

# Rapport de stage

dans le cadre du  
stage de fin d'études  
à  
Supélec

stage effectué au  
Groupe des problèmes inverses

Laboratoire des signaux et systèmes

ayant pour sujet:  
**Approche Bayésienne pour la séparation de sources**

présenté par:  
Hichem SNOUSSI<sup>1</sup>

responsable de stage:  
Ali MOHAMMAD-DJAFARI<sup>2</sup>

rapport édité le 21 juin 2000

1. Supélec, 91 192 Gif-sur-Yvette cedex, *snoussi@lss.supelec.fr*  
2. Supélec, 91 192 Gif-sur-Yvette cedex, *djafari@lss.supelec.fr*

# REMERCIEMENTS

Avant de commencer ce rapport, je tiens à remercier Ali MOHAMMAD-DJAFARI pour son encadrement judicieux et le temps qu'il m'a consacré.

Je tiens aussi à remercier Le laboratoire des signaux et systèmes **L2S** qui m'a accueilli au sein de son unité de recherche: le groupe des problèmes inverses **GPI**.

Merci à toute l'équipe du GPI pour la sympathie qu'ils m'ont offerte, cela m'a permis d'effectuer ce stage dans une bonne atmosphère.

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Présentation du problème: . . . . .	3
1.1.1	Modèle de mélange . . . . .	4
1.1.2	Indéterminations fondamentales . . . . .	5
1.2	Présentation de quelques méthodes pour le modèle carré non bruité: . . . . .	5
1.3	Limitations des méthodes classiques: . . . . .	7
<b>2</b>	<b>Approche Bayésienne pour la séparation de sources</b>	<b>8</b>
2.1	Approche bayésienne . . . . .	8
2.2	Algorithme JMAP . . . . .	9
2.3	Cas des sources indépendantes spatialement et temporellement . . . . .	10
2.4	Résultats de simulations . . . . .	11
2.4.1	Exemple: QAM 16 . . . . .	12
2.5	Conclusion . . . . .	13
<b>3</b>	<b>Variables cachées du mélange gaussien</b>	<b>14</b>
3.1	Variables cachées . . . . .	14
3.2	loi <i>a posteriori</i> des sources . . . . .	14
3.3	Algorithme <b>JMAPc</b> . . . . .	16
3.4	Conclusion . . . . .	16
<b>4</b>	<b>Séparation de sources avec estimation des hyperparamètres</b>	<b>17</b>
4.1	Introduction et notations . . . . .	17
4.1.1	Loi du bruit . . . . .	17
4.1.2	Loi <i>a priori</i> des éléments de la matrice de mélange . . . . .	17
4.1.3	Loi <i>a priori</i> des sources . . . . .	18
4.1.4	Loi <i>a priori</i> des hyperparamètres . . . . .	18
4.2	Estimation des hyperparamètres du mélange gaussien . . . . .	18
4.3	Algorithme <b>JMAP</b> . . . . .	20
4.4	Résultats des simulations: . . . . .	21
4.5	Conclusion . . . . .	27
<b>5</b>	<b>Algorithme EM pénalisé pour la séparation de sources</b>	<b>28</b>
5.1	Equations de ré-estimation: . . . . .	28
5.2	Résultats de simulations . . . . .	30
5.3	Conclusion . . . . .	35
<b>6</b>	<b>Conclusion</b>	<b>36</b>

# Chapitre 1

## Introduction

### 1.1 Présentation du problème :

Imaginons que, dans une chambre, deux personnes parlent simultanément. On place deux micros à deux endroits différents de la chambre. On obtient alors deux signaux temporels qu'on note  $x_1(t)$  et  $x_2(t)$  où  $t$  représente le temps. Chaque signal enregistré est une superposition des deux signaux originaux notés  $s_1(t)$  et  $s_2(t)$ . On peut exprimer cette superposition par un système d'équations linéaires :

$$\begin{aligned}x_1(t) &= a_{11}s_1 + a_{12}s_2 \\x_2(t) &= a_{21}s_1 + a_{22}s_2\end{aligned}\tag{1.1}$$

Avec  $a_{11}$ ,  $a_{12}$ ,  $a_{21}$  et  $a_{22}$  des coefficients qui dépendent des distances entre les microphones et les personnes qui parlent. On s'intéresse alors à reconstituer les paroles  $s_1(t)$  et  $s_2(t)$  seulement à partir des signaux enregistrés  $x_1(t)$  et  $x_2(t)$  ("cocktail-party problem"). Pour l'instant, on suppose qu'il n'y ait pas de retard de propagation et que les signaux enregistrés ne sont pas entachés de bruit.

Comme illustrations, on peut voir sur la figure 1 les signaux originaux  $s_1(t)$  et  $s_2(t)$ , les signaux mélangés  $x_1(t)$  et  $x_2(t)$  avec une matrice de mélange  $\mathbf{A} = \begin{pmatrix} 1 & 0.4 \\ -0.61 & \end{pmatrix}$  et les signaux reconstruits  $\hat{s}_1(t)$  et  $\hat{s}_2(t)$ .

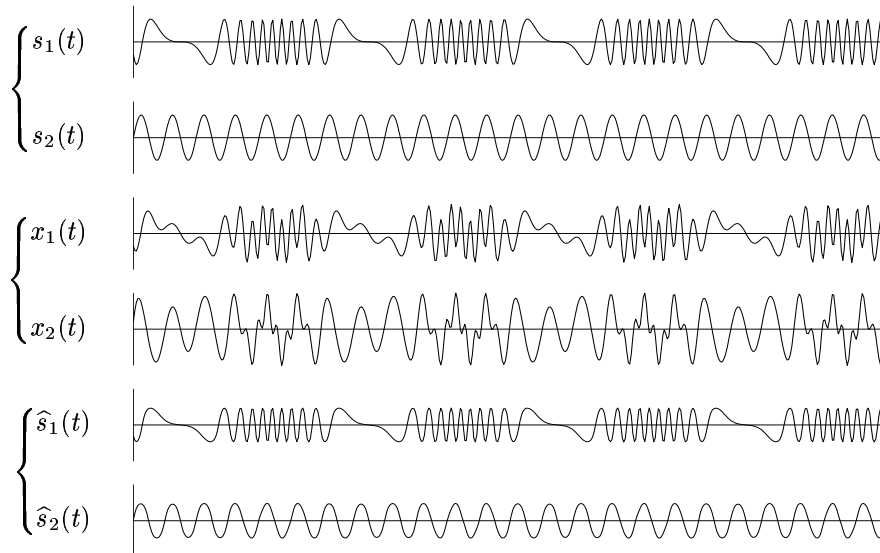


Fig. 1: Résultats de la séparation utilisant seulement les données observées

Dans le cas où on connaît les coefficients de la matrice de mélange, la séparation des sources revient simplement à résoudre le système linéaire (1.1) avec des méthodes classiques. Mais, en général,

la matrice de mélange n'est pas connue. Il s'agit alors de reconstituer les sources et d'identifier la matrice de mélange en utilisant des outils statistiques.

Il existe bien d'autres domaines où les applications de séparation de sources sont nombreuses. On peut citer par exemple le traitement d'antenne, l'identification aveugle de canal de communication, le traitement de signaux biomédicaux (ECG, EEG, ...) ou de paroles. Il existe aussi des applications en traitement d'images. Il s'agit de filtrer les images naturelles qui ont été mélangées avec du bruit. Prenons, par exemple, une image de la terre prise par un satellite. L'image numérisée est une matrice  $\mathbf{Z}$  dont les éléments sont des niveaux de gris. Cette image est la superposition d'une image originale  $\mathbf{X}$  représentant l'endroit de la terre qu'on veut observer et des nuages  $\mathbf{N}$ . Si on observe la terre pendant des périodes différentes, on obtient plusieurs images  $\mathbf{Z}_t$  qui sont des superpositions de l'image originale  $\mathbf{X}$  et des nuages  $\mathbf{N}_t$  :

$$\mathbf{Z}_t = \mathbf{X} + \alpha_t \mathbf{N}_t, \quad t = 1..N$$

On retrouve alors un mélange linéaire instantané de sources. Le but de la séparation de sources est alors de retrouver l'image originale.

On mentionne aussi une application qui prend de plus en plus d'ampleur : Les télécommunications numériques. Un exemple important est l'utilisation des techniques de séparation aveugle pour séparer le signal d'utilisateur et les signaux d'interférence des autres utilisateurs en CDMA (Code-Division Multiple Access). Les techniques ne sont pas totalement aveugles puisqu'on possède des informations *a priori* sur la structure des données du modèle CDMA. Cependant, le nombre de paramètres est tellement grand qu'on préfère utiliser des techniques de séparation aveugle en prenant en compte les propriétés statistiques qu'on peut connaître *a priori*.

### 1.1.1 Modèle de mélange

On considère  $m$  capteurs qui recueillent le mélange linéaire instantané de  $n$  sources. L'observation peut être corrompue par un bruit additif. Ce bruit peut représenter les erreurs de mesures ou éventuellement les erreurs de modélisation. A chaque instant  $t$ , on peut écrire :

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \epsilon(t) \quad (1.2)$$

où  $\mathbf{s}(t)$  désigne le vecteur source dont les composantes sont les signaux scalaires à séparer.  $\mathbf{A}$  est la matrice de mélange de dimension  $m \times n$ .  $\epsilon(t)$  représente le bruit qui a affecté la mesure.

La plupart des méthodes de séparation de sources qui existent supposent l'indépendance des sources. On peut exprimer l'indépendance par la relation suivante :

$$\forall t \in \mathbf{Z}, p_t(\mathbf{s}) = p(\mathbf{s}) = \prod_{j=1}^n p_j(s_j)$$

Les colonnes de  $\mathbf{A}$  sont supposés linéairement indépendantes. On suppose aussi que le bruit  $\epsilon$  est statistiquement indépendant des sources. Dans le problème de séparation de sources, on peut distinguer deux sous-problèmes :

- Reconstruire les sources.
- Identifier la matrice de mélange

On peut donc procéder de trois manières différentes :

1. On suppose qu'on connaît les sources (séquence d'apprentissage) et on cherche à identifier la matrice de mélange.
2. On suppose qu'on connaît la matrice de mélange. C'est le cas où soit on a procédé auparavant à l'identification de cette matrice connaissant une séquence d'apprentissage soit on maîtrise la structure physique du problème et donc on connaît parfaitement la matrice  $\mathbf{A}$ . Dans ce cas, l'estimateur des sources est tout simplement :

$$\hat{\mathbf{s}}(t) = \mathbf{A}^\# \mathbf{x}(t)$$

où  $\mathbf{A}^\# = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T$  désigne la matrice pseudo-inverse de  $\mathbf{A}$ .

3. On peut aussi procéder à une estimation conjointe des sources  $\mathbf{s}$  et de la matrice de mélange  $\mathbf{A}$ .

## 1.1.2 Indéterminations fondamentales

### Indétermination d'échelle

La reconstitution des sources et l'identification de la matrice de mélange s'effectuent avec une indétermination d'échelle. On peut se rendre compte de cette indétermination directement sur le modèle (4.1). En effet, si on multiplie la  $j$  ème colonne de la matrice  $\mathbf{A}$  par un réel non nul  $\alpha_j$  et on divise la  $j$  ème composante de la source  $\mathbf{s}_j$  par ce même réel  $\alpha_j$ , la donnée observée  $\mathbf{x}$  demeure inchangée :

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \epsilon(t) = \sum_{j=1}^n \mathbf{s}_j(t)\mathbf{a}_j + \epsilon(t) = \sum_{j=1}^n \alpha_j \mathbf{s}_j(t) \frac{\mathbf{a}_j}{\alpha_j} + \epsilon(t)$$

avec  $\mathbf{a}_j$  désignant la  $j$  ème colonne de  $\mathbf{A}$ .

### Indétermination d'ordre

Il est clair qu'en modifiant la numérotation des sources et en même temps celle des colonnes de la matrice  $\mathbf{A}$  la donnée observée  $\mathbf{x}$  ne change pas. Autrement dit, si on applique une permutation  $\mathbf{P}$  au vecteur source et aux colonnes de  $\mathbf{A}$  on ne change pas les données :

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \epsilon(t) = (\mathbf{A}\mathbf{P})(\mathbf{P}\mathbf{s}(t)) + \epsilon(t)$$

Donc, on ne peut espérer reconstruire les sources  $\mathbf{s}$  qu'à un facteur multiplicatif et une permutation près.

## 1.2 Présentation de quelques méthodes pour le modèle carré non bruité:

Le problème de séparation de sources a été largement étudié ces dernières années. La plupart des méthodes existantes consiste à :

- Dans un premier temps, fixer un critère en se basant sur certaines connaissances et propriétés statistiques préalables.
- Dans un deuxième temps, minimiser ce critère par des algorithmes itératifs ou de type gradient. La minimisation s'effectue par blocs ou d'une manière adaptative.

