Multivariate Convolutional Sparse Coding for Electromagnetic Brain Signals

Thomas Moreau
INRIA Saclay

Joint work with Dupré La Tour T., Mainak J., Gramfort A.

PARIETAL

Inria
inventors for the digital world

Context: Functional Neuroimaging

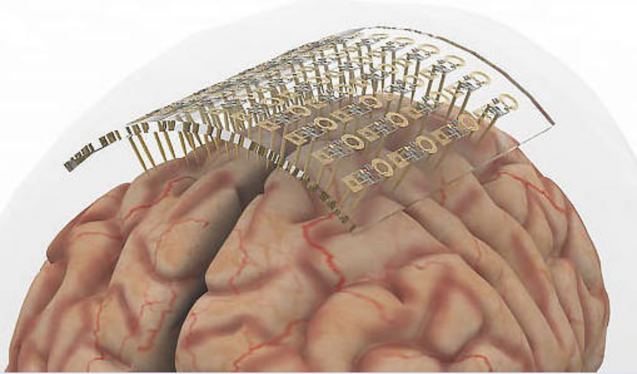
Goal: Study the brain mechanisms while it is functioning.

Outputs:

- **Functional Atlases:** Link areas of the brain to specific cognitive functions.
- **Functional Connectivity:** Highlight the information flow in the brain.
- **Healthcare:** Develop bio-markers for neurological disorders.
- ...

How to record living brains activity: **Electrophysiology**

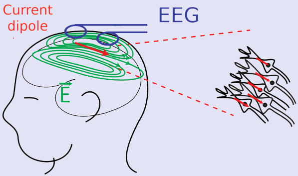
Direct measurement of electrical activity.

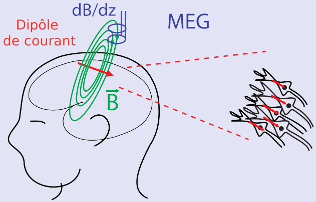


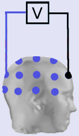
High Localization **Low Resolution** **Invasive**

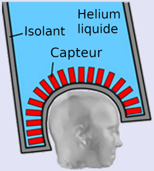
How to record living brains activity: **Electrophysiology**

Remote measurement of the electrical activity.

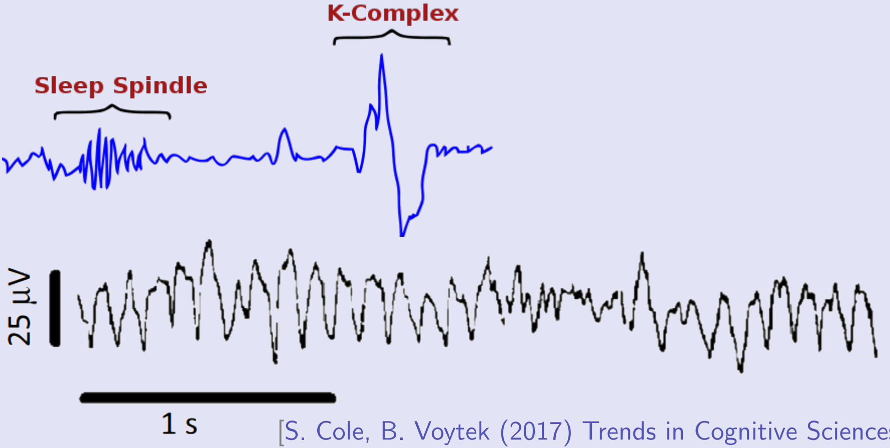




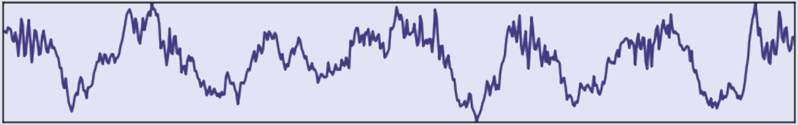




No Localization **Global** **Non Invasive**

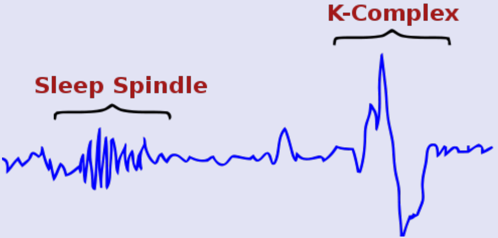


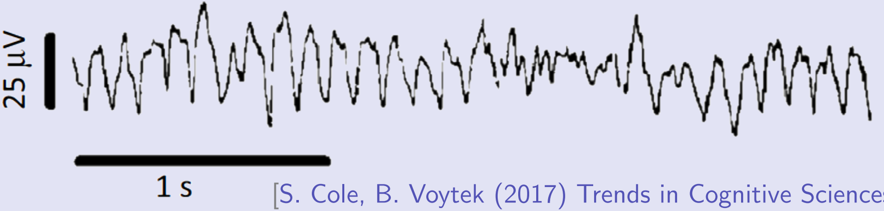
[S. Cole, B. Voytek (2017) Trends in Cognitive Sciences]



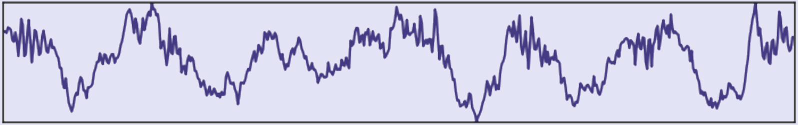
[Dupré la Tour, Tallot, Grabot, Doyère, van Wassenhove, Grenier, Gramfort (2017) PLOS Computational biology]

Neural signals exhibit diverse and complex morphologies

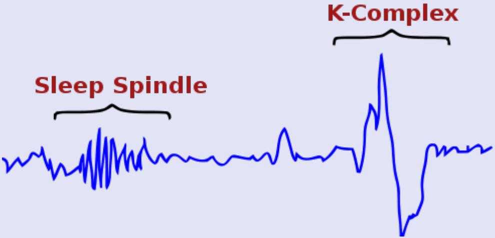




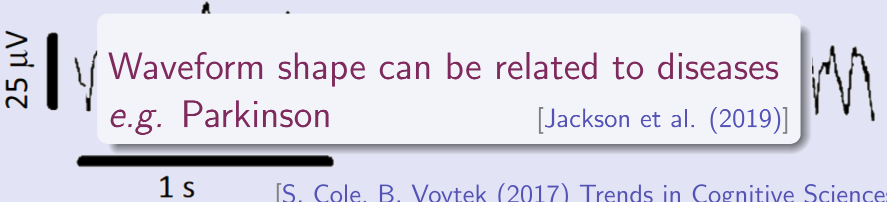
[S. Cole, B. Voytek (2017) Trends in Cognitive Sciences]



