Université Aix-Marseille 2018-2019
Master Mathématiques appliquées et statistiques 2ème année

**Représentation parcimonieuse des signaux**

Notes de cours et Td : Parcimonie et algorithmes de décomposition parcimonieuse.
Algorithme de Matching-Pursuit

# Notations

- On note $\langle .,. \rangle$ le produit scalaire usuel sur $\mathbb{C}^N$ tel que $\langle x, y\rangle = \sum_{n=0}^{N-1} x_n \overline{y_n}$ pour $x = (x_i)_{i=0,\dots N-1} \in \mathbb{C}^N$ et $y = (y_i)_{i=0,\dots N-1} \in \mathbb{C}^N$. On note $|x|_2^2 = \sum_{n=0}^{N-1} |x_n|^2$ pour $x \in \mathbb{C}^N$.
- On note $\delta^\ell$ pour $\ell \in \{0, \dots N-1\}$ le $\ell$-ième vecteur de la base canonique de $\mathbb{C}^N$. Il s'agit du vecteur de $\mathbb{C}^N$ tel que pour $n \in \{0, \dots N-1\}$ on a $\delta_n^\ell = 1$ si $n = \ell$ et 0 sinon. La collection de vecteurs $\{\delta^\ell, \ell \in \{0, \dots, N-1\}\}$ constitue la base canonique de $\mathbb{C}^N$.
- la famille de vecteurs $\widetilde{\mathcal{E}} = \{\frac{1}{\sqrt{N}} e^\ell \in \mathbb{C}^N, \ell \in \mathbb{Z}\}$ tels que pour $\ell \in \mathbb{Z}$ et pour $n \in \{0, \dots N-1\}$ la $n$-ième coordonnée du vecteur $e^\ell$ s'écrive
$$e_n^\ell = e^{\frac{2i\pi\ell n}{N}}$$
est appelée base de Fourier orthonormalisée discrète.
- Si $x$ est un signal dans $\mathbb{R}^N = \mathbb{R}^{2^n}$ et $\Phi$ une base d'ondelettes de $\mathbb{R}^N$ à $n_0$ niveaux de décomposition on note $D_k(x)$ les coefficients dits d'« approximation », ou coefficients d'« échelle » pour $k = 0, \dots 2^{n-n_0} - 1$ et $C_{j,k}(x)$ pour $j = n - n_0, \dots n-1$ et $k = 0, \dots 2^j - 1$ les coefficients en ondelettes de $x$ ou coefficients de « détail ».

# 1 Introduction

Il n'est pas toujours facile ou possible de trouver une base dans laquelle un signal $x \in \mathbb{C}^N$ a une décomposition parcimonieuse. Considérons en effet l'exemple suivant :

$$x = a_0 \delta^{k_0} + a_1 \frac{e^{k_1}}{\sqrt{N}} \tag{1}$$

Ce signal n'est parcimonieux ni dans la base canonique, ni dans la base de Fourier discrète orthonormalisée.

Cependant si on considère la réunion des deux bases $\mathcal{D} = \{\delta^k, k = 0, \dots N-1\} \cup \{\frac{e^k}{\sqrt{N}}, k = 0, \dots N-1\}$ alors le signal $x$ admet une décomposition très parcimonieuse sur $\mathcal{D}$, qui est (1).

Par contre il nous faut constater que $x$ admet une infinité de décompositions possibles sur les vecteurs de $\mathcal{D}$. La question qui se pose est de trouver la décomposition de $x$ la plus parcimonieuse parmi toutes les décompositions possibles de $x$ sur les vecteurs de $\mathcal{D}$.

Nous allons voir que l'algorithme dit de Matching-Pursuit nous permet de répondre à la question posée.

# 2 Algorithme de Matching-Pursuit

Nous nous plaçons dans $\mathbb{C}^N$ et considérons un système générateur noté $\mathcal{D} = \{\phi^0, \dots \phi^{K-1}\}$ comportant $K$ vecteurs de $\mathbb{C}^N$, tels que pour tout $k = 0, \dots K-1$ on a $|\phi^k|_2 = 1$. Un tel système est appelé dictionnaire de $\mathbb{C}^N$.

