Ph.D. Defense – École Normale Supérieure Paris-Saclay

Thomas Moreau

Représentations Convolutives

Dec. 19, 2017

Committee:

Stéphanie Allassonière	Examiner	Alexandre Gramfort	Examiner
Pierre-Paul Vidal	Examiner	René Vidal	Referee
Stéphane Mallat	Referee	Julien Mairal	Referee
Nicolas Vayatis	Supervisor	Laurent Oudre	Co-supervisor

Context

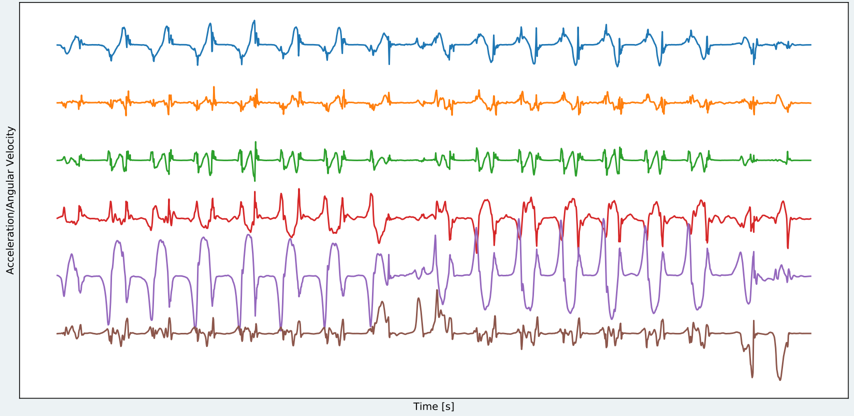
Collaboration with the Cognac-G laboratory:

- Started in 2014
- Gathers MDs and mathematicians
- Goal:

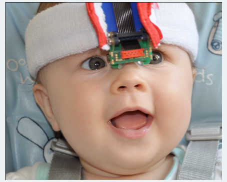
  $\Rightarrow$ Quantification of human and animal phenome (behavior, movement,...).

Signals from human walking

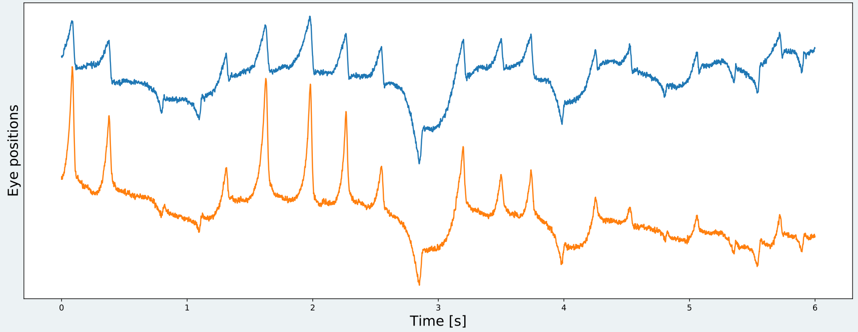
- Accelerometer
- Gyrometer
- Magnetometer



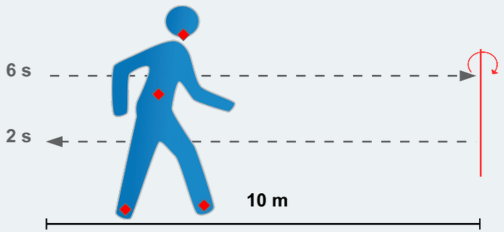
Oculometric signals

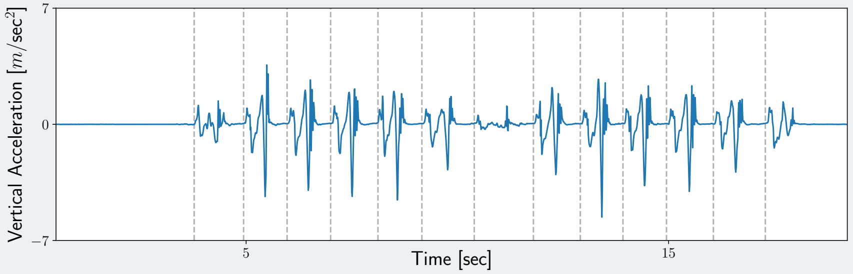


- Eye tracker
- Photoreflectometry Infra-rouge



Signals from human walking





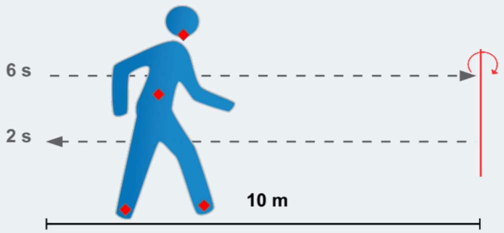
Properties

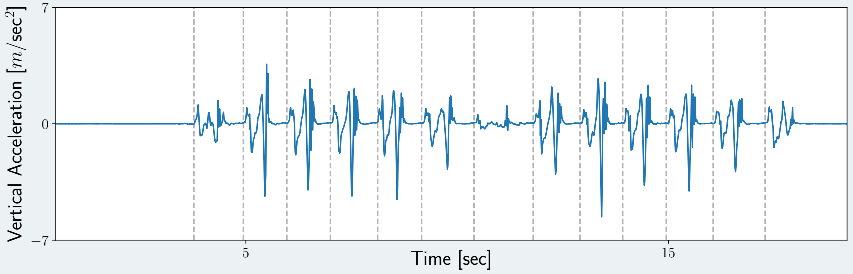
- Routine test
- Standardized protocol
- Signal with 24 channels (4x6)
- Minutes of signal recorded at 100Hz (10^4 samples)

Requirements

- Automatized analysis
- Robustness on a large basis
- Quick results
- Interpretability

Signals from human walking





Challenges

- High-variability:
  Healthy, Parkinsonian, Strokes,...
- Non-stationary:
  Multiple exercises in one signal
- Need interpretability:
  Link to medical analysis
- Possibly long signals:
  Ambulatory studies

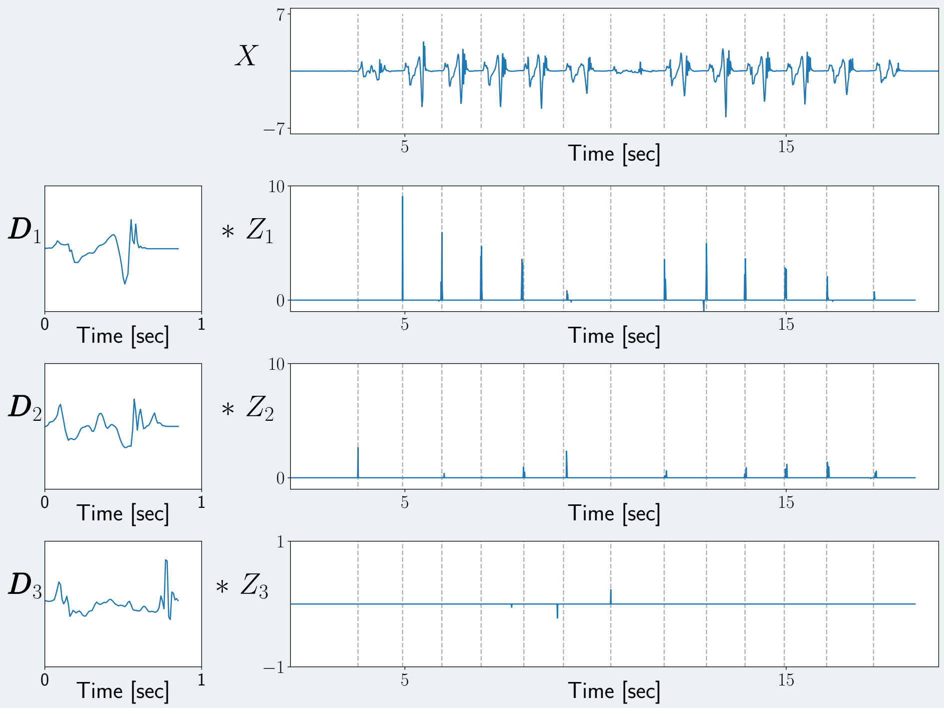
Notation

- X is a signal of length T
- $\mathcal{E}$ is a noise signal of length T
- $\boldsymbol{D}$ is a set of K patterns of length W
- Z is a signal of length $L = T - W + 1$ in $\mathbb{R}^K$

Sparse Convolutional model:

$$X[t] = \sum_{k=1}^{K} (\boldsymbol{D}_k * Z_k)[t] + \mathcal{E}[t]$$

with Z sparse. Few of its coefficients are non-zero.



Notation

- x is a vector in $\mathbb{R}^T$
- ϵ is a noise vector in $\mathbb{R}^T$
- D is a matrix in $\mathbb{R}^{T \times LK}$
- z is a coding vector in $\mathbb{R}^{LK}$

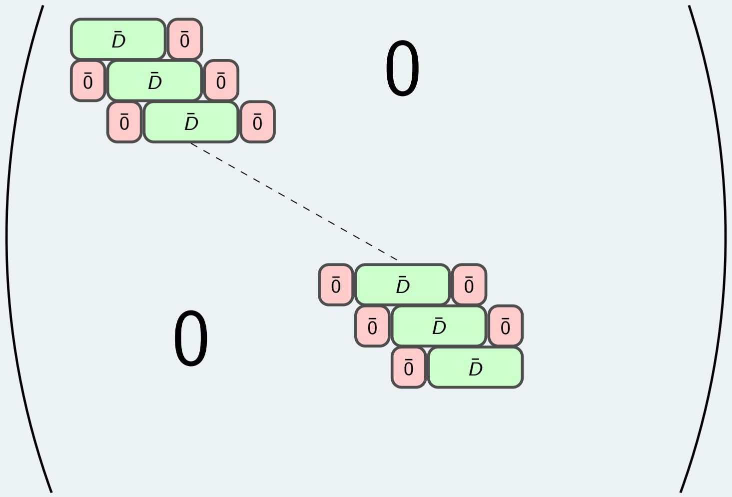
Sparse Linear model:

$$x = Dz + \epsilon$$

with z sparse. Few of its coefficients are non-zero.

Link with convolutional model

$D =$



Convolutional Sparse Model

Dictionary learning optimization problem for $\{X^{[n]}\}_{n=1}^N$

$$(Z^*, \boldsymbol{D}^*) = \operatorname*{argmin}_{Z,\boldsymbol{D}} \frac{1}{N}\sum_{n=1}^{N} \underbrace{\|X^{[n]} - \sum_{k=1}^{K} \boldsymbol{D}_k * Z_k^{[n]}\|_2^2}_{E(Z)\text{ data fit}} + \underbrace{\lambda\|Z^{[n]}\|_1 + \mathbf{1}_\Omega(D)}_{\text{penalizations}}$$

with a constraint set Ω and a regularization parameter $\lambda > 0$.

This problem is non-convex and is generally solved using an **alternate minimization**.

[Engan et al., 1999; Grosse et al., 2007]

Convolutional Sparse Model

Dictionary learning optimization problem for $\{X^{[n]}\}_{n=1}^N$

$$(Z^*, \boldsymbol{D}^*) = \underset{Z,\boldsymbol{D}}{\operatorname{argmin}} \frac{1}{N}\sum_{n=1}^N \underbrace{\|X^{[n]} - \sum_{k=1}^K \boldsymbol{D}_k * Z_k^{[n]}\|_2^2}_{E(Z) \text{ data fit}} + \underbrace{\lambda\|Z^{[n]}\|_1 + \mathbf{1}_\Omega(D)}_{\text{penalizations}}$$

with a constraint set Ω and a regularization parameter $\lambda > 0$.

This problem is non-convex and is generally solved using an **alternate minimization**.