### Analyse en composantes indépendantes: ICA

Parmi les approches les plus connues dans ce domaine, on peut citer l'ICA (analyse en composantes indépendantes). Cette technique se base sur l'indépendance spatiale des sources (composantes indépendantes). Le principe est d'appliquer un filtrage linéaire ( $\mathbf{B}$ ) et non linéaire ( $\mathbf{g}(\cdot)$ ) aux données observées  $\mathbf{x}(t)$  de telle manière qu'à la sortie les composantes soient le plus possible indépendantes :

$$\mathbf{s} \longrightarrow \boxed{\mathbf{A}} \longrightarrow \mathbf{x} \qquad \mathbf{x} \longrightarrow \boxed{\mathbf{B}} \longrightarrow \mathbf{y} \longrightarrow \boxed{\mathbf{g}(\cdot)} \longrightarrow \hat{\mathbf{s}}$$

Ensuite, il faut définir des critères mesurant l'indépendance des signaux recueillis à la sortie du filtre. La minimisation du critère se fait par rapport à la matrice séparatrice  $\mathbf{B}$ . Parmi les techniques utilisées, on peut citer :

1. Maximum de vraisemblance (ML):

On suppose connues les densités  $p_j(\mathbf{s}_j)$ . L'indépendance s'exprime par  $p(\mathbf{s}) = \prod_{j=1}^n p_j(\mathbf{s}_j)$ . La vraisemblance de la matrice de mélange  $\mathbf{A}$  est:

$$p(\mathbf{x} | \mathbf{A}) = \frac{1}{|\det(\mathbf{A})|} \prod_{j=1}^n p_j([\mathbf{A}^{-1}\mathbf{x}]_j)$$

ou encore, en introduisant la matrice séparatrice:

$$p(\mathbf{x} | \mathbf{B}) = | \det(\mathbf{B}) | \prod_{j=1}^n p_j([\mathbf{B}\mathbf{x}]_j) = | \det(\mathbf{B}) | \prod_{j=1}^n p_j(\mathbf{y}_j)$$

L'estimée de la matrice  $\hat{\mathbf{B}}$  est l'argument maximum de la vraisemblance:

$$\hat{\mathbf{B}} = \arg \max_{\mathbf{B}} \{ \log p(\mathbf{x} | \mathbf{B}) \} = \arg \max_{\mathbf{B}} \left\{ \sum_j r_j(\mathbf{y}_j) - \log | \det(\mathbf{B}) | \right\}$$

avec  $r_j(\mathbf{y}_j) = \log p_j(\mathbf{y}_j)$

## 2. Infomax

On utilise, dans ces méthodes, l'entropie comme mesure d'indépendance de  $\mathbf{y} = g(\mathbf{B}\mathbf{x} + \mathbf{b})$ :

$$\mathcal{S} = - \sum_j p_j(\mathbf{y}_j) \log p_j(\mathbf{y}_j)$$

On va maximiser  $\mathcal{S}$  comme fonction du couple  $(\mathbf{B}, \mathbf{b})$ :

$$(\hat{\mathbf{B}}, \hat{\mathbf{b}}) = \arg \max_{(\mathbf{B}, \mathbf{b})} \{ \mathcal{S}(\mathbf{B}, \mathbf{b}) \}$$

## 3. M-estimation:

Il s'agit d'annuler une fonction de la matrice séparatrice  $\mathbf{B}$ :

$$\frac{1}{T} \sum_t \mathbf{H}(\mathbf{y}(t)) = [0] \quad \text{avec} \quad \mathbf{y}(t) = \mathbf{B}\mathbf{x}(t)$$

La fonction  $\mathbf{H}$  peut être définie par:

$$\mathbf{H}(\mathbf{y}) = E \{ f(\mathbf{y})g(\mathbf{y})^t \}$$

où  $f$  et  $g$  deux fonctions non linéaires. En prenant  $\mathbf{H}(\mathbf{y}) = \frac{\partial \log p_s(\mathbf{y})}{\partial \mathbf{B}}$ , cette technique se confond avec celle du maximum de vraisemblance (ML).

## 4. Fonction de contraste:

On définit une fonction de contraste  $c(\mathbf{y}) = c(\mathbf{B}\mathbf{x})$  qui prend son maximum quand la matrice  $\mathbf{B}$  est une matrice séparatrice. Un exemple de fonction de contraste est la mesure de l'indépendance des composantes de  $\mathbf{y}$  par la distance Kullback:

$$c(\mathbf{B}) = KL \left( p(\mathbf{y}) : \prod_{j=1}^n p_j(\mathbf{y}_j) \right) = \int p(\mathbf{y}) \log \frac{p(\mathbf{y})}{\prod_{j=1}^n p_j(\mathbf{y}_j)} d\mathbf{y}$$

Ensuite, il s'agit d'optimiser  $c(\mathbf{B})$  sous la contrainte de la blancheur de  $\mathbf{y}$ :

$$\frac{1}{T} \sum_t \mathbf{y}(t)\mathbf{y}^t(t) = \mathbf{I}$$

## 5. Statistiques d'ordres supérieures (HOS):

La mesure de l'indépendance des composantes du vecteur  $\mathbf{y}$  nécessite la connaissance des lois  $p_j(\mathbf{y}_j)$  qui ne sont pas, en général, connues. On approxime alors les expressions des probabilités par des mesures statistiques d'ordres supérieures à deux comme les cumulants. Ceci est obtenu en développant en série de Taylor la fonction caractéristique. En reprenant  $c(\mathbf{B}) = KL \left( p(\mathbf{y}) : \prod_{j=1}^n p_j(\mathbf{y}_j) \right)$ , on montre que:

$$\frac{\partial c(\mathbf{B})}{\partial \mathbf{B}} \propto E \left\{ \frac{p'_j(\mathbf{y}_j)}{p_j(\mathbf{y}_j)} \right\} \propto \text{cumulants de } (\mathbf{y}_j)$$

### 1.3 Limitations des méthodes classiques:

Les limitations de ces méthodes sont:

- Ces méthodes ne prennent pas en compte la présence du bruit. Or on sait que les mesures sont toujours entachées de bruit dû au mauvais fonctionnement des capteurs et à la distortion apportée par la propagation. Le bruit est aussi nécessaire pour exprimer notre incertitude sur le modèle.
- Ces méthodes supposent que la matrice de mélange  $\mathbf{A}$  est inversible. Ce qui justifie le passage par une matrice séparatrice pour reconstituer les sources et identifier la matrice de mélange (inverse de la matrice séparatrice). Supposer que  $\mathbf{A}$  est inversible revient à imposer un nombre de capteurs égal au nombre de sources, ce qui n'est pas toujours le cas. On peut avoir plus de capteurs que de sources (cas sur-déterminé) ou plus de sources que de capteurs (cas sous-déterminé).
- Une hypothèse fondamentale dans ces méthodes est l'indépendance spatiale et temporelle des sources. Cette hypothèse n'est pas toujours vérifiée surtout dans le domaine du traitement d'images où les pixels sont presque toujours corrélés.
- Ces techniques se basent sur l'identification d'une matrice séparatrice en adoptant le schéma suivant:

$$s \rightarrow \boxed{\mathbf{A}} \rightarrow x \qquad x \rightarrow \boxed{\mathbf{B}} \rightarrow y \rightarrow \boxed{g(\cdot)} \rightarrow \hat{s}$$

Tout au long de notre travail, nous allons traiter le problème de séparation de sources en adoptant une approche bayésienne et nous allons voir comment elle nous permet de dépasser les limitations des méthodes classiques. Les travaux menés pendant le stage sont fortement inspirés par l'article [1] écrit par Ali Mohammad-Djafari.



## Chapitre 2

# Approche Bayésienne pour la séparation de sources

Dans ce chapitre, on va essayer de résoudre le problème de séparation de sources en adoptant une approche bayésienne. Cette approche nous permet d'introduire des informations *a priori* qu'on peut avoir sur les sources et sur la matrice de mélange. Ce qui revient à régulariser le problème de la séparation qui est un problème en général mal posé. On va traiter plus spécialement le cas où les sources possèdent une densité de probabilité comme mélange de gaussiennes [2]. Cette classe de densités est bien adaptée à certains cas réels surtout en communications numériques. La loi de la matrice de mélange est supposée gaussienne. L'originalité de ce travail est de montrer une manière d'introduire nos connaissances *a priori* sur la matrice de mélange. Supposer que la loi est gaussienne revient à dire qu'on connaît la moyenne de chaque élément de la matrice de mélange avec une certaine incertitude.

### 2.1 Approche bayésienne

Le modèle du problème de séparation de sources qu'on va adopter est le suivant:

$$\mathbf{x}(t) = \mathbf{A} \mathbf{s}(t) + \boldsymbol{\epsilon}(t), \quad t = 1, \dots, T, \quad (2.1)$$

$\mathbf{s}(t) = (\mathbf{s}_1(t), \dots, \mathbf{s}_n(t))$  est le vecteur source non connu, il est de dimension  $n$ .

$\mathbf{x}(t) = (\mathbf{x}_1(t), \dots, \mathbf{x}_m(t))$  est le vecteur des observations à l'instant  $t$ , il est de dimension  $m$ .

Il est à remarquer que dans notre approche  $n$  peut être supérieur ou inférieur à  $m$ .

$\mathbf{A}$  est la matrice de mélange supposé donc linéaire, elle est rectangulaire de dimension  $(m \times n)$ .

$\boldsymbol{\epsilon}(t)$  est un bruit additif qui peut représenter le bruit de mesure ou l'incertitude de modèle.

On note dans toute la suite  $\mathbf{s}_{1..T} = (\mathbf{s}_1, \dots, \mathbf{s}_T)$  la matrice  $(n \times T)$  des sources de  $T$  échantillons, ainsi que  $\mathbf{x}_{1..T} = (\mathbf{x}_1, \dots, \mathbf{x}_T)$  la matrice  $(m \times T)$  des observations de  $T$  échantillons.

Le mélange est donc supposé instantané. On peut montrer qu'une convolution se ramène à un modèle de mélange instantané comme (4.1):

Soit un système de mélange convolutif défini comme suit:

$$\mathbf{x}(t) = \sum_{k=1}^{L-1} \mathbf{A}(k) \mathbf{s}(t-k) + \boldsymbol{\epsilon}(t), \quad t = 1, \dots, T, \quad (2.2)$$

On définit alors la matrice de mélange comme:

$$\mathbf{A} = [\mathbf{A}(L-1), \dots, \mathbf{A}(0)] \quad (2.3)$$

Et le nouveau vecteur source:

$$\bar{\mathbf{s}}(\mathbf{t}) = (\mathbf{s}^T(\mathbf{t} - \mathbf{L} + 1), \dots, \mathbf{s}^T(\mathbf{t}))^T \quad (2.4)$$

En utilisant les deux équations précédentes, on peut réécrire le modèle convolutif de la manière suivante:

$$\mathbf{x}(t) = \mathbf{A} \tilde{\mathbf{s}}(t) + \boldsymbol{\epsilon}(t), \quad t = 1, \dots, T, \quad (2.5)$$

On a ainsi ramené le mélange convolutif en un mélange instantané de la même forme que le modèle (4.1).

L'approche bayésienne consiste essentiellement à pondérer la vraisemblance  $p(\mathbf{x}_{1..T} | \mathbf{A}, \mathbf{s}_{1..T})$  par les lois *a priori* de  $\mathbf{A}$  et  $\mathbf{s}$  pour obtenir la probability distribution fonction *a posteriori* de  $\mathbf{A}$  et  $\mathbf{s}$  sachant les données observées  $\mathbf{x}$ . En supposant que les sources  $\mathbf{s}$  sont indépendantes de la matrice de mélange  $\mathbf{A}$  (qui est une hypothèse très réaliste), on peut écrire:

$$p(\mathbf{A}, \mathbf{s}_{1..T} | \mathbf{x}_{1..T}) \propto p(\mathbf{x}_{1..T} | \mathbf{A}, \mathbf{s}_{1..T}) p(\mathbf{A}) p(\mathbf{s}_{1..T}) \quad (2.6)$$

On a donc trois directions :

- Estimer conjointement la matrice de mélange et les sources en optimisant le critère  $J(\mathbf{A}, \mathbf{s}_{1..T})$ :

$$(\hat{\mathbf{A}}, \hat{\mathbf{s}}_{1..T}) = \arg \max_{(\mathbf{A}, \mathbf{s}_{1..T})} \{J(\mathbf{A}, \mathbf{s}_{1..T}) = \ln p(\mathbf{A}, \mathbf{s}_{1..T} | \mathbf{x}_{1..T})\} \quad (2.7)$$

- Calculer la marginale en  $\mathbf{A}$  en intégrant par rapport à  $\mathbf{s}$  pour obtenir  $p(\mathbf{A} | \mathbf{x}_{1..T})$  puis estimer la matrice de mélange:

$$\hat{\mathbf{A}} = \arg \max_{\mathbf{A}} \{J(\mathbf{A}) = \ln p(\mathbf{A} | \mathbf{x}_{1..T})\} \quad (2.8)$$

- Calculer la marginale en  $\mathbf{s}$  en intégrant par rapport à  $\mathbf{A}$  pour obtenir  $p(\mathbf{s} | \mathbf{x}_{1..T})$  puis estimer les sources:

$$\hat{\mathbf{s}} = \arg \max_{\mathbf{s}} \{J(\mathbf{s}) = \ln p(\mathbf{s} | \mathbf{x}_{1..T})\} \quad (2.9)$$

Dans ce papier, on va s'intéresser à la première méthode c'est-à-dire à une estimation conjointe de  $\mathbf{A}$  et  $\mathbf{s}$ .

## 2.2 Algorithme JMAP

L'algorithme proposé est un algorithme itératif qui estime l'optimum du critère joint alternativement au cours des itérations. A l'itération  $k$ , on optimise le critère par rapport à  $\mathbf{s}$  sachant la matrice de mélange à l'itération  $k - 1$ . Après on se sert de cet estimé de  $\mathbf{s}$  pour optimiser par rapport à  $\mathbf{A}$ :

$$\begin{cases} \hat{\mathbf{s}}_{1..T}^{(k)} &= \arg \max_{\mathbf{s}_{1..T}} \left\{ \ln p \left( \hat{\mathbf{A}}^{(k-1)}, \mathbf{s}_{1..T} | \mathbf{x}_{1..T} \right) \right\} \\ \hat{\mathbf{A}}^{(k)} &= \arg \max_{\mathbf{A}} \left\{ \ln p \left( \mathbf{A}, \hat{\mathbf{s}}_{1..T}^{(k)} | \mathbf{x}_{1..T} \right) \right\} \end{cases} \quad (2.10)$$

Les deux principales différences entre l' algorithme **JMAP** et l'algorithme **EM** utilis dans les mthodes de sparation de sources ( fondes sur le principe de maximum de vraisemblance ) sont:

- La première étape de l'algorithme **EM** est l'estimation (Expectation) des sources  $\mathbf{s}$  sachant les données  $\mathbf{x}$  et la matrice de mélange  $\mathbf{A}$ . Tandis que la première étape de l'algorithme **JMAP** est la maximisation de la probability distribution fonctiondes sources sachant aussi les données et la matrice de mélange. Cette différence est capitale dès que le moment d'ordre un et l'argument du maximum de la probability distribution fonctionne coincident pas. C'est le cas par exemple du mélange gaussien pour lequel ces deux valeurs ne coincident pas. En plus, il est clairement plus judicieux de choisir la maximisation (quand elle est possible bien sûr) plutôt qu' une simple moyenne.
- Dans la deuxième étape de l'algorithme **EM**, l'estimé de  $\mathbf{A}$  est le maximum de vraisemblance, alors que dans **JMAP**, l'estimé de  $\mathbf{A}$  est le maximum *a posteriori* .