[Dupré la Tour, Tallot, Grabot, Doyère, van Wassenhove, Grenier, Gramfort (2017) PLOS Computational biology]

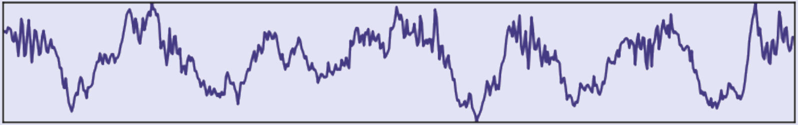


Neural signals exhibit diverse and complex morphologies



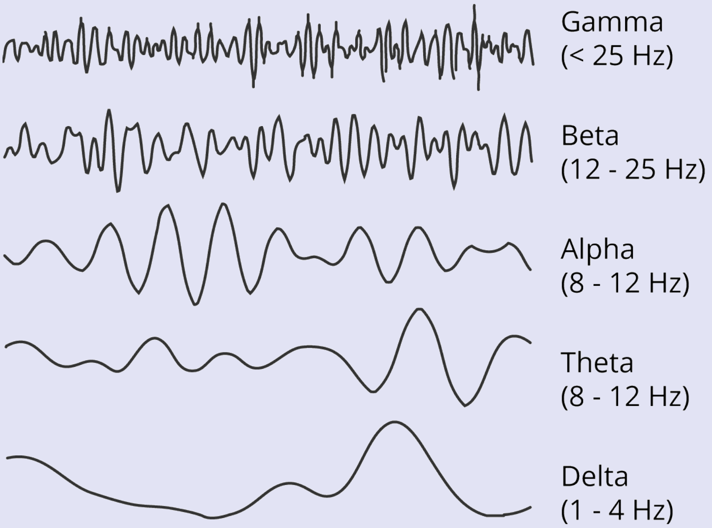
Waveform shape can be related to diseases
e.g. Parkinson [Jackson et al. (2019)]

[S. Cole, B. Voytek (2017) Trends in Cognitive Sciences]

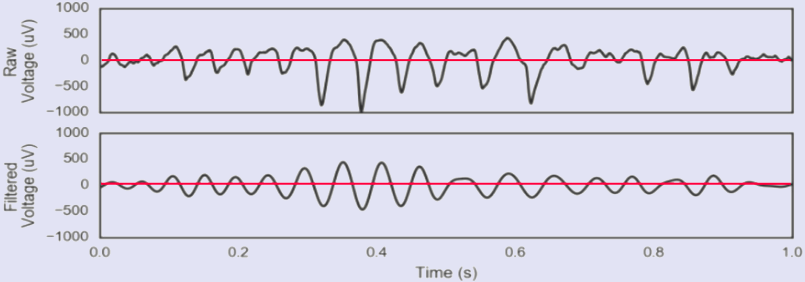


[Dupré la Tour, Tallot, Grabot, Doyère, van Wassenhove, Grenier, Gramfort (2017) PLOS Computational biology]

"Textbook" brain rythm



After Linear filters, everything looks like a sinusoïd.



⇒ Lose the asymmetry and the shape information.

Fourier Fallacy

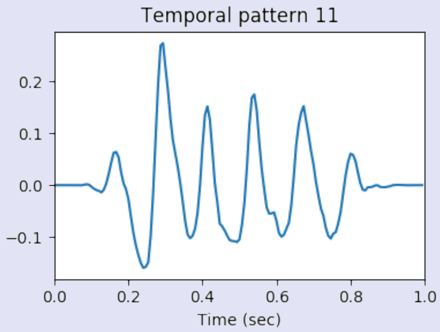
"Even though it may be possible to analyze the complex forms of brain waves into a **number of different sine-wave** frequencies, this may lead only to what might be termed a "**Fourier fallacy**", if one assumes **ad hoc** that all of the necessary frequencies actually occur as periodic phenomena in **cell groups** within the brain."

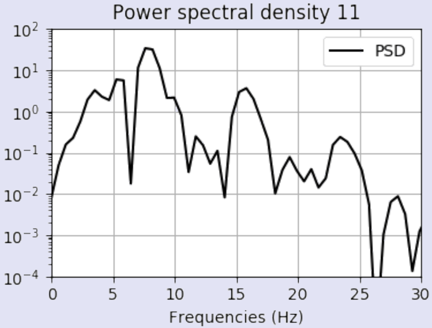
[Jasper (1948)]

Fourier Fallacy

"Even though it may be possible to analyze the complex forms of brain waves into a **number of different sine-wave** frequencies, this may lead only to what might be termed a "**Fourier fallacy**", if one assumes **ad hoc** that all of the necessary frequencies actually occur as periodic phenomena in **cell groups** within the brain."

[Jasper (1948)]

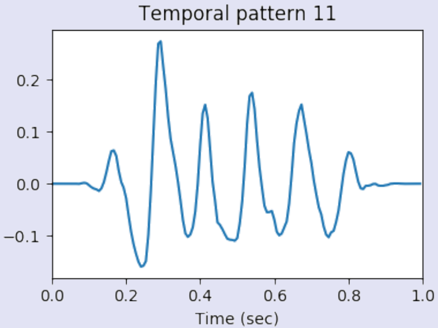


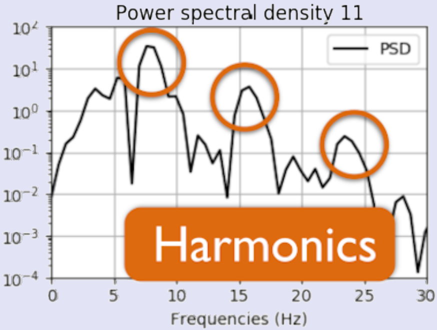


Fourier Fallacy

"Even though it may be possible to analyze the complex forms of brain waves into a **number of different sine-wave** frequencies, this may lead only to what might be termed a "**Fourier fallacy**", if one assumes **ad hoc** that all of the necessary frequencies actually occur as periodic phenomena in **cell groups** within the brain."