[Engan et al., 1999; Grosse et al., 2007]

$\boldsymbol{D}$-step: Dictionary updates

$\rightarrow$ Z fixed, update $\boldsymbol{D}$

$$\boldsymbol{D}^* = \underset{\boldsymbol{D}}{\operatorname{argmin}} \frac{1}{N}\sum_{n=1}^N \|X^{[n]} - \sum_{k=1}^K \boldsymbol{D}_k * Z_k^{[n]}\|_2^2 + \mathbf{1}_\Omega(D)$$

Related Algorithms:

- Proximal Gradient Descent (PDG) [Rockafellar, 1976]
- Accelerated PGD [Nesterov, 1983]
- Block Coordinate Descent [Mairal et al., 2009]
- Alternated Direction Method of Multiplier (ADMM) [Gabay and Mercier, 1976]

Convolutional Sparse Model

Dictionary learning optimization problem for $\{X^{[n]}\}_{n=1}^N$

$$(Z^*, \boldsymbol{D}^*) = \underset{Z,\boldsymbol{D}}{\operatorname{argmin}} \frac{1}{N} \sum_{n=1}^N \underbrace{\|X^{[n]} - \sum_{k=1}^K \boldsymbol{D}_k * Z_k^{[n]}\|_2^2}_{E(Z) \text{ data fit}} + \underbrace{\lambda \|Z^{[n]}\|_1 + \mathbf{1}_\Omega(D)}_{\text{penalizations}}$$

with a constraint set Ω and a regularization parameter $\lambda > 0$.

This problem is non-convex and is generally solved using an **alternate minimization**.

[Engan et al., 1999; Grosse et al., 2007]

Z-step: Sparse coding

$\rightarrow$ $\boldsymbol{D}$ fixed, update Z

$$Z^{[n],*} = \underset{Z^{[n]}}{\operatorname{argmin}} \|X^{[n]} - \sum_{k=1}^K \boldsymbol{D}_k * Z_k^{[n]}\|_2^2 + \lambda \|Z^{[n]}\|_1$$

$\Rightarrow$ Independent for each $n \in [\![1, N]\!]$

Related Algorithms:

- Iterative Soft-Thresholding Algorithm (ISTA) [Daubechies et al., 2004]
- Fast ISTA [Beck and Teboulle, 2009]
- Alternated Direction Method of Multiplier (ADMM) [Gabay and Mercier, 1976]
- Coordinate Descent (CD) [Friedman et al., 2007]

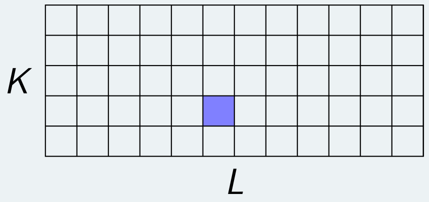
Coordinate Descent [Friedman et al., 2007]

Select a coordinate (k, t) and update it to the value

$$Z'_k[t] = \operatorname*{argmin}_{Z_k[t]} \|X - \sum_{k=1}^{K} \boldsymbol{D}_k * Z_k\|_2^2 + \lambda\|Z\|_1$$

with all other coordinates fixed.

Update one coordinate $Z_k[t]$



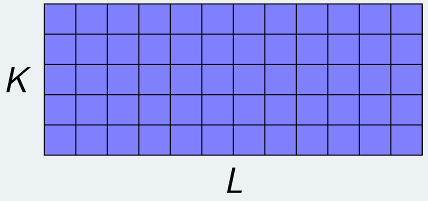
ISTA [Daubechies et al., 2004]

Proximal Gradient descent for Sparse Coding:

$$Z^{(q+1)} = \mathrm{Sh}\left(Z^{(q)} - \alpha\nabla E(Z^{(q)}), \alpha\lambda\right)$$

with $\mathrm{Sh}(Z_k[t], \lambda) = \mathrm{sign}(Z_k[t])\max(|Z_k[t]| - \lambda, 0)$.

Update all coordinates of Z



Part I: Distributed Coordinate Descent

Coordinate descent only update one coordinate at each iteration.

$\Rightarrow$ Not efficient for convolutional model.

We could update M coefficients in **parallel**.

General Parallel Coordinate Descent:

- Synchronous: Scherrer et al. [2012], Bradley et al. [2011].
- Asynchronous: Yu et al. [2012], Low et al. [2012].

Can we do better with the structure of our problem?

Part I: Distributed Coordinate Descent

Coordinate descent only update one coordinate at each iteration.

$\Rightarrow$ Not efficient for convolutional model.

We could update M coefficients in **parallel**.

General Parallel Coordinate Descent:

- Synchronous: Scherrer et al. [2012], Bradley et al. [2011].
- Asynchronous: Yu et al. [2012], Low et al. [2012].

Can we do better with the structure of our problem?

Part II: Adaptive Optimization

We have to solve N independent problems with a common structure $\boldsymbol{D}$,

$$Z^{[n],*} = \underset{Z^{[n]}}{\operatorname{argmin}} \|X^{[n]} - \sum_{k=1}^{K} \boldsymbol{D}_k * Z_k^{[n]}\|_2^2 + \lambda \|Z^{[n]}\|_1$$

Can we use this structure to accelerate the resolution?

Yes, with the Learned ISTA. [Gregor and Lecun, 2010]

Why does it work?

PART I

Accelerating Convolutional Sparse Coding: DICOD

References

Moreau, T., Oudre, L., and Vayatis, N. (2015a). Distributed Convolutional Sparse Coding via Message Passing Interface (MPI). In *NIPS Workshop Nonparametric Methods for Large Scale Representation Learning*

Moreau, T., Oudre, L., and Vayatis, N. (2017). Distributed Convolutional Sparse Coding. *arXiv preprint*, arXiv:1705(10087)

Parallel Coordinate Descent

Coordinate descent only update one coordinate at each iteration.

$\Rightarrow$ Not efficient for convolutional model.

We could update M coefficients in **parallel**.

Existing Parallel Coordinate Descent:

- Synchronous: Scherrer et al. [2012], Bradley et al. [2011].
- Asynchronous: Yu et al. [2012], Low et al. [2012].

Can we do better with the structure of our problem?

- Asynchronous updates
- Communication efficient
- Parameter-free
- Optimal coordinate updates

Coordinate Descent (CD)

Minimize

$$Z^* = \operatorname*{argmin}_Z \|X - \sum_{k=1}^{K} \boldsymbol{D}_k * Z_k\|_2^2 + \lambda \|Z\|_1$$

Update one coordinate at each iteration.

1. Select a coordinate (k_0, t_0) to update.

Three algorithms:

- Cyclic updates; $\mathcal{O}(1)$ [Friedman et al., 2007]
- Random updates; $\mathcal{O}(1)$ [Nesterov, 2012]
- Greedy updates; $\mathcal{O}(KL)$ [Osher and Li, 2009]

Coordinate Descent (CD)

Minimize

$$Z^* = \underset{Z}{\operatorname{argmin}} \|X - \sum_{k=1}^{K} \boldsymbol{D}_k * Z_k\|_2^2 + \lambda \|Z\|_1$$

Update one coordinate at each iteration.

1. Select a coordinate (k_0, t_0) to update.
2. Compute a new value $Z'_{k_0}[t_0]$ for this coordinate

For convolutional CD, we can use optimal updates:

$$Z'_{k_0}[t_0] = \frac{1}{\|\boldsymbol{D}_{k_0}\|_2^2} \mathrm{Sh}(\beta_{k_0}[t_0], \lambda),$$

with $\mathrm{Sh}(y, \lambda) = \mathrm{sign}(y)(|y| - \lambda)_+$.

Kavukcuoglu et al. [2010] showed this can be done efficiently, with $\mathcal{O}(KW)$ operations.

$\Rightarrow$ Local operations

Coordinate Descent (CD)

Minimize

$$Z^* = \underset{Z}{\operatorname{argmin}} \|X - \sum_{k=1}^{K} \boldsymbol{D}_k * Z_k\|_2^2 + \lambda \|Z\|_1$$

Update one coordinate at each iteration.

1. Select a coordinate (k_0, t_0) to update.
2. Compute a new value $Z'_{k_0}[t_0]$ for this coordinate

$\Rightarrow$ Converges to the optimal point for CSC problem.

For convolutional CD, we can use optimal updates:

$$Z'_{k_0}[t_0] = \frac{1}{\|\boldsymbol{D}_{k_0}\|_2^2} \operatorname{Sh}(\beta_{k_0}[t_0], \lambda),$$

with $\operatorname{Sh}(y, \lambda) = \operatorname{sign}(y)(|y| - \lambda)_+$.

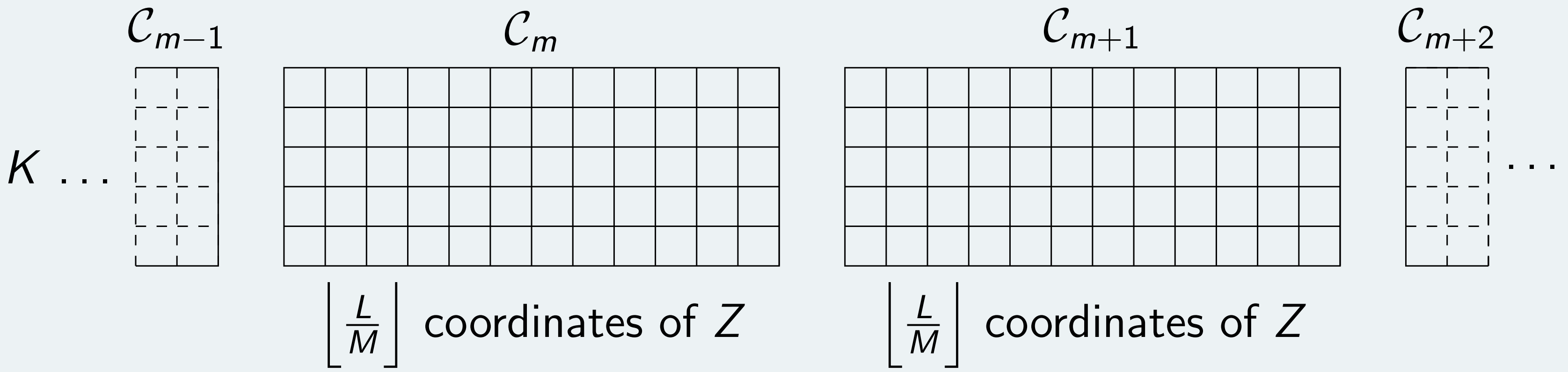
Kavukcuoglu et al. [2010] showed this can be done efficiently, with $\mathcal{O}(KW)$ operations.

$\Rightarrow$ Local operations

Our algorithm DICOD with M cores, principles

Z is the coding signal of length L.

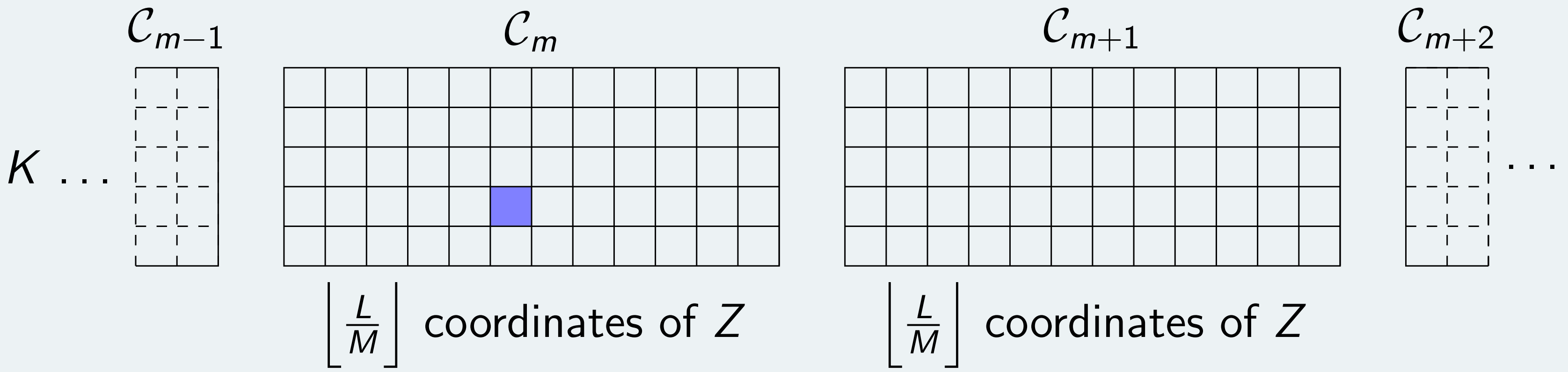
Each core $\mathcal{C}_m$ is responsible for the updates of a segment $\left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}$.