### Remarque:

Dans la plupart des méthodes de séparation de sources qui existent déjà, on transforme le problème de l'identification de la matrice de mélange  $\mathbf{A}$  en la recherche d'une matrice de séparation  $\mathbf{B}$  qui est supposé être l'inverse de  $\mathbf{A}$  ce qui restreint le problème aux matrices carrées et donc un nombre de sources égal au nombre de capteurs, ce qui n'est pas toujours le cas en réalité. Or dans l'approche bayésienne, on peut se permettre de traiter tous les cas: Carré, sur-déterminé, sous-déterminé, puisque c'est la recherche de la matrice  $\mathbf{A}$  qui est effectuée et que nos informations *a priori* porte naturellement sur cette matrice de mélange non sur une autre matrice  $\mathbf{B}$  qui peut ne pas exister!!

Dans toute la suite, le bruit est supposé blanc additif gaussien (BBGA) de moyenne nulle et de variance  $\Sigma = \sigma_c^2 I$

## 2.3 Cas des sources indépendantes spatialement et temporellement

On va traiter dans cette partie des sources dont les composantes sont des mélanges de gaussiennes. Ces composantes sont supposées indépendantes entre elles et au cours du temps. La première hypothèse est réaliste surtout dans le domaine des communications numériques, plus précisément en modulation **QAM**. Tandis que la deuxième hypothèse est une hypothèse simplificatrice qui va nous permettre de comparer nos résultats avec les méthodes déjà existantes pour lesquelles l'indépendance est une hypothèse fondamentale. Or dans l'approche bayésienne, on se permet de traiter des sources corrélées spatialement et temporellement et donc de traiter des cas réalistes.

Les critères à optimiser à l'étape  $k + 1$  sont:

$$\begin{cases} J(\mathbf{s}_{1..T}) &= \ln p\left(\hat{\mathbf{A}}^{(k)}, \mathbf{s}_{1..T} | \mathbf{x}_{1..T}\right) \\ J(\mathbf{A}) &= \ln p\left(\mathbf{A}, \hat{\mathbf{s}}_{1..T}^{(k+1)} | \mathbf{x}_{1..T}\right) \end{cases} \quad (2.11)$$

L'indépendance temporelle donne:

$$\begin{cases} J(\mathbf{s}_{1..T}) &= \sum_{t=1}^T \ln p\left(\mathbf{x}(t) | \hat{\mathbf{A}}^{(k)}, \mathbf{s}(t)\right) + \sum_{t=1}^T \ln p(\mathbf{s}(t)) + cte \\ J(\mathbf{A}) &= \sum_{t=1}^T \ln p\left(\mathbf{x}(t) | \mathbf{A}, \hat{\mathbf{s}}^{k+1}(t)\right) + \ln p(\mathbf{A}) + cte \end{cases} \quad (2.12)$$

Donc l'optimisation par rapport à  $\mathbf{s}_{1..T}$  se fait de façon indépendante vecteur par vecteur:

$$\hat{\mathbf{s}}(t)^{(k+1)} = \arg \max_{\mathbf{s}} \left\{ \ln p\left(\mathbf{x}(t) | \hat{\mathbf{A}}^{(k)}, \mathbf{s}\right) + \ln p(\mathbf{s}) \right\} \quad (2.13)$$

Tandis que l'optimisation par rapport à  $\mathbf{A}$  prend en compte tous les échantillons observés. L'indépendance des composantes du vecteur source s'exprime par la propriété suivante de la probability distribution fonctionjointe:

$$\ln p(\mathbf{s}(t)) = \sum_{j=1}^n \ln p_j(\mathbf{s}_j(t)) \quad (2.14)$$

Dans le cas d'un mélange gaussien et d'indépendance des sources, la probability distribution fonction *posteriori* des sources sachant les observations  $\mathbf{x}$  et la matrice de mélange est aussi un mélange gaussien. Mais le nombre de gaussiennes constituant la probability distribution fonction de tout le vecteur source est le produit du nombre de gaussiennes d'une composante par le nombre de composantes. Donc afin de garder les expressions simples, on va traiter les sources composante par

composante. L'algorithme **JMAP** va se trouver légèrement modifié: L'optimisation par rapport au vecteur  $\mathbf{s}$  va se faire aussi alternativement:

$$\begin{cases} \widehat{\mathbf{s}}_j(t)^{(k+1)} &= \arg \max_{\mathbf{s}_j} \left\{ \ln p \left( \mathbf{s}_j | \mathbf{x}(t), \widehat{\mathbf{A}}^{(k)}, \widehat{\mathbf{s}}_{l \neq j}(t)^{(k)} \right) \right\} \\ j = 1..n \\ \widehat{\mathbf{A}}^{(k+1)} &= \arg \max_{\mathbf{A}} \left\{ \ln p \left( \mathbf{A} | \widehat{\mathbf{s}}_{1..T}^{(k+1)}, \mathbf{x}_{1..T} \right) \right\} \end{cases} \quad (2.15)$$

$\ln p \left( \mathbf{s}_j | \mathbf{x}(t), \widehat{\mathbf{A}}^{(k)}, \widehat{\mathbf{s}}_{l \neq j}(t)^{(k)} \right)$  est un mélange de gaussiennes avec le même nombre de gaussiennes que la loi *a priori* mais les coefficients, les moyennes et les variances vont dépendre de l'itération, des données  $\mathbf{x}$ , des  $\mathbf{s}(t)^k$  et de  $\widehat{\mathbf{A}}^{(k)}$  de l'itération précédente. La première étape de l'algorithme **JMAP** consiste alors à prendre la moyenne de la gaussienne dominante comme estimé de  $\mathbf{s}_j$  à l'étape  $k+1$ . Le calcul n'est pas reporté ici pour raison de brièveté.

Le critère de  $\mathbf{A}$  à maximiser est le suivant:

$$J(\mathbf{A}) = \sum_{t=1}^T \ln p \left( \mathbf{x}(t) | \mathbf{A}, \widehat{\mathbf{s}}^{k+1}(t) \right) + \ln p(\mathbf{A}(t)) + cte \quad (2.16)$$

On va prendre un *a priori* gaussien pour la matrice de mélange  $\mathbf{A}$  de moyenne  $M$ . On définit la matrice  $\Sigma_a$  dont l'élément  $\Sigma_a(i,j)$  représente la variance de l'élément  $\mathbf{A}(i,j)$  autour de sa moyenne  $M(i,j)$ . Autrement dit, c'est comme on connaît les éléments de  $\mathbf{A}$  avec un certain degré de confiance.

Le gradient de  $J(\mathbf{A})$  possède une forme simple:

$$\frac{\partial J(\mathbf{A})}{\partial \mathbf{A}_{i,j}} = \sum_{t=1}^T \frac{1}{\sigma_{\epsilon}^2} \widehat{\mathbf{s}}_j^{k+1}(t) \left( \mathbf{x}_i(t) - \left[ \mathbf{A} \widehat{\mathbf{s}}^{k+1}(t) \right]_i \right) - \frac{1}{\Sigma_{a;i,j}} (\mathbf{A}_{i,j} - M_{i,j})$$

En annulant ce gradient et en définissant  $\mu_{i,j} = \frac{\sigma_{\epsilon}^2}{\Sigma_{a;i,j}}$  on trouve la relation suivante:

$$\left[ \sum_{t=1}^T \left( \mathbf{x}(t) - \mathbf{A} \widehat{\mathbf{s}}^{k+1}(t) \right) \widehat{\mathbf{s}}^{k+1}(t)^T \right]_{i,j} - \mu_{i,j} (\mathbf{A}_{i,j} - M_{i,j}) = 0$$

Soit maintenant l'opérateur **Vect** qui transforme une matrice en un vecteur colonne en concaténant les transposés de ses lignes. On définit de la même façon l'opérateur inverse **Mat** qui à un vecteur associe une matrice. En appliquant l'opérateur **Vect** à la dernière relation, on aboutit à:

$$Vect \left( \mathbf{x}_{1..T} (\widehat{\mathbf{s}}_{1..T}^{k+1})^T \right) + \mu Vect(M) = (\mu + S^*) Vect \mathbf{A}$$

Avec  $\mu$  la matrice diagonale  $nm * nm$  dont la diagonale est le vecteur  $Vect((\mu_{i,j})_{i=1..m, j=1..n})$  et  $S^*$  la matrice dont les blocs diagonaux sont les  $\widehat{\mathbf{s}}_{1..T} \widehat{\mathbf{s}}_{1..T}^T$  de l'itération  $k+1$ . Finalement on obtient l'estimée de la matrice de mélange à l'itération  $k+1$ :

$$\widehat{\mathbf{A}}^{k+1} = Mat \left( [\mu + S^*]^{-1} \left[ \mu Vect(M) + Vect \left( \mathbf{x}_{1..T} (\widehat{\mathbf{s}}_{1..T}^{k+1})^T \right) \right] \right)$$

## 2.4 Résultats de simulations

Dans la suite, on va essayer de reconstituer des signaux qui ont été mélangés. Ces sources sont engendrées en utilisant les *a priori* utilisés dans l'algorithme **JMAP**, ce qui suppose qu'on possède de bons *a priori* sur les sources. On suppose aussi qu'on possède un bon *a priori* sur la matrice de mélange.

### 2.4.1 Exemple : QAM 16

Dans cet exemple, on considère deux sources. Chacune est un mélange de 4 gaussiennes de moyennes respectives  $-3$ ,  $-1$ ,  $1$  et  $3$  pondérées respectivement par  $0.3$ ,  $0.1$ ,  $0.4$  et  $0.2$ . La matrice de mélange est  $\mathbf{A} = \begin{pmatrix} 1 & -0.6 \\ 0.6 & 1 \end{pmatrix}$  et le bruit est blanc gaussien de variance 1. La matrice  $\mu$  de l'*a priori* de  $\mathbf{A}$  est  $\mu = \begin{pmatrix} 150 & 0 & 0 & 0 \\ 0 & 0.009 & 0 & 0 \\ 0 & 0 & 0.009 & 0 \\ 0 & 0 & 0 & 150 \end{pmatrix}$  et la moyenne de  $\mathbf{A}$  est prise égale à  $M = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ . Ce qui veut dire qu'on connaît les termes diagonaux avec une bonne incertitude tandis que les termes anti-diagonaux ne sont pas bien connus. Les figures suivantes illustrent bien qu'on arrive à séparer les sources avec des erreurs négligeables:

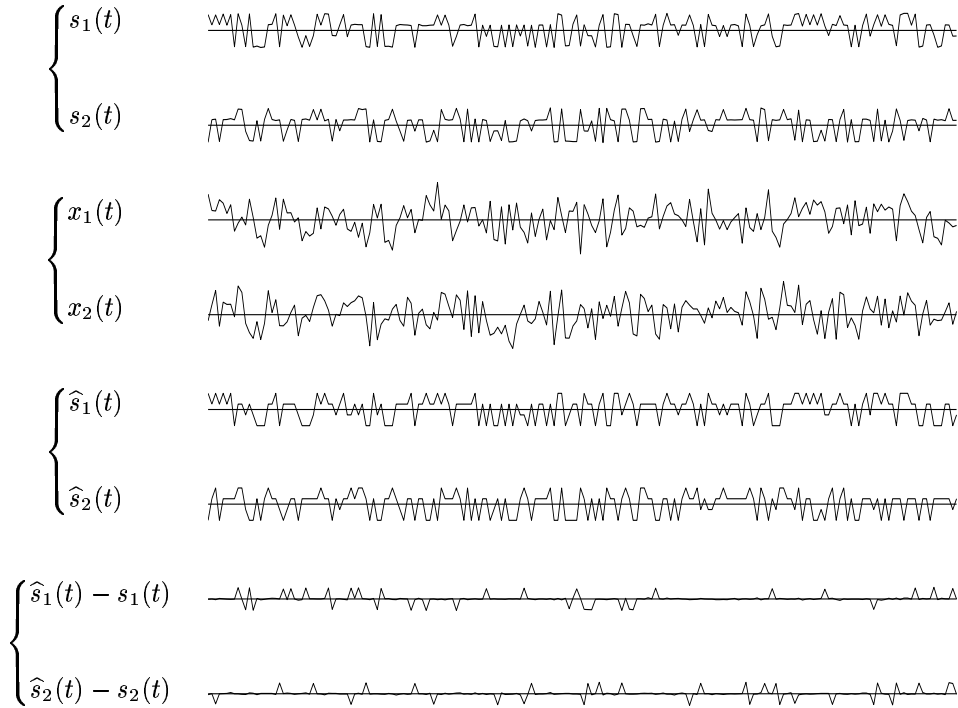


Fig. 1: Résultats de la séparation avec QAM-16

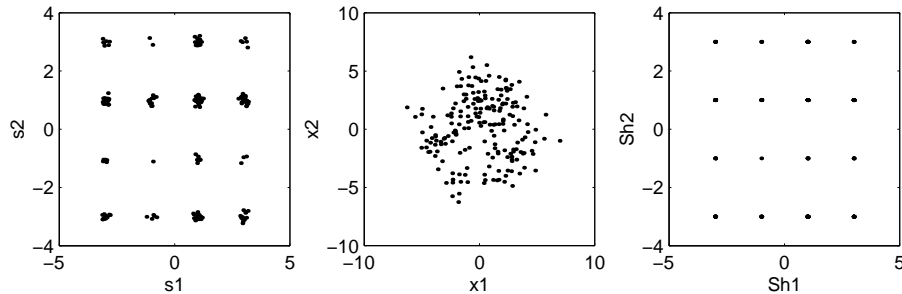


Fig. 2: Résultats de la séparation avec QAM-16: distribution dans l'espace des phases des sources, des signaux mélangés et des sources séparées

