Unsupervised Learning for Time-series: Extracting Patterns in Brain Recordings

Thomas Moreau – Parietal – Inria

Who am I?

Parcours

2014-2017: PhD – ENS Cachan (N. Vayatis & L. Oudre)

Applied Maths

Distributed optimization
Unsupervised learning
Representation learning

Healthcare Data

Gait analysis
Oculomotor recordings

2018-2019: Post-doc – Inria (Parietal)

Convolutional dictionary learning for MEG

$\Rightarrow$ **Local structure** analysis for signals

2019-∞: CR – Inria (Parietal)

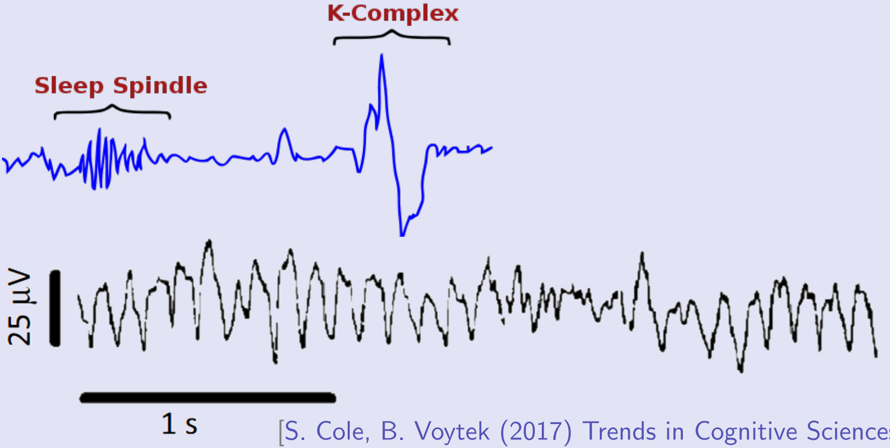
Unsupervised learning for brain signals

$\Rightarrow$ Finding structure in brain recordings!

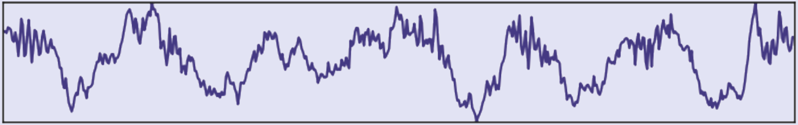
Unsupervised Learning for Electrophysiology
Convolutional Dictionary Model

References

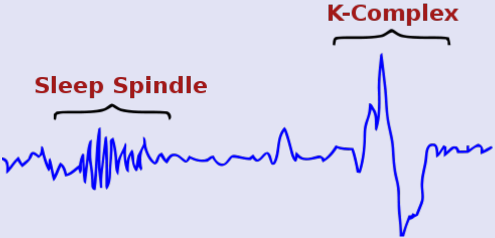
-
-



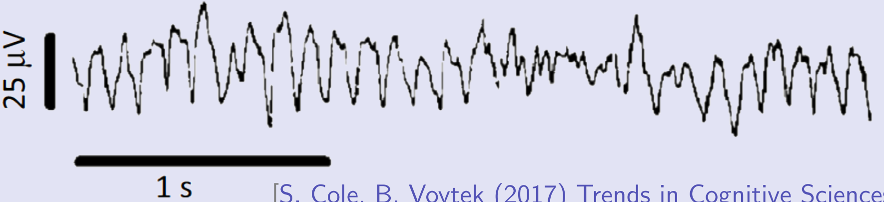
[S. Cole, B. Voytek (2017) Trends in Cognitive Sciences]



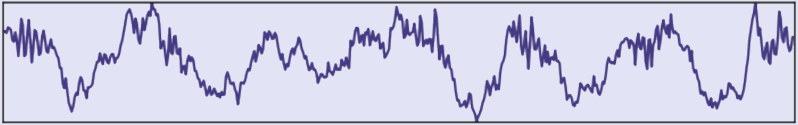
[Dupré la Tour, Tallot, Grabot, Doyère, van Wassenhove, Grenier, Gramfort (2017) PLOS Computational biology]