Our algorithm DICOD with M cores, principles

Z is the coding signal of length L.

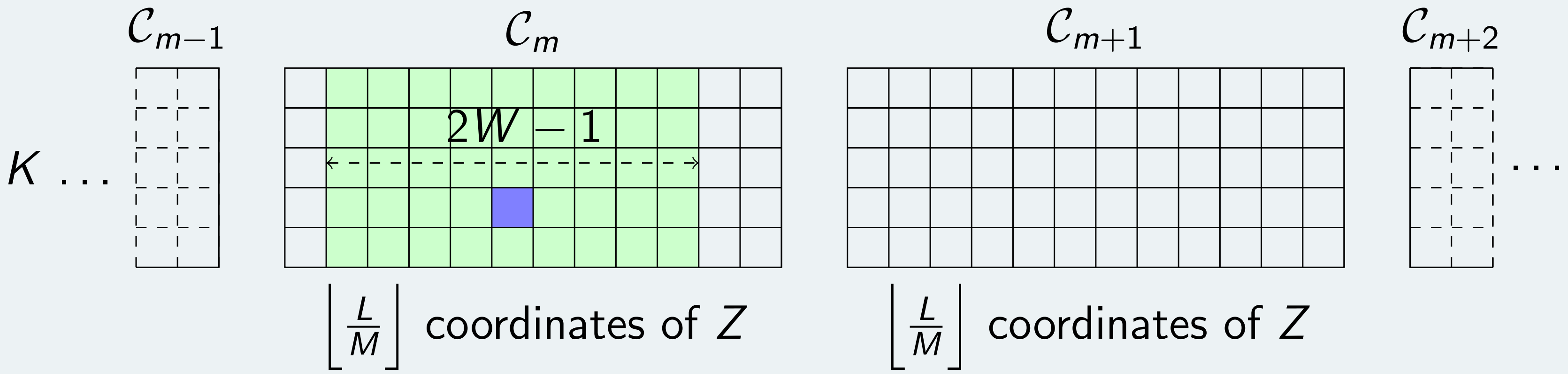
Each core $\mathcal{C}_m$ is responsible for the updates of a segment $\left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}$.



Our algorithm DICOD with M cores, principles

Z is the coding signal of length L.

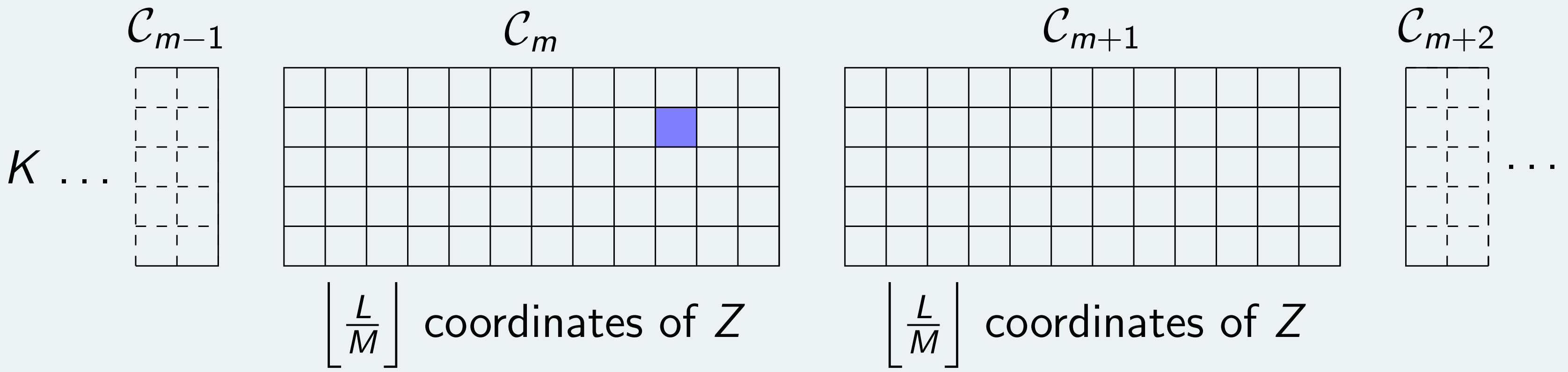
Each core $\mathcal{C}_m$ is responsible for the updates of a segment $\left\{m\left\lfloor\frac{L}{M}\right\rfloor, \dots (m+1)\left\lfloor\frac{L}{M}\right\rfloor - 1\right\}$.



Our algorithm DICOD with M cores, principles

Z is the coding signal of length L.

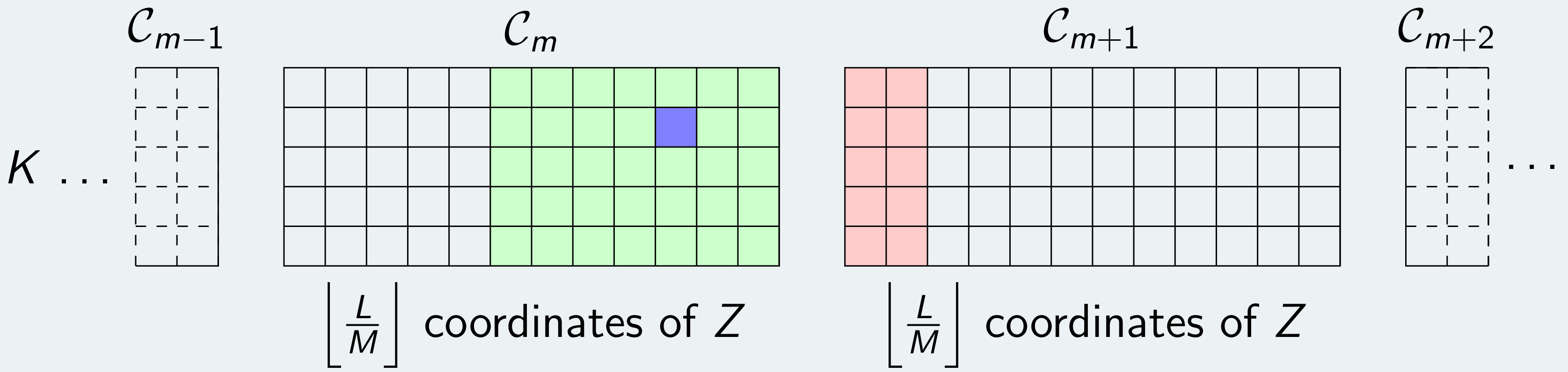
Each core $\mathcal{C}_m$ is responsible for the updates of a segment $\left\{m\left\lfloor\frac{L}{M}\right\rfloor, \dots (m+1)\left\lfloor\frac{L}{M}\right\rfloor - 1\right\}$.



Our algorithm DICOD with M cores, principles

Z is the coding signal of length L.

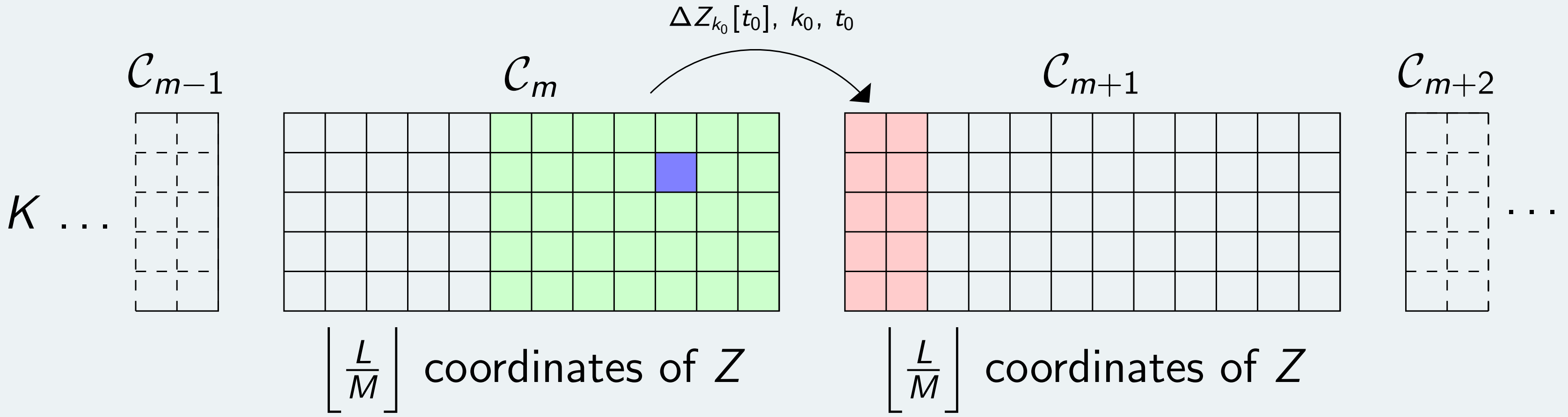
Each core $\mathcal{C}_m$ is responsible for the updates of a segment $\left\{ m\left\lfloor\frac{L}{M}\right\rfloor, \dots (m+1)\left\lfloor\frac{L}{M}\right\rfloor - 1\right\}$.



Our algorithm DICOD with M cores, principles

Z is the coding signal of length L.

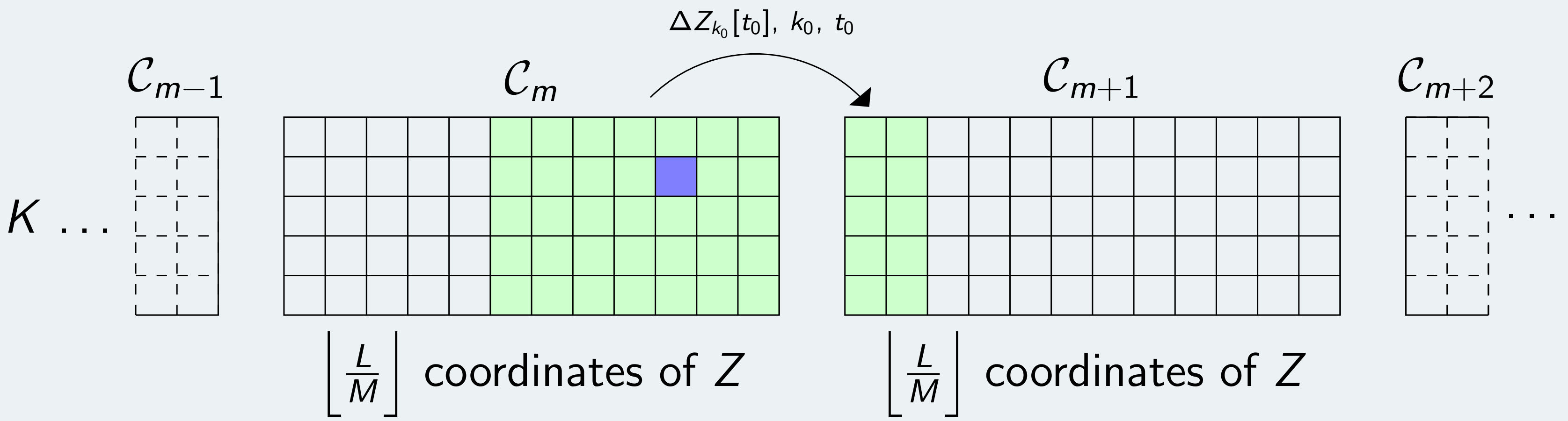
Each core $\mathcal{C}_m$ is responsible for the updates of a segment $\left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}$.



Our algorithm DICOD with M cores, principles

Z is the coding signal of length L.