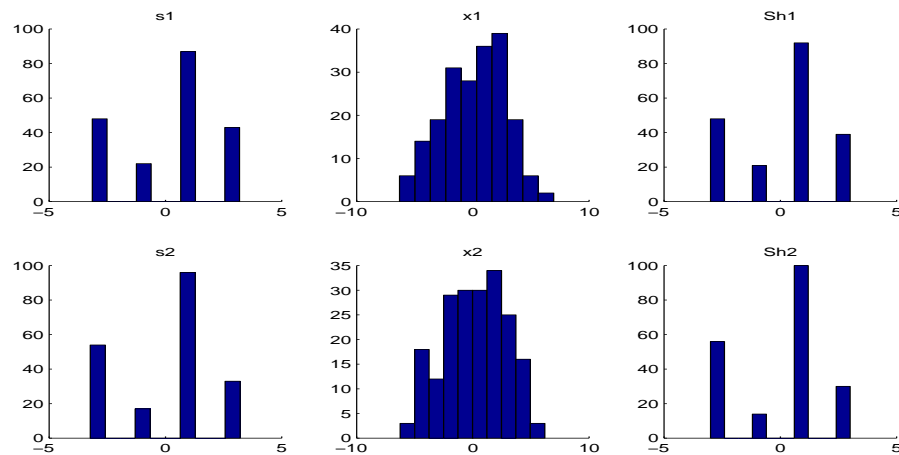


Fig. 3: Résultats de la séparation avec QAM-16: histogrammes des sources, signaux mélangés et des sources séparées

On note qu'on arrive à séparer les sources avec des erreurs négligeables. On retrouve aussi la matrice de mélange  $\hat{\mathbf{A}} = \begin{pmatrix} 1.00 & -0.57 \\ 0.65 & 1.04 \end{pmatrix}$ . D'autres simulations ont été réalisées et donnent de bons résultats quand la matrice  $\mathbf{A}$  est bien conditionnée. On note aussi la robustesse de l'algorithme face au bruit qui affecte les observations: Même avec une grande variance du bruit on arrive à retrouver les sources.

## 2.5 Conclusion

Les résultats de l'algorithme sont assez satisfaisants pour des cas bien appropriés, c'est-à-dire connaissant les bons paramètres du mélange gaussien des sources ainsi que les bons moments de la loi gaussienne de la matrice de mélange. On va essayer, désormais, ou bien d'estimer ces paramètres directement à partir des données ou bien d'effectuer une inférence bayésienne connaissant un certain *a priori* sur ces paramètres.

## Chapitre 3

# Variables cachées du mélange gaussien

Dans ce chapitre, on va apporter une modification à l'algorithme **JMAP** précédemment expliqué. Le cadre traité est toujours un mélange de gaussiennes comme un *a priori* sur les sources. L'algorithme précédent repose sur deux étapes: La première est l'estimation des sources  $\mathbf{s}$  comme le maximiseur de la loi *a posteriori* connaissant les données  $\mathbf{x}$  et la matrice de mélange  $\hat{\mathbf{A}}$  de l'itération précédente et la deuxième étape est l'estimation de  $\mathbf{A}$  par le maximum *a posteriori* connaissant les données  $\mathbf{x}$  et l'estimé des sources  $\hat{\mathbf{s}}$  de cette même étape. On va désormais ajouter une autre étape qui consiste à estimer les variables cachées par le maximum *a posteriori*. Cette étape est ensuite suivie des deux étapes de l'algorithme précédent, à savoir estimer les sources et la matrice de mélange mais connaissant cette fois les variables cachées estimées à cette itération de l'algorithme.

### 3.1 Variables cachées

Dans le cas du mélange gaussien, chaque composante  $s_j$  du vecteur source  $\mathbf{s}$  possède une probability distribution fonction de la forme suivante:

$$p(s_j) = \sum_{i=1}^{q_j} \alpha_{ji} \mathcal{N}(m_{ji}, \sigma_{ji}^2)$$

On considère maintenant la variable cachée  $z_j$  qui prend ses valeurs dans l'ensemble fini  $\mathcal{Z}_j = \{1, \dots, q_j\}$  avec  $\alpha_{ji}$  la probabilité que cette variable prend la valeur  $i$ :  $\alpha_{ji} = p(z_j = i)$ . Dans ce cas, la loi de  $s_j$  sachant  $z_j = i$  est la loi normale  $\mathcal{N}(m_{ji}, \sigma_{ji}^2)$ .

On peut étendre cette considération au cas vectoriel:

Soit alors le vecteur  $\mathbf{z} = [z_1, \dots, z_n]$  qui prend ses valeurs dans l'ensemble  $\mathcal{Z} = \prod_{j=1}^n \mathcal{Z}_j$ . La loi du vecteur source  $\mathbf{s}$  sachant  $\mathbf{z}$  est une loi normale  $p(\mathbf{s}|\mathbf{z}) = \mathcal{N}(\mathbf{m}_z, \mathbf{\Gamma}_z)$  avec:

$$\mathbf{m}_z = [m_{1z_1}, m_{2z_2}, \dots, m_{nz_n}]$$
$$\mathbf{\Gamma}_z = \text{diag}(\sigma_{1z_1}^2, \sigma_{2z_2}^2, \dots, \sigma_{nz_n}^2)$$

On peut alors écrire la loi *a priori* marginale de  $\mathbf{s}$  comme somme de  $\prod_{j=1}^n q_j$  gaussiennes:

$$p(\mathbf{s}) = \sum_{\mathbf{z} \in \mathcal{Z}} p(\mathbf{z}) p(\mathbf{s}|\mathbf{z})$$

### 3.2 loi *a posteriori* des sources

On suppose pour le calcul suivant que la matrice de mélange  $\mathbf{A}$  est connue, donc toutes les lois sont conditionnelles par rapport à cette matrice.

La loi jointe de  $\mathbf{s}$ ,  $\mathbf{z}$  et  $\mathbf{x}$  se factorise de deux manières:

$$p(\mathbf{s}, \mathbf{z}, \mathbf{x}|\mathcal{I}) = p(\mathbf{x}|\mathbf{s}) p(\mathbf{s}|\mathbf{z}) p(\mathbf{z})$$

ou

$$p(\mathbf{s}, \mathbf{z}, \mathbf{x} | \mathcal{I}) = p(\mathbf{s} | \mathbf{x}, \mathbf{z}) p(\mathbf{z} | \mathbf{x}) p(\mathbf{x})$$

D'où la loi *a posteriori* marginale de  $\mathbf{s}$  sachant  $\mathbf{x}$ :

$$p(\mathbf{s} | \mathbf{x}, \mathcal{I}) = \sum_{\mathbf{z} \in \mathcal{Z}} \frac{p(\mathbf{z}) p(\mathbf{x} | \mathbf{s}) p(\mathbf{s} | \mathbf{z})}{p(\mathbf{x})}$$

ou

$$p(\mathbf{s} | \mathbf{x}, \mathcal{I}) = \sum_{\mathbf{z} \in \mathcal{Z}} p(\mathbf{z} | \mathbf{x}) p(\mathbf{s} | \mathbf{x}, \mathbf{z})$$

D'après la deuxième forme, on voit bien que la loi *a posteriori* de  $\mathbf{s}$  est aussi un mélange de gaussiennes pondérées par la loi *a posteriori* de  $\mathbf{z}$  sachant  $\mathbf{x}$ . Notre démarche consiste alors à estimer cette variable cachée sachant les données observées.

### Calcul de la loi $p(\mathbf{s} | \mathbf{x}, \mathbf{z})$

Cette loi est normale de moyenne  $\boldsymbol{\theta}_{xz}$  et de variance  $\mathbf{V}_{xz}$ :

$$\boldsymbol{\theta}_{xz} = \mathbf{m}_z + \boldsymbol{\Gamma}_z \mathbf{A}^t \mathbf{R}_z (\mathbf{x} - \mathbf{A} \mathbf{m}_z)$$

$$\mathbf{V}_{xz} = \boldsymbol{\Gamma}_z - \boldsymbol{\Gamma}_z \mathbf{A}^t \mathbf{R}_z \mathbf{A} \boldsymbol{\Gamma}_z$$

Avec

$$\mathbf{R}_z = (\mathbf{A} \boldsymbol{\Gamma}_z \mathbf{A}^t + \mathbf{R}_n)^{-1}$$

$\mathbf{R}_n$  représente la matrice de covariance du bruit.

### Calcul de la loi $p(\mathbf{z} | \mathbf{x})$

Le calcul de la loi *a posteriori* de  $\mathbf{z}$  s'obtient en intégrant par rapport à  $\mathbf{s}$  la loi *a posteriori* conjointe de  $\mathbf{z}$  et  $\mathbf{s}$ :

$$\begin{aligned} p(\mathbf{z} | \mathbf{x}) &= \int p(\mathbf{z}, \mathbf{s} | \mathbf{x}) d\mathbf{s} \\ &\propto p(\mathbf{z}) \int p(\mathbf{x} | \mathbf{s}) p(\mathbf{s} | \mathbf{z}) d\mathbf{s} \end{aligned} \quad (3.1)$$

La dernière intégrale est celle d'une fonction gaussienne en  $\mathbf{s}$ . Le résultat est alors immédiat:

$$p(\mathbf{z} | \mathbf{x}) \propto p(\mathbf{z}) |\boldsymbol{\Gamma}_z|^{-\frac{1}{2}} |\mathbf{V}_{xz}|^{\frac{1}{2}} \exp[\mathbf{K}_{zx}]$$

où:

$$\begin{aligned} \mathbf{K}_{zx} &= -\frac{1}{2} (\mathbf{A} \mathbf{m}_z - \mathbf{x})^t \mathbf{Q}_{xz} (\mathbf{A} \mathbf{m}_z - \mathbf{x}) \\ \mathbf{Q}_{xz} &= (\mathbf{I} - \mathbf{R}_z \mathbf{A} \boldsymbol{\Gamma}_z \mathbf{A}^t) \mathbf{R}_n^{-1} (\mathbf{I} - \mathbf{A} \boldsymbol{\Gamma}_z \mathbf{A}^t \mathbf{R}_z) + \mathbf{R}_z \mathbf{A} \boldsymbol{\Gamma}_z \mathbf{A}^t \mathbf{R}_z \end{aligned} \quad (3.2)$$

Si maintenant on considère tous les échantillons observés, la loi de  $\mathbf{z}_{1..T}$  est:

$$p(\mathbf{z}_{1..T} | \mathbf{x}_{1..T}) \propto p(\mathbf{z}_{1..T}) \int p(\mathbf{x}_{1..T} | \mathbf{s}_{1..T}) p(\mathbf{s}_{1..T} | \mathbf{z}_{1..T}) d\mathbf{s}_{1..T}$$

En supposant que les  $\mathbf{z}(t)$  sont *a priori* indépendants ainsi que les sources et le bruit, la dernière relation devient:

$$p(\mathbf{z}_{1..T} | \mathbf{x}_{1..T}) \propto \prod_{t=1}^T \left\{ p(\mathbf{z}(t)) \int p(\mathbf{x}(t) | \mathbf{s}(t)) p(\mathbf{s}(t) | \mathbf{z}(t)) d\mathbf{s}(t) \right\}$$

L'estimé de  $\mathbf{z}_{1..T}$  se fait donc échantillon par échantillon:

$$\arg \max_{\mathbf{z}_{1..T}} \{p(\mathbf{z}_{1..T} | \mathbf{x}_{1..T})\} = \left( \arg \max_{\mathbf{z}(t)} \{p(\mathbf{z}(t) | \mathbf{x}(t))\} \right)_{t=1..T}$$



### 3.3 Algorithme JMAPc

L'algorithme **JMAPc** comporte trois étapes:

- (i) La première phase est l'estimation de la variable cachée  $\hat{\mathbf{z}}_{MAP}$  par le principe du maximum *a posteriori* sachant les données  $\mathbf{x}$  et la matrice de mélange de l'itération précédente:

$$\hat{\mathbf{z}}_{MAP}^{(k)}(t) = \arg \max_{\mathbf{z}(t)} \left\{ p(\mathbf{z}(t) | \mathbf{x}(t), \hat{\mathbf{A}}^{(k-1)}) \right\}$$

Ceci pour chaque instant  $t$  comme on l'a démontré dans le paragraphe précédent. Cette optimisation est assez triviale puisque elle opère sur l'ensemble fini  $\mathcal{Z}$  de  $\prod_{j=1}^n q_j$  éléments.

- (ii) La deuxième phase est l'estimation des sources. Cette étape se trouve beaucoup plus simplifiée par rapport à l'algorithme **JMAP**. En effet, sachant l'estimée  $\hat{\mathbf{z}}_{MAP}^{(k)}$ , le vecteur source suit une loi *a posteriori* gaussienne:  $\mathcal{N}(\boldsymbol{\theta}_{x\hat{\mathbf{z}}_{MAP}^{(k)}}, \mathbf{V}_{x\hat{\mathbf{z}}_{MAP}^{(k)}})$ .

L'estimé MAP de  $\mathbf{s}$  coïncide alors avec le minimiseur de l'erreur quadratique moyenne qui est égal à la moyenne de la loi, à savoir  $\boldsymbol{\theta}_{x\hat{\mathbf{z}}_{MAP}^{(k)}}$ .