Neural signals exhibit diverse and complex morphologies

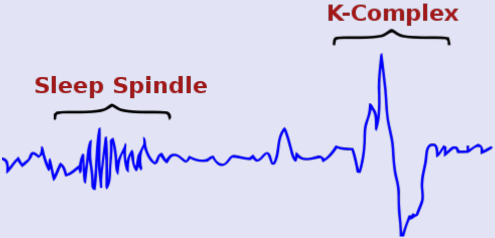


[S. Cole, B. Voytek (2017) Trends in Cognitive Sciences]



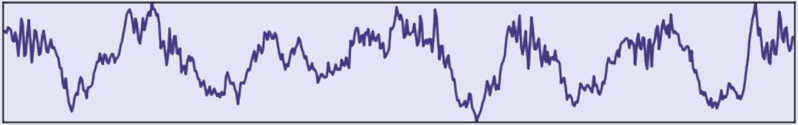
[Dupré la Tour, Tallot, Grabot, Doyère, van Wassenhove, Grenier, Gramfort (2017) PLOS Computational biology]

Neural signals exhibit diverse and complex morphologies



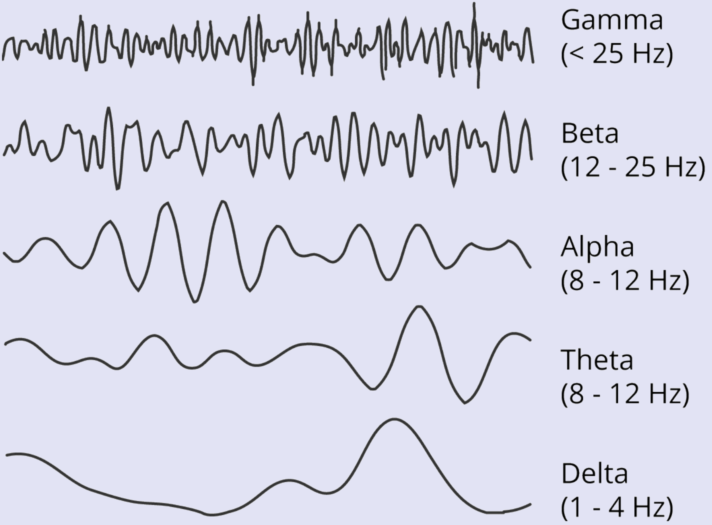
Waveform shape is related to disease
e.g. Parkinson [Jackson et al. (2019)]

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



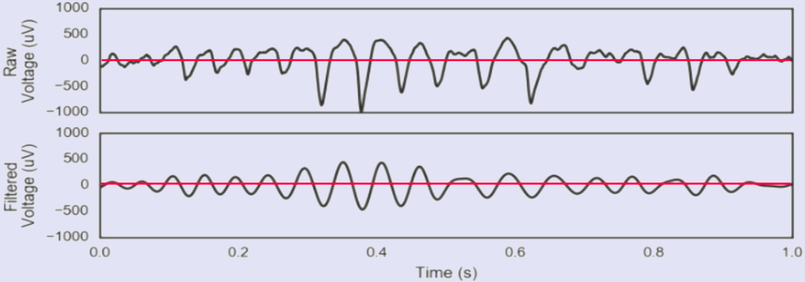
[Dupré la Tour, Tallot, Grabot, Doyère, van Wassenhove, Grenier, Gramfort (2017) PLOS Computational biology]

"Textbook" brain rythm



Linear filtering

After Linear filters, everything looks like a sinusoïd.



⇒ Lose the asymmetry and the shape information.

