Distributed sparse BSS for large-scale datasets

Tobias Liaudat, Jérôme Bobin, Christophe Kervazo

CEA Saclay, Irfu/SEDI, France | christophe.kervazo@cea.fr

DE LA RECHERCHE À L'INDUSTRIE

cea

Abstract

Blind Source Separation (BSS) is widely used to analyze multichannel data stemming from origins as wide as astrophysics to medicine. However, existent methods do not efficiently handle large datasets. In this work, we propose a new method coined dGMCA (distributed Generalized Morphological Component Analysis) in which the original BSS problem is decomposed into subproblems which can be tackled in parallel, alleviating the large-scale issue. We use the RCM (Riemannian Center of Mass) to *aggregate* during the iterative process the estimations yielded by the different subproblems. We further robustify our large-scale sparse BSS method through the introduction of distributed automatic hyper-parameter choices.

Context

Sparse BSS

- Some multichannel data $\mathbf{X}$ composed of m row observations are assumed to be the linear combination, entached of noise $\mathbf{N}$, of n unknown elementary sources $\mathbf{S}$ of t samples (Comon 2010):

$$\mathbf{X} = \mathbf{AS} + \mathbf{N} \tag{1}$$

- Goal: estimate *physical* $\mathbf{A}$ and $\mathbf{S}$ from the sole $\mathbf{X}$. Ill-posed unsupervised matrix separation problem mandating additional information: here, focus on sparse sources (Bobin 2007).

Large-scale BSS and data deluge

- Ever growing datasets in many fileds. In astronomy: SKA, Euclid, LSST $\rightarrow$ up to $t = 10^9$ samples!
- Two challenges: i) Reduce compution time; ii) Alleviate the memory burden.

We aim at performing sparse BSS on *large-scale* datasets $\mathbf{X}$ in a scalable and reliable way.

Problem formulation and state-of-art

Sparse BSS as an optimization problem

- A way to perform sparse BSS is to look for a minimizer of (Zibulevski 2001):

$$\underset{\mathbf{A},\mathbf{S}}{\text{minimize}} \frac{1}{2}\|\mathbf{X}-\mathbf{AS}\|_F^2 + \left\|\mathbf{R_S} \odot \mathbf{S}\mathbf{\Phi}^{\mathbf{T}}\right\|_1 + \sum_{j\in[1,n]} i_{\|\mathbf{A}^j\|_2=1}(\mathbf{A}) \tag{2}$$

- 3 terms: 1) data fidelity stemming form Gaussian noise assumption; 2) a l_1 sparsity constraint on $\mathbf{S}$: $\mathbf{R_S}$ controls the trade-off with 2), $\mathbf{\Phi}$ is a sparsifying transform; 3) oblique constraint on $\mathbf{A}$, used to avoid degenerated solutions.

Related works

No fully satisfying works for large-scale sparse BSS:

- *Small-scale sparse BSS algorithms:* (Bobin 2007) based on projected Alternating Least Square (pALS), enabling an automatic choice of $\mathbf{R_S}$. Works well on small-scale data only.
- *General large-scale sparse matrix factorization algorithms:* (Mairal 2010, Davis 2016) Highly scalable as based on mini-batches. But do not use pALS, implying low-reliability for sparse BSS (Kervazo 2018).

$\rightarrow$ Proposed method: merge the best of the above two-worlds by **introducing mini-batches in the pALS scheme** of (Bobin 2007).

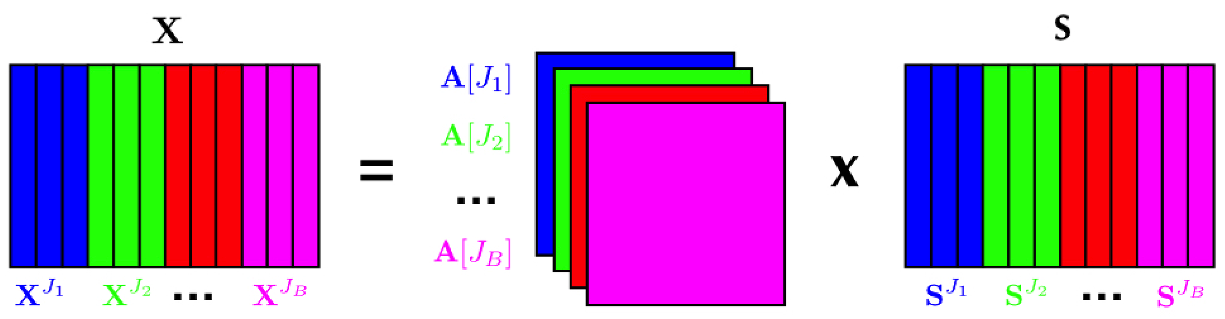
Distributing the GMCA algorithm: overview