- (iii) La troisième étape est l'estimation de la matrice de mélange sachant les données  $\mathbf{x}$  et l'estimée  $\hat{\mathbf{s}}^k$  de cette même itération. On adopte le même principe à savoir le maximum *a posteriori* pour évaluer  $\hat{\mathbf{A}}^{(k)}$ . Les calculs sont les mêmes que dans l'algorithme **JMAP**.

### 3.4 Conclusion

Cette modification apportée à l'algorithme de départ (raisonnement avec les variables cachées), nous permet d'avoir une "bonne" loi *a posteriori* du vecteur source lors de la deuxième étape. Nous entendons par une "bonne" loi une probability distribution function mono-modale qui permet une estimation par point qui minimise l'erreur quadratique moyenne. En plus, on peut bien étudier le comportement asymptotique de l'algorithme quand le nombre d'échantillons tend vers l'infini et on peut prouver que l'estimée  $\hat{\mathbf{A}}$  de la matrice de mélange tend vers la bonne matrice  $\mathbf{A}$ .

## Chapitre 4

# Séparation de sources avec estimation des hyperparamètres

L'algorithm **JMAP** proposé précédemment repose sur une bonne connaissance des paramètres des lois *a priori* des sources et ceux de la matrice de mélange à estimer. Autrement dit, la convergence des algorithmes vers les bonnes solutions dépend du bon choix de ces hyperparamètres à fixer. Ce qui peut sembler, à premier abord, comme une limitation sévère de notre approche bayésienne pour la séparation de sources. Nous montrons dans cette partie du travail, qu'il est possible avec une approche bayésienne d'estimer conjointement les sources, la matrice de mélange et les paramètres du mélange gaussien qui représente l'*a priori* choisi pour modéliser la classe des distributions des sources. Ceci n'évite pas, bien entendu, d'inclure de nouveaux "hyper-hyperparamètres" qui paramétrisent les lois *a priori* des hyperparamètres, mais nous montrerons, lors les simulations, la robustesse du nouvel algorithme vis-à-vis de ces nouveaux paramètres de nuisance. Ce travail est bas sur l'article [3]

### 4.1 Introduction et notations

Le modèle du problème de séparation de sources est le suivant:

$$\mathbf{x}(t) = \mathbf{A} \mathbf{s}(t) + \boldsymbol{\epsilon}(t), \quad t = 1, \dots, T, \quad (4.1)$$

$\mathbf{s}(t) = (\mathbf{s}_1(t), \dots, \mathbf{s}_n(t))$  est le vecteur source non connu, il est de dimension  $n$ .

$\mathbf{x}(t) = (\mathbf{x}_1(t), \dots, \mathbf{x}_m(t))$  est le vecteur des observations à l'instant  $t$ , il est de dimension  $m$ .

$\mathbf{A}$  est la matrice de mélange supposé donc linéaire, elle est rectangulaire de dimension  $(m \times n)$ .

$\boldsymbol{\epsilon}(t)$  est un bruit additif qui peut représenter le bruit de mesure ou l'incertitude de modèle.

Il est à remarquer que dans notre approche  $n$  peut être supérieur ou inférieur à  $m$ .

On note dans toute la suite  $\mathbf{s}_{1..T} = (\mathbf{s}_1, \dots, \mathbf{s}_T)$  la matrice  $(n \times T)$  des sources de  $T$  échantillons, ainsi que  $\mathbf{x}_{1..T} = (\mathbf{x}_1, \dots, \mathbf{x}_T)$  la matrice  $(m \times T)$  des observations de  $T$  échantillons.

#### 4.1.1 Loi du bruit

Le bruit est supposé blanc gaussien de moyenne nulle et de variance  $\mathbf{R}_n = \sigma_\epsilon^2 \mathbf{I}$ , on notera  $\phi_\epsilon = \frac{1}{\sigma_\epsilon^2}$  qu'on supposera connue par la suite:

$$\boldsymbol{\epsilon}(t) \sim \mathcal{N}\left(0, \frac{1}{\phi_\epsilon} \mathbf{I}\right)$$

#### 4.1.2 Loi *a priori* des éléments de la matrice de mélange

Nous imposons à la matrice de mélange  $\mathbf{A}$  une loi gaussienne de moyenne  $\mathbf{M}$  et de variance  $\mathbf{R}_a = (\sigma_{a,ij}^2)_{i=1,\dots,m,j=1,\dots,n}$ . On définit alors la matrice  $\boldsymbol{\Lambda}$  comme:  $\boldsymbol{\Lambda}_{ij} = \frac{\sigma_{a,ij}^2}{\sigma_{a,ij}^2}$  Ce qui permet de

dire qu'on connaît les éléments de  $\mathbf{A}$  ( moyenne  $\mathbf{M}$  ) avec une certaine incertitude ( variance  $\mathbf{R}_a$  ). Notons que la matrice  $\mathbf{A}$  est supposée constante au cours du temps.

### 4.1.3 Loi *a priori* des sources

Les composantes du vecteur source  $\mathbf{s}(t)$  sont supposées, dans la suite, indépendantes entre elles et au cours du temps. Nous supposons que chaque composante  $\mathbf{s}_j$  possède une densité de probabilité comme mélange de gaussiennes:

$$\mathbf{s}_j \sim \sum_{z=1}^{q_j} \Pi_{jz} \mathcal{N} \left( m_{jz}, \frac{1}{\psi_{jz}} \right)$$

Nous modélisons cette loi d'une manière hiérarchique en considérant la variable cachée  $z_j$  qui prend ses valeurs dans l'ensemble fini  $\mathcal{Z}_j = (1, \dots, q_j)$  avec  $\Pi_{ji}$  la probabilité que cette variable prend la valeur  $i$ :  $\Pi_{ji} = p(z_j = i)$ . Dans ce cas, la loi de  $\mathbf{s}_j$  sachant  $z_j = i$  est la loi normale  $\mathcal{N}(m_{ji}, \frac{1}{\psi_{jz}})$ .

### 4.1.4 Loi *a priori* des hyperparamètres

#### Moyennes du mélange gaussien

La moyenne  $m_{j,z}$  de la  $z$ ème gaussienne relative à la  $j$ ème composante du vecteur source suit une loi normale de moyenne  $\mu_{jz}$  et de variance  $\frac{1}{\gamma_{jz}}$ :

$$m_{jz} \sim \mathcal{N} \left( \mu_{jz}, \frac{1}{\gamma_{jz}} \right)$$

#### Variances du mélange gaussien

La variance  $\frac{1}{\psi_{j,z}}$  de la  $z$ ème gaussienne relative à la  $j$ ème composante du vecteur source suit une loi Gamma inverse:

$$\psi_{jz} \sim \mathcal{G}(\alpha, \beta)$$

## 4.2 Estimation des hyperparamètres du mélange gaussien

Dans ce paragraphe, nous détaillons la stratégie employée pour estimer les moyennes et les variances du mélange gaussien à partir des observations bruitées  $\mathbf{x}(t)$  pour  $t = 1..T$ . Cette méthode est aussi valable lorsqu'on peut observer directement les sources bruitées.

L'idée est la suivante: Conditionnellement aux variables cachées  $(z_j)_{1..T} = [z_j(1), \dots, z_j(T)]$ , l'estimation des hyperparamètres  $m_{jz}$  et  $\psi_{jz}$  pour  $z \in \mathcal{Z}_j = (1, \dots, q_j)$  revient tout simplement à l'estimation des hyperparamètres d'une loi *a priori* gaussienne. En fait, connaissant le vecteur  $(z_j)_{1..T} = [z_j(1), \dots, z_j(T)]$  on peut effectuer une partition de l'ensemble  $\mathcal{T} = [1, \dots, T]$  en sous ensembles  $\mathcal{T}_z$  tel que:

$$\mathcal{T}_z = \{ t \mid z_j(t) = z \}, z \in \mathcal{Z}_j$$

Supposons maintenant que la matrice de mélange  $\mathbf{A}$  et les composantes  $\mathbf{s}_{i \neq j}$  sont fixées et nous allons nous intéresser à l'estimation de  $m_{jz}$  et  $\psi_{jz}$ . Soit  $\theta_{jz} = (m_{jz}, \psi_{jz})$ .

La loi *a posteriori* conjointe pour  $\mathbf{s}_j$  et  $\theta_{jz}$  sachant la donnée  $\mathbf{x}$  et  $z_j$  à l'instant  $t$  est:

$$p(\mathbf{s}_j, \theta_{jz} \mid \mathbf{x}, z_j) \propto p(\mathbf{x} \mid \mathbf{s}) p(\mathbf{s}_j \mid \theta_{jz}, z_j) p(\theta_{jz} \mid z_j)$$

$p(\mathbf{s}_j \mid \theta_{jz}, z_j)$  est une gaussienne de moyenne  $m_{jz}$  et de variance l'inverse de  $\psi_{jz}$ .

$p(\theta_{jz} \mid z_j) = p(\theta_{jz}) = p(m_{jz}) p(\psi_{jz})$  est l'*a priori* des hyperparamètres. La loi *a posteriori* marginale de  $\theta_{jz}$  s'obtient donc par l'intégration de la relation précédente par rapport à  $\mathbf{s}_j$ . Cette intégrale se calcule aisément puisque l'expression est gaussienne en  $\mathbf{s}_j$ :

$$p(\theta_{jz} | \mathbf{x}, z_j) \propto p(\theta_{jz}) \int_{\mathbf{s}_j} p(\mathbf{x} | \mathbf{s}) p(\mathbf{s}_j | \theta_{jz}, z_j) d\mathbf{s}_j.$$

L'expression à l'intérieur de l'intégrale est proportionnelle à la loi *a posteriori* conjointe de  $(\mathbf{s}_j, z_j)$  sachant  $\mathbf{x}$  et  $\theta_{jz}$  et donc:

$$p(\theta_{jz} | \mathbf{x}, z_j) \propto p(\theta_{jz}) p(z_j | \mathbf{x}, \theta_{jz}).$$

Le calcul de  $p(z_j | \mathbf{x}, \theta_{jz})$  a été effectué dans la partie précédente et donne:

$$p(z_j | \mathbf{x}, \theta_{jz}) \propto \sqrt{\frac{\phi_j \psi_{jz}}{\phi_j + \psi_{jz}}} \exp \left[ -\frac{1}{2} \frac{\phi_j \psi_{jz}}{\phi_j + \psi_{jz}} (m_j - m_{jz})^2 \right] \quad (4.2)$$

Avec les quantités  $\phi_j$  et  $m_j$  qui dépendent des données observées et la loi du bruit:

$$\phi_j = \frac{1}{\sigma_j^2} = \frac{\sum_{i=1}^m \mathbf{A}_{ij}^2}{\sigma_\epsilon^2}$$

et

$$\begin{cases} m_j &= \frac{\sum_{i=1}^m \mathbf{A}_{ij} (\mathbf{x}_i - \hat{\mathbf{x}}_i)}{\sum_{i=1}^m \mathbf{A}_{ij}^2} \\ \hat{\mathbf{x}}_i &= \sum_{l \neq j} \mathbf{A}_{il} \mathbf{s}_l \end{cases} \quad (4.3)$$

Il est clair que l'expression (??) représente la vraisemblance des hyperparamètres  $\theta_{jz}$  et on montrera lors des simulations que l'estimation par maximum de vraisemblance du paramètre  $\psi_{jz}$  ne donne pas de résultats satisfaisants et que la solution diverge. D'où la nécessité de la régularisation par des *a priori* convenables.

### Estimation MAP des moyennes du mélange gaussien

La loi *a posteriori* de  $m_{jz}$  tenant compte de tous les échantillons s'écrit de la manière suivante:

$$p(m_{jz} | \mathbf{x}_{1..T}, (\mathbf{s}_{l \neq j})_{1..T}, \mathbf{A}, z_{1..T}, \psi_{jz}) \propto \mathcal{N} \left( \mu_{jz}, \frac{1}{\gamma_{jz}} \right) \prod_{t \in \mathcal{T}_z} \mathcal{N} \left( m_j(t), \frac{\phi_j + \psi_{jz}}{\phi_j \psi_{jz}} \right)$$

En annulant le gradient du logarithme de cette expression, on trouve l'estimé MAP de  $m_{jz}$ :

$$\widehat{m}_{jz}^{MAP} = \frac{\gamma_{jz} \mu_{jz} + \frac{\phi_j \psi_{jz}}{\phi_j + \psi_{jz}} \sum_{t \in \mathcal{T}_z} m_j(t)}{\gamma_{jz} + T_z \frac{\phi_j \psi_{jz}}{\phi_j + \psi_{jz}}}$$

Avec  $T_z$  le cardinal de l'ensemble  $\mathcal{T}_z$ .

