Sketching for Large-Scale Learning of Mixture Models

Nicolas Keriven, Anthony Bourrier, Rémi Gribonval, Patrick Perez

SKETCHING FOR LARGE-SCALE LEARNING OF MIXTURE MODELS

Nicolas Keriven[⋆] *Anthony Bourrier*[†] *Rémi Gribonval*[⋆] *Patrick Pérez*[‡]

[⋆] INRIA Rennes-Bretagne Atlantique, Campus de Beaulieu, 35042 Rennes, France
[†] Gipsa-Lab, 11 Rue des Mathématiques, 38400 St-Martin-d'Hères, France
[‡] Technicolor, 975 Avenue des Champs Blancs, 35576 Cesson Sévigné, France

ABSTRACT

Learning parameters from voluminous data can be prohibitive in terms of memory and computational requirements. We propose a "compressive learning" framework where we first *sketch* the data by computing random generalized moments of the underlying probability distribution, then estimate mixture model parameters from the sketch using an iterative algorithm analogous to greedy sparse signal recovery. We exemplify our framework with the sketched estimation of Gaussian Mixture Models (GMMs). We experimentally show that our approach yields results comparable to the classical Expectation-Maximization (EM) technique while requiring significantly less memory and fewer computations when the number of database elements is large. We report large-scale experiments in speaker verification, where our approach makes it possible to fully exploit a corpus of 1000 hours of speech signal to learn a universal background model at scales computationally inaccessible to EM.

Index Terms— Gaussian mixture models, compressive sensing, database sketch, compressive learning.

1. INTRODUCTION

Learning from large scale data is an essential challenge in data analysis [1]. In this paper, we propose to compress data –prior to learning– into a representation, called *sketch*, whose dimension does not depend on the number of data elements. This compression is reminiscent of compressive sensing (CS) [2], where one seeks a dimensionality-reducing linear operator $\mathbf{M}$ such that certain signals can be reconstructed from their observations by $\mathbf{M}$.

Although initially stated for sparse vectors, CS has been considered for a variety of models which are often unions of low-dimensional subspaces [3]. Such models also intervene in learning procedures: a standard example is *mixture models* which comprise distributions of the form $\sum_{k=1}^{K} \alpha_k p_k$, where the p_k's are probability measures taken in a certain set and the α_k's are the weights of the mixture. This mixture model Σ is therefore included in a constrained subset of a union of K-dimensional subspaces in the space E of signed finite measures over a set X.

Similarly to compressive sensing, one can define a linear compressive operator $\mathcal{A} : E \to \mathbb{C}^m$ which computes *generalized moments* of a measure $p \in E$:

$$\mathcal{A} : p \mapsto \left[\int_X M_1 dp, \dots, \int_X M_m dp \right], \tag{1}$$

with M_j's some well-chosen functions on X. When p is a probability measure, the integrals are simply expectations of $M_j(\mathbf{x})$ with $\mathbf{x} \sim p$ and can be approximated given training data $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\} \overset{i.i.d.}{\sim} p$, yielding an approximation of $\mathcal{A}p$ by a *data sketch* $\hat{\mathbf{z}}$ typically computed as $\hat{\mathbf{z}} = \mathcal{A}\hat{p}$, where $\hat{p} = \frac{1}{N}\sum_{i=1}^{N} \delta_{\mathbf{x}_i}$ is the empirical distribution of the data and $\delta_{\mathbf{x}_i}$ is a unit mass at $\mathbf{x}_i$. Searching for the probability mixture in the model Σ whose sketch is closest to the data sketch $\hat{\mathbf{z}}$ yields a moment matching problem:

$$\operatorname*{argmin}_{q \in \Sigma} \|\hat{\mathbf{z}} - \mathcal{A}q\|. \tag{2}$$

By analogy with sparse reconstruction, we design an iterative algorithm targeting the solution of this problem, described in Section 3. The compressive learning framework is instantiated by designing sketches adapted to Gaussian Mixture Models (GMMs) in Section 4. Numerical experiments are described in Section 5.

2. RELATED WORKS

Mixture model estimation considered as a linear inverse problem has been investigated in [4, 5]. Theoretical guarantees are provided in the case of finite and incoherent sets of densities, which do not apply to continuously indexed models like GMMs. In [6], the authors considered GMM estimation via sketching, applied to isotropic Gaussians with fixed known variance. Here we consider unknown diagonal covariances, and propose a different algorithm and a modified sketching operator that prove to be more stable and robust.