Fourier Fallacy

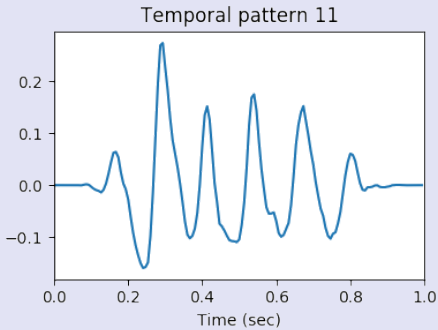
"Even though it may be possible to analyze the complex forms of brain waves into a **number of different sine-wave** frequencies, this may lead only to what might be termed a "**Fourier fallacy**", if one assumes **ad hoc** that all of the necessary frequencies actually occur as periodic phenomena in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

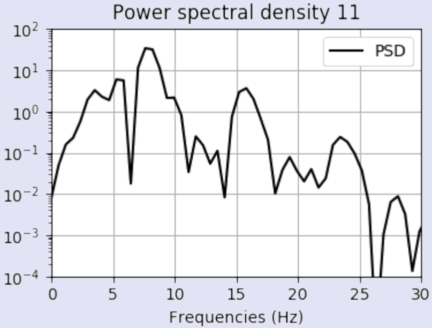
[Jasper (1948)]

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)]

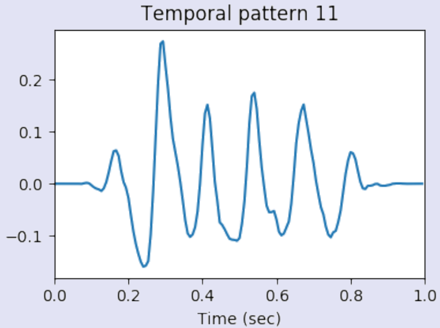


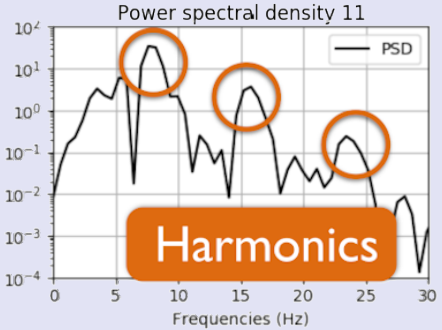


Fourier Fallacy

"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)]

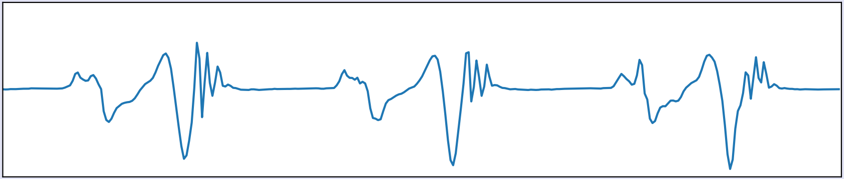




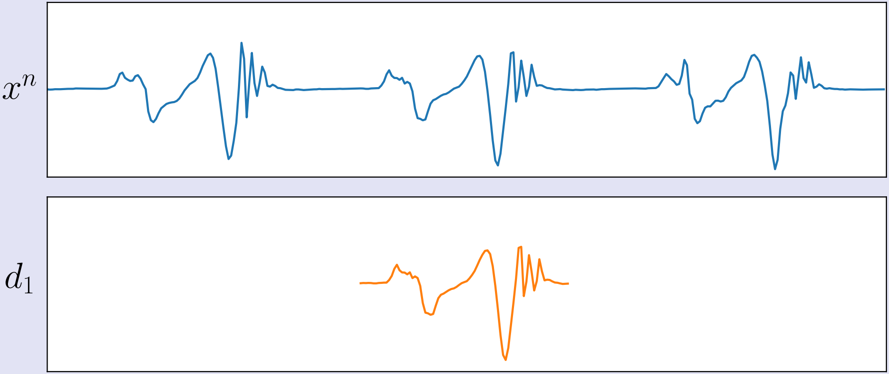
Can't we learn the waveforms?