L'algorithme de Matching-Pursuit fonctionne de la façon suivante :

1. Initialisation : $m := 0$, $R^0(x) = x$.
2. Par récurrence sur $m \in \mathbb{N}$
   (a) Choix du meilleur vecteur $k_m = \arg\max_k |\langle R^m(x), \phi^k \rangle|$
   (b) Mise à jour du « résidu » $R^{m+1}(x) = R^m(x) - \langle R^m(x), \phi^{k_m} \rangle \phi^{k_m}$

Nous avons les propriétés suivantes

**Proposition 1**

*Soit $x \in \mathbb{C}^N$ et $\mathcal{D}$ un dictionnaire de $\mathbb{C}^N$. Avec les notations qui précèdent pour tout $m \in \mathbb{N}$ et $M \in \mathbb{N}$*

1. *$x = \sum_{m=0}^{M-1} \langle R^m(x), \phi^{k_m} \rangle \phi^{k_m} + R^M(x)$.*
2. *$R^{m+1}(x)$ et $\phi^{k_m}$ sont orthogonaux, et on a*

$$|x|_2^2 = \sum_{m=0}^{M-1} |\langle R^m(x), \phi^{k_m} \rangle|^2 + |R^M(x)|_2^2 \tag{2}$$

3. *la suite $(|R^m(x)|_2)_{m\in\mathbb{N}}$ est décroissante.*

**Remarque 1**

*Lorsque $\mathcal{D}$ est une base orthonormale de $\mathbb{C}^N$ l'algorithme retrouve la décomposition de $x$ sur cette base en $N$ itérations.*

Nous admettrons le résultat suivant (voir [1] chapitre 12, section 12.3 Théorème 12.6, pour ceux qui veulent étudier la preuve)

**Théorème 1**

*Avec les notations précédentes et $x \in \mathbb{C}^N$ il existe $0 < \beta(\mathcal{D}) \leq 1$ avec $\beta(\mathcal{D}) = \inf_{x\in\mathbb{C}^N, |x|_2=1} \sup_{j\in\{0,\ldots K-1\}} |\langle x, \phi^j \rangle|$ tel que le résidu à l'étape $m$ vérifie*

$$|R^m(x)|_2 \leq (1 - \beta(\mathcal{D})^2)^m |x|_2^2.$$

*En particulier $|R^m(x)|_2 \to 0$ quand $m \to +\infty$, l'algorithme de Matching Pursuit converge et on a $x = \sum_{m=0}^{\infty} \langle R^m(x), \phi^{k_m} \rangle \phi^{k_m}$.*

**Exemple :**
Supposons que $x$ est une combinaison linéaire avec peu de coefficients non nuls de vecteurs de la base canonique et de vecteurs de la base $\tilde{\mathcal{E}}$. Testons sur un exemple numérique et voyons quelle décomposition nous produit l'algorithme.

Nous programmons une version de l'algorithme 1 (abrégé MP) dont la formulation est adaptée à ce que nous voulons faire : calculer une décomposition de $x$ sur les vecteurs de $\mathcal{D}$. Le vecteur sortie $\alpha$ correspond en effet aux coordonnées de $x$ sur les vecteurs de $\mathcal{D}$ calculées par l'algorithme, autrement dit l'algorithme calcule une décomposition $(\alpha_0, \alpha_1, \ldots, \alpha_{K-1})$ telle que $x = \sum_{j=0}^{K-1} \alpha_j \phi^j + R(x)$ où $R(x)$ est un reste que l'on voudrait le plus petit possible et en particulier on souhaiterait $|R(x)|_2 < \varepsilon$ où $\varepsilon > 0$ est une précision fixée à l'avance.