Sketching is a classical technique in the database literature [7, 8]. Closer to machine learning, random linear sketches have been considered [9] for histogram estimation in dimension 2. However, this method suffers from the curse of dimensionality as the number of histogram bins increases exponentially with the data dimension.

Some approaches to compressive learning compress *each element* of the database with random projections [10] prior to learning, for classification [11] or regression [12]. Here, the *whole data collection* is compressed to a fixed-size sketch independent of the number of items in the collection, leading to substantial memory savings.

The formulation (2) that we consider is a special case of the Generalized Method of Moments (GeMM) [13]. This method is typically used in the estimation of probability models without explicit likelihoods, while we consider here the collection of moments as a *voluntarily compressed* representation of the training data.

3. RECONSTRUCTION ALGORITHM

3.1. Notations and analogy with compressive sensing

Let E be the space of signed finite measures over a measurable space $(X, \mathcal{B})$, and $\mathcal{P} = \left\{p \in E; p \geq 0, \int_X dp = 1\right\}$ the set of probability measures over X. We consider a set of distributions $\mathcal{G} = \{p_{\boldsymbol{\theta}} \in \mathcal{P}; \boldsymbol{\theta} \in \mathcal{T}\}$ indexed by a parameter $\boldsymbol{\theta} \in \mathcal{T}$. For $K \in \mathbb{N}^*$,

This work was supported in part by the European Research Council, PLEASE project (ERC-StG- 2011-277906).

a distribution $p \in \mathcal{P}$ is said *K-sparse* if it satisfies $p = p_{\Theta,\boldsymbol{\alpha}} := \sum_{k=1}^{K} \alpha_k p_{\boldsymbol{\theta}_k}$, with $p_{\boldsymbol{\theta}_k} \in \mathcal{G}$ and $\alpha_k \geq 0$ for all k and $\sum_{k=1}^{K} \alpha_k = 1$. If such a decomposition of p is unique, $\Theta = \{\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_K\}$ and $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K)$ can be referred to as the *support* and *weights* of p. In the case of GMM estimation, which is considered in Section 4, there is indeed uniqueness of this decomposition.

Given a linear *sketching operator* $\mathcal{A} : E \to \mathbb{C}^m$, the sketch $\mathbf{z} = \sum_{k=1}^{K} \alpha_k \mathcal{A} p_{\boldsymbol{\theta}_k}$ is a limited combination of *atoms* selected from the *dictionary* $\{\mathcal{A} p_{\boldsymbol{\theta}}; \boldsymbol{\theta} \in \mathcal{T}\}$.

3.2. Orthogonal Matching Pursuit with Replacement

In the case of sparse decomposition, the problem (2) is expressed as the (typically highly nonconvex) optimization problem

$$\min_{\Theta,\boldsymbol{\alpha}} \|\hat{\mathbf{z}} - \mathcal{A} p_{\Theta,\boldsymbol{\alpha}}\|_2^2, \tag{3}$$

which has been proven to yield instance optimal decoders for general models in CS under certain hypotheses [3, 14]. However, the minimization (3) may not allow for an efficient direct optimization.

As a heuristic, we propose a greedy approach inspired by Orthogonal Matching Pursuit (OMP) [15] and its variant OMP with Replacement (OMPR) [16], which iteratively extend the support by choosing at each iteration the atom most correlated with the residual. OMPR runs with more iterations than OMP (typically $2K$ instead of K): in the spirit of CoSAMP [2], it extends the support *further than* the desired sparsity before enforcing it at each iteration with a Hard Thresholding step.

As detailed below, Algorithm 1 involves several modifications to OMPR to handle the generalized framework considered here.

Non-negativity. The compressive mixture estimation framework imposes a non-negativity constraint on the weights $\boldsymbol{\alpha}$. Thus **step 1** maximizes the real part of the correlation instead of its modulus, to avoid negative correlation between atom and residual. Similarly, in **step 4** we perform a Non-Negative Least-Squares (NNLS) minimization instead of a classical Least-Squares minimization.

Continuous dictionary. The space of atoms is often continuously indexed and cannot be exhaustively searched. We therefore perform the maximization in **step 1** with a gradient ascent $\texttt{maximize}_{\boldsymbol{\theta}}$ randomly initialized, leading to a –local– maximum of the correlation between atom and residual. Note that the atoms are normalized during the search, as is often the case with OMP.