Local structure in signals

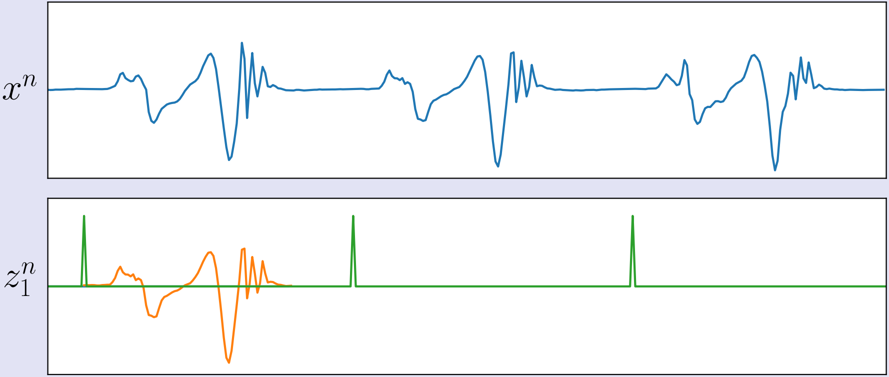
x^n



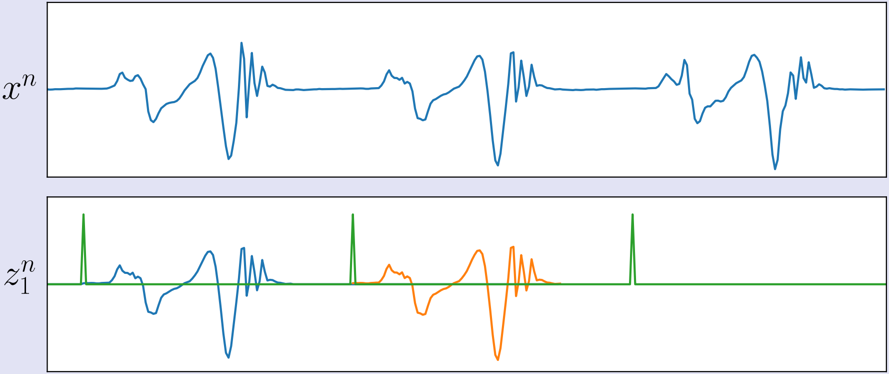
Local structure in signals



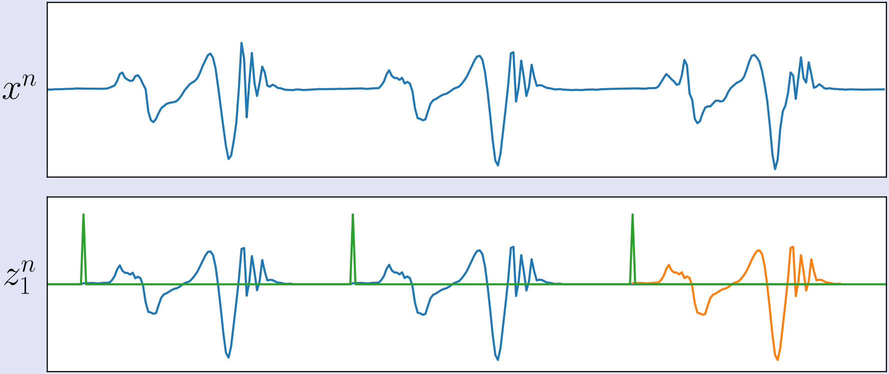
Local structure in signals



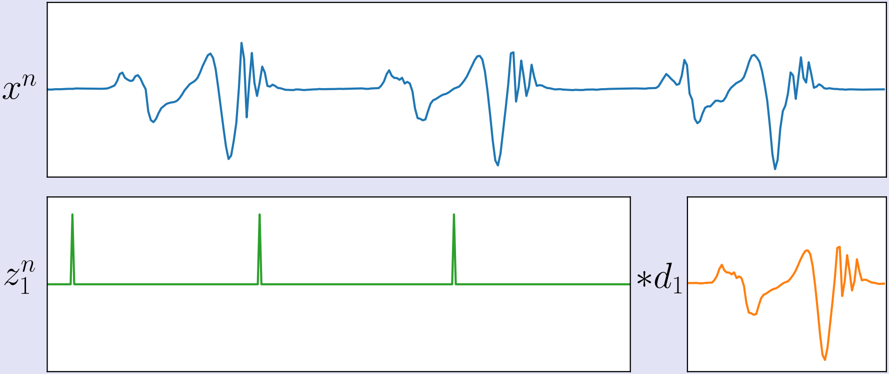
Local structure in signals



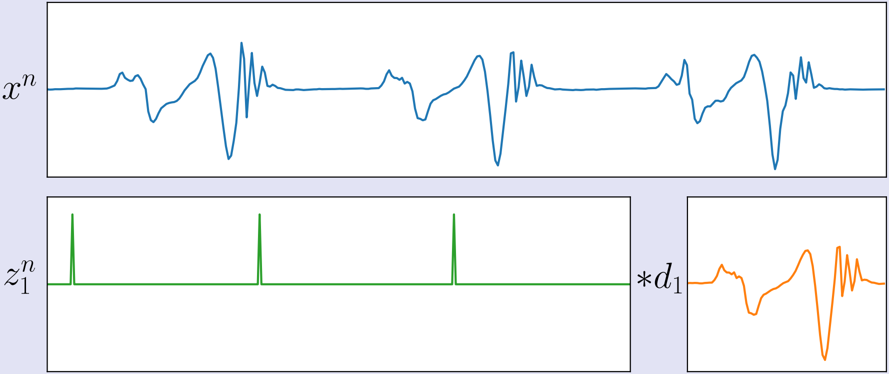
Local structure in signals



Local structure in signals



Local structure in signals