Introducing mini-batch pALS: difficulty

- Main idea: re-use the GMCA scheme of (Bobin 2007) but split initial dataset $\mathbf{X}$ into B mini-batches $\mathbf{X}^{J_b}, b \in [1, B]$. The indices of the columns are denoted as J_b (here, $\#J_b = t/b$).

But:

- Issue i) Several estimations $\mathbf{A}[J_b] \in \mathbb{R}^{m\times n}$ of the *same* $\mathbf{A} \in \mathbb{R}^{m\times n}$



$\rightarrow$ Use an **aggregation** method to get a final estimate $\hat{\mathbf{A}}$

- Issue ii) The automatic regularization parameter $\mathbf{R_S}$ choice which made the strength of GMCA uses the *whole* source distribution.

$\rightarrow$ We **introduce a highly parrallelizable adaptative parameter $\mathbf{R_S}$ choice**.

dGMCA: algorithm structure

procedure DGMCA($\mathbf{X}$,$\mathbf{A}^{(0)}$)
    for $k = 1, \cdots, K$ **do**
        Choose $J_1, J_2, \cdots, J_B$ as a partition of $[1, t]$
        for $b = 1, \cdots, B$ **do**
            $\tilde{\mathbf{S}}^{J_b(k)} = \hat{\mathbf{A}}^{(k-1)\dagger}\mathbf{X}^{J_b(k)}$
            $\mathbf{R_S}^{(k)} = \texttt{DISTRIBUTED\_}\mathbf{R_S}\texttt{\_CHOICE}$
            $\mathbf{w}[J_b(k)] = \texttt{COMPUTE\_WEIGHT}(\tilde{\mathbf{S}}^{J_b(k)})$
            $\hat{\mathbf{S}}^{J_b(k)} = \mathcal{S}_{\mathbf{R_S}^{(k)}}(\tilde{\mathbf{S}}^{J_b(k)})$
            $\mathbf{A}[J_b(k)] = \Pi_{\|.\|_2=1}(\mathbf{X}^{J_b(k)}\hat{\mathbf{S}}^{J_b(k)\dagger})$
        end for
        $\hat{\mathbf{A}}^{(k)} = \texttt{AGGREGATE}(\mathbf{A}[J_1(k)], \mathbf{A}[J_2(k)], ..., \mathbf{A}[J_B(k)], \mathbf{w}[J_1(k)], ..., \mathbf{w}[J_B(k)])$
    end for
    return $\hat{\mathbf{A}}^{(K)}, \hat{\mathbf{S}}^{(K)}$
end procedure

Detailling the main steps

Aggregation step

- Simplistic solution to aggregate the different $\mathbf{A}[J_b], b \in [1, B]$: Euclidean mean.
- $\hat{\mathbf{A}}$ must respect the oblique constraint $\rightarrow$ **aggregate using a Riemannian Center of Mass – RCM** (Afsari 2011).
- Compute using (Afsari 2011) the RCM by finding the point on the manifold minimizing a sum of the weighted square geodesic lengths d:

$$\forall j \in [1, n], \quad \hat{\mathbf{A}}^j = \underset{\mathbf{a}}{\text{argmin}} \sum_{b\in[1,B]} \mathbf{w}[J_b]^j \, d^2\left(\mathbf{a}, \mathbf{A}[J_b]^j\right), \tag{3}$$

- Robustification: weights $\mathbf{w}[J_b]^j$ are used to penalize noisy sources estimated from a given mini-batch b:

$$\forall j \in [1, n], \quad \forall b \in [1, B]; \quad \mathbf{w}[J_b]^j = \frac{\left([\mathbf{A}[J_b]^+]_j \mathbf{\Sigma_N} [\mathbf{A}[J_b]^+]_j^T\right)^{-1}}{\sum_{b=1}^n \left([\mathbf{A}[J_b]^+]_j \mathbf{\Sigma_N} [\mathbf{A}[J_b]^+]_j^T\right)^{-1}} \tag{4}$$