We also *add* **step 5**, to further reduce the cost function with a gradient descent $\texttt{minimize}_{\Theta,\boldsymbol{\alpha}}$ initialized with the current parameters $(\Theta, \boldsymbol{\alpha})$. In practice, the need for this additional step stems from the lack of incoherence between the elements of the uncountable dictionary: when a new atom is introduced, all previously selected atoms may need to be adjusted.

Overall, similar to classical OMPR, we derive two algorithms from Algorithm 1, depending on the number of iterations:

- ***Compressive Learning OMP*** (CLOMP) if run with $T = K$ iterations (i.e. *without* Hard Thresholding);
- ***CLOMP with Replacement*** (CLOMPR) if run with $T = 2K$ iterations.

Algorithm 1 is suitable for any sketching operator $\mathcal{A}$ and any mixture model of parametric distributions $p_{\boldsymbol{\theta}}$, as long as the optimization schemes in steps 1 and 5 can be performed. In the case of a continuously indexed dictionary like with GMMs, efficient implementation can only be conducted if $\mathcal{A} p_{\boldsymbol{\theta}}$ and its gradient with respect to $\boldsymbol{\theta}$ have closed form expressions.

Algorithm 1: Basic algorithm for CLOMP (if $T = K$) and CLOMPR (if $T = 2K$)

Data: Sketch $\hat{\mathbf{z}}$, sketching operator $\mathcal{A}$, parameters $K, T \geq K$
Result: Support Θ, weights $\boldsymbol{\alpha}$
$\hat{\mathbf{r}} \leftarrow \hat{\mathbf{z}}$; $\Theta \leftarrow \emptyset$;
for $t \leftarrow 1$ **to** T **do**
  Step 1: Find a normalized atom highly correlated with the residual
    $\boldsymbol{\theta} \leftarrow$
    $\texttt{maximize}_{\boldsymbol{\theta}} \left(\mathrm{Re} \left\langle \frac{\mathcal{A} p_{\boldsymbol{\theta}}}{\|\mathcal{A} p_{\boldsymbol{\theta}}\|_2}, \hat{\mathbf{r}} \right\rangle_2, \texttt{init} = \texttt{rand} \right)$
  end
  Step 2: Expand support
    $\Theta \leftarrow \Theta \cup \{\boldsymbol{\theta}\}$
  end
  Step 3: Enforce sparsity by Hard Thresholding if needed
    if $|\Theta| > K$ **then**
      $\boldsymbol{\beta} \leftarrow \arg\min_{\boldsymbol{\beta} \geq 0} \left\| \hat{\mathbf{z}} - \sum_{k=1}^{|\Theta|} \beta_k \frac{\mathcal{A} p_{\boldsymbol{\theta}_k}}{\|\mathcal{A} p_{\boldsymbol{\theta}_k}\|_2} \right\|_2$
      Select K largest entries $\beta_{i_1}, ..., \beta_{i_K}$
      Reduce the support $\Theta \leftarrow \{\boldsymbol{\theta}_{i_1}, ..., \boldsymbol{\theta}_{i_K}\}$
    end
  end
  Step 4: Project to find $\boldsymbol{\alpha}$
    $\boldsymbol{\alpha} \leftarrow \arg\min_{\boldsymbol{\alpha} \geq 0} \left\| \hat{\mathbf{z}} - \sum_{k=1}^{|\Theta|} \alpha_k \mathcal{A} p_{\boldsymbol{\theta}_k} \right\|_2$
  end
  Step 5: Perform a gradient descent *initialized with current parameters*
    $\Theta, \boldsymbol{\alpha} \leftarrow \texttt{minimize}_{\Theta,\boldsymbol{\alpha}} \Big(\left\| \hat{\mathbf{z}} - \sum_{k=1}^{|\Theta|} \alpha_k \mathcal{A} p_{\boldsymbol{\theta}_k} \right\|_2,$
      $\texttt{init} = (\Theta, \boldsymbol{\alpha}), \texttt{constraint} = \{\boldsymbol{\alpha} \geq 0\} \Big)$
  end
  Update residual: $\hat{\mathbf{r}} \leftarrow \hat{\mathbf{z}} - \sum_{k=1}^{|\Theta|} \alpha_k \mathcal{A} p_{\boldsymbol{\theta}_k}$
end
Normalize $\boldsymbol{\alpha}$ such that $\sum_{k=1}^{K} \alpha_k = 1$

4. SKETCHING GAUSSIAN MIXTURES

When considering Gaussian Mixture Models (GMM), the basic distributions $p_{\boldsymbol{\theta}}$ are Gaussian densities on $X = \mathbb{R}^n$:

$$p_{\boldsymbol{\theta}}(\mathbf{x}) = \frac{1}{(2\pi)^{n/2} |\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right) \tag{4}$$

where $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$ represents the parameters of the Gaussian with mean $\boldsymbol{\mu} \in \mathbb{R}^n$ and covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{n \times n}$.