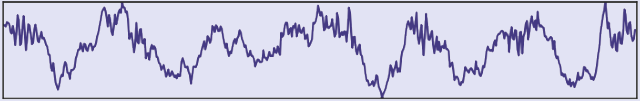


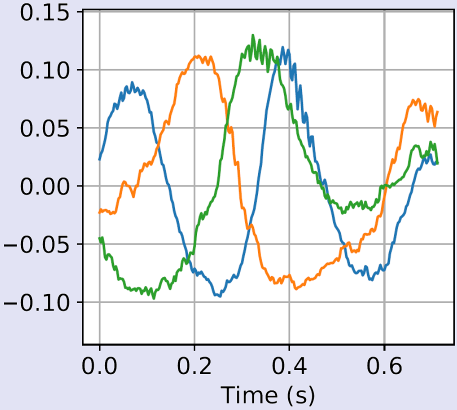
$$\boxed{x^n}[t] = \sum_{k=1}^{K} (\boxed{z_k^n} * \boxed{d_k})[t] + \varepsilon[t]$$

Learned atoms

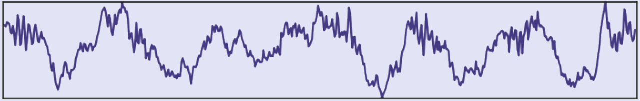
[Jas et al. (2017)]

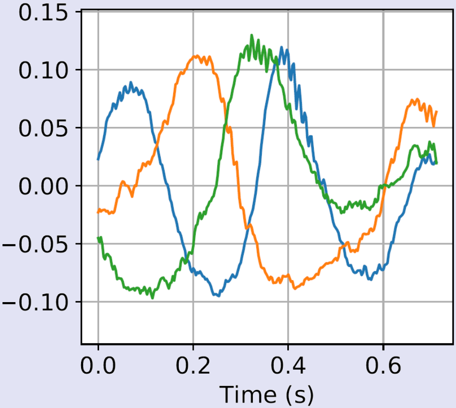
Data:





Data:

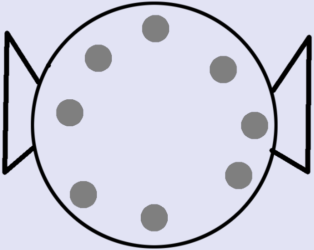




What to do in the case of multivariate signals?