[Jasper (1948)]





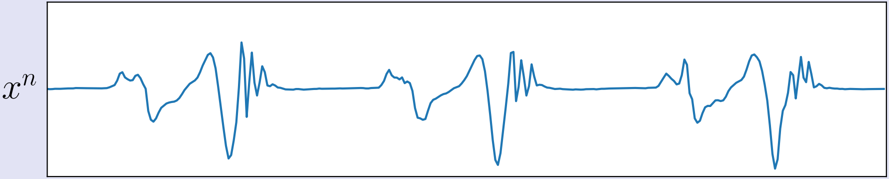
Learning the waveform: Convolutional Dictionary Learning

References

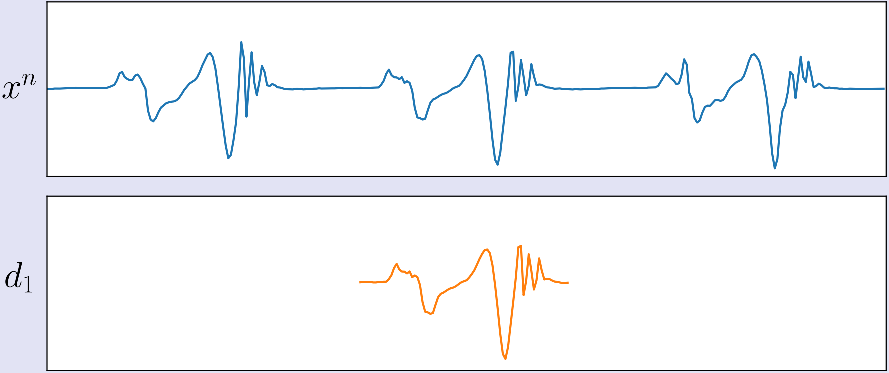
- Grosse, R., Raina, R., Kwong, H., and Ng, A. Y. (2007). Shift-Invariant Sparse Coding for Audio Classification.