A K-GMM is then naturally parametrized by the weights $\boldsymbol{\alpha} \in \mathbb{R}^K$ and parameters $\boldsymbol{\theta}_k = (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$. In this work, we only consider Gaussians with diagonal covariances, which is known to be sufficient for many applications [17], and denote $\boldsymbol{\Sigma}_k = \mathrm{diag}(\sigma_{k,1}^2, ..., \sigma_{k,n}^2)$.

4.1. Sketching by sampling the characteristic function

Gaussians, as well as their sparse mixtures, are somewhat spatially localized. When their variances are small they even approximate well Dirac masses. Inspired by Random Fourier Sampling [18], which is known to be adapted for compressive sensing of combinations of Diracs, one can thus design sketching operators by sampling the characteristic function of the distribution p [6]. This function has

a closed form expression for GMMs, from which the gradients of the cost functions in steps 1 and 5 of Algorithm 1 can be easily derived.

For a frequency $\boldsymbol{\omega} \in \mathbb{R}^n$, we denote $\psi_p(\boldsymbol{\omega}) := \mathbb{E}_{\mathbf{x}\sim p}\left(e^{\mathrm{i}\boldsymbol{\omega}^T\mathbf{x}}\right)$ the characteristic function of a distribution $p \in \mathcal{P}$. Given frequencies $\{\boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_m\}$ in $\mathbb{R}^n$, the generalized moment functions (1) are therefore $M_j(\mathbf{x}) = e^{\mathrm{i}\boldsymbol{\omega}_j^T\mathbf{x}}$ corresponding to a sketching operator $\mathcal{A}p = [\psi_p(\boldsymbol{\omega}_j)]_{j=1,...,m}$. Given a data collection $\mathcal{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ in $\mathbb{R}^n$, the empirical sketch is $\hat{\mathbf{z}} = \mathcal{A}\hat{p} = \left[\frac{1}{N}\sum_{i=1}^N e^{\mathrm{i}\boldsymbol{\omega}_j^T\mathbf{x}_i}\right]_{j=1,...,m} \approx \mathcal{A}p$.

4.2. Designing the frequency sampling pattern

To fully specify the sketching operator $\mathcal{A}$, we need to indicate how the frequencies $\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_m$ are chosen.

In the spirit of Random Fourier Sampling, they are drawn at random, $(\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_m) \overset{i.i.d.}{\sim} \Lambda$, according to some probability distribution Λ. We consider three heuristic distributions to sketch a single Gaussian $p_{\boldsymbol{\theta}} = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ of known parameters $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. We will deal in due time with mixtures, and with unknown parameters.

Gaussian distribution. The characteristic function of a Gaussian distribution $p_{\boldsymbol{\theta}}$ is $\psi_{\boldsymbol{\theta}}(\boldsymbol{\omega}) = \exp\left(\mathrm{i}\boldsymbol{\omega}^T\boldsymbol{\mu}\right)\exp\left(-\frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\Sigma}\boldsymbol{\omega}\right)$. Since its modulus is $\exp\left(-\frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\Sigma}\boldsymbol{\omega}\right)$ an intuitive choice could be to directly use this Gaussian shape as a frequency distribution [6]:

$$\Lambda_{\boldsymbol{\Sigma}}^{(G)} = \mathcal{N}\left(0, \boldsymbol{\Sigma}^{-1}\right).$$

However, as the dimension n increases, this choice "undersamples" low frequencies: for instance, if $\boldsymbol{\Sigma} = \mathbf{I}$, then $\|\boldsymbol{\omega}\|_2^2$ follows a χ^2 distribution with n degrees of freedom which quickly concentrates on a sphere of radius growing with n [19]. The amplitude $|\psi_{\boldsymbol{\theta}}(\boldsymbol{\omega})| = e^{-\frac{1}{2}\|\boldsymbol{\omega}\|_2^2}$ becomes negligible at all selected frequencies.