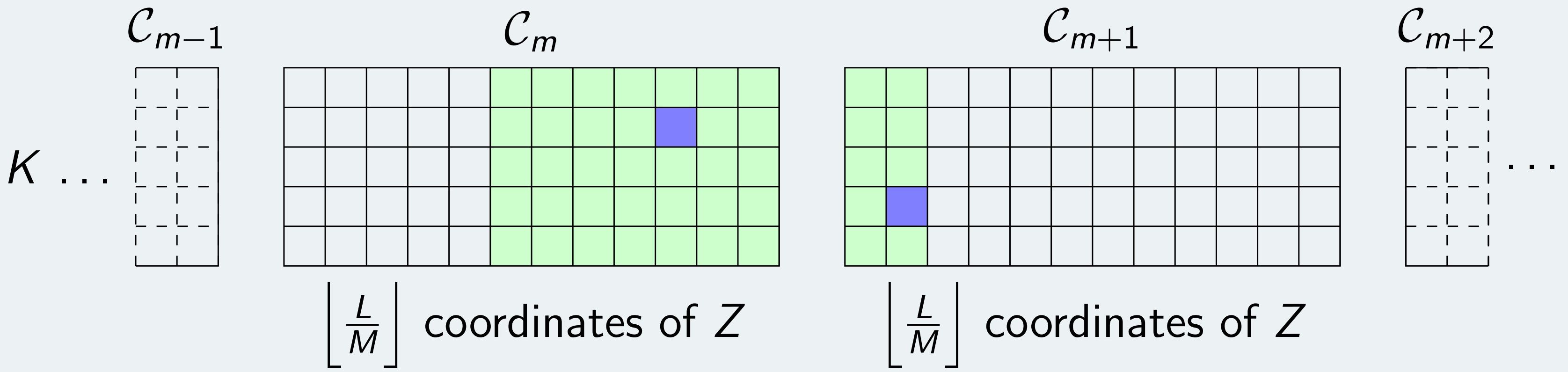
Each core $\mathcal{C}_m$ is responsible for the updates of a segment $\left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}$.



Our algorithm DICOD with M cores, principles

Z is the coding signal of length L.

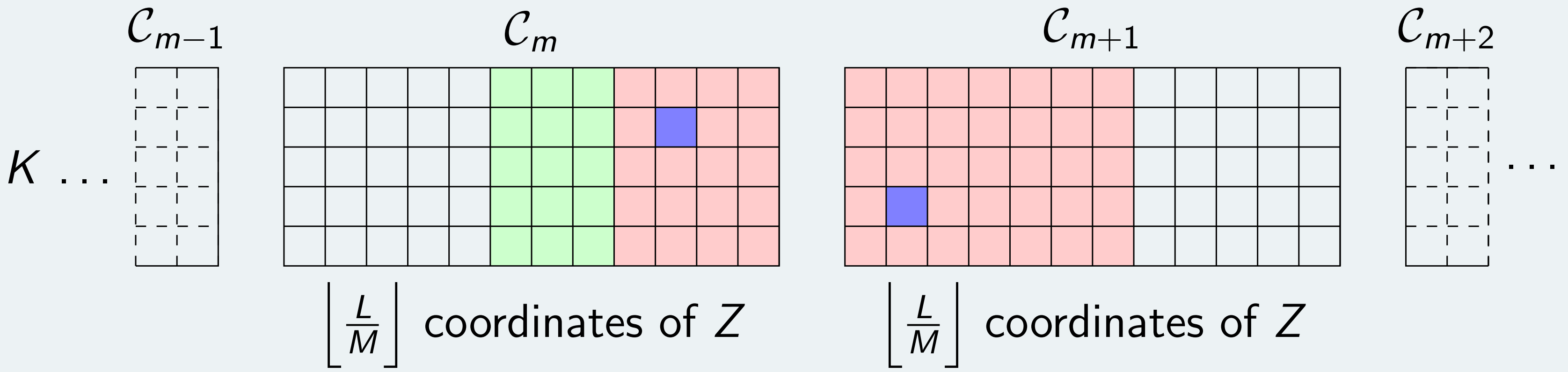
Each core $\mathcal{C}_m$ is responsible for the updates of a segment $\left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}$.



Our algorithm DICOD with M cores, principles

Z is the coding signal of length L.

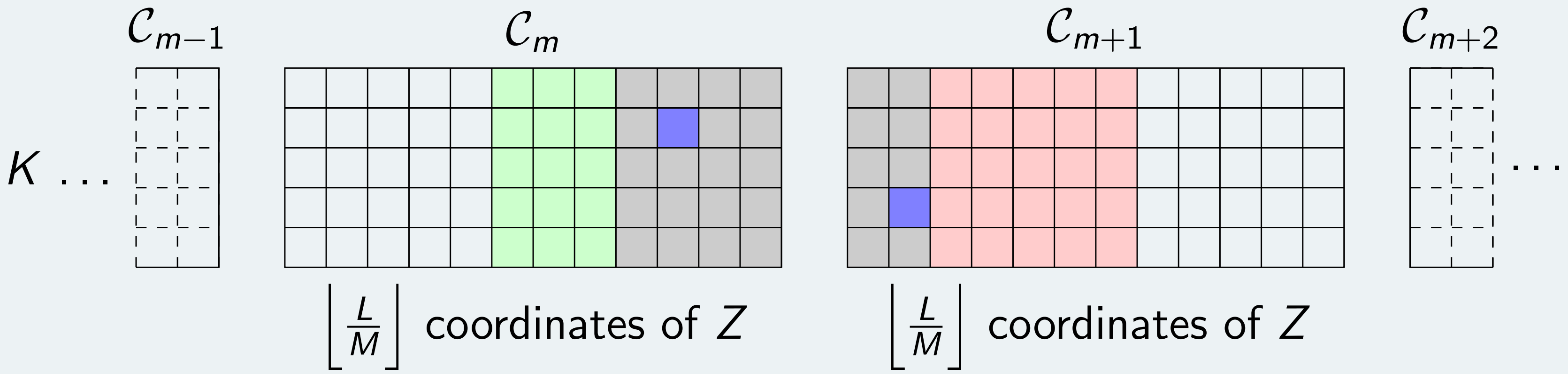
Each core $\mathcal{C}_m$ is responsible for the updates of a segment $\left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}$.



Our algorithm DICOD with M cores, principles

Z is the coding signal of length L.

Each core $\mathcal{C}_m$ is responsible for the updates of a segment $\left\{ m \left\lfloor \frac{L}{M} \right\rfloor, \dots (m+1) \left\lfloor \frac{L}{M} \right\rfloor - 1 \right\}$.



DICOD, algorithm and convergence result

DICOD pseudo-code

Input: $\boldsymbol{D}, X$, parameter $\delta > 0$
In parallel for $m = 1 \cdots M$
For all (k, t) in $\mathcal{C}_m$, initialize $\beta_k[t]$ and $Z_k[t]$
repeat
  Receive messages and update β
  $\forall (k, t) \in \mathcal{C}_m$, compute $Z'_k[t]$
  Choose $(k_0, t_0) = \arg \max_{(k,t)\in\mathcal{C}_m} |\Delta Z_k[t]|$
  Update β and $Z_{k_0}[t_0] \leftarrow Z'_{k_0}[t_0]$
  if $t_0 - mL_M < W$ **then**
    Send $(k_0, t_0, \Delta Z_{k_0}[t_0])$ to core $m - 1$
  if $(m+1)L_M - t_0 < W$ **then**
    Send $(k_0, t_0, \Delta Z_{k_0}[t_0])$ to core $m + 1$
until for all cores, $|\Delta Z_{k_0}[t_0]| < \delta$

Theorem (Convergence of DICOD)

We consider the following assumptions:

H1: If the cross correlation between atoms of $\boldsymbol{D}$ is strictly smaller than 1.

H2: No cores stop before all its coefficients are optimal.

H3: If the delay in communication between the processes is inferior to the update time.

Under these assumptions, the DICOD algorithm converges asymptotically to the optimal solution Z^*. 2.

Numerical Experiments

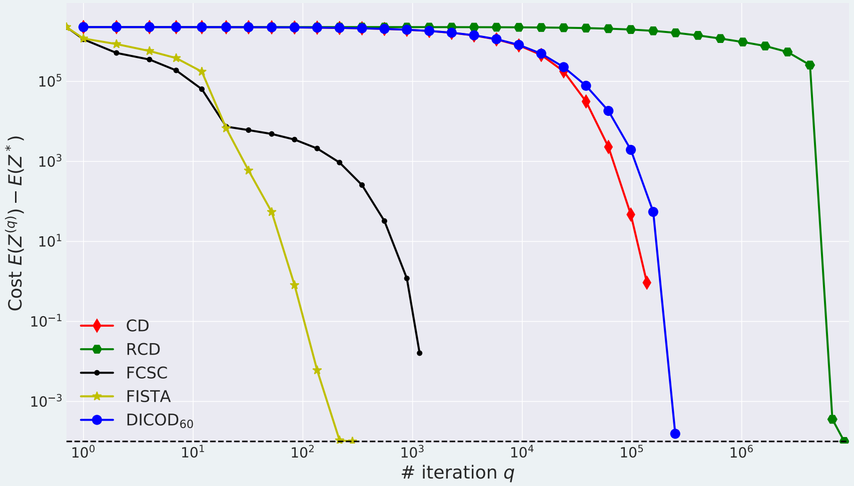
Test on long signals generated with Bernoulli-Gaussian coding signal Z and a Gaussian dictionary $\boldsymbol{D}$.

Fixed $K = 25$, $W = 200$ and $T = 600 * W$,

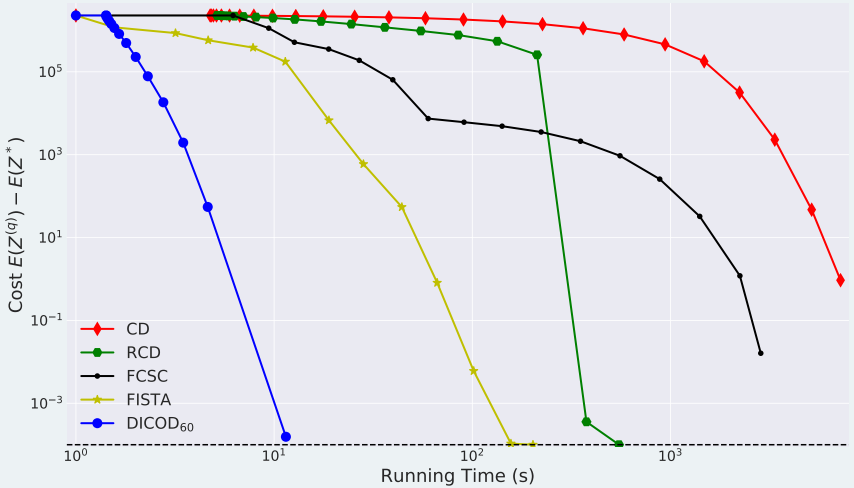
Compare the evolution of the cost function with the number of iteration and the time.