  Cortex, 8:9

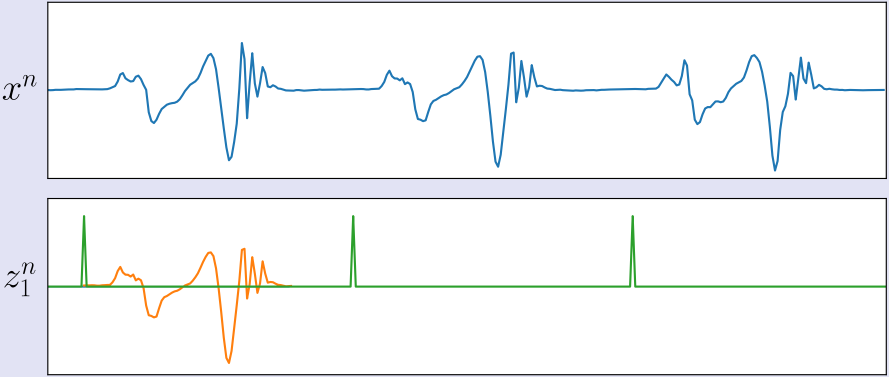
Local structure in signals



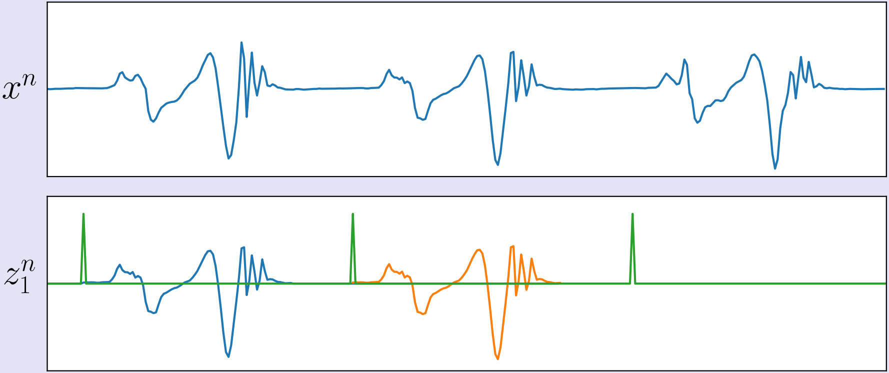
Local structure in signals



Local structure in signals

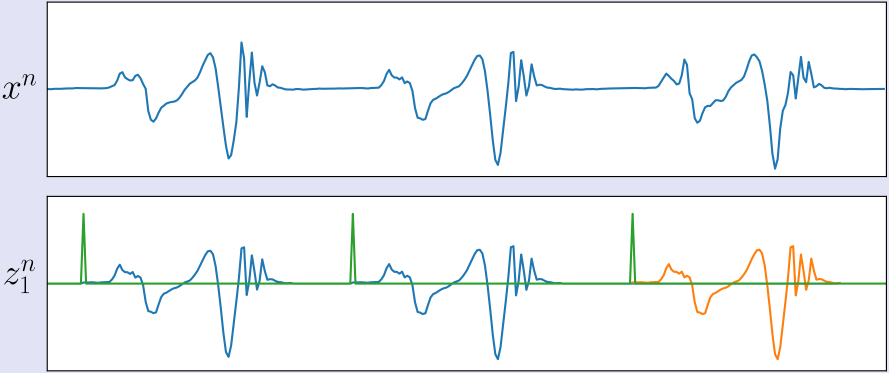


Local structure in signals

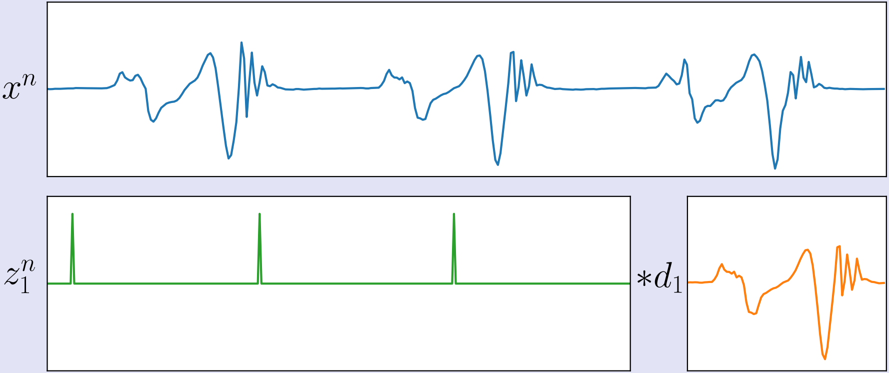


Local structure in signals

Key idea: decouple the localization of the patterns and their shape

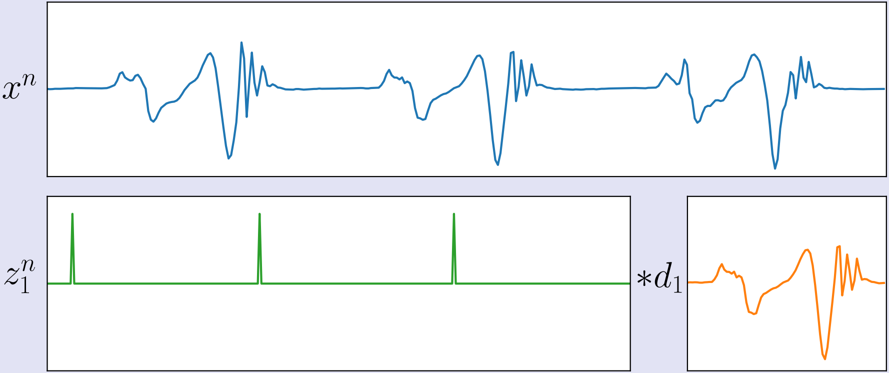


Key idea: decouple the localization of the patterns and their shape



Local structure in signals

Key idea: decouple the localization of the patterns and their shape



$$x^n[t] = \sum_{k=1}^{K} (z_k^n * d_k)[t] + \varepsilon[t]$$

For a set of N univariate signals x^n, solve

$$\min_{d,z} \sum_{n=1}^{N} \frac{1}{2} \left\| \boxed{x^n} - \sum_{k=1}^{K} \boxed{z_k^n} * \boxed{d_k} \right\|_2^2 + \lambda \sum_{k=1}^{K} \|\boxed{z_k^n}\|_1,$$
$$\text{s.t.} \quad \|\boxed{d_k}\|_2^2 \leq 1$$

Hypothesis: patterns d_k are not present everywhere in the signal. They are localized in time.

$\Rightarrow$ Sparse activation signals z

Technical hypothesis: the patterns are in the ℓ_2-ball: $\|d_k\|_2^2 \leq 1$.

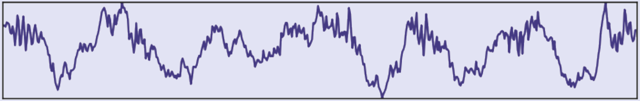
Optimization strategy

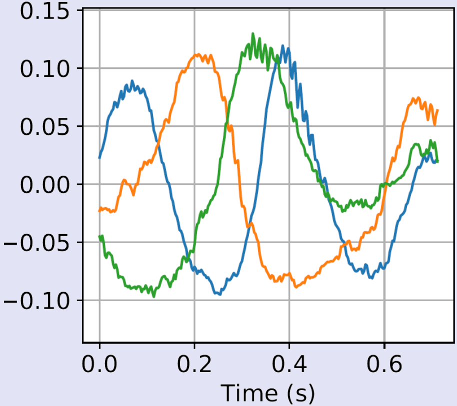
Bi-convex: The problem is not jointly convex in z_k^n, and d_k but it is convex in each block of coordinate.

Alternate minimization (*a.k.a.* Bloc Coordinate Descent):

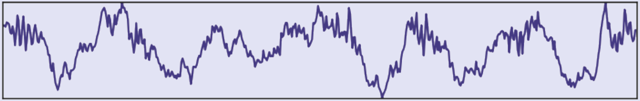
- **Z-step:** given a fixed estimate of the atom, compute the activation signal z_k^n associated to each signal x^n.
- **D-step:** given a fixed estimate of the activation, update the atoms in the dictionary d_k.

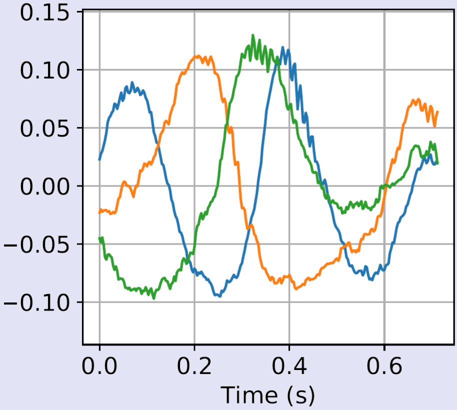
Data:





Data:





What to do in the case of multivariate signals?

How to extend CSC to multivariate signals?

We can just use multivariate convolution,

$$\underbrace{X[t]}_{\in\mathbb{R}^P} = \sum_{k=1}^{K} \left(z_k * D_k\right)[t] = \sum_{k=1}^{K}\sum_{\tau=1}^{L} z_k[t-\tau]\,\underbrace{D_k[\tau]}_{\in\mathbb{R}^P}$$

with:

- X a multivariate signal of length T in $\mathbb{R}^P$
- D_k a multivariate signal of length L in $\mathbb{R}^P$
- z_k a univariate activation signal of length $\widetilde{T} = T - L + 1$

However, this model does not account for the physics of the problem.

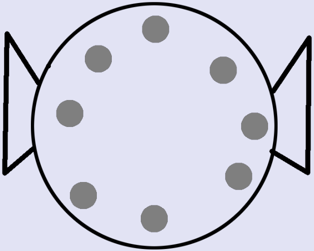
Rank-1 constrained dictionary learning

References

- Dupré la Tour, T., Moreau, T., Jas, M., and Gramfort, A. (2018).
  Multivariate Convolutional Sparse Coding for Electromagnetic Brain Signals.
  In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 3296–3306, Montreal, Canada

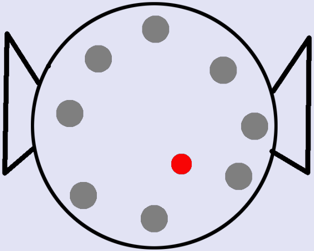
EM wave diffusion

- Recording here with 8 sensors



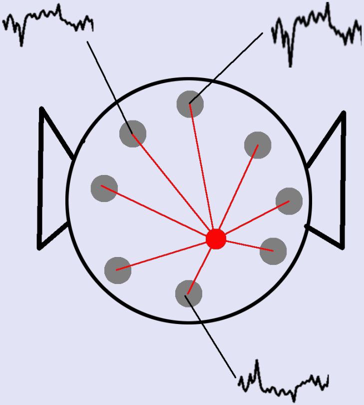
EM wave diffusion

- Recording here with 8 sensors
- EM activity in the brain



EM wave diffusion

- Recording here with 8 sensors
- EM activity in the brain
- The electric field is spread **linearly** and **instantaneously** over all sensors (Maxwell equations)