Gaussian radius distribution. To better control the amplitude of $\psi_{\boldsymbol{\theta}}$, one can choose $\boldsymbol{\omega}$ as:

$$\boldsymbol{\omega} = \boldsymbol{\Sigma}^{-\frac{1}{2}}\boldsymbol{\varphi}R, \tag{5}$$

where $\boldsymbol{\varphi}$ is uniformly drawn on the unit sphere $\mathcal{S}_{n-1}$, and $R \in \mathbb{R}_+$ is a random radius chosen independently. We now have $|\psi_{\boldsymbol{\theta}}(\boldsymbol{\omega})| = e^{-\frac{1}{2}R^2}$, which may suggest choosing R with respect to $\mathcal{N}^+(0,1)$ (i.e. Gaussian with absolute value, referred to as *folded* Gaussian). The decomposition (5) then yields a frequency distribution $\Lambda_{\boldsymbol{\Sigma}}^{(FGr)}$ referred to as *Folded Gaussian radius* frequency distribution. Though we will see in experiments that it yields decent results, this choice produces too many frequencies with low radius R, which carry a limited quantity of information about the original distribution since all characteristic functions equal 1 at the origin.

Adapted radius distribution. Intuitively, the selected frequencies should properly discriminate Gaussians with different parameters, which suggests drawing more frequencies where the norm of the gradient $\|\nabla_{\boldsymbol{\theta}}\psi_{\boldsymbol{\theta}}\|$ is maximal. After some simple computations that we do not detail here, we obtain a radius density function that follows this heuristic:

$$p(R) \propto \left(R^2 + \frac{R^4}{4}\right)^{\frac{1}{2}} e^{-\frac{1}{2}R^2}. \tag{6}$$

Using this distribution of R with the decomposition (5) yields a distribution $\Lambda_{\boldsymbol{\Sigma}}^{(Ar)}$ referred to as *Adapted radius* frequency distribution.

These frequency distributions selected for estimating a single Gaussian can be naturally extended to the case of a mixture $p_{\Theta,\boldsymbol{\alpha}}$ by mixing the frequency distributions corresponding to each Gaussian (still supposing the parameters $(\Theta, \boldsymbol{\alpha})$ are known):

$$\Lambda_{\Theta,\boldsymbol{\alpha}}^{(\cdot)} = \sum_{k=1}^K \alpha_k \Lambda_{\boldsymbol{\Sigma}_k}^{(\cdot)}. \tag{7}$$

4.3. Practical choice of frequencies

In practice, even if $\mathcal{X}$ is actually drawn from a GMM, one has no immediate access to its parameters $(\Theta, \boldsymbol{\alpha})$. The proposed approach consists in using a first partial pass on the dataset $\mathcal{X}$ to estimate some of its characteristics that will drive the selection of the frequency distribution used to sketch it.

The idea is to estimate the average variance $\bar{\sigma}^2$ of the components in the GMM – note that this parameter may be significantly different from the global variance of the data, for instance in the case of well-separated components with small variances. Since the characteristic function of the GMM has an amplitude which *approximately* follows $e^{-\frac{1}{2}\bar{\sigma}^2\|\boldsymbol{\omega}\|_2^2}$, we use a first sketch computed on a small subset of $N_0 \ll N$ items from the database, then perform a simple regression to estimate $\bar{\sigma}^2$. From there, a frequency distribution corresponding to a single isotropic Gaussian $\Lambda_{\bar{\sigma}^2\mathbf{I}}^{(\cdot)}$ (choosing one of the three possibilities described in the previous section) is selected to compute the final sketch.

This two-stage approach can be related to a line of work referred to as adaptive (or *distilled*) sensing [20], in which a portion of the computational budget is used to crudely design the measurement operator while the rest is used to actually measure the signal. To obtain a method that strictly requires *one pass* over the data collection for the whole estimation, the remaining $N - N_0$ samples are used to compute the final sketch.