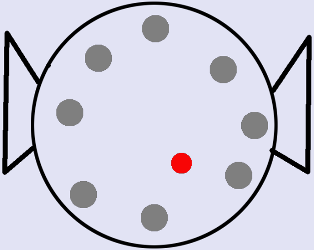
EM wave diffusion

- Recording here with 8 sensors



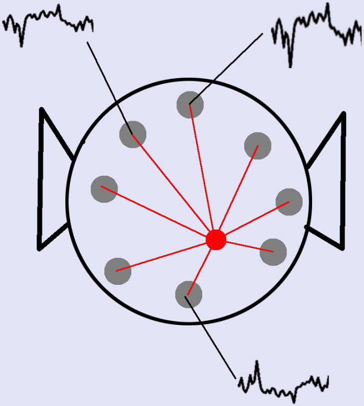
EM wave diffusion

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



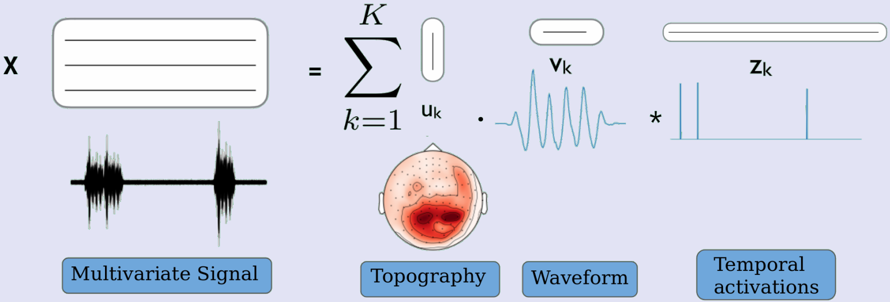
EM wave diffusion

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

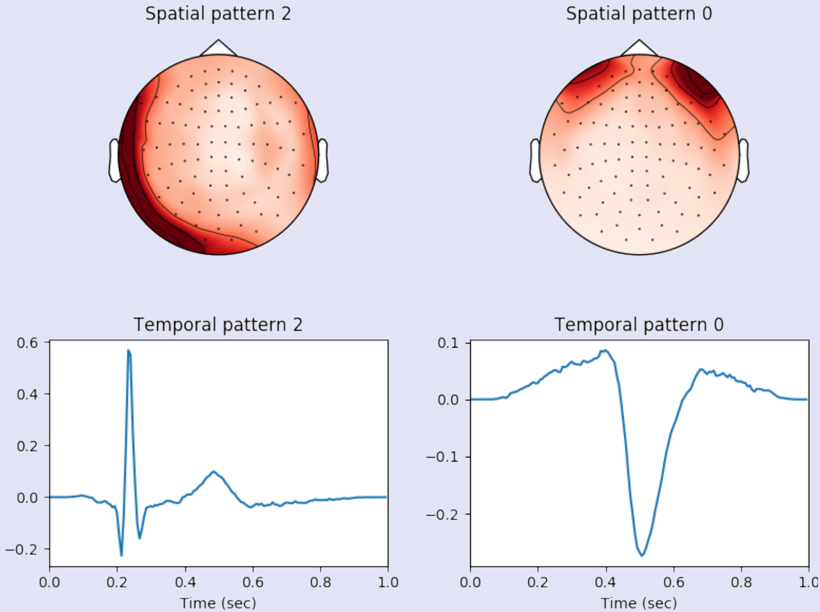


EM wave diffusion

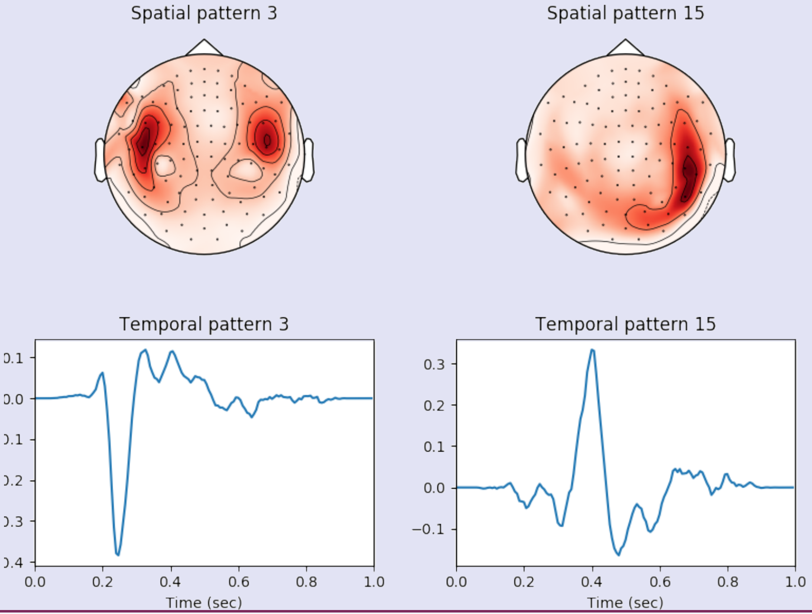
- Recording here with 8 sensors
- EM activity in the brain