Multivariate CSC with rank-1 constraint

Idea: Impose a rank-1 constraint on the dictionary atoms D_k

To make the problem tractable, we decided to use auxiliary variables u_k and v_k *s.t.* $D_k = u_k v_k^\top$.

$$\min_{u_k, v_k, z_k^n} \sum_{n=1}^{N} \frac{1}{2} \left\| X^n - \sum_{k=1}^{K} z_k^n * (u_k v_k^\top) \right\|_2^2 + \lambda \sum_{k=1}^{K} \left\| z_k^n \right\|_1 , \tag{1}$$
$$\text{s.t.} \quad \|u_k\|_2^2 \leq 1 \ , \ \|v_k\|_2^2 \leq 1 \text{ and } z_k^n \geq 0 \ .$$

Here,

- $u_k \in \mathbb{R}^P$ is the spatial pattern of our atom
- $v_k \in \mathbb{R}^L$ is the temporal pattern of our atom

Optimization strategy

Tri-convex: The problem is not jointly convex in z_k^n, u_k and v_k but it is convex in each block of coordinate.

We can use a block coordinate descent, aka alternate minimization, to converge to a local minima of this problem. The 3 following steps are applied alternatively:

- **Z-step:** given a fixed estimate of the atom, compute the activation signal z_k^n associated to each signal X^n.
- **u-step:** given a fixed estimate of the activation and temporal pattern, update the spatial pattern u_k.
- **v-step:** given a fixed estimate of the activation and spatial pattern, update the temporal pattern v_k.

Z-step: Locally greedy coordinate descent (LGCD)

N independent problem such that

$$\min_{z_k^n \geq 0} \frac{1}{2} \left\| X^n - \sum_{k=1}^{K} z_k^n * D_k \right\|_2^2 + \lambda \sum_{k=1}^{K} \left\| z_k^n \right\|_1 \quad .$$

This problem is convex in z_k and can be solved with different techniques:

- Greedy CD [Kavukcuoglu et al., 2010]
- Fista [Chalasani et al., 2013]
- ADMM [Bristow et al., 2013]
- L-BFGS [Jas et al., 2017]

⇒ These methods can be slow for long signals as the complexity of each iteration is at least linear in the length of the signal.

Z-step: Locally greedy coordinate descent (LGCD)

For the Greedy Coordinate Descent, only 1 coordinate is updated at each iteration: [Kavukcuoglu et al., 2010]

1. The coordinate $z_{k_0}[t_0]$ is updated to its optimal value $z'_{k_0}[t_0]$ when all other coordinate are fixed.

$$z'_k[t] = \max\left(\frac{\beta_k[t] - \lambda}{\|D_k\|_2^2}, 0\right),$$

with $\beta_k[t] = \left[D_k^{\Lsh} * \left(X - \sum_{l=1}^K z_l * D_l + z_k[t]e_t * D_k\right)\right][t]$

For each coordinate update, it is possible to maintain the value of β with $\mathcal{O}(KL)$ operations.

Z-step: Locally greedy coordinate descent (LGCD)

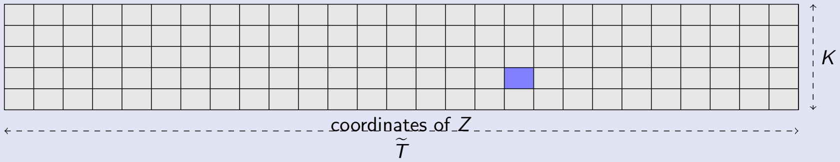
For the Greedy Coordinate Descent, only 1 coordinate is updated at each iteration: [Kavukcuoglu et al., 2010]

1. The coordinate $z_{k_0}[t_0]$ is updated to its optimal value $z'_{k_0}[t_0]$ when all other coordinate are fixed.

2. The updated coordinate is chosen

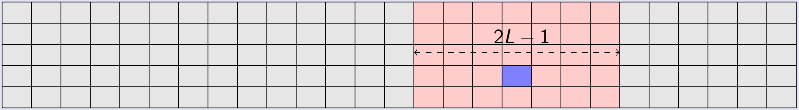
- Cyclic selection: $\mathcal{O}(1)$ [Friedman et al., 2007]
- Randomized selection: $\mathcal{O}(1)$ [Nesterov, 2010]
- Greedy selection: $\mathcal{O}(K\widetilde{T})$ [Osher and Li, 2009]
  by maximizing $|z_k[t] - z'_k[t]|$

We introduced the LGCD method which is an extension of GCD.



GCD has $\mathcal{O}(K\widetilde{T})$ computational complexity.

We introduced the LGCD method which is an extension of GCD.

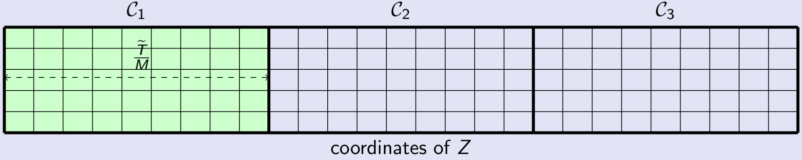


coordinates of Z

GCD has $\mathcal{O}(K\widetilde{T})$ computational complexity.

But the update itself has complexity $\mathcal{O}(KL)$

We introduced the LGCD method which is an extension of GCD.

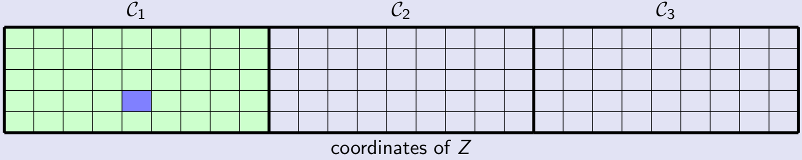


With a partition $\mathcal{C}_m$ of the signal domain $[1, K] \times [0, \widetilde{T}[$,

$$\mathcal{C}_m = [1, K] \times [\frac{(m-1)\widetilde{T}}{M}, \frac{m\widetilde{T}}{M}[$$

Locally Greedy Coordinate Descent [Moreau et al., 2018]

We introduced the LGCD method which is an extension of GCD.