5. EXPERIMENTS

5.1. Experiments on synthetic data

The compressive method is extensively tested against synthetic data. For each experiments, a K-GMM is generated by drawing the means $\boldsymbol{\mu}_k \in \mathbb{R}^n$ $i.i.d.$ from a Gaussian $\mathcal{N}\left(0, K^{\frac{1}{n}}\mathbf{I}\right)$ (so that the expected volume of a ball containing them is K times that of a single Gaussian with identity covariance) and the variances $\sigma_{k,i}^2 \in \mathbb{R}_+^*$ $i.i.d.$ uniformly between 0.25 and 1.75. We use the VLFeat toolbox [21] to perform the Expectation-Maximization (EM) algorithm for GMM estimation with diagonal covariances. Reconstruction performance is evaluated using a symmetric version [6] of the classical KL-divergence. The results are obtained by taking the mean of the logarithm of the KL-divergence over 40 experiments.

The results presented in Table 1 show that the Gaussian frequency distribution indeed yields poor reconstruction results in high dimension ($n = 20$), while the Adapted radius frequency distribution outperforms the Folded Gaussian radius. The use of the approximate $\Lambda_{\bar{\sigma}^2\mathbf{I}}^{(\cdot)}$ instead of the ideal $\Lambda_{\Theta,\boldsymbol{\alpha}}^{(\cdot)}$ is shown to have little effect, especially for the Adapted radius distribution. All following experiments are performed with an Adapted radius distribution $\Lambda_{\bar{\sigma}^2\mathbf{I}}^{(Ar)}$.

Reconstruction precision. In Fig. 1, we compare reconstruction results with respect to the database size N, for EM and three compressive algorithms: the IHT algorithm used in [6] (adapted to non-isotropic Gaussians with unknown variances), CLOMP and CLOMPR. With few Gaussians $K = 5$, both CLOMP and CLOMPR yield results close to the precision achieved by EM. With more Gaussians $K = 20$, CLOMPR clearly outperforms CLOMP

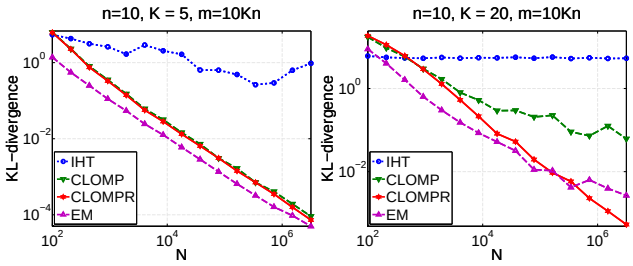

Fig. 1. KL-divergence with respect to the number of samples N.

	(G)	(FGr)	(Ar)
$\Lambda^{(.)}_{\Theta,\boldsymbol{\alpha}}$	7.26	0.116	**0.022**
$\Lambda^{(.)}_{\sigma^2\mathbf{I}}$	8.73	0.595	**0.026**

Table 1. KL-divergence results on synthetic data for $n = 20$, $K = 10$ and $m = 1000$, using either the exact or approximate distribution, and the three frequency distributions: Gaussian [6] (G), Folded Gaussian radius (FGr) or Adapted radius (Ar).

and indisputably matches EM at large N. The IHT algorithm is here often observed to converge to an undesired local minimum in which all Gaussians in the GMM are equal.

Computation time and memory usage. In Fig. 2, computation time and memory usage for CLOMPR and EM are presented with respect to the database size N, using an *Intel Core i7-4600U 2.1 GHz* CPU with 8 GB of RAM. In terms of time complexity (resp. memory usage), the EM algorithm scales in $\mathcal{O}(nKNT)$ for a fixed number of iterations T (resp. $\mathcal{O}(Nn)$), while CLOMPR scales in $\mathcal{O}(mnK^2)$ (resp. $\mathcal{O}(mn)$). The computation of the sketch, which scales in $\mathcal{O}(nmN)$, may seem a heavy operation since it requires performing dot products between each vector $\mathbf{x}_i$ and each frequency $\boldsymbol{\omega}_j$. However, it requires *only one pass* over the data, and the dot products can be computed independently, allowing for *massive parallelization* (e.g. with GPU) and distributed computing.

At large N the EM algorithm indeed becomes substantially slower than CLOMPR. We also keep in mind that we compare a MATLAB implementation of the compressive methods with a state-of-the-art C++ implementation of EM[1] [21]. Similarly, at large N the compressive algorithms outperforms EM by several orders of magnitude in terms of memory usage.

5.2. Application: speaker verification

We tested our algorithm on a speaker verification task, with a classical approach referred to as Universal Background Model (GMM-UBM) [17], which requires to train a GMM on a database with millions of items. The details can be found in the original paper [17].