On considère ici le signal $x = (x_0, \ldots, x_{N-1})$ tel que pour tout $n = 0, \ldots N-1$ avec $N = 1024$ $x_n = 4\delta_n^{256} + 0.25\delta_n^{700} + \frac{16}{\sqrt{N}} \cos\left(2\pi * 25 * \frac{n}{N}\right)$. L'algorithme converge vite car au bout de 14 itérations on a $|R^m(x)|_2 \leq 10^{-6}$. De plus on constate que l'algorithme de Matching-Pursuit retrouve les composantes de la décomposition parcimonieuse de $x$. Nous nous proposons maintenant de démontrer que ce sera toujours le cas si certaines conditions sont réunies.

**Algorithme 1 :** Matching Pursuit.

**Data :** dictionnaire $D = [\phi^0, \ldots, \phi^{K-1}] \in \mathbb{C}^{N,K}$, $x \in \mathbb{R}^N$, $\varepsilon > 0$, $M_{max} \in \mathbb{N}$
**initialisation** :
$r \leftarrow x$ [Initialisation du résidu];
$\alpha \leftarrow 0_K$ [Initialisation de $\alpha$];
$m \leftarrow 0_K$ [Initialisation des itérations];
**while** $m \leq M_{max}$ *et* $|r|_2 > \varepsilon$ **do**
  $m \leftarrow m + 1$;
  $\forall m, c_m \leftarrow \langle r, \phi^m \rangle$;
  $k_m \leftarrow \arg \max\limits_{m \leq K-1} |\langle r, \phi^m \rangle|$;
  $\alpha(k_m) \leftarrow \alpha(k_m) + c_{k_m}$;
  $r \leftarrow r - c_{k_m} \phi^{k_m}$
**Result :** $\alpha \in \mathbb{C}^K$, $M$ nombre d'itérations.

# 3 Conditions de recouvrement de décomposition parcimonieuse

Soit $\mathcal{D} = \{\phi^0, \ldots, \phi^{K-1}\}$ un dictionnaire de $\mathbb{C}^N$.

On note $\mu(\mathcal{D}) = \max\limits_{i \neq j} |\langle \phi^i, \phi^j \rangle|$ la cohérence du dictionnaire. Plus $\mu(\mathcal{D})$ est petite plus le dictionnaire est dit « incohérent » et plus les vecteurs $\phi^j$ sont étrangers les uns aux autres. En particulier si $\mu(\mathcal{D}) = 0$ cela signifie que le dictionnaire forme une base orthonormale.

Dans le cas où $\mathcal{D}$ est la réunion des vecteurs de la base canonique et de ceux de la base de Fourier $\mu(\mathcal{D}) = \frac{1}{\sqrt{N}}$.

Nous allons démontrer le résultat suivant.

**Théorème 2**

*Soit $\mathcal{D} = \{\phi^0, \ldots, \phi^{K-1}\}$ un dictionnaire de $\mathbb{C}^N$ et $x \in \mathbb{C}^N$ tel que*

$$x = \sum_{\ell \in \mathcal{L}} \alpha_\ell \phi^\ell \tag{3}$$

*où $\mathcal{L}$ est un sous-ensemble de $\{0, \ldots, K-1\}$ tel que $\{\phi^\ell, \ell \in \mathcal{L}\}$ forme un système libre et $\alpha_\ell \neq 0$ pour tout $\ell \in \mathcal{L}$.*

*Si $card(\mathcal{L})\mu(\mathcal{D}) < \frac{1}{2}$ alors l'algorithme de Matching-Pursuit sélectionne à chaque itération un vecteur $\phi^\ell$ avec $\ell \in \mathcal{L}$ et converge vers la décomposition (3).*

Nous allons démontrer ce théorème.
**Indications pour la démonstration du théorème 2 :** On considère $x \in \mathbb{C}^N$ qui vérifie (3) avec $\mathcal{L}$ tel que $\{\phi^\ell, \ell \in \mathcal{L}\}$ forme un système libre.

Nous faisons une récurrence sur $m$, les itérations de l'algorithme.

On indique comment raisonner pour $m = 0$.
Considérons $i_0$ le plus petit indice dans $\mathcal{L}$ tel que $|\alpha_{i_0}| = \max\limits_{\ell \in \mathcal{L}} |\alpha_\ell|$.
Soit $j \in \{0, \ldots K-1\}$. Calculons $|\langle R^0(x), \phi^j \rangle|$.