With a partition $\mathcal{C}_m$ of the signal domain $[1, K] \times [0, \widetilde{T}[$,

$$\mathcal{C}_m = [1, K] \times [\frac{(m-1)\widetilde{T}}{M}, \frac{m\widetilde{T}}{M}[$$

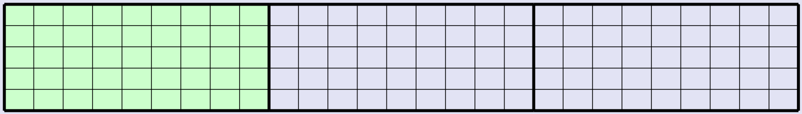
The coordinate to update is chosen greedily on a sub-domain $\mathcal{C}_m$

$$\frac{\widetilde{T}}{M} = 2L - 1 \quad \Rightarrow \quad \mathcal{O}(\text{Coordinate selection}) = \mathcal{O}(\text{Coordinate Update})$$

The overall iteration complexity is $\mathcal{O}(KL)$ instead of $\mathcal{O}(K\widetilde{T})$.

$\Rightarrow$ Efficient for sparse Z

We introduced the LGCD method which is an extension of GCD.



With a partition $\mathcal{C}_m$ of the signal domain $[1, K] \times [0, \widetilde{T}[$,

$$\mathcal{C}_m = [1, K] \times [\frac{(m-1)\widetilde{T}}{M}, \frac{m\widetilde{T}}{M}[$$

The coordinate to update is chosen greedily on a sub-domain $\mathcal{C}_m$

$$\frac{\widetilde{T}}{M} = 2L - 1 \quad \Rightarrow \quad \mathcal{O}(\text{Coordinate selection}) = \mathcal{O}(\text{Coordinate Update})$$

The overall iteration complexity is $\mathcal{O}(KL)$ instead of $\mathcal{O}(K\widetilde{T})$.

$\Rightarrow$ Efficient for sparse Z $\Rightarrow$ Can be efficiently parallelized.

D-step: solving for the atoms

The dictionary update is performed by minimizing

$$\min_{\|D_k\|_2 \leq 1} E(D) \triangleq \sum_{n=1}^{N} \frac{1}{2} \|X^n - \sum_{k=1}^{K} z_k^n * D_k\|_2^2 \quad . \tag{2}$$

Computing $\nabla_{d_k} E(\{d_k\}_k)$ can be done efficiently

$$\nabla_D E(D) = \sum_{n=1}^{N} (z_k^n)^{\urcorner} * \left(x^n - \sum_{l=1}^{K} z_l^n * D_l \right) = \Phi_k - \sum_{l=1}^{K} \Psi_{k,l} * D_l \ ,$$

⇒ Save with Projected Gradient Descent (PGD) with an Armijo backtracking line-search for the D-step [Wright and Nocedal, 1999].

D-step: solving for the atoms

We use the projected gradient descent with an Armijo backtracking line-search Wright and Nocedal [1999] for both u-step and v-step for

$$\min_{\substack{\|u_k\|_2 \leq 1 \\ \|v_k\|_2 \leq 1}} E(u_k, v_k) \triangleq \sum_{n=1}^{N} \frac{1}{2} \|X^n - \sum_{k=1}^{K} z_k^n * (u_k v_k^\top)\|_2^2 \quad . \tag{3}$$

One important computation trick is for fast computation of the gradient.

$$\nabla_{u_k} E(u_k, v_k) = \nabla_{D_k} E(u_k, v_k) v_k \quad \in \mathbb{R}^P ,$$
$$\nabla_{v_k} E(u_k, v_k) = u_k^\top \nabla_{D_k} E(u_k, v_k) \quad \in \mathbb{R}^L ,$$

Computing $\nabla_{D_k} E(u_k, v_k)$ can be done efficiently

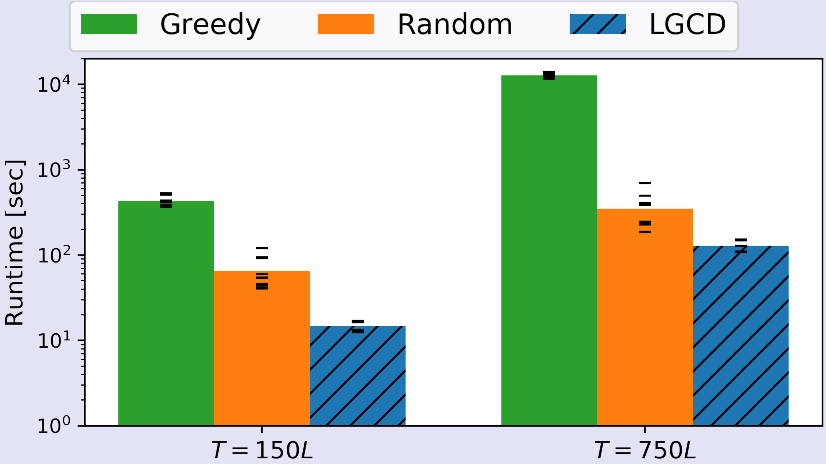
$$\nabla_{D_k} E(u_k, v_k) = \sum_{n=1}^{N} (z_k^n)^{\Lsh} * \left(X^n - \sum_{l=1}^{K} z_l^n * D_l \right) = \Phi_k - \sum_{l=1}^{K} \Psi_{k,l} * D_l \ ,$$

Experiments

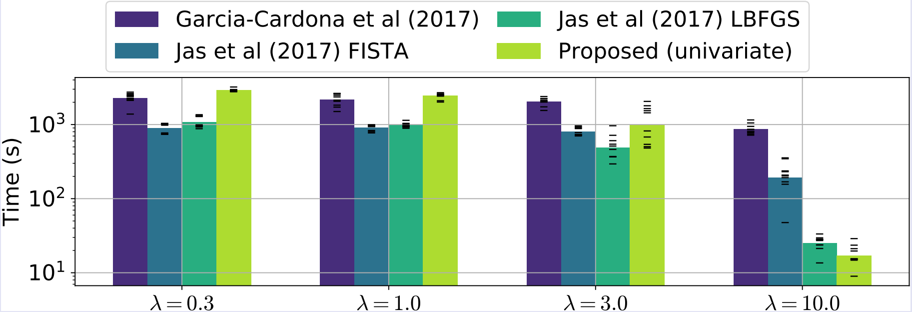
Good time to wake-up if you got lost in the previous section!