Experiments were performed on the NIST SRE 2005 database [23], in which $50GB$ of speech are available to train the GMM-UBM. We emphasize the fact that our goal is not to attain state-of-the-art results in speaker verification, but rather to compare the results obtained with CLOMPR and EM on this classical approach. In Table 2, results are presented in terms of Equal Error Rate (EER) between False Positive and False Negative rates.

When using $N = 3.10^5$ items (uniformly selected in the whole database to cover all speakers), the results obtained with CLOMPR

[1]There exists an incremental variant of EM (often referred to as online EM, or recursive EM) [22], whose computational cost may be closer to our method despite its different approach. However, to our knowledge there is no optimized implementation of online EM readily available for multivariate GMMs, and we leave comparisons with this approach for future work.

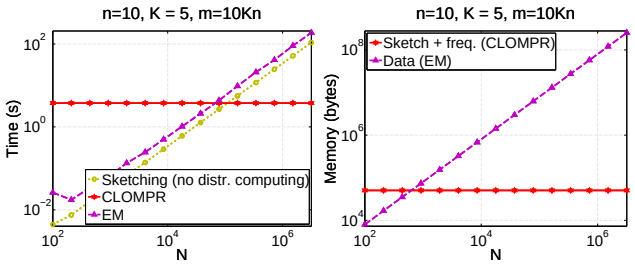

Fig. 2. Time (left) and memory (right) usage of CLOMPR and EM with respect to the size of the database N.

	CLOMPR			EM
	$m = 10^3$	$m = 10^4$	$m = 10^5$	
$N = 3.10^5$	37.15	30.03	28.87	28.69
$N = 3.10^7$	36.57	**29.23**	**28.26**	n/a

Table 2. Speaker verification results in terms of EER, for $K = 64$.

approach those obtained with EM as the number of frequencies increases. When using the entire training database ($N \approx 3.10^7$), at memory scales where EM cannot be performed on a machine with 8 GB of RAM, the results obtained with CLOMPR outperforms those previously obtained with EM ($m = 10^4$ corresponds to a 36000-fold compression of the database).

6. CONCLUSION AND OUTLOOKS

We presented a method for probability mixture estimation on a large database exploiting a *sketch* of the data instead of the data itself. The sketch is a structure that leads to considerable gains in terms of memory, is convenient to compute with possible massive parallelization, and can be easily updated in streaming scenarios.

Inspired by greedy methods for sparse reconstruction, reconstruction algorithms both efficient and stable were defined. In the case of GMM, we designed a heuristic to select generalized moments through random sampling of the empirical characteristic function. Excellent results were observed on synthetic data, and the method was successfully applied to a large-scale speaker verification task.

As mentioned earlier, the method can readily be applied to other mixture models and sketching operators. From a theoretical point of view, the sketching procedure can be related to probability measures embedding in Reproducing Kernel Hilbert Space (RKHS) [24] combined with random features [25], which is expected to lead to theoretical guarantees and new sketching operators.

Greedy approaches such as CLOMPR incur an $O(K^2)$ computational cost to handle K-mixtures, calling for alternatives to deal with large K. Fortunately, there exists many fast algorithms *specific to GMM*, such as the fast hierarchical EM used in [26], which can be modified to be applicable on sketches. Preliminary experiments show that the resulting algorithm is very fast and equally effective for GMM-UBM and its application to speaker verification.

Sketched learning seems particularly adapted for tasks, such as GMM-UBM learning for speaker verification, where one encounters large amounts of distributed training data collected by decentralized devices. In fact, besides their intrinsic amenability to distributed computing and easy update in streaming settings, sketches of controlled size somewhat preserve data privacy. For instance, one could imagine training a GMM-UBM in a real-life environment with sketches aggregated from local sketches maintained on each device, without sharing or transmitting the individual spoken fragments, possibly of sensitive nature.

References

[1] Konstantinos Slavakis, Georgios B. Giannakis, and Gonzalo Mateos, "Modeling and Optimization for Big Data Analytics," *Signal Processing Magazine*, vol. 31, no. 5, pp. 18–31, 2014.

[2] Simon Foucart and Holger Rauhut, *A Mathematical Introduction to Compressive Sensing*, Applied and Numerical Harmonic Analysis. Springer New York, New York, NY, 2013.

[3] Thomas Blumensath, "Sampling and reconstructing signals from a union of linear subspaces," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4660–4671, 2011.