— 1er cas $j \notin \mathcal{L}$ on a

$$\begin{aligned}
|\langle R^0(x), \phi^j\rangle| &= |\langle x, \phi^j\rangle| = \left|\left\langle \sum_{\ell\in\mathcal{L}} \alpha_\ell \phi^\ell, \phi^j \right\rangle\right| \\
&= \left|\sum_{\ell\in\mathcal{L}} \alpha_\ell \left\langle \phi^\ell, \phi^j \right\rangle\right| \\
&\leq \sum_{\ell\in\mathcal{L}} |\alpha_\ell| |\langle \phi^\ell, \phi^j\rangle| \\
&\leq |\alpha_{i_0}| card(\mathcal{L})\mu < \frac{|\alpha_{i_0}|}{2} \qquad (4)
\end{aligned}$$

— 2ème cas : $j = i_0$

$$\begin{aligned}
|\langle R^0(x), \phi^{i_0}\rangle| &= |\langle x, \phi^{i_0}\rangle| = \left|\left\langle \sum_{\ell\in\mathcal{L}} \alpha_\ell \phi^\ell, \phi^{i_0} \right\rangle\right| \\
&= \left|\sum_{\ell\in\mathcal{L}} \alpha_\ell \langle \phi^\ell, \phi^{i_0}\rangle\right| = \left|\alpha_{i_0} + \sum_{\ell\in\mathcal{L}, \ell\neq i_0} \alpha_\ell \langle \phi^\ell, \phi^{i_0}\rangle\right| \\
&\geq |\alpha_{i_0}| - \left|\sum_{\ell\in\mathcal{L}, \ell\neq i_0} \alpha_\ell \langle \phi^\ell, \phi^{i_0}\rangle\right| \qquad (5) \\
&\geq |\alpha_{i_0}| - \sum_{\ell\in\mathcal{L}, \ell\neq i_0} |\alpha_\ell| |\langle \phi^\ell, \phi^{i_0}\rangle| \qquad (6) \\
&> |\alpha_{i_0}| - \frac{|\alpha_{i_0}|}{2} = \frac{|\alpha_{i_0}|}{2} \qquad (7)
\end{aligned}$$

Donc l'algorithme sélectionne à la première itération un vecteur $\phi^{k_0}$ avec $k_0 \in \mathcal{L}$.

# Travaux dirigés

## Démonstrations des résultats du cours.

**Exercice 1**

*Démontrer la proposition 1.*

**Exercice 2**

*Détailler la remarque 1.*

**Exercice 3**

*Écrire entièrement la démonstration du théorème 2.*

## Exercices d'applications.

**Exercice 4**

*Montrer que $\mu(\mathcal{D}) = \frac{1}{\sqrt{N}}$ si $\mathcal{D}$ est la réunion de la base canonique et de la base de Fourier orthonormalisée.*

**Exercice 5**

*Soit $\{\phi^0, \ldots, \phi^{J-1}\}$ un système de $J$ vecteurs de $\mathbb{C}^N$.*

*Montrer que la matrice de Gram $G$ de coefficients $G_{i,j} = \langle \phi^i, \phi^j \rangle$ est positive.*

*Montrer que $G$ est définie positive, donc inversible si et seulement si les $\{\phi^j, j = 0, \ldots J\}$ forment un système libre.*

**Exercice 6**

*On se place dans $\mathbb{C}^N$.*

*Montrer que si on rajoute un vecteur $h$ de norme 1 à une base orthonormale $\mathcal{B}$ alors on obtient un dictionnaire $\mathcal{D} = \mathcal{B} \cup \{h\}$ dont la cohérence satisfait $\mu(\mathcal{D}) \geq \frac{1}{\sqrt{N}}$. En déduire que si $\mathcal{D}$ est un dictionnaire qui contient une base orthonormale on a toujours $\mu(\mathcal{D}) \geq \frac{1}{\sqrt{N}}$.*