### Estimation MAP des variances du mélange gaussien

La loi *a posteriori* de  $\psi_{jz}$  tenant compte de tous les échantillons s'écrit de la manière suivante:

$$\begin{aligned} p(\psi_{jz} | \mathbf{x}_{1..T}, (\mathbf{s}_{l \neq j})_{1..T}, \mathbf{A}, z_{1..T}, \widehat{m}_{jz}^{MAP}) &\propto \mathcal{G}(\alpha, \beta) \left( \frac{\phi_j \psi_{jz}}{\phi_j + \psi_{jz}} \right)^{\frac{T_z}{2}} \times \\ &\exp \left[ -\frac{1}{2} \frac{\phi_j \psi_{jz}}{\phi_j + \psi_{jz}} \sum_{t \in \mathcal{T}_z} (m_j(t) - \widehat{m}_{jz}^{MAP})^2 \right] \end{aligned} \quad (4.4)$$

**Remarque:** L'*a priori* Gamma inverse pour la variance n'est pas une loi conjuguée pour la vraisemblance qui est une loi Gamma en  $\lambda_{jz} = \frac{\phi_j \psi_{jz}}{\phi_j + \psi_{jz}}$ . Donc, afin d'avoir des expressions faciles à traiter, nous proposons deux solutions:

- Connaissant l'ordre de grandeur de  $\psi_{jz}$  par rapport  $\phi_j$ , on effectue un développement au premier ordre de l'expression de la vraisemblance qui devient une loi Gamma en  $\psi_{jz}$ . On aboutit donc à une loi *a posteriori* Gamma pour  $\psi_{jz}$ :

$$p(\psi_{jz} | \mathbf{x}_{1..T}, (\mathbf{s}_{l \neq j})_{1..T}, \mathbf{A}, z_{1..T}, \widehat{m}_{jz}^{MAP}) \propto \mathcal{G}(\alpha^{posteriori}, \beta^{posteriori})$$

L'estimée MAP de  $\psi_{jz}$  est alors:

$$\widehat{\psi}_{jz}^{MAP} = \frac{\alpha^{posteriori} - 1}{\beta^{posteriori}}$$

- Une deuxième possibilité est d'ajouter un *a priori*  $\mathcal{G}(\alpha, \beta)$  en  $\lambda_{jz} = \frac{\phi_j \psi_{jz}}{\phi_j + \psi_{jz}}$ . Autrement dit:

$$p(\psi_{jz}) \propto \left( \frac{\phi_j \psi_{jz}}{\phi_j + \psi_{jz}} \right)^{\alpha-1} \exp \left[ -\beta \frac{\phi_j \psi_{jz}}{\phi_j + \psi_{jz}} \right]$$

Et l'expression de l'estimé MAP devient:

$$\widehat{\psi}_{jz}^{MAP} = \frac{\frac{T_z}{2} + \alpha - 1}{\beta + \frac{\sum_{t \in \mathcal{T}_z} (m_j(t) - \widehat{m}_{jz}^{MAP})^2}{2} - \frac{(\frac{T_z}{2} + \alpha - 1)}{\phi_j}}$$

Dans la suite, on va adopter la première solution qui permet un réglage simple des paramètres  $\alpha$  et  $\beta$ .

### 4.3 Algorithme JMAP

On va présenter dans ce paragraphe l'algorithme itératif proposé **JMAP** qui permet de séparer les sources, identifier la matrice de mélange et estimer les hyperparamètres du mélange gaussien de manière conjointe:

#### I. Paramètres:

$$M, \mathbf{\Lambda}, \sigma_\epsilon^2, (\Pi_{jz})_{j=1..n, z=1..q_j}, \mathbf{q} = (q_j)_{j=1..n}, \alpha, \beta, (\mu_{jz})_{j=1..n, z=1..q_j}, (\gamma_{jz})_{j=1..n, z=1..q_j}.$$

#### II. Données:

$$\mathbf{x}(t), \text{ pour } t = 1..T$$

#### III. Conditions initiales:

- $\mathbf{A}^{(0)} = \mathbf{I}$
- $m_{jz}^{(0)}$
- $\psi_{jz}^{(0)}$
- $\mathbf{s}^{(0)}$  générées selon la loi  $\mathbf{s}_j^{(0)} \sim \sum_{z=1}^{q_j} \Pi_{jz} \mathcal{N}(m_{jz} 0, \frac{1}{\psi_{jz} 0})$

#### IV. Etapes de la boucle de l'algorithme:

l'algorithme se compose de cinq étapes:

##### (a) Classification des données:

On possède une expression explicite de la loi *a posteriori* des variables  $z_j(t)$  à chaque instant  $t$ . Cette loi est conditionnelle par rapport à l'observation  $\mathbf{x}(t)$  et des estimées des composantes  $\mathbf{s}_{t \neq j}$ , de la matrice  $\mathbf{A}$  et des hyperparamètres  $(m_{jz}, \psi_{jz})$  à l'itération précédente. Ceci nous fournit alors les estimées  $(\widehat{z_j})_{1..T}^{MAP}$  par maximum *a posteriori*:

$$(\widehat{z_j})_{1..T}^{MAP} = (\arg \max_{z_j} \{p(z_j | \mathbf{x}(t), m_{jz}, \psi_{jz}, \mathbf{A}, \mathbf{s}_{t \neq j})\})_{1..T}$$

On effectue après, une estimation de la répartition:

$$\widehat{\mathcal{T}}_z = \left\{ t \mid (\widehat{z_j})_{1..T}^{MAP}(t) = z \right\}$$

(b) **Estimation des hyperparamètres :**

Ayant les estimées des sous ensembles de la répartition lors de la première étape, on estime les hyperparamètres par MAP  $\hat{\psi}_{jz}^{MAP}$  et  $\hat{m}_{jz}^{MAP}$  comme c'est détaillé dans le paragraphe précédent.

Les trois prochaines étapes sont les mêmes que dans l'algorithme précédent **JMAP** expliqué dans le travail précédent. En effet, on possède désormais les hyperparamètres estimés et donc les deux algorithmes se rejoignent.

(c) **Estimation des variables cachées :**

Après la remise à jour des hyperparamètres, on remet aussi à jour les variables cachées:  $(\hat{z}_j)_{1..T}^{MAP}$ .

(d) **Estimation des sources :**

Connaissant la donnée  $\mathbf{x}(t)$ , la variable cachée  $z(t)$ , la matrice de mélange  $\mathbf{A}$  et les hyperparamètres du mélange gaussien, la loi *a posteriori* de  $\mathbf{s}(t)$  est une loi normale de moyenne  $\boldsymbol{\theta}_{xz}$  et de variance  $\mathbf{V}_{xz}$ :

$$\boldsymbol{\theta}_{xz} = \mathbf{m}_z + \boldsymbol{\Gamma}_z \mathbf{A}^t \mathbf{R}_z (\mathbf{x} - \mathbf{A} \mathbf{m}_z)$$

$$\mathbf{V}_{xz} = \boldsymbol{\Gamma}_z - \boldsymbol{\Gamma}_z \mathbf{A}^t \mathbf{R}_z \mathbf{A} \boldsymbol{\Gamma}_z$$

Avec

$$\begin{cases} \mathbf{R}_z &= (\mathbf{A} \boldsymbol{\Gamma}_z \mathbf{A}^t + \mathbf{R}_n)^{-1} \\ \mathbf{m}_z &= [m_{1z_1}, m_{2z_2}, \dots, m_{nz_n}] \\ \boldsymbol{\Gamma}_z &= \text{diag}(\sigma_{1z_1}^2, \sigma_{2z_2}^2, \dots, \sigma_{nz_n}^2) \end{cases} \quad (4.5)$$

$\mathbf{R}_n$  représente la matrice de covariance du bruit.

Donc l'estimée de  $\mathbf{s}(t)$  est simplement la moyenne de la gaussienne, à savoir  $\boldsymbol{\theta}_{xz}$ .

(e) **Estimation de la matrice de mélange :**

Les calculs sont les mêmes que dans la version précédente de l'algorithme **JMAP**: L'estimation de la matrice de mélange  $\mathbf{A}$  s'effectue en annulant le gradient de la loi *a posteriori* de  $\mathbf{A}$  sachant les données  $\mathbf{x}(t)$  et les estimés des sources  $\mathbf{s}(t)$  et des hyperparamètres  $\boldsymbol{\theta}_{xz}$  de cette même itération. Rappelons que l'*a priori* de  $\mathbf{A}$  est choisi comme une loi gaussienne.

## 4.4 Résultats des simulations:

### Génération de données:

On a considéré deux sources, chacune est un mélange de deux gaussiennes centrées en  $\pm 1$  et  $\psi = 100$  la même pour toutes les gaussiennes. Ces sources sont mélangées avec une matrice  $\mathbf{A} = \begin{pmatrix} 1 & -0.6 \\ 0.4 & 1 \end{pmatrix}$ . Un bruit de variance  $\sigma_\epsilon^2 = 0.03$  a été ajouté. Le nombre d'échantillons est égal à 1000.

### Paramètres de l'algorithme:

$$M = \mathbf{I}, \Lambda = \begin{pmatrix} 150 & 0.01 \\ 0.01 & 150 \end{pmatrix}, \Pi = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}, \mathbf{q} = [2, 2], \alpha = 200, \beta = 2, \mu = \begin{pmatrix} -1 & 1 \\ -1 & 1 \end{pmatrix}, \gamma = \begin{pmatrix} 0.001 & 0.001 \\ 0.001 & 0.001 \end{pmatrix}.$$

**Remarque:** Les valeurs des éléments de  $Mu$  indiquent que l'incertitude est très petite sur les termes diagonaux de  $\mathbf{A}$  et qu'elle est très grande pour les termes anti-diagonaux. Le réglage de  $\alpha$  et  $\beta$  est fait en prenant en compte des considérations sur le rapport **SNR** signal sur bruit qui est de l'ordre de 15 dB pour cet exemple.

### Conditions initiales:

$$\psi^{(0)} = \begin{pmatrix} 80 & 80 \\ 80 & 80 \end{pmatrix}, m^{(0)} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \text{ le reste: Voir plus haut.}$$

## Résultats

Les sources sont reconstruites avec des erreurs quadratiques moyennes très faibles:  $EQM(s_1) = 0.0094$ ,  $EQM(s_2) = 0.0097$ . On a représenté sur la figure suivante les sources, les mélanges, les sources reconstruites et les différences entre les sources et leurs estimées pour les 100 premiers échantillons:

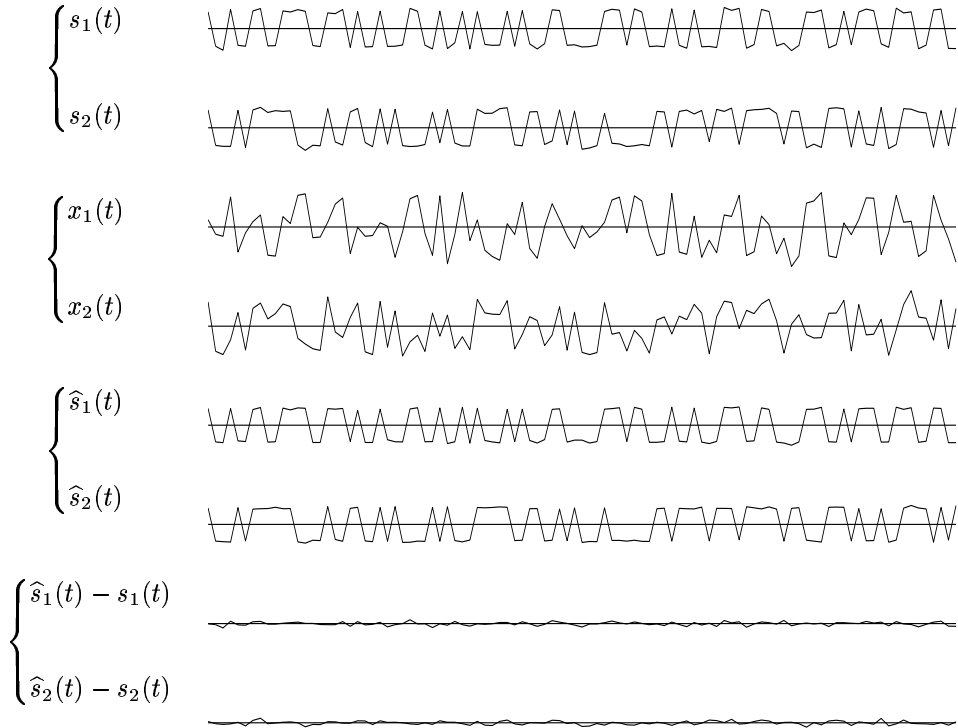


Fig. 1: Résultats de la séparation avec  $SNR = 15\text{ dB}$

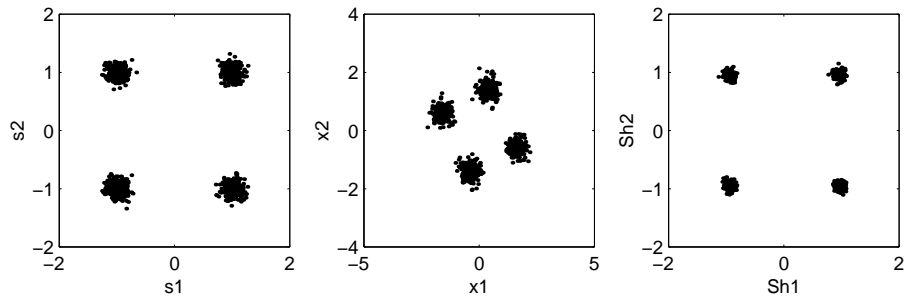


Fig. 2: Résultats de la séparation avec  $SNR = 15\text{ dB}$ : distribution dans l'espace des phases des sources, des signaux mélangés et des sources séparées

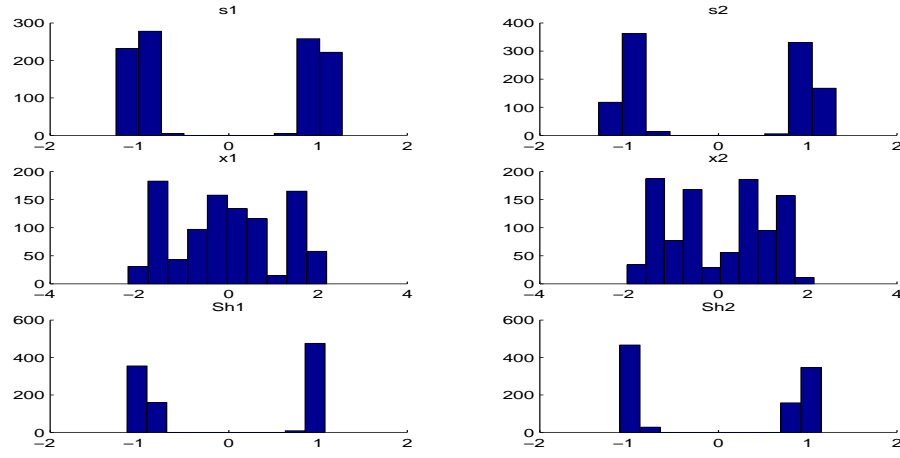


Fig. 3: Résultats de la séparation avec  $SNR = 15 \text{ dB}$ : histogrammes des sources, signaux mélangés et des sources séparées

On a ensuite repris un critère qui a été défini dans [?] afin de pouvoir comparer les résultats de notre algorithme **JMAP** avec ceux des algorithmes **PWS** et **NS** proposés par O.Macchi et E.Moreau et ceux de l'algorithme **EASI** proposé par J.F.Cardoso:

$$ind(S = \hat{\mathbf{A}}^{-1} \mathbf{A}) = \frac{1}{2} \left[ \sum_i \left( \sum_j \frac{|S_{ij}|^2}{\max_l |S_{il}|^2} - 1 \right) + \sum_j \left( \sum_i \frac{|S_{ij}|^2}{\max_l |S_{lj}|^2} - 1 \right) \right]$$

La figure 4 représente l'évolution de ce critère de performance au cours des itérations. On remarque que l'algorithme converge bien vers une valeur satisfaisante de  $-45 \text{ dB}$  et ceci à partir de la dixième itération. Pour le même rapport signal sur bruit et même matrice de mélange, les algorithmes **PWS**, **NS** et **EASI** ne fournissent pas des valeurs au dessous de  $-35 \text{ dB}$  même pour un nombre d'échantillons supérieur à 6000.

