathbf{z} - \mathcal{A}p\|_2$$

- Does $\mathbf{z}$ contains "enough" information to recover $p \in \Sigma$?

Information preservation guarantees ?

$$p \xrightarrow{\mathcal{A}} \mathbf{z} = \mathcal{A}p \xrightarrow{\text{Best algo. possible}} \bar{p} \in \arg\min_{p \in \Sigma} \|\mathbf{z} - \mathcal{A}p\|_2$$

- Does $\mathbf{z}$ contains "enough" information to recover $p \in \Sigma$?
- Is it stable if $p \notin \Sigma$?

Information preservation guarantees ?

$$p \xrightarrow{\mathcal{A}} \mathbf{z} = \mathcal{A}p \xrightarrow{\text{Best algo. possible}} \bar{p} \in \arg\min_{p\in\Sigma} \|\mathbf{z} - \mathcal{A}p\|_2$$

- Does $\mathbf{z}$ contains "enough" information to recover $p \in \Sigma$?
- Is it stable if $p \notin \Sigma$?
- Is it stable to use $\hat{\mathbf{z}}$ instead of $\mathbf{z}$?

Information preservation guarantees ? Yes !

$$\hat{p} \xrightarrow{\mathcal{A}} \hat{\mathbf{z}} = \mathcal{A}\hat{p} \xrightarrow{\text{Best algo. possible}} \bar{p} \in \arg\min_{p \in \Sigma} \|\hat{\mathbf{z}} - \mathcal{A}p\|_2$$

Main result

(for a compact Σ, under some hypothesis on Λ)

- W.h.p. on $(\mathbf{x}_1, ..., \mathbf{x}_N) \overset{i.i.d.}{\sim} p^*$ and $(\boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_m) \overset{i.i.d.}{\sim} \Lambda$,

$$\gamma_\Lambda(p^*, \bar{p}) \leq 5 d_{TV}(p^*, \Sigma) + \mathcal{O}\left(N^{-\frac{1}{2}}\right) + \eta,$$

  - γ_Λ "kernel" metric *[Sriperumbudur 2010]*
  - d_{TV} total variation distance between p^* and the model Σ
  - η additive error in m

Application to GMMs with compact set of parameters.

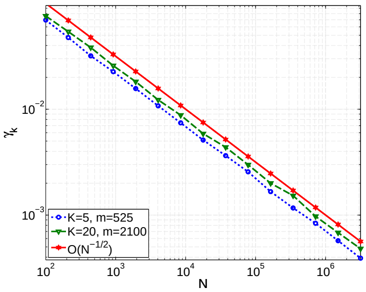
- $K = 1$ (toy):
  - $\eta = \mathcal{O}(\beta^{-m})$: Good !

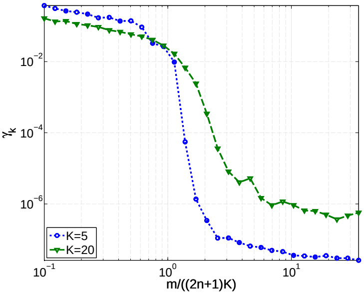
Application to GMMs with compact set of parameters.

- $K = 1$ (toy):
  - $\eta = \mathcal{O}(\beta^{-m})$: Good !
- $K \geq 2$:
  - $\eta = \mathcal{O}\left(m^{-\frac{1}{2}}\right)$: Worst possible !
  - Global error in $\mathcal{O}\left(N^{-\frac{1}{2}} + m^{-\frac{1}{2}}\right)$: "compressive" approach ?

Application to GMMs with compact set of parameters.

- $K = 1$ (toy):
  - $\eta = \mathcal{O}(\beta^{-m})$: Good !
- $K \geq 2$:
  - $\eta = \mathcal{O}\left(m^{-\frac{1}{2}}\right)$: Worst possible !
  - Global error in $\mathcal{O}\left(N^{-\frac{1}{2}} + m^{-\frac{1}{2}}\right)$: "compressive" approach ?
  - **Conjecture**: it is in fact much better !





Recent results (unpublished yet...)

- $\eta = \mathcal{O}(\beta^{-m})$ for K-GMMs with fixed known Σ and $\|\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k'}\|_2 \geq \mathcal{O}(\ln k)$
  - May need more layers for unknown Σ ("sketching the sketches...") : CNN !
- Can relate the "kernel" metric γ_Λ to traditional excess risk in Machine Learning !

Outline

Conclusion

Summary

Effective method to learn GMMs from a sketch, using greedy algorithms and an efficient heuristic to design the sketching operator. Empirical and theoretical motivations.

More...

- Faster algorithm for GMM with large K
- More on theoretical guarantees

Future Work

- Application to other Mixture Models (α-stable...)
- Generalized theoretical guarantees
- Application to other kernel methods *[Sutherland 2015]* (classification...)

Questions ?

Keriven et al., **Sketching for Large-Scale Learning of Mixture Models**, *ICASSP 2016*

Keriven et al., **Sketching for Large-Scale Learning of Mixture Models**, *arXiv:1606.02838*

Soon : sketching toolbox