Comparison of the coordinate selection strategy for CD on simulated signals

We set $K = 10$, $L = 150$, $\lambda = 0.1\lambda_{\max}$

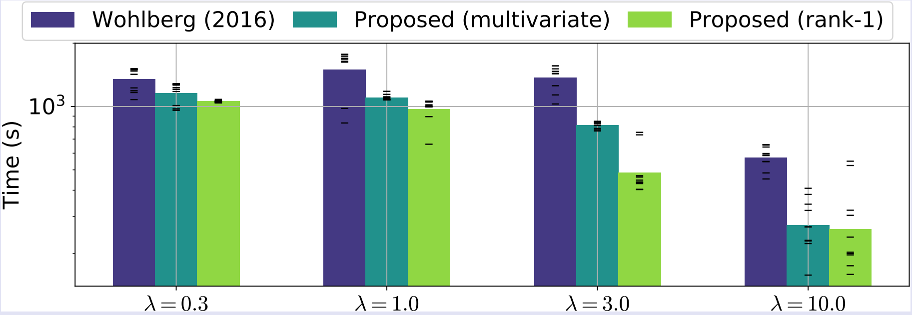


Comparison with univariate methods on somato dataset with $T = 134,700$, $K = 8$ and $L = 128$



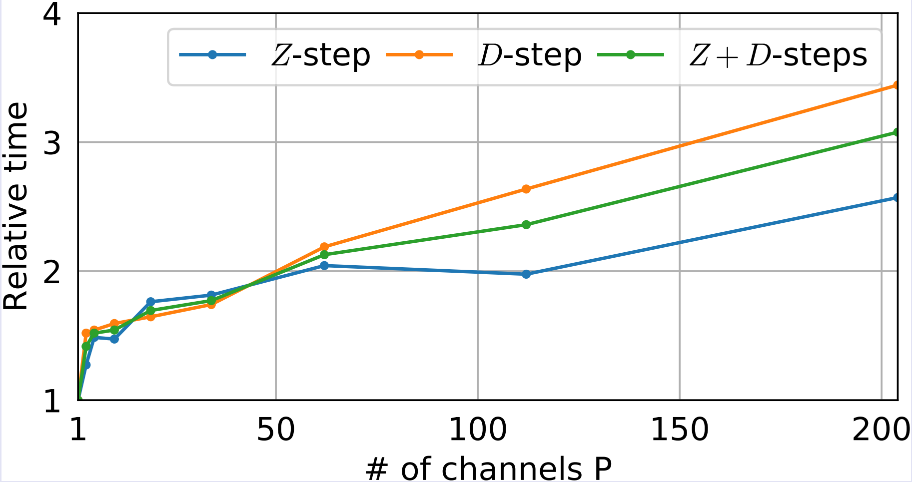
Fast optimization

Comparison with multivariate methods on somato dataset with
$T = 134,700$, $K = 8$, $P = 5$ and $L = 128$



Good scaling in the number of channels P

Scaling relative to P on somato dataset with $T = 134,700$, $K = 2$, and $L = 128$



Pattern recovery

Test the pattern recovery capabilities of our method on simulated data,

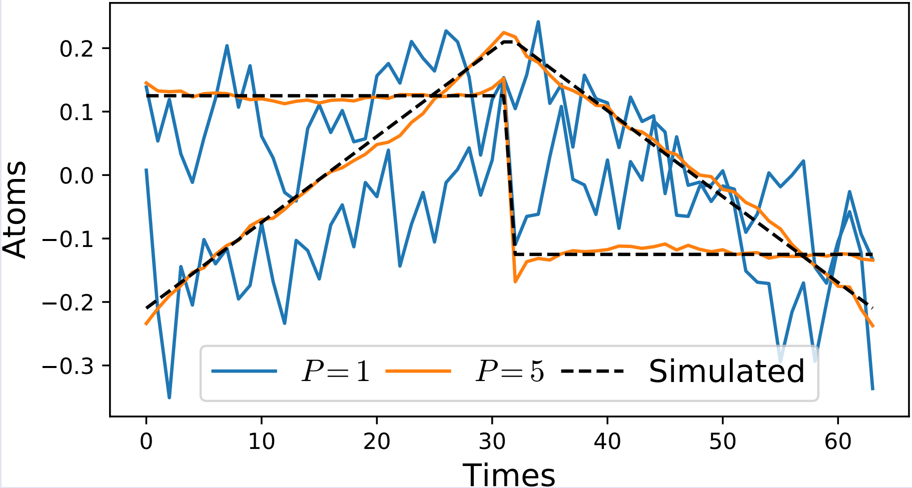
$$X^n = \sum_{k=1}^{2} z_k * (u_k v_k^\top) + \mathcal{E}$$

where (u_k, v_k) are chosen patterns of rank-1 and the activated coefficient $z_k^n[t]$ are drawn uniformly and their value are uniform in $[0, 1]$.

The noise $\mathcal{E}$ is generated as a gaussian white noise with variance σ.

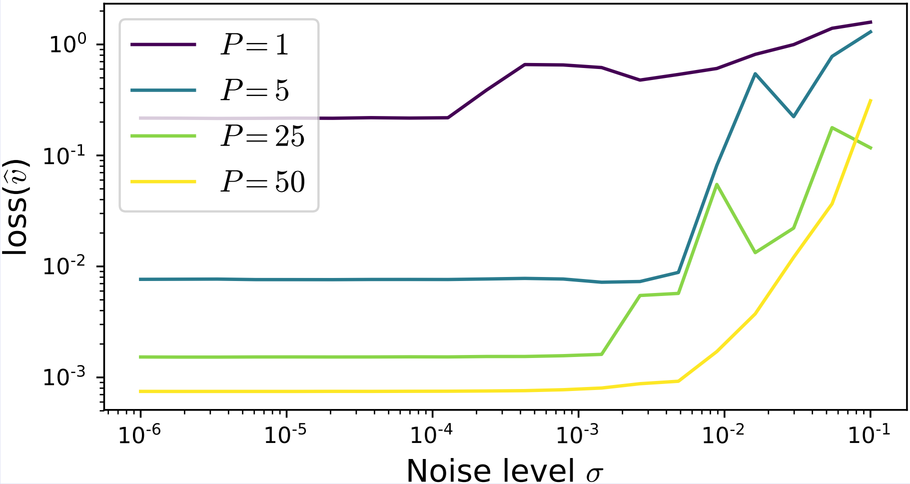
We set $N = 100$, $L = 64$ and $\widetilde{T} = 640$

Patterns recovered with $P = 1$ and $P = 5$. The signals were generated with the two simulated temporal patterns and with $\sigma = 10^{-3}$.



Pattern recovery

Evolution of the recovery loss with σ for different values of P. Using more channels improves the recovery of the original patterns.