## Exercice pour aller plus loin : l'algorithme de Matching-Pursuit orthogonal

L'algorithme dit de Matching Pursuit Orthogonal est comme l'algorithme original de Matching Pursuit un algorithme de décomposition d'un vecteur $x \in \mathbb{C}^N$ sur les vecteurs d'un dictionnaire $\mathcal{D} = \{\varphi^0, .., \varphi^{K-1}\}$ avec $K \geq N$.
Dans le Matching Pursuit Orthogonal on sélectionne les vecteurs du dictionnaire de la même façon que dans le Matching Pursuit classique. Par contre le résidu est calculé à partir de la projection du vecteur à décomposer sur l'espace vectoriel engendré par les vecteurs déjà sélectionnés. Ce qui donne :

**Algorithme 2 :** Matching Pursuit orthogonal.

**Data :** dictionnaire $D = [\phi^0, \ldots, \phi^{K-1}] \in \mathbb{C}^{N,K}$, $x \in \mathbb{C}^N$, $\varepsilon > 0$, $M_{max} \in \mathbb{N}$
**initialisation** :
$r \leftarrow x$ [Initialisation du résidu];
$\alpha \leftarrow 0_K$ [Initialisation de $\alpha$];
$m \leftarrow 0_K$ [Initialisation des itérations];
$\Lambda \leftarrow \emptyset$ [Initialisation de l'ensemble des indices sélectionnés];
$S \leftarrow \emptyset$ [Initialisation de la matrice des vecteurs sélectionnés du dictionnaire.];
**while** $m \leq M_{max}$ *et* $|r|_2 > \varepsilon$ **do**
    $m \leftarrow m + 1$;
    $\forall m, c_m \leftarrow \langle r, \phi^m \rangle$;
    $k_m = \arg \max_{m \leq K-1} |\langle r, \phi^m \rangle|$;
    $\Lambda \leftarrow \Lambda \cup \{k_m\}$;
    $S \leftarrow [S \; \varphi_{k_m}]$ [mise à jour de la matrice des vecteurs du dictionnaire séléctionnés];
    $\alpha^* \leftarrow \arg \min_{\alpha} |x - S\alpha|_2$ [Résolution d'un problème aux moindres carrés];
    $x^* \leftarrow S\alpha^*$;
    $r \leftarrow x - x^*$;
**Result :** $M$ nombre d'itérations, approximation $x^*$, coefficients de décomposition $\alpha^*$, $\Lambda$ les indices des vecteurs sélectionnés pour la décomposition.

**Exercice 7**

1. *Expliquer pourquoi l'étape de projection orthogonale peut être résolue grâce à la résolution d'un problème aux moindres carrés. Comment résout-on ce problème aux moindres carrés ?*
2. *Montrer que l'algorithme de Matching-Pursuit orthogonal converge en au plus $N$ itérations.*
3. *Programmer l'algorithme dans une fonction* `MPorthogonal` *qui prend en entrée le vecteur $x$ à décomposer, le dictionnaire $\mathcal{D}$, le nombre $N$ d'itérations maximal, la précision $\varepsilon$ sur le résidu et donne en sortie une approximation $x^\star$ de $x$, $\Lambda$ les indices des vecteurs séléctionnés du dictionnaire, les coordonnées stockés dans $\alpha$.*

4. *Tester cette fonction sur le signal* $x \in \mathbb{C}^{1024}$ *avec* $x$ *de l'exemple détaillé dans la partie cours en prenant par exemple* $\varepsilon = 10^{-6}$.
5. *À partir de combien d'itérations l'algorithme a-t-il convergé ? La reconstruction obtenue est-elle satisfaisante, ou même parfaite ? L'algorithme a-t-il séléctionné les vecteurs qui entraient en compte dans la combinaison linéaire de départ ?*

Des conditions sur la cohérence du dictionnaire vont permetttre de montrer un résultat comme le théorème 2. Cela est détaillé dans [1], chapitre 12.5.

## Références


[1] Mallat, Stéphane. *A wavelet tour of signal processing.* Academic Press, (2008)