Algorithms implemented for benchmark

- Coordinate Descent (CD) [Kavukcuoglu et al., 2010]
- Randomized Coordinate Descent (RCD) [Nesterov, 2012]
- Fast Convolutional Sparse Coding (FCSC) [Bristow et al., 2013]
- Fast Ierative Soft-Thresholding Algorithm (FISTA) [Chalasani et al., 2013; Wohlberg, 2016]
- DICOD with 60 cores



Cost as a function of the iterations



Cost as a function of the runtime

Speed up analysis

Two sources of acceleration:

- Perform M updates in parallel,
- Each update is computed on a segment of size $\frac{L}{M}$
  Iteration complexity of $\mathcal{O}\left(K\frac{L}{M}\right)$ instead of $\mathcal{O}\left(KL\right)$

Limitations:

- Interfering updates, with probability $\alpha^2 = \left(\frac{WM}{T}\right)^2$

$$\mathbb{E}[Q_{dicod}] \underset{\alpha\to 0}{\gtrsim} M(1-2\alpha^2M^2+\mathcal{O}(\alpha^4M^4)) \ .$$

- Cost of the update of β in $\mathcal{O}\left(KW\right)$

Speed up analysis

Two sources of acceleration:

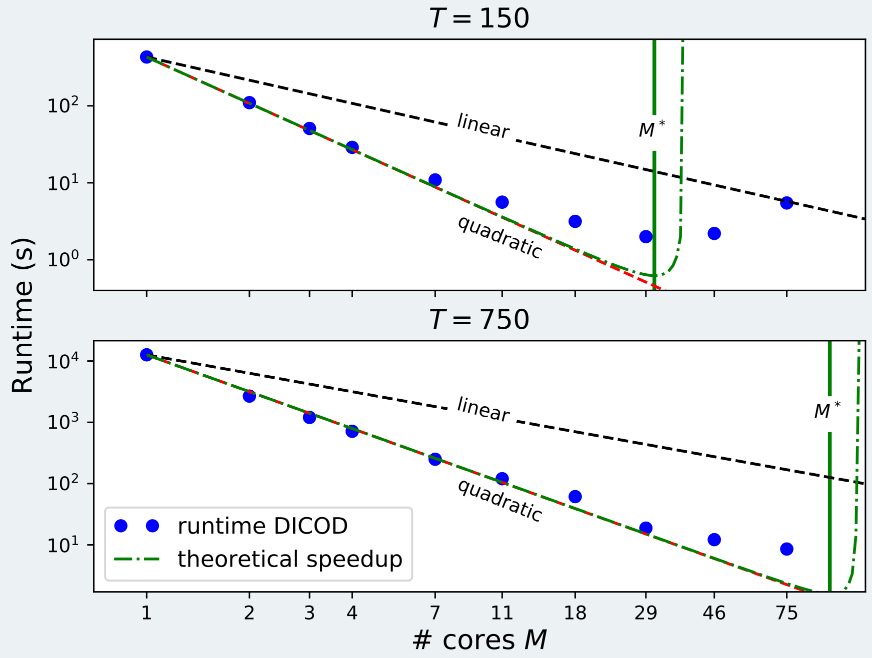
- Perform M updates in parallel,
- Each update is computed on a segment of size $\frac{L}{M}$
  Iteration complexity of $\mathcal{O}\left(K\frac{L}{M}\right)$ instead of $\mathcal{O}\left(KL\right)$

Limitations:

- Interfering updates, with probability $\alpha^2 = \left(\frac{WM}{T}\right)^2$

$$\mathbb{E}[Q_{dicod}] \underset{\alpha\to 0}{\gtrsim} M(1-2\alpha^2M^2+\mathcal{O}(\alpha^4M^4)) \ .$$

- Cost of the update of β in $\mathcal{O}\left(KW\right)$



Runtime as a function of the number of cores M

Contributions

- A novel algorithm DICOD: distributed algorithm efficient to solve the CSC problem,
- Theoretical guarantees: convergence to the optimal solution,
- Complexity analysis: achieves a super-linear speedup

Future work

- 2D convolutions: extension of this algorithm to images,
- Local penalization: extension of this algorithm for localized penalties.

PART II

Adaptive Sparse Coding: FacNet

Reference

Moreau, T. and Bruna, J. (2017). Understanding Neural Sparse Coding with Matrix Factorization. In *International Conference on Learning Representation (ICLR)*

Adaptive Optimization

We have to solve N problems with a common structure $\boldsymbol{D}$.

$$Z^{[n],*} = \underset{Z^{[n]}}{\operatorname{argmin}} \|X^{[n]} - \sum_{k=1}^{K} \boldsymbol{D}_k * Z_k^{[n]}\|_2^2 + \lambda \|Z^{[n]}\|_1$$

Can we use this structure to accelerate the resolution?

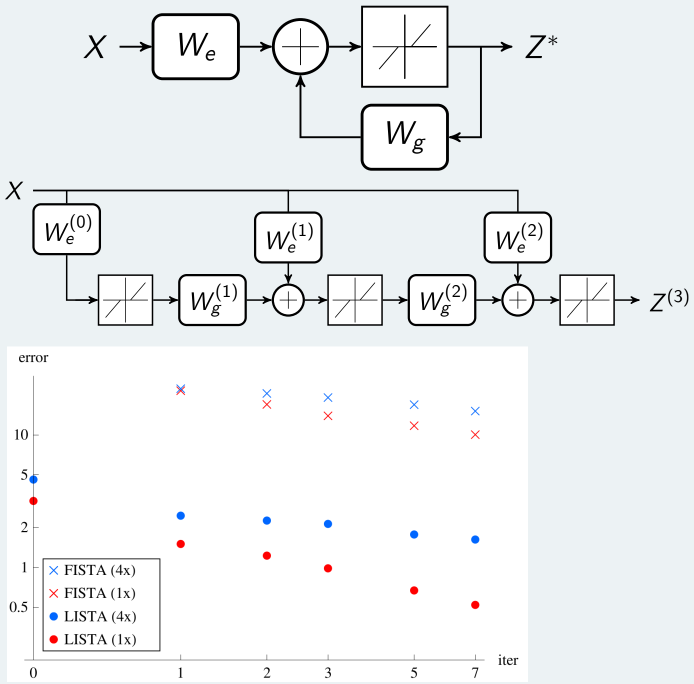
Adaptive Optimization

We have to solve N problems with a common structure $\boldsymbol{D}$.

$$Z^{[n],*} = \underset{Z^{[n]}}{\operatorname{argmin}} \|X^{[n]} - \sum_{k=1}^{K} \boldsymbol{D}_k * Z_k^{[n]}\|_2^2 + \lambda \|Z^{[n]}\|_1$$

Can we use this structure to accelerate the resolution?

Yes, with the Learned ISTA Gregor and Lecun [2010]



Adapted from Gregor and Lecun [2010]

Adaptive Optimization

We have to solve N problems with a common structure $\boldsymbol{D}$.

$$Z^{[n],*} = \underset{Z^{[n]}}{\operatorname{argmin}} \|X^{[n]} - \sum_{k=1}^{K} \boldsymbol{D}_k * Z_k^{[n]}\|_2^2 + \lambda \|Z^{[n]}\|_1$$

Can we use this structure to accelerate the resolution?

Yes, with the Learned ISTA Gregor and Lecun [2010]

Open problem: Why does it work?

- Can we leverage the structure of $\boldsymbol{D}$?
- Can we get a non-asymptotic acceleration of ISTA?
- How to explain LISTA performance?

[Giryes et al., 2016; Xin et al., 2016]

Notations

Consider the sparse coding problem with a dictionary D.

$$z^* = \operatorname*{argmin}_z F(z) = \underbrace{\|x - Dz\|_2^2}_{E(z)} + \lambda\|z\|_1$$

We denote $B = D^\top D$ is the Gram matrix of D.

Quadratic form: $Q_S(u, v) = \frac{1}{2}(u - v)^\top S(u - v) + \lambda\|u\|_1$.

Note that $F(z) = Q_B(z, D^\dagger x)$

For S is diagonal, $\operatorname{argmin}_u Q_S(u, v)$ can be efficiently minimized as the problem is separable on each coordinate:

$$\operatorname*{argmin}_{u_i} \frac{s_i}{2}(u_i - v_i)^2 + \lambda\|u_i\|$$

$\Rightarrow$ Scaled soft thresholding

$$u_i^* = \frac{\operatorname{sign}(v_i)}{s_i}\max(0, |v_i| - \lambda)$$

Notations

Consider the sparse coding problem with a dictionary D.

$$z^* = \operatorname*{argmin}_z F(z) = \underbrace{\|x - Dz\|_2^2}_{E(z)} + \lambda\|z\|_1$$

We denote $B = D^\top D$ is the Gram matrix of D.

We introduce a novel class of algorithms – FacNet – based on a sparse factorization of B.

Quadratic form: $Q_S(u, v) = \frac{1}{2}(u - v)^\top S(u - v) + \lambda\|u\|_1$.

Note that $F(z) = Q_B(z, D^\dagger x)$

For S is diagonal, $\operatorname{argmin}_u Q_S(u, v)$ can be efficiently minimized as the problem is separable on each coordinate:

$$\operatorname*{argmin}_{u_i} \frac{s_i}{2}(u_i - v_i)^2 + \lambda\|u_i\|$$

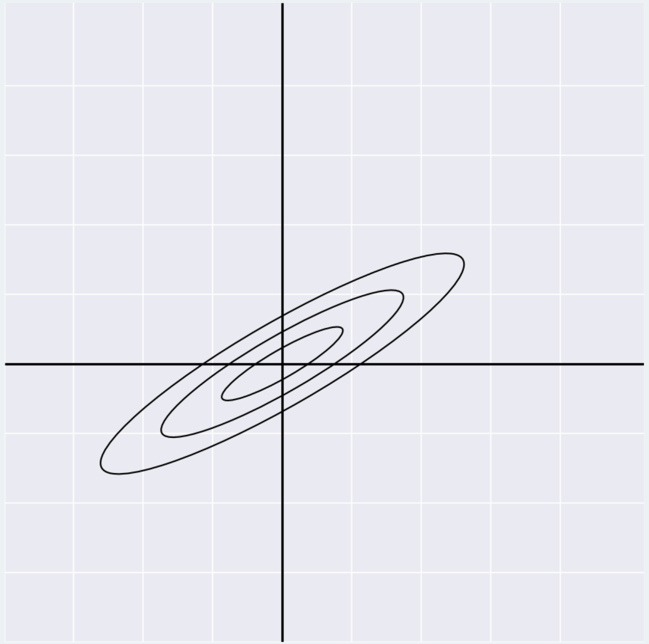
$\Rightarrow$ Scaled soft thresholding

$$u_i^* = \frac{\operatorname{sign}(v_i)}{s_i}\max(0, |v_i| - \lambda)$$

Toward an adaptive procedure

Given an estimate $z^{(q)}$ of z^* at iteration q, we can write:

$$\begin{aligned} F(z) &= E(z) + \lambda\|z\|_1 \\ &= E(z^{(q)}) + \left\langle \nabla E(z^{(q)}), z - z^{(q)} \right\rangle + Q_B(\ z, \ \ z^{(q)}) \ , \end{aligned}$$



Toward an adaptive procedure

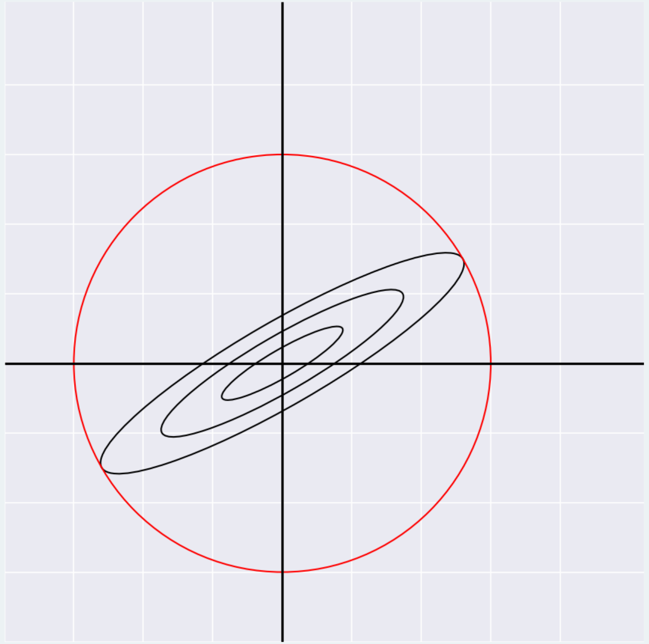
Given an estimate $z^{(q)}$ of z^* at iteration q, we can write:

$$\begin{aligned} F(z) &= E(z) + \lambda \|z\|_1 \\ &= E(z^{(q)}) + \left\langle \nabla E(z^{(q)}), z - z^{(q)} \right\rangle + Q_B(\quad z, \quad z^{(q)}) \ , \end{aligned}$$

<u>ISTA:</u> Replace B by diagonal matrix $S = \|B\|_2 \boldsymbol{I}_K$

$$F_q(z) = E(z^{(q)}) + \left\langle \nabla E(z^{(q)}), z - z^{(q)} \right\rangle + Q_{\color{blue}S} \left(\quad z, \quad z^{(q)} \right) \ ,$$

$$\min_z F_q(z) \Leftrightarrow \min_z Q_S \left(\quad z, \quad z^{(q)} - \quad S^{-1} \ \nabla E(z^{(q)}) \right)$$



Studying physiological signals

Convolutional Dictionary Learning

Adaptive Sparse Coding

Motivations

Adaptive ISTA: FacNet

Understanding LISTA

Conclusion

Toward an adaptive procedure

Given an estimate $z^{(q)}$ of z^* at iteration q, we can write:

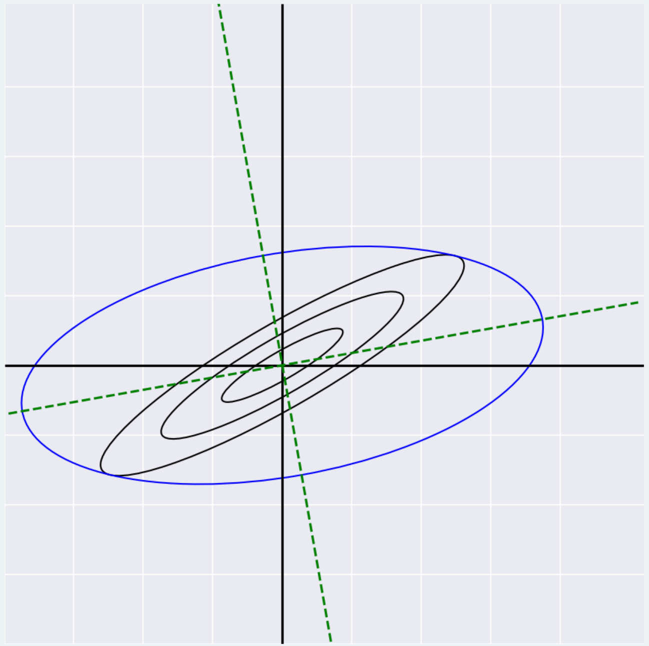
$$\begin{aligned} F(z) &= E(z) + \lambda\|z\|_1 \\ &= E(z^{(q)}) + \left\langle \nabla E(z^{(q)}), z - z^{(q)} \right\rangle + Q_B(\ z,\ \ z^{(q)})\ , \end{aligned}$$

ISTA: Replace B by diagonal matrix $S = \|B\|_2 \boldsymbol{I}_K$

FacNet: Replace B by $A^\top SA$ (S diagonal, A unitary)

$$\widetilde{F}_q(z) = E(z^{(q)}) + \left\langle \nabla E(z^{(q)}), z - z^{(q)} \right\rangle + Q_{\boldsymbol{S}_q}(\boldsymbol{A}_q z, \boldsymbol{A}_q z^{(q)})\ ,$$

$$\min_z \widetilde{F}_q(z) \Leftrightarrow \min_z Q_{S_q}\left(A_q z, A_q z^{(q)} - S_q^{-1} A_q \nabla E(z^{(q)})\right)$$



Studying physiological signals

Convolutional Dictionary Learning

Adaptive Sparse Coding

Motivations

Adaptive ISTA: FacNet

Understanding LISTA

Conclusion

Toward an adaptive procedure

Given an estimate $z^{(q)}$ of z^* at iteration q, we can write:

$$\begin{aligned} F(z) &= E(z) + \lambda\|z\|_1 \\ &= E(z^{(q)}) + \left\langle \nabla E(z^{(q)}), z - z^{(q)} \right\rangle + Q_B(\quad z, \quad z^{(q)}) \ , \end{aligned}$$

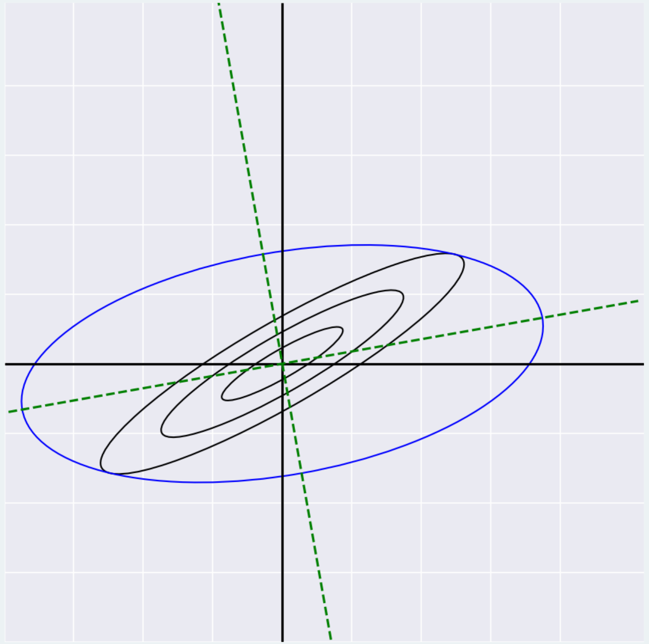
<u>ISTA:</u> Replace B by diagonal matrix $S = \|B\|_2 \boldsymbol{I}_K$

<u>FacNet:</u> Replace B by $A^\top SA$ (S diagonal, A unitary)

$$\widetilde{F}_q(z) = E(z^{(q)}) + \left\langle \nabla E(z^{(q)}), z - z^{(q)} \right\rangle + Q_{S_q}(\boldsymbol{A}_q z, \boldsymbol{A}_q z^{(q)}) \ ,$$

$$\min_z \widetilde{F}_q(z) \Leftrightarrow \min_z Q_{S_q}\left(A_q z, A_q z^{(q)} - S_q^{-1} A_q \nabla E(z^{(q)})\right)$$

Can we choose A_q, S_q to accelerate the optimization compared to ISTA?



Similar iterative procedure with steps adapted to the problem topology.

$$\widetilde{F}_q(z) = F(z) + (z - z^{(q)})^\mathsf{T} R(z - z^{(q)}) + \delta_A(z)$$

Tradeoff between:

- Rotation to align the norm $\|\cdot\|_B$ and the norm $\|\cdot\|_1$,
  Computation
  $$R = A^\mathsf{T} S A - B$$
- Deformation of the ℓ_1-norm with the rotation A .
  Accuracy
  $$\delta_A(z) = \lambda \left(\|Az\|_1 - \|z\|_1 \right)$$

One step improvement

Suppose that $R = A^\mathsf{T} SA - B \succ 0$ is positive definite, and define

$$z^{(q+1)} = \arg\min_z \widetilde{F}_q(z) \ ,$$

Then

$$F(z^{(q+1)}) - F(z^*) \leq \frac{1}{2}(z^{(q)} - z^*)^\mathsf{T} R(z^{(q)} - z^*) \\ + \delta_A(z^*) - \delta_A(z^{(q+1)}) \ .$$

We are interested in factorization (A, S) for which $\|R\|_2$ and δ_A are small.

Theoretical results

- We showed that FacNet has the same asymptotic convergence rate as ISTA in $\mathcal{O}(\frac{1}{q})$.
- The constant factors are different and can be improved. If the factorization (A_q, S_q) at iteration q verifies

$$\|R_q\|_2 + 2\frac{L_{A_q}(z^{(q+1)})}{\|z^* - z^{(q)}\|_2} \le \frac{\|B\|_2}{2}$$

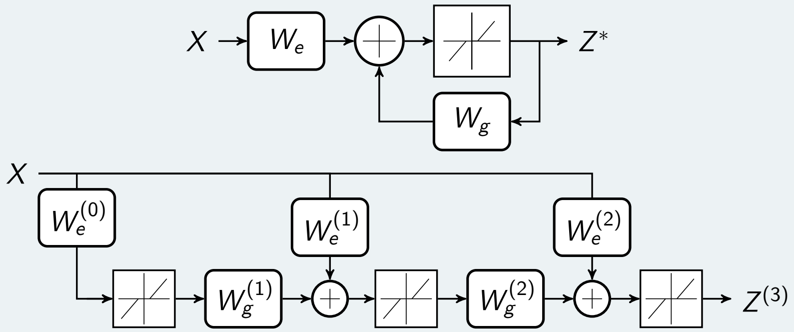
and $A_p = \boldsymbol{I}_K, S_p = \|B\|_2\boldsymbol{I}_K$ for $p > q$, then the procedure has improved convergence rate compared to ISTA.

$\Rightarrow$ There is a phase transition when $\|z^{(q)} - z^*\|_2 \to 0$

- We consider the **generic dictionaries**, uniformly sampled from $\mathcal{S}^{p-1}$.
- We derive **sufficient conditions** on the problem setting for the existence of a factorization (A_q, S_q) of B which improves **the performance of one step** of FacNet compared to one step of ISTA, **in expectation over the generic dictionaries**.

$$\lambda\mathbb{E}_z\left[\|z^{(q+1)}\|_1 + \|z^*\|_1\right] \le \sqrt{\frac{K(K-1)}{p}}\mathbb{E}_z\left[\|z^{(q)} - z^*\|_2^2\right]$$

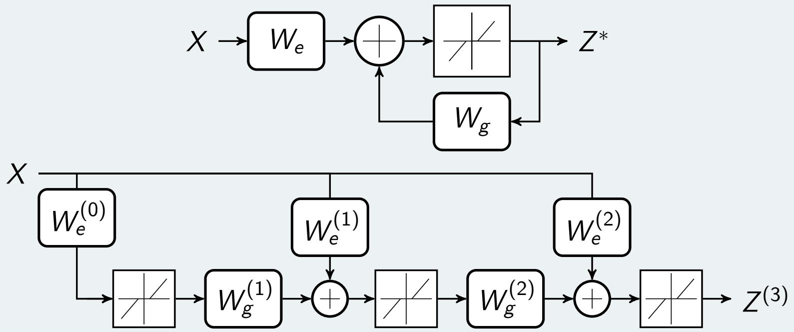
Learned ISTA [Gregor and Lecun, 2010]



Network architecture for ISTA/LISTA. LISTA is the unfolded version of the RNN of ISTA, trainable with back-propagation.

With $W_e = \frac{D^\top}{\|B\|_2}$ and $W_g = I - \frac{B}{\|B\|_2}$, this network computes exactly 2 iterations of ISTA.

Learned ISTA [Gregor and Lecun, 2010]



Network architecture for ISTA/LISTA. LISTA is the unfolded version of the RNN of ISTA, trainable with back-propagation.

With $W_e = \frac{D^\top}{\|B\|_2}$ and $W_g = I - \frac{B}{\|B\|_2}$, this network computes exactly 2 iterations of ISTA.

FacNet

Specialization of LISTA

$$z^{(q+1)} = A^\top \operatorname{prox}_S(Az^{(q)} - S^{-1}AB(z^{(q)} - y)) ,$$

with A unitary and S diagonal.
Same architecture with more constraints on the parameter space:

$$\begin{cases} W_e & = S^{-1}AD^\top \\ W_g & = A^\top - S^{-1}ABA^\top \end{cases}$$

$\Rightarrow$ LISTA can be at least as good as this model.