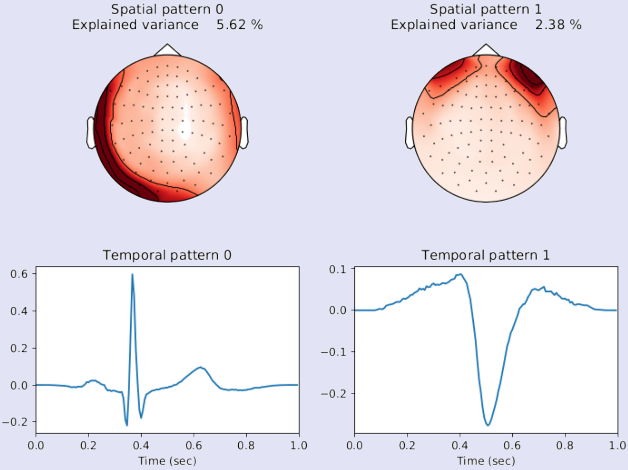


Experiments on MEG data

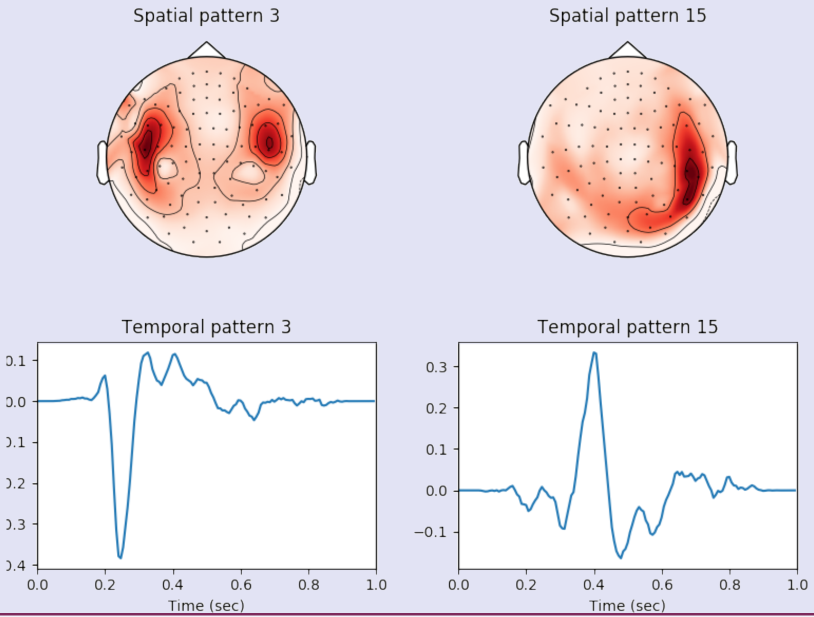
Even better time to wake-up!

MNE somatosensory data

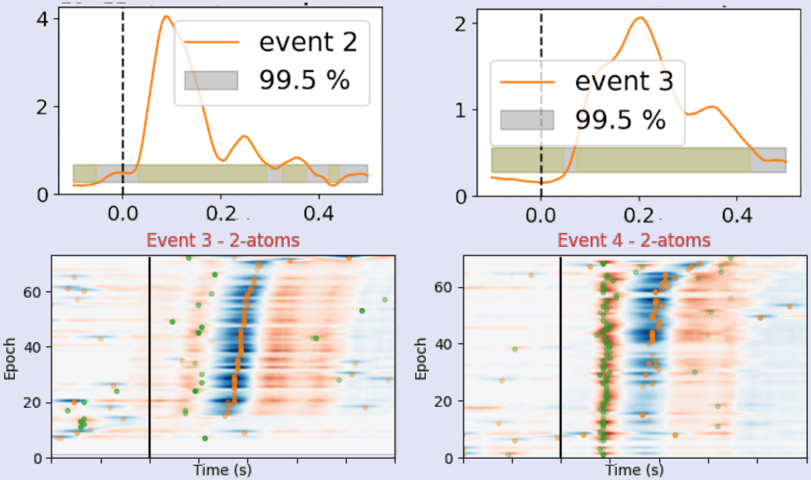
A selection of temporal waveforms of the atoms learned on the MNE sample dataset.



Learned atoms – Evoked response



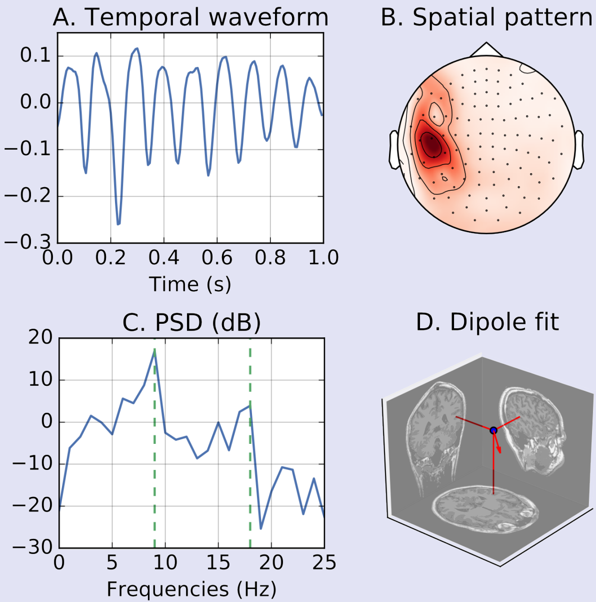
Learned atoms – Evoked response



Learned atoms – Evoked response

Work in progress!

Learned atoms – Complex waveforms



alphaCSC: Convolution sparse coding for time-series

build passing codecov 82%

This is a library to perform shift-invariant sparse dictionary learning, also known as time-series data. It includes a number of different models:

1. univariate CSC
2. multivariate CSC
3. multivariate CSC with a rank-1 constraint [1]
4. univariate CSC with an alpha-stable distribution [2]

A mathematical descriptions of these models is available in the documentation.

Python code online:
https://alphacsc.github.io

pip install alphacsc

Installation

To install this package, the easiest way is using `pip`. It will install this package and depends on `numpy` and `cython` for the installation so it is advised to install them please run one of the two commands:

Examples reproduce figures from this talk!

(Latest stable version)

```
pip install numpy cython
pip install alphacsc
```

Thanks for your attention!

Code available online:

alphacsc : alphacsc.github.io

DiCoDiLe : github.com/tommoral/dicodile

Slides are on my web page:

tommoral.github.io @tomamoral