Distributing the $\mathbf{R_S}$ parameter choice

- In GMCA, $\mathbf{R_S}$ is chosen as an increasing percentile of the estimated sources $\rightarrow$ requires the whole source distribution.
- Here, we use parametrized exponentially decaying $\mathbf{R_S}$, which is *easily parallelizable*:

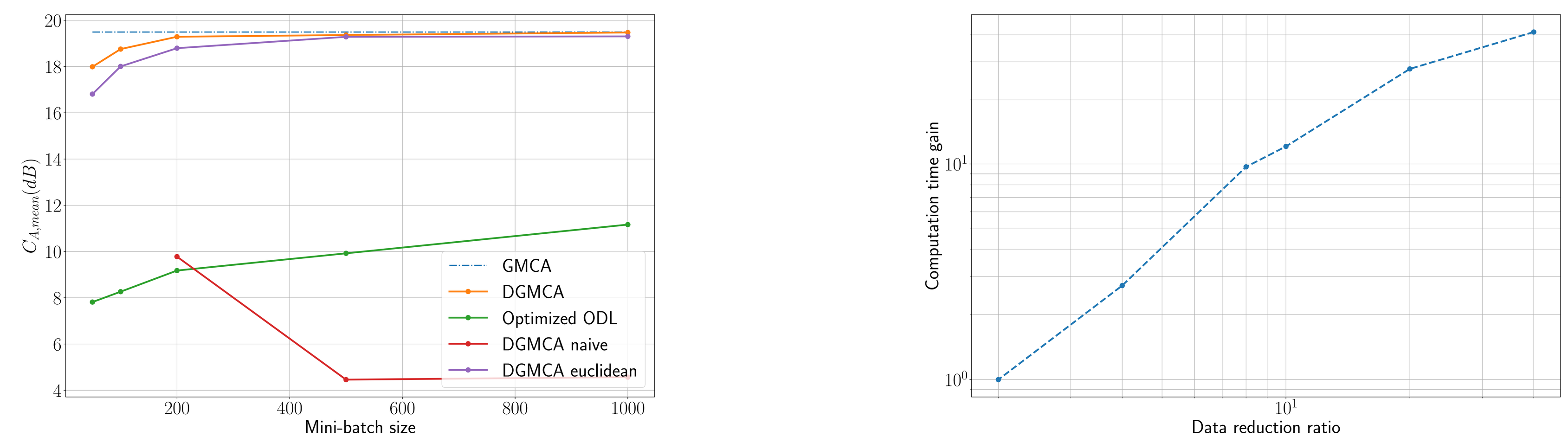
$$\mathbf{R}^i_{i\mathbf{S}} = K\sigma_i + \left(\left\|\hat{\mathbf{S}}_i^{(k)}\right\|_\infty - K\sigma_i\right)\exp(-k\alpha_i), \tag{5}$$

where σ_i is an estimation of the back-projected noise on source $\mathbf{S}_i$ std (estimated through the Median Absolute Deviation operator), $K = 3$ is a parameter (set according to a fixed point argument, see (Bobin 2007)), $\left\|\hat{\mathbf{S}}_i^{(k)}\right\|_\infty$ is the maximum absolute value of $\hat{\mathbf{S}}_i^{(k)}$ and α_i is a parameter controlling the exponential decay speed.

- Parameter α_i can be chosen fitting a generalized Gaussian to the sources during the first iterations. However, in practice the results are quite insensible to α_i values, enabling to set them beforehand.

Numerical experiments: simulations

- Sources follow a Generalized Gaussian distribution with $\beta \in [0.35, 1.4]$. The parameters are $n = 10$, $m = 20$, $t = 10000$, $SNR = 15dB$. The matrix $\mathbf{A}$ is random with a condition number of 10.