- The electric field is spread **linearly** and **instantaneously** over all sensors (Maxwell equations)



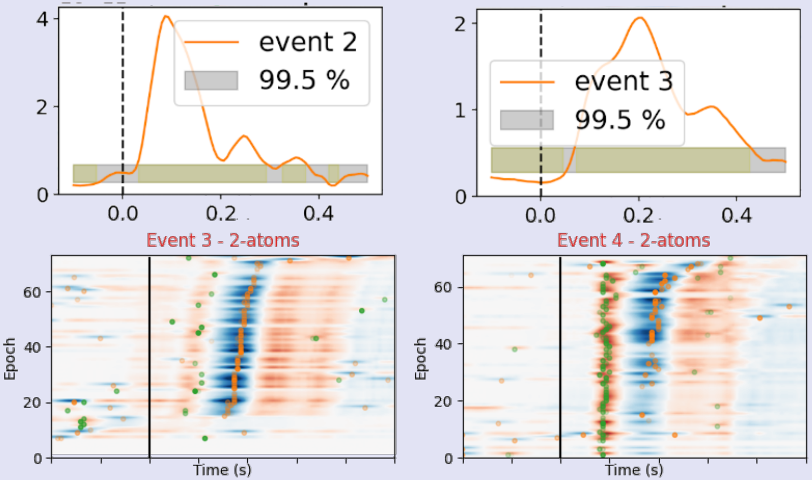
Learned atoms – Artifacts



Learned atoms – Evoked response

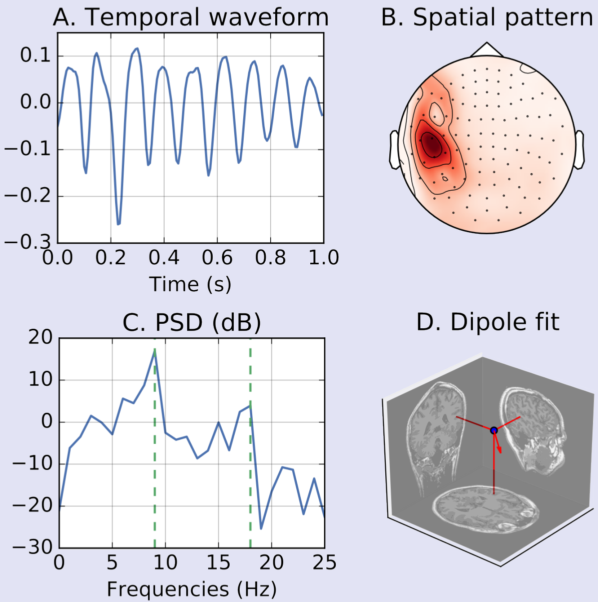


Learned atoms – Evoked response



Learned atoms – Evoked response

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 convolutional sparse coding (CSC), on 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.