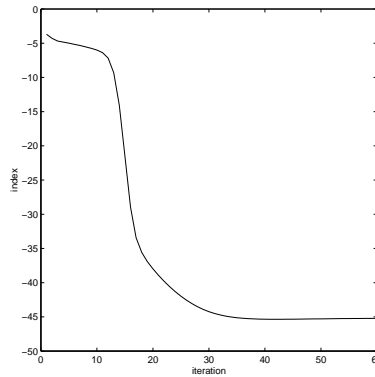


Fig. 4: Résultats de l'identification de la matrice de mélange pour 1000 échantillons et  $SNR = 15 \text{ dB}$

La figure 5 représente l'évolution de l'hyperparamètre  $\psi(1,1)$  au cours des itérations. On remarque la convergence vers une valeur satisfaisante. En fait, il ne faut pas oublier que:

$$\sigma = \frac{1}{\sqrt{\psi}} \Rightarrow d\sigma = -\frac{d\psi}{2\sqrt{\psi}\psi} \quad (4.6)$$



Et donc si  $\psi$  est de l'ordre de 100, l'erreur sur  $\sigma$  va être celle de  $\psi$  divisée par 2000!

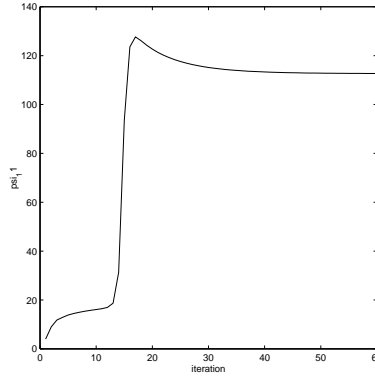


Fig. 5: Résultats de l'estimation de l'hyperparamètre  $\psi(1,1)$  pour 1000 échantillons et  $SNR = 15 dB$

La figure 6 représente l'évolution de l'hyperparamètre  $m(1,1)$  au cours des itérations. On note bien la convergence vers la valeur voulue en rappelant qu'on est parti d'une condition initiale nulle et que l'*a priori* sur les moyennes est presque uniforme (variance de l'ordre de 1000).

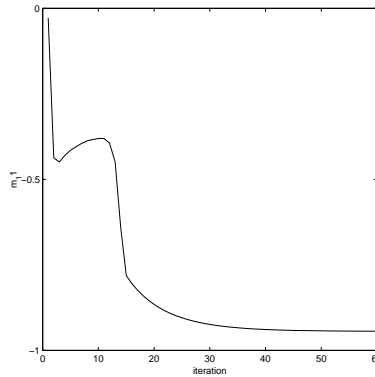


Fig. 6: Résultats de l'estimation de l'hyperparamètre  $m(1,1)$  pour 1000 échantillons et  $SNR = 15 dB$

Afin de valider l'utilisation de la classification, nous avons tracé les erreurs cumulées sur la partition estimée, autrement dit, le nombre d'échantillons mal classés. La figure 7 représente l'évolution de l'erreur de partition au cours des itérations. On note bien la convergence de l'erreur vers la valeur nulle. Donc l'estimation des hyperparamètres s'effectue sur les bons échantillons et elle est donc la plus fidèle possible.

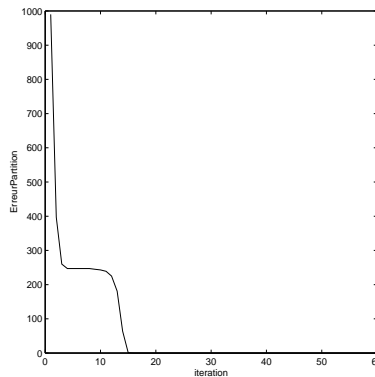


Fig. 7: Evolution des erreurs cumulées au cours des itérations pour 1000 échantillons et  $SNR = 15 dB$

### Résultats pour $SNR = 0 dB$

On a simulé le même exemple en augmentant le bruit jusqu'à atteindre  $SNR = 0 dB$ . Il a fallu modifier  $\alpha$  et  $\beta$  mais l'algorithme reste toujours robuste par rapport à des fluctuations de ces paramètres. Le développement limité de la vraisemblance de l'hyperparamètre  $\psi$  doit aussi remise

à jour puisque l'ordre de grandeur de  $\psi$  par rapport à  $\phi_j$  change aussi. Nous partons toujours des mêmes conditions initiales.

$EQM(s_1) = 0.64$  et  $EQM(s_2) = 0.50$ .

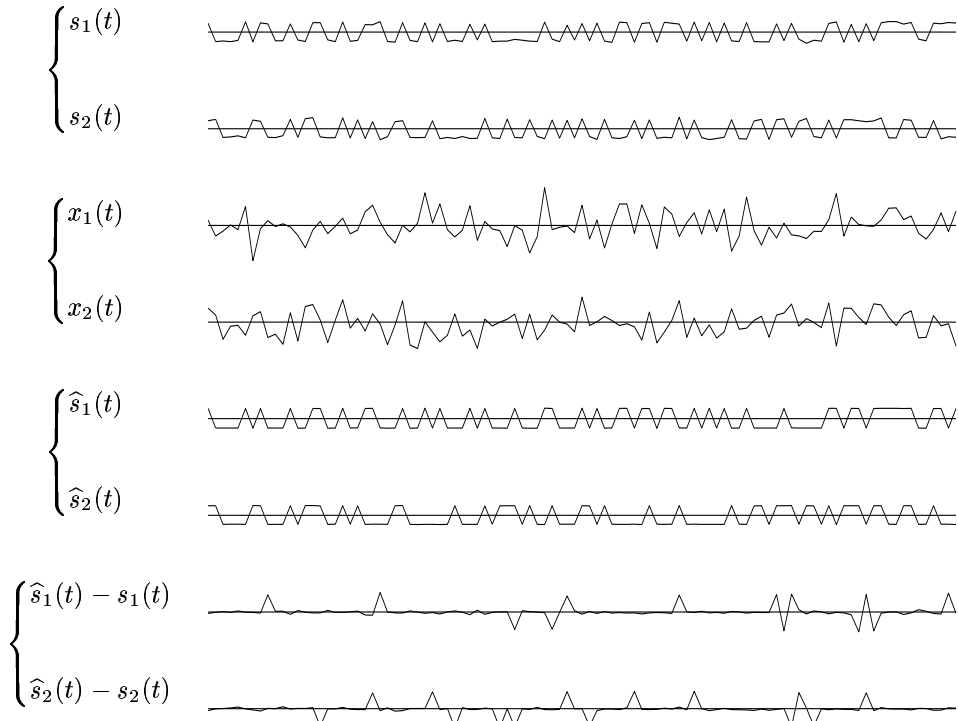


Fig. 8: Résultats de la séparation avec  $SNR = 0\text{ dB}$

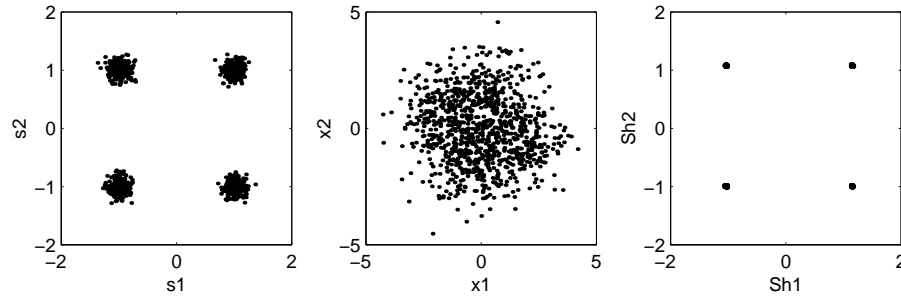


Fig. 9: Résultats de la séparation avec  $SNR = 0\text{ dB}$ : distribution dans l'espace des phases des sources, des signaux mélangés et des sources séparées

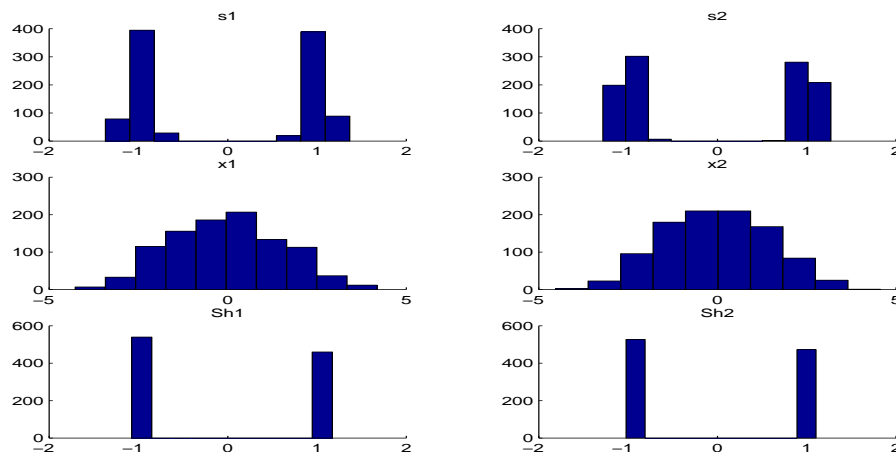


Fig. 10: Résultats de la séparation avec  $SNR = 0 dB$ : histogrammes des sources, signaux mélangés et des sources séparées

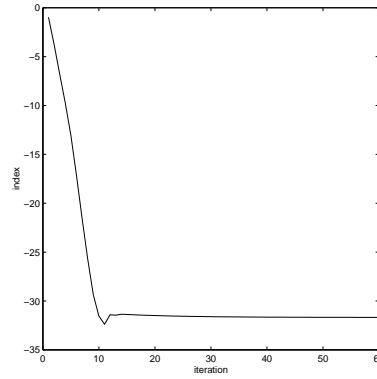


Fig. 11: Résultats de l'identification de la matrice de mélange pour 1000 échantillons avec  $SNR = 0 dB$

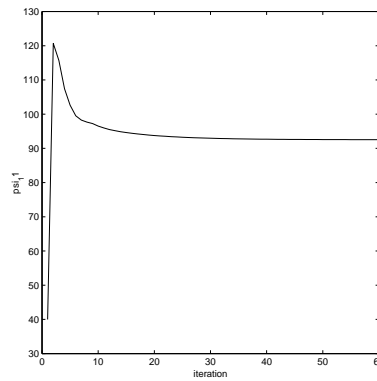


Fig. 12: Résultats de l'estimation de l'hyperparamètre  $\psi(1,1)$  pour 1000 échantillons avec  $SNR = 0 dB$

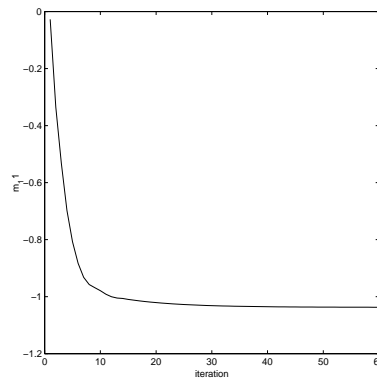


Fig. 13: Résultats de l'estimation de l'hyperparamètre  $m(1,1)$  pour 1000 échantillons avec  $SNR = 0 dB$

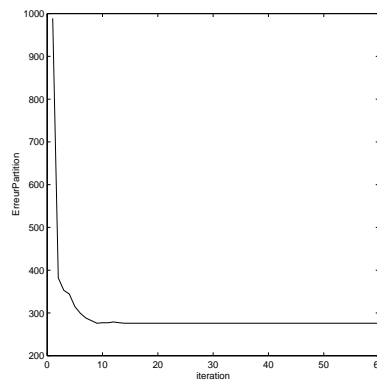


Fig. 14: Evolution des erreurs cumulées au cours des itérations pour 1000 échantillons et  $SNR = 0 dB$

## 4.5 Conclusion

Dans cette partie, nous avons proposé une méthode pour estimer conjointement les paramètres d'intérêts (les sources et la matrice de mélange) et les hyperparamètres (les variances du mélange gaussien). Tous les estimateurs sont basés sur le MAP connaissant tous les autres paramètres déjà estimés que ce soit dans l'itération précédente ou dans les étapes précédentes de la même itération. L'estimation des hyperparamètres consiste à estimer d'abord les partitions de l'ensemble  $\mathcal{T} = (1..T)$  à partir de l'estimation des variables cachées. Puis, en introduisant des *a priori* convenables (Gamma inverse pour les variances), on estime ces grandeurs en annulant les gradients des lois *a posteriori*.