[4] Florentina Bunea, Alexandre B. Tsybakov, Marten H. Wegkamp, and Adrian Barbu, "SPADES and mixture models," *The Annals of Statistics*, vol. 38, no. 4, pp. 2525–2558, Aug. 2010.

[5] Karine Bertin, Erwan Le Pennec, and Vincent Rivoirard, "Adaptive Dantzig density estimation," *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, vol. 47, no. 1, pp. 43–74, Feb. 2011.

[6] Anthony Bourrier, Rémi Gribonval, and Patrick Pérez, "Compressive gaussian mixture estimation," *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pp. 6024–6028, 2013.

[7] Graham Cormode and S. Muthukrishnan, "An improved data stream summary: the count-min sketch and its applications," *Journal of Algorithms*, vol. 55, no. 1, pp. 58–75, Apr. 2005.

[8] Graham Cormode and Marios Hadjieleftheriou, "Methods for finding frequent items in data streams," *The VLDB Journal*, vol. 19, no. 1, pp. 3–20, Dec. 2009.

[9] Nitin Thaper, Sudipto Guha, Piotr Indyk, and Nick Koudas, "Dynamic multidimensional histograms," *Proceedings of the 2002 ACM SIGMOD international conference on Management of data - SIGMOD '02*, p. 428, 2002.

[10] Dimitris Achlioptas, "Database-friendly random projections," *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems - PODS '01*, pp. 274–281, 2001.

[11] Hugo Reboredo, Francesco Renna, Robert Calderbank, and Miguel D. Rodrigues, "Compressive Classification," *2013 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2013 - Proceedings*, , no. 1, pp. 5, Feb. 2013.

[12] Oldaric-Ambrym Maillard and Rémi Munos, "Compressed Least-Squares Regression.," *NIPS*, pp. 1–9, 2009.

[13] Alastair R. Hall, *Generalized method of moments*, Oxford University Press, 2005.

[14] Anthony Bourrier, Mike E. Davies, Tomer Peleg, Patrick Pérez, and Rémi Gribonval, "Fundamental performance limits for ideal decoders in high-dimensional linear inverse problems," *IEEE Transaction on Information Theory*, vol. 60, no. 12, pp. 7928–7946, 2014.

[15] Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, pp. 40–44, 1993.

[16] Prateek Jain, Ambuj Tewari, and Inderjit S. Dhillon, "Orthogonal matching pursuit with replacement," *Advances in Neural Information Processing Systems 24 (2011)*, pp. 1–9, 2011.

[17] Douglas A. Reynolds, Thomas F. Quatieri, and Robert B. Dunn, "Speaker Verification Using Adapted Gaussian Mixture Models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19–41, Jan. 2000.

[18] Emmanuel Candes, Justin K. Romberg, and Terence Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 480–509, Sept. 2006.

[19] Sanjoy Dasgupta, "Learning mixtures of Gaussians," *Foundations of Computer Science, 1999. 40th Annual Symposium*, , no. May, 1999.

[20] Jarvis Haupt, Rui Castro, and Robert Nowak, "Distilled Sensing : Adaptive Sampling for Sparse Detection and Estimation," *IEEE Transactions on Information Theory*, vol. 57, no. 9, 2011.

[21] Andrea Vedaldi and Brian Fulkerson, "VLFeat - An open and portable library of computer vision algorithms," *Design*, vol. 3, no. 1, pp. 1–4, 2010.

[22] Olivier Cappé and Eric Moulines, "Online EM Algorithm for Latent Data Models," *Journal of the Royal Statistical Society*, vol. 71, no. 3, pp. 593–613, 2009.

[23] "The NIST year 2005 speaker recognition evaluation plan," 2005.

[24] Bharath K. Sriperumbudur, Arthur Gretton, Kenji Fukumizu, Bernhard Schölkopf, and Gert R. G. Lanckriet, "Hilbert space embeddings and metrics on probability measures," *The Journal of Machine Learning Research*, vol. 11, pp. 1517–1561, 2010.

[25] Ali Rahimi and Ben Recht, "Random Features for Large Scale Kernel Machines," *Advances in Neural Information Processing Systems*, , no. 1, pp. 1–8, 2007.

[26] Seyed Omid Sadjadi, Malcolm Slaney, and Larry Heck, "Msr identity toolbox v1.0: A matlab toolbox for speaker-recognition research," *Speech and Language Processing Technical Committee Newsletter*, November 2013.