Installation

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

(Latest stable version)

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

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]

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

`pip install alphacsc`

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

Installation

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

(Latest stable version)

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

alphacsc 0.3 Models Examples API GitHub Site Search

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
```

Recurring Patterns extraction

Convolutional Dictionary Learning

- Flexible pattern extraction technique,
- Computationally tractable for more and more problems,
- Some application are already beginning to emerge.

Challenges

- Theoretical challenges remains (convergence, recoverability),
- The evaluation (and thus the parameter choices) is still not clear,
- Can give some insight for deep learning models?

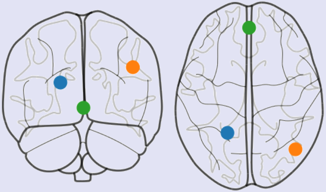
How to highlight temporal dependency?

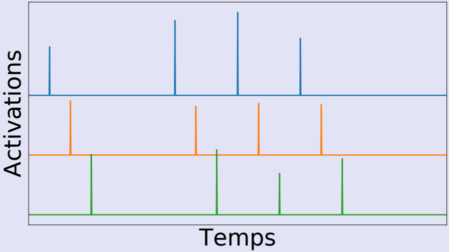
Model inter-atom interactions:

Pénalité ℓ_1

$\Rightarrow$ independent activations

Independent activations





How to highlight temporal dependency?

Model inter-atom interactions:

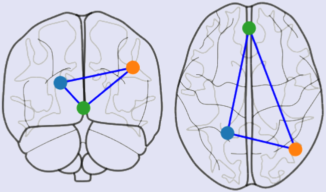
Pénalité ℓ_1

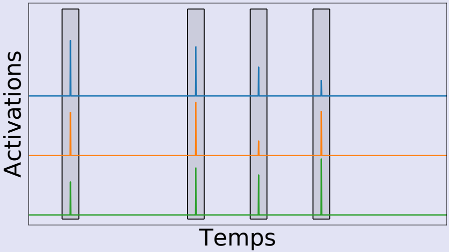
$\Rightarrow$ independent activations

Group penalty (*e.g.* $\ell_{1,2}$)

$\Rightarrow$ simultaneous activations

Simultaneous activations





How to highlight temporal dependency?

Model inter-atom interactions:

Pénalité ℓ_1

$\Rightarrow$ independent activations

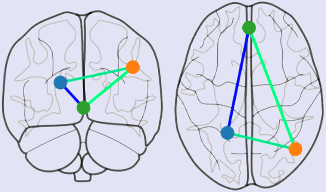
Group penalty (*e.g.* $\ell_{1,2}$)

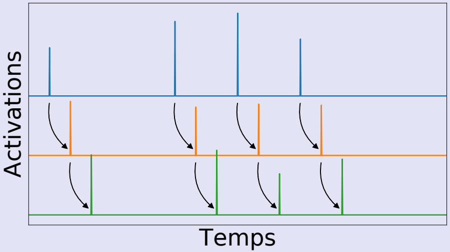
$\Rightarrow$ simultaneous activations

Point process ponctuels (*e.g.* Hawkes)

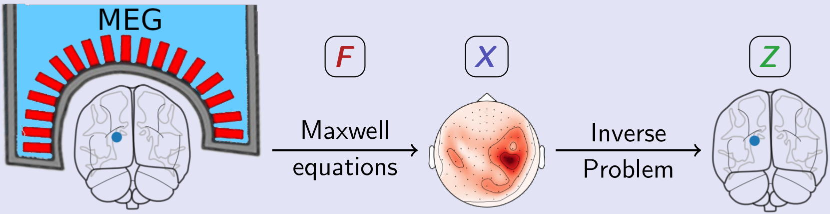
$\Rightarrow$ temporal interactions

Temporal interactions





Deep Learning for Inverse problem



Inverse Problem: $\operatorname{argmin}_{Z} \|X - FZ\|_2^2 + \mathcal{R}(Z)$

- Fast iterative solvers to compute Z [Massias et al. (2017)]
- Use of deep learning to inverse this system [Moreau et al. (2017); Ablin et al. (2019)]

Thanks!

Code available online:

alphacsc : `alphacsc.github.io`

Slides are on my web page:

`tommoral.github.io` @tomamoral