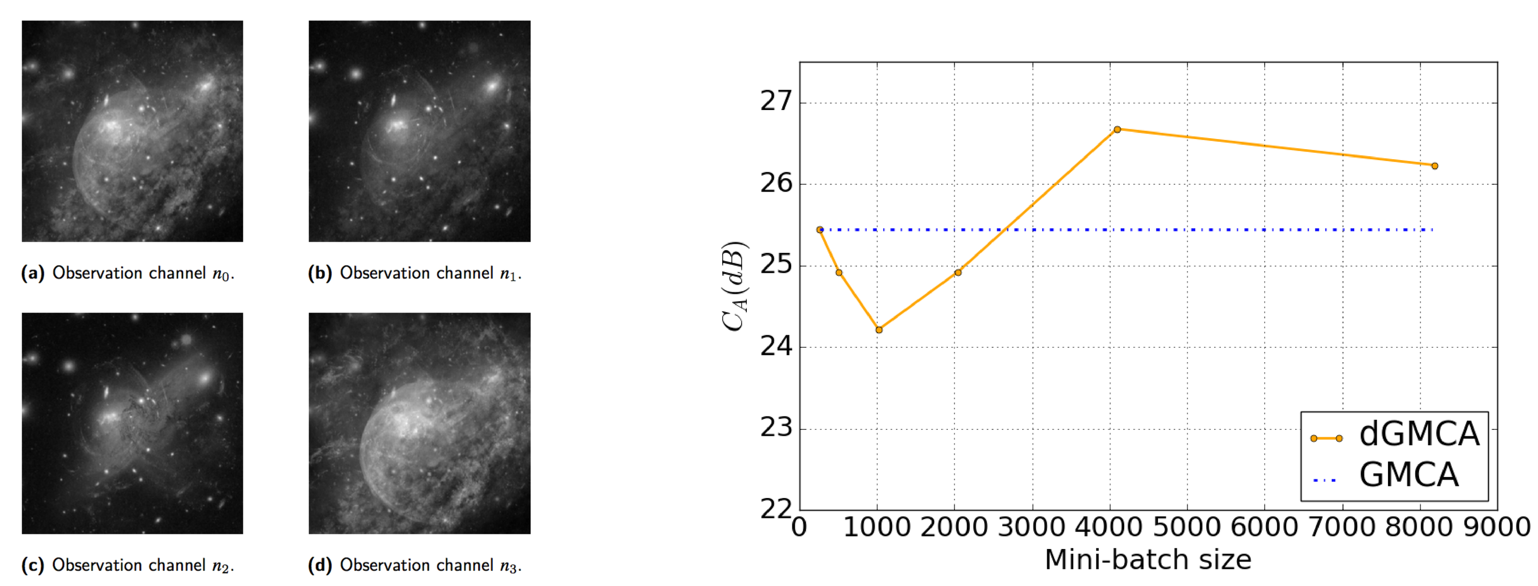


Left: separation quality in terms of $C_A = -10\log(\|\mathbf{P}\hat{\mathbf{A}}\mathbf{A} - \mathbf{I}\|_{\ell_1})$, with $\mathbf{P}$ correcting permutations and scale; Right: Computational time gain between GMCA and dGMCA.

- dGMCA enables to maintain a high separation quality while providing an almost linear gain in computation time.

Numerical experiments: realistic sources

- $n = 5$ realistic astrophysical sources, mixed through a $\mathbf{A}$ matrix resembling power laws. There are $m = 250$ observations, $t = 32768$.
- The sources are sparse in the *starlet domain* (Starck 2010)
- The transform is applied on each mini-batch. With relatively large mini-batches, the support of the transform is small enough to limit border effects.



Left: examples of observations; Right: Separation results for different mini-batch sizes.

Conclusion and perspectives

We introduced in this work the dGMCA algorithm, which enables to perform large-scale sparse BSS through a parrallelization of the GMCA algorithm. The method is based on a mini-batch pALS optimization scheme. The originality lies in the use of the Riemannian Center of Mass to aggregate the different estimates during the estimation process, enabling to take into account the geometry of the problem. The approach is further robustified both by a clever choice of the weights of the RCM based on the estimated SNR and a parrallelized heuristic regularization parameter choice. Numerical experiments on both simulated and realistic sources show that the proposed approach is able to handle large-scale problems, as it enables a linear acceleration and alleviate the memory burden, while getting results almost as good as GMCA.