Generic Dictionary

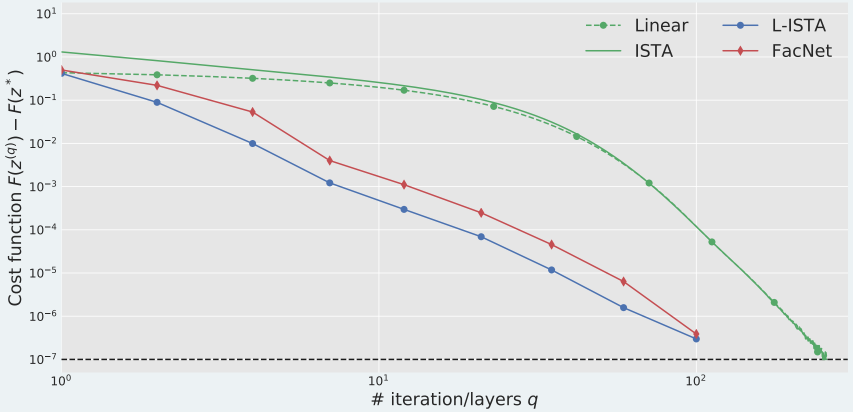
- Generic dictionary uniformly sample in unit ball,

$$D \sim \mathcal{S}^{p-1} \ ,$$

- Sparse code generated with Bernouilli-Gaussian model, *s.t.*

$$z_k = b_k a_k, \qquad b_k \sim \mathcal{B}(\rho) \text{ and } a \sim \mathcal{N}(0, \sigma \boldsymbol{I}_K)$$

Fixed: $K = 100,\ P = 64,\ \sigma = 10$ and $\lambda = 0.01$



$$\rho = {}^1/_{20}.$$

Evolution of the cost function $F(z^{(q)}) - F(z^*)$ with the number of layers/iterations q with a denser model

Adversarial dictionary

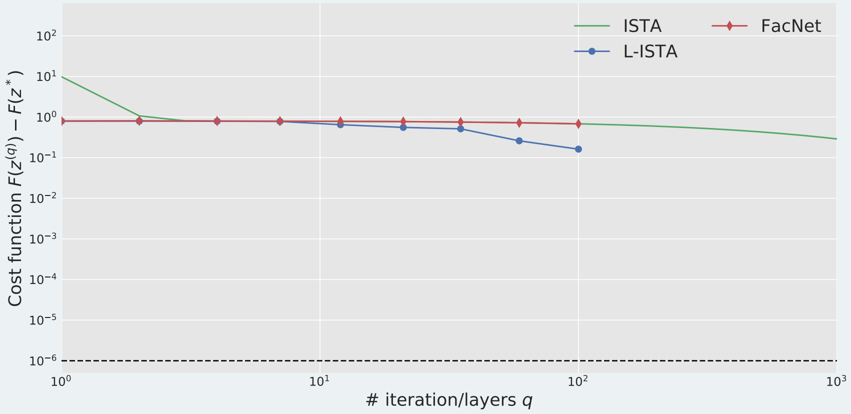
The dictionary is constructed such that it eigen-vectors are sampled from the Fourier basis, with

$$D_{k,j} = e^{-2i\pi k \zeta_j}$$

for a random subset of frequencies

$$\{\zeta_i\}_{0 \le i \le p} \sim \mathcal{U}\left\{\frac{m}{K}; 0 \le m \le \frac{K}{2}\right\}$$

Diagonalizing B implies large deformation of the ℓ_1-norm.



Evolution of the cost function $F(z^{(q)}) - F(z^*)$ with the number of layers/iterations q with an adversarial dictionary.

Contributions

- Theoretical analysis: Non asymptotic acceleration of ISTA is possible based on the structure of $\boldsymbol{D}$,
- FacNet Algorithm: Sufficient analysis to explain LISTA acceleration,
- Adversarial Example: Empirically showed the structure of D is necessary for LISTA.

Future work

- Direct factorization: Improve the factorization formulation for direct optimization,
- Performance quantification: Second order analysis for generic dictionary,
- Sparse PCA: Link the sparse eigenvectors properties to our factorization.

A diverse work

- Technical contributions:
  → Theoretical study of LISTA; DICOD
- Exploratory contributions:
  → Link SSA to CSC; Post-training
- Collaboration with Medical doctors for clinical research publications,
- Contribution to open source software with the python library `loky`.

A collaborative work

- Co-authors: L. Oudre, N. Vayatis, J. Audiffren, R. Barrois, J. Bruna.
- Medical doctors: S. Buffat, D. Ricard, M. Robert, P-P. Vidal, C. de Waele, A. Yelnik.
- Open-source: O. Grisel.

Publications and preprints

Moreau, T., Oudre, L., and Vayatis, N. (2015b). Groupement automatique pour l ' analyse du spectre singulier. In *Groupe de Recherche et d'Etudes en Traitement du Signal et des Images (GRETSI)*

Oudre, L., Moreau, T., Truong, C., Barrois-Müller, R., Dadashi, R., and Grégory, T. (2015). Détection de pas à partir de données d'accélérométrie. In *Groupe de Recherche et d'Etudes en Traitement du Signal et des Images (GRETSI)*

Moreau, T., Oudre, L., and Vayatis, N. (2015a). Distributed Convolutional Sparse Coding via Message Passing Interface (MPI). In *NIPS Workshop Nonparametric Methods for Large Scale Representation Learning*

Moreau, T. and Audiffren, J. (2016). Post Training in Deep Learning with Last Kernel. *arXiv preprint*, arXiv:1611(04499)

Moreau, T. and Bruna, J. (2017). Understanding Neural Sparse Coding with Matrix Factorization. In *International Conference on Learning Representation (ICLR)*

Moreau, T., Oudre, L., and Vayatis, N. (2017). Distributed Convolutional Sparse Coding. *arXiv preprint*, arXiv:1705(10087)

Barrois, R., Oudre, L., Moreau, T., Truong, C., Vayatis, N., Buffat, S., Yelnik, A., de Waele, C., Gregory, T., Laporte, S., and Others (2015). Quantify osteoarthritis gait at the doctor's office: a simple pelvis accelerometer based method independent from footwear and aging. *Computer methods in biomechanics and biomedical engineering*, 18(Sup1):1880–1881

Barrois, R., Gregory, T., Oudre, L., Moreau, T., Truong, C., Pulini, A. A., Vienne, A., Labourdette, C., Vayatis, N., Buffat, S., Yelnik, A., De Waele, C., Laporte, S., Vidal, P. P., and Ricard, D. (2016). An automated recording method in clinical consultation to rate the limp in lower limb osteoarthritis. *PLoS ONE*, 11(10):e0164975

Robert, M., Contal, E., Moreau, T., Vayatis, N., and Vidal, P.-P. (2015). The Why and How of Recording Eye Movement from Very Early Childhood. Oral Presentation, Gordon Research Conference on Eye Movement

Thanks!

Experiment

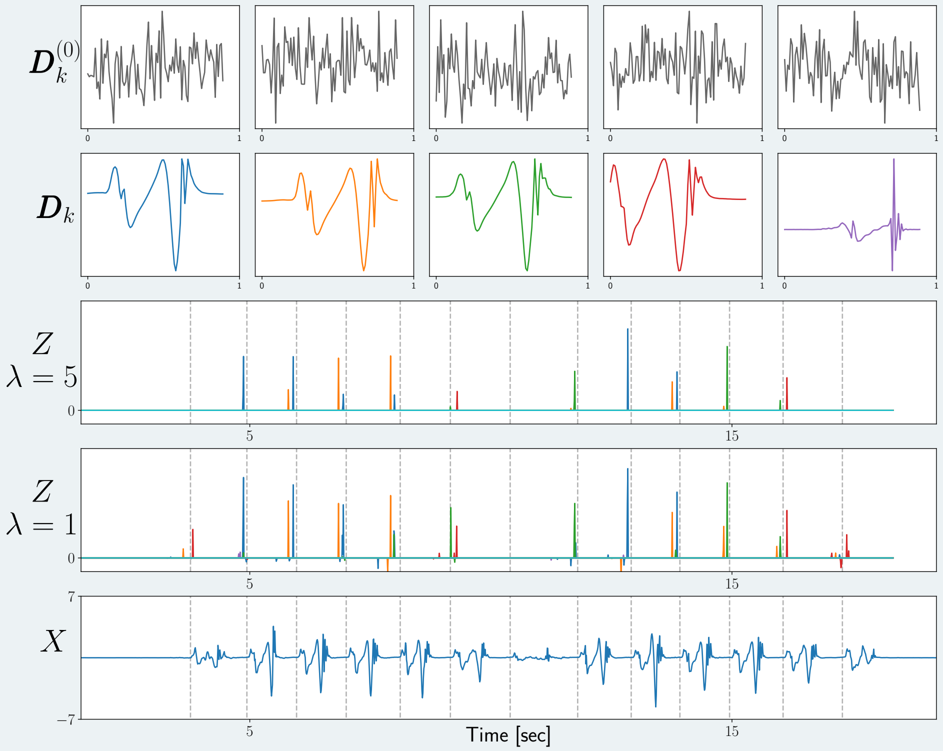
Create a dictionary with 25 Gaussian patterns ($W = 90$)

$$\boldsymbol{D}_k^{(0)} \sim \mathcal{N}(0, I_{90})$$

Use the Convolutional Dictionary Learning with DICOD to learn a dictionary $\boldsymbol{D}$ on a set of 50 recording of healthy subjects walking.

Challenges

- Alignment of the patterns,
- Detect steps of different amplitude,
- Handle multivariate signals.



Auxiliary Slides

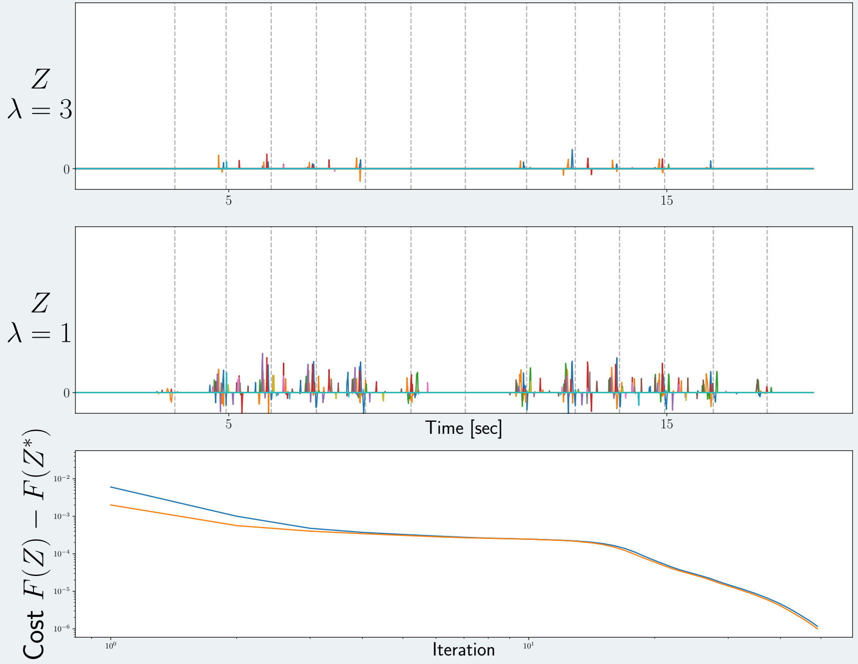
Physiological Signals

FacNet

DICOD

Singular Spectrum Analysis (SSA)

Post-training for Deep Learning



- Giryes et al. [2016]: Propose the inexact projected gradient descent and conjecture that LISTA accelerate the LASSO resolution by learning the sparsity pattern of the input distribution.

- Xin et al. [2016]: Study the Hard-thresholding Algorithm and its capacity to recover the support of a sparse vector.
  The paper relax the RIP conditions for the dictionary.

Generic Dictionaries

A dictionary $D \in \mathbb{R}^{p\times K}$ is a generic dictionary when its columns D_i are drawn uniformly over the ℓ_2 unit sphere $\mathcal{S}^{p-1}$.

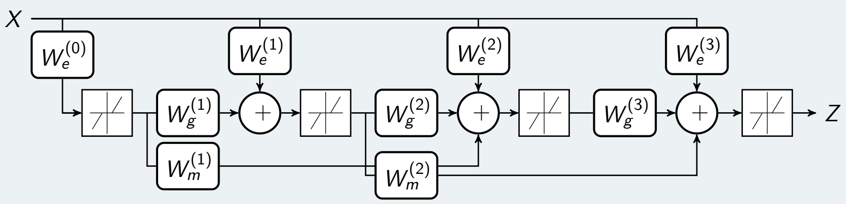
Theorem (Generic Acceleration)

In **expectation over the generic dictionary** D, the factorization algorithm using a diagonally dominant matrix $A \subset \mathcal{E}_\delta$, has better performance for iteration $q+1$ than the normal ISTA iteration – which uses the identity – when

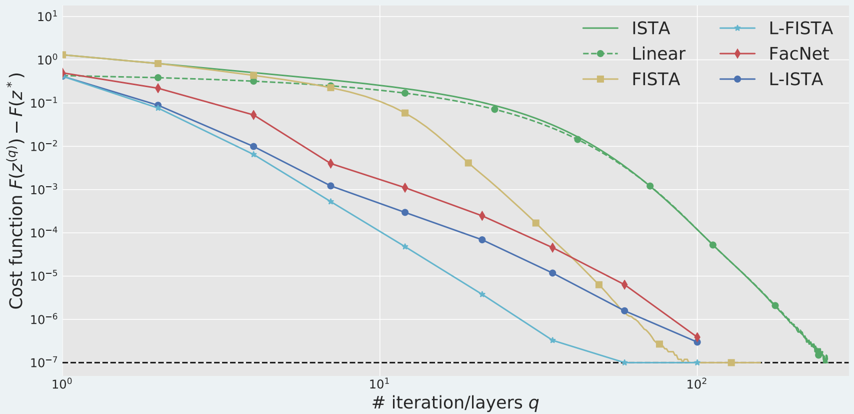
$$\lambda \mathbb{E}_z\left[\|z^{(q+1)}\|_1 + \|z^*\|_1\right] \le \sqrt{\frac{K(K-1)}{p}}\underbrace{\mathbb{E}_z\left[\|z^{(q)} - z^*\|_2^2\right]}_{\text{expected resolution at iteration } q}$$

FacNet can improve the performances compared to ISTA when this is verified.

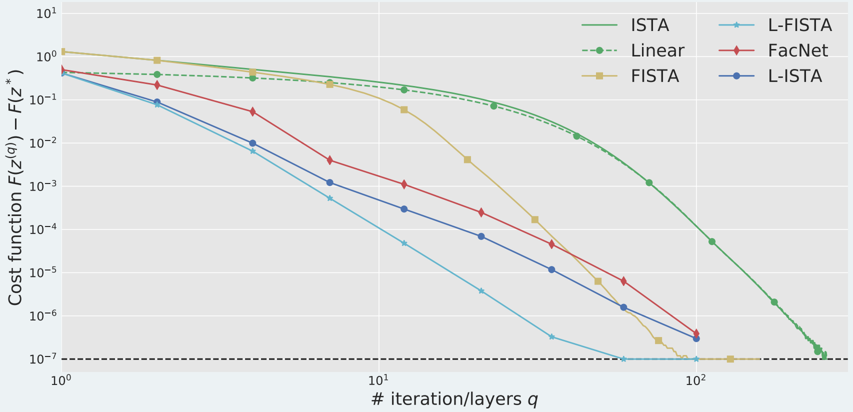
L-FISTA



Network architecture for L-FISTA.

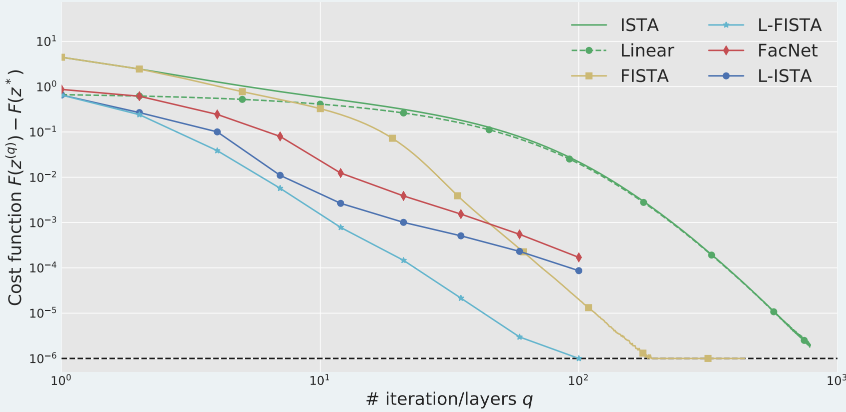


Evolution of the cost function $F(z^{(q)}) - F(z^*)$ with the number of layers/iterations q with a denser model



$\rho = {}^1/_{20}.$

Evolution of the cost function $F(z^{(q)}) - F(z^*)$ with the number of layers/iterations q with a denser model



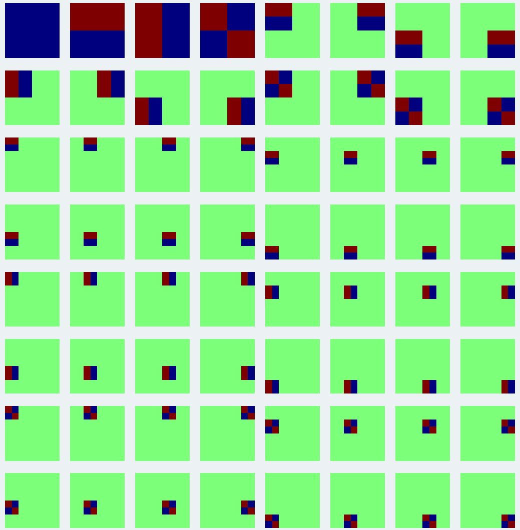
$\rho = {}^1/_{4}.$

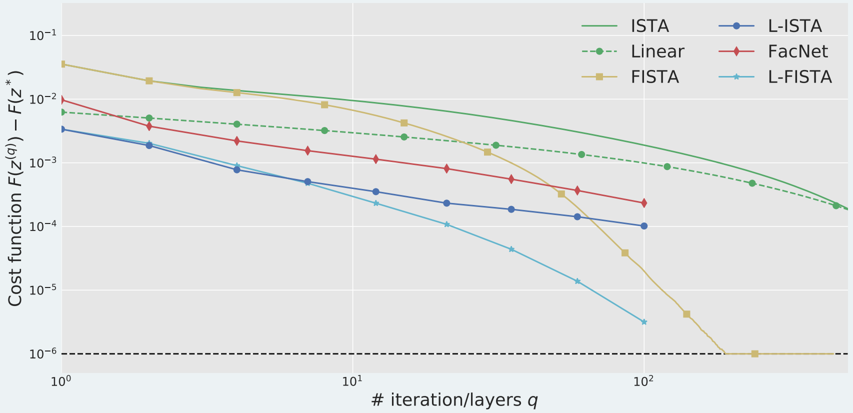
Evolution of the cost function $F(z^{(q)}) - F(z^*)$ with the number of layers/iterations q with a denser model

PASCAL 08

Sparse coding for the PASCAL 08 datasets over the Haar wavelets family.

Patch size: 8x8; $K = 267$; *train/test:* 500/100

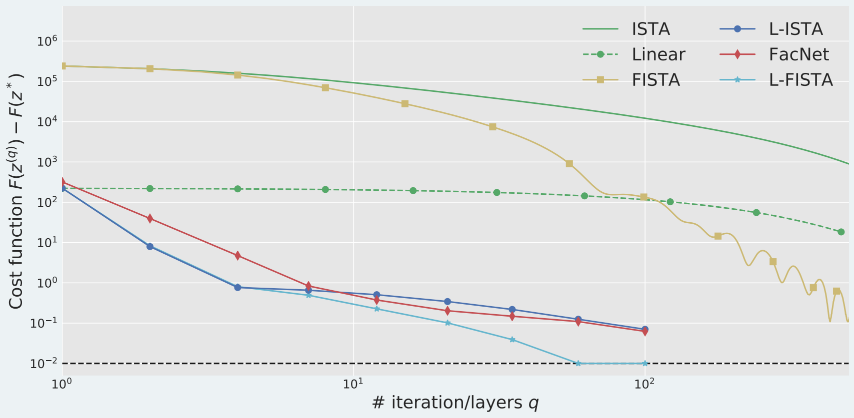




Evolution of the cost function $F(z^{(q)}) - F(z^*)$ with the number of layers or the number of iteration q for Pascal VOC 2008.

MNIST

Dictionary D with $K = 100$ atoms learned on 10 000 MNIST samples (17x17) with dictionary learning. LISTA trained with MNIST training set and tested on MNIST test set.

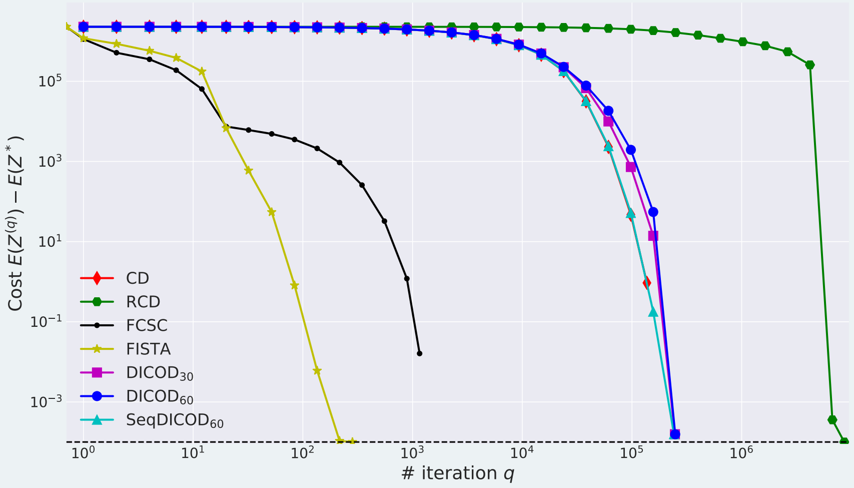


Evolution of the cost function $F(z^{(q)}) - F(z^*)$ with the number of layers or the number of iteration q for MNIST.

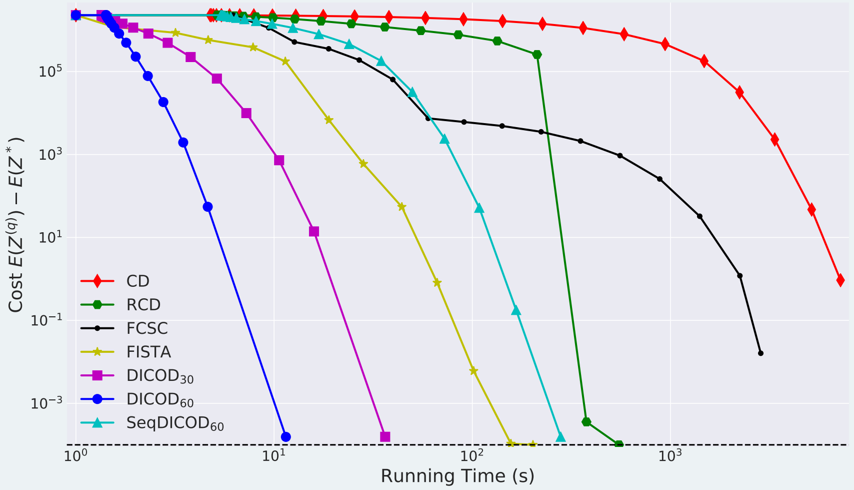
Finishing the process in a linear grid?

Non trivial point: **How to decide that the algorithm has converged?**

- Neighbors paused is not enough!
- Define a master 0 and send probes.
  Wait for M probes return.
- Uses the notion of message queue and network flow.
  Maybe we can have better way?



Cost as a function of the iterations



Cost as a function of the time

Sinuglar Spectrum Analysis

- Choose a window size K and extract sub series, $\longrightarrow$ K-trajectory matrix $\boldsymbol{X}^{(K)}$
- Reconstruct a low rank estimate of all the K-length sub series, $\longrightarrow$ Singular Value decomposition $X^{(K)} = \sum_{k=1}^{K} \lambda_k U_k V_k^T$
- Decomposition of the series as a sum of "low rank" components. $\longrightarrow$ Average along anti-diagonals

$\Longrightarrow$ Extract components linked to trend and oscillations

Linked to the convolutional least square

$$Z^*, \boldsymbol{D}^* = \arg\min_{Z, \boldsymbol{D}} \frac{1}{2} \left\| X - \sum_{k=1}^{K} z_k * D_k \right\|_2^2 , \qquad (1)$$

with constraints $\langle D_i, D_j \rangle = \delta_{i,j}$

- $\boldsymbol{D}$ is the dictionary with $K = W$ patterns in $\mathbb{R}$ of length W
- Z is an activation signal, or coding signal in $\mathbb{R}^K$ of length $L = T - W + 1$

Issues

- Same pattern present in different components,
- Representation is "dense", no localization,
- Different representation for each signal,
- A grouping step necessary to clean the extracted patterns.

Post-training

Paper with J. Audiffren: arxiv:1611.04499

Use the idea to split the representation learning and the task resolution:

- *Post-training* step: only train the last layer,
- Easy problem: this problem is often convex,
- Link with kernel: close form solution for optimal last layer,
- Experiments: consistent performance boost with multiple architecture.

Remarks

- No gain if we are in a local minima,
- Should be use with early stopping.