## Chapitre 5

# Algorithme EM pénalisé pour la séparation de sources

Dans ce chapitre, nous présentons un algorithme EM pénalisé pour la séparation de sources. Cet algorithme itératif est souvent utilisé en estimation au sens du maximum de vraisemblance [4]. On trouve dans [5] l'implémentation de cet algorithme pour l'identification de la matrice de mélange et pour l'estimation des hyperparamètres. Nous allons reprendre la même démarche mais en introduisant un *a priori* gaussien pour les éléments de la matrice de mélange par une approche bayésienne. Dans [6], il est montré que l'estimation de la variance du mélange gaussien, par l'algorithme EM, échoue en tendant vers des points singuliers. On y trouve, dans le même papier, une solution bayésienne à ce problème de dégénérescence. Nous allons incorporer cette méthode dans notre algorithme. Nous allons aussi estimer conjointement les sources. Une classification des données est aussi estimée afin de réduire le coût de calcul lors de l'estimation de la matrice de mélange.

### 5.1 Equations de ré-estimation :

Les notations sont les mêmes que dans les parties précédentes. On note  $\theta$  l'ensemble des hyperparamètres du mélange gaussien. La première étape de l'algorithme EM (expectation) consiste à prendre le logarithme de la densité conditionnelle jointe des données observées  $\mathbf{x}(t)$  et des données cachées  $\mathbf{s}(t)$  sachant les paramètres à estimer ( $\log p(\mathbf{x}, \mathbf{s} | \mathbf{A}, \theta)$ ) et à calculer son espérance conditionnellement aux données  $\mathbf{x}(t)$  et aux paramètres courants ( $\mathbf{A}', \theta'$ ). La deuxième étape (Maximisation) consiste à maximiser la fonctionnelle obtenue par rapport à ces paramètres :

1. Etape-E :

$$Q(\mathbf{A}, \theta | \mathbf{A}', \theta') = E[\log p(\mathbf{x}, \mathbf{s} | \mathbf{A}, \theta) | \mathbf{x}, \mathbf{A}', \theta']$$

2. Etape-M :

$$(\hat{\mathbf{A}}, \hat{\theta}) = \arg \max_{(\mathbf{A}, \theta)} \{Q(\mathbf{A}, \theta | \mathbf{A}', \theta')\}$$

Les deux étapes de l'algorithme EM pénalisé que nous proposons deviennent :

1. Etape-E :

$$Q(\mathbf{A}, \theta | \mathbf{A}', \theta') = E[\log p(\mathbf{x}, \mathbf{s}, | \mathbf{A}, \theta) + \log p(\mathbf{A}) + \log p(\theta) | \mathbf{x}, \mathbf{A}', \theta']$$

2. Etape-M :

$$(\hat{\mathbf{A}}, \hat{\theta}) = \arg \max_{(\mathbf{A}, \theta)} \{Q(\mathbf{A}, \theta | \mathbf{A}', \theta')\}$$

Où nous avons inclus les *a priori* gaussien pour la matrice de mélange  $\mathbf{A}$  et gamma inverse pour les variances du mélange gaussien. La distribution jointe  $p(\mathbf{x}, \mathbf{s}, \mathbf{A}, \theta)$  se factorise de la manière suivante :

$$p(\mathbf{x}, \mathbf{s}, \mathbf{A}, \theta) = p(\mathbf{x} | \mathbf{A}, \mathbf{s}) p(\mathbf{A}) p(\mathbf{s} | \theta) p(\theta)$$

La fonctionnelle se trouve ainsi séparée en deux termes, un terme comprenant le paramètre d'intérêt qu'est la matrice de mélange et un terme comprenant les hyperparamètres :

$$Q(\mathbf{A}, \boldsymbol{\theta} | \mathbf{A}', \boldsymbol{\theta}') = Q_a(\mathbf{A} | \mathbf{A}', \boldsymbol{\theta}') + Q_h(\boldsymbol{\theta} | \mathbf{A}', \boldsymbol{\theta}')$$

Avec :

$$\begin{cases} Q_a(\mathbf{A} | \mathbf{A}', \boldsymbol{\theta}') &= E[\log p(\mathbf{x} | \mathbf{A}, \mathbf{s}) + \log p(\mathbf{A}) | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}'] \\ Q_h(\boldsymbol{\theta} | \mathbf{A}', \boldsymbol{\theta}') &= E[\log p(\mathbf{s} | \boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}'] \end{cases} \quad (5.1)$$

## Ré-estimation de $\mathbf{A}$

La fonctionnelle  $Q_a$  s'écrit :

$$Q_a = \frac{-1}{2\sigma_\epsilon^2} \sum_{t=1}^T E[(\mathbf{x}(t) - \mathbf{A} \mathbf{s}(t))^T (\mathbf{x}(t) - \mathbf{A} \mathbf{s}(t)) | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}'] + \log p(\mathbf{A}).$$

La dérivée de cette expression par rapport aux éléments de  $\mathbf{A}$  donne :

$$\frac{\partial Q_a}{\partial \mathbf{A}_{i,j}} = \frac{T}{\sigma_\epsilon^2} (\widehat{R}_{xs} - \mathbf{A} \widehat{R}_{ss})_{i,j} - \frac{1}{\sigma_{aij}^2} (\mathbf{A}_{I,J} - \mathbf{M}_{I,J}). \quad (5.2)$$

Où :

$$\begin{cases} \widehat{R}_{xs} &= \frac{1}{T} \sum_{t=1}^T E[\mathbf{x}(t) \mathbf{s}(t)^T | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}'] \\ \widehat{R}_{ss} &= \frac{1}{T} \sum_{t=1}^T E[\mathbf{s}(t) \mathbf{s}(t)^T | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}'] \end{cases} \quad (5.3)$$

Le calcul de  $\widehat{R}_{xs}$  et  $\widehat{R}_{ss}$  fait intervenir des sommes pondérées par les lois *a posteriori* de toutes les étiquettes. En effet :

$$E[f(\mathbf{s}) | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}'] = \sum_{z_1, \dots, z_n} E[f(\mathbf{s}) | \mathbf{x}, \mathbf{z} = (z_1, \dots, z_n), \mathbf{A}', \boldsymbol{\theta}'] p(\mathbf{z} | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}').$$

Ce calcul est extrêmement coteux. Il fait intervenir la somme de  $\prod_{j=1}^n q(j)$  termes correspondant toutes les combinaisons possibles des étiquettes. Donc, afin d'alléger le calcul de ces espérances, nous allons estimer d'abord les étiquettes par MAP ( $\widehat{\mathbf{z}}^{MAP}$ ). L'espérance précédente devient alors :

$$E[f(\mathbf{s}) | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}'] = E[f(\mathbf{s}) | \mathbf{x}, \mathbf{z} = \widehat{\mathbf{z}}^{MAP}, \mathbf{A}', \boldsymbol{\theta}'].$$

Sachant ces estimés, la source  $\mathbf{s}(t)$  suit *a posteriori* une loi normale de moyenne  $\boldsymbol{\theta}_{xz}$  et de variance  $\mathbf{V}_{xz}$  ( $\widehat{\mathbf{z}}^{MAP}$  est remplacé par  $\mathbf{z}$  pour alléger les expressions suivantes) :

$$\boldsymbol{\theta}_{xz} = \mathbf{m}_z + \boldsymbol{\Gamma}_z \mathbf{A}^t \mathbf{R}_z (\mathbf{x} - \mathbf{A} \mathbf{m}_z)$$

$$\mathbf{V}_{xz} = \boldsymbol{\Gamma}_z - \boldsymbol{\Gamma}_z \mathbf{A}^t \mathbf{R}_z \mathbf{A} \boldsymbol{\Gamma}_z$$

Avec

$$\begin{cases} \mathbf{R}_z &= (\mathbf{A} \boldsymbol{\Gamma}_z \mathbf{A}^t + \mathbf{R}_n)^{-1} \\ \mathbf{m}_z &= [m_{1z_1}, m_{2z_2}, \dots, m_{nz_n}] \\ \boldsymbol{\Gamma}_z &= \text{diag}(\sigma_{1z_1}^2, \sigma_{2z_2}^2, \dots, \sigma_{nz_n}^2) \end{cases} \quad (5.4)$$

$\mathbf{R}_n$  représente la matrice de covariance du bruit. On peut réécrire la variance de la loi gaussienne sous la forme :

$$\mathbf{V}_{xz} = (\mathbf{A}^t \mathbf{R}_n^{-1} \mathbf{A} + \boldsymbol{\Gamma}_z^{-1})^{-1}.$$

L'estimée de la source est alors  $\boldsymbol{\theta}_{xz}$  et les expressions de  $\widehat{R}_{xs}$  et  $\widehat{R}_{ss}$  deviennent :

$$\widehat{R}_{xs} = \frac{1}{T} \sum_{t=1}^T \mathbf{x}(t) \widehat{\mathbf{s}}(t)^T$$

et

$$\widehat{R}_{ss} = \frac{1}{T} \sum_{t=1}^T \widehat{s}(t) \widehat{s}(t)^T + \frac{1}{T} \sum_{t=1}^T (\mathbf{A}^t \mathbf{R}_n^{-1} \mathbf{A} + \mathbf{\Gamma}_z^{-1})^{-1}$$

En reprenant les opérations matricielles définies dans la première partie et en annulant le gradient (5.2), nous obtenons une expression de l'estimée de  $\mathbf{A}$ :

$$\widehat{\mathbf{A}}^{k+1} = \text{Mat} \left( \left[ \mathbf{\Lambda} + T \widehat{R}_{ss}^* \right]^{-1} \left[ \mathbf{\Lambda} \text{Vect}(M) + T \text{Vect}(\widehat{R}_{xs}) \right] \right) \quad (5.5)$$

## Ré-estimation de $\theta$

La loi *a priori* des moyennes est uniforme. L'estimée est alors celle du maximum de vraisemblance. L'équation de ré-estimation est obtenu en annulant le gradient de la fonctionnelle  $Q_h$  par rapport à  $m_{jz}$ :

$$\widehat{m}_{jz} = \frac{\sum_{t=1}^T \theta_{jz}(t) p(z(t) | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}')}{\sum_{t=1}^T p(z(t) | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}')} \quad (5.6)$$

Il est montr dans ?? que la fonction de vraisemblance pour les variances du mlange gaussien n'est pas borne. Ceci est principalement d au fait que la variance se trouve au dnominateur dans la fonction de vraisemblance. On a donc une dgnrescence de l'estimateur vers la valeur singulire 0 qui se trouve au bord de l'ensemble des variances admissibles ( $\sigma > 0$ ). Une solution bayssienne propos dans ?? consiste pnaliser la vraisemblance par un *a priori* conjugue gamma inverse  $\mathcal{G}(\alpha, \beta)$  ( $\alpha > 0$  et  $\beta > 1$ ). Les points singuliers sont ainsi limins et le critre devient born. L'équation de r-estimation est la suivante:

$$\widehat{\sigma}_{jz} = \frac{2\beta + \sum_{t=1}^T (V_{jz} + \theta_{jz}^2 - 2\widehat{m}_{jz}\theta_{jz} + \widehat{m}_{jz}^2) p(z(t) | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}')}{\sum_{t=1}^T p(z(t) | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}') + 2(\alpha - 1)} \quad (5.7)$$

Les poids du mélange gaussien sont aussi ré-estimés à chaque iération:

$$\widehat{\Pi}_{jz} = \frac{1}{T} \sum_{t=1}^T p(z_j(t) | \mathbf{x}, \mathbf{A}', \boldsymbol{\theta}') \quad (5.8)$$

## Boucle de l'algorithme

L'algorithme que nous proposons comporte les étapes suivantes:

1. Estimation des hyperparamètres selon les équations (5.6), (5.7) et (5.8) établies plus haut.
2. Remise à jour de la classification des données en estimant  $\widehat{\mathbf{z}}^{MAP}$ . Le détail de cette procédure se trouve dans la partie précédente de ce travail.
3. Sachant cette classification, les sources suivent *a posteriori* une loi gaussienne donc l'estimé par MAP se confond avec la moyenne *a posteriori*. Les expressions deviennent explicites et le calcul ais grce cette classification.
4. Estimation de la matrice de mélange selon l'équation de ré-estimation (5.5). L'estimé de la classification est celle de l'étape 2 de cette même itération.

## 5.2 Résultats de simulations

Afin de pouvoir comparer avec l'algorithme JMAP proposé dans les parties précédentes, nous allons reprendre les mêmes paramètres et la même génération de données pour un rapport signal à bruit SNR de l'ordre de 15 dB:

### Génération de données :

On a considéré deux sources, chacune est un mélange de deux gaussiennes centrées en  $\pm 1$  et  $\psi = 100$  la même pour toutes les gaussiennes. Ces sources sont mélangées avec une matrice  $\mathbf{A} = \begin{pmatrix} 1 & -0.6 \\ 0.4 & 1 \end{pmatrix}$ . Un bruit de variance  $\sigma_\epsilon^2 = 0.03$  a été ajouté. Le nombre d'échantillons est égal à 1000.

### Paramètres de l'algorithme :

$$M = \mathbf{I}, \Lambda = \begin{pmatrix} 150 & 0.01 \\ 0.01 & 150 \end{pmatrix}, \Pi = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}, \mathbf{q} = [2, 2], \alpha = 200, \beta = 2, \mu = \begin{pmatrix} -1 & 1 \\ -1 & 1 \end{pmatrix}, \gamma = \begin{pmatrix} 0.001 & 0.001 \\ 0.001 & 0.001 \end{pmatrix}.$$

### Sensibilité aux conditions initiales :

Pour montrer la sensibilité de l'algorithme EM aux conditions initiales, nous avons fixé les mêmes conditions initiales que dans l'exemple du chapitre précédent :  $\psi^{(0)} = \begin{pmatrix} 80 & 80 \\ 80 & 80 \end{pmatrix}$ ,  $m^{(0)} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$ . L'algorithme ne donne pas de résultats satisfaisants dans ces conditions. Les figures 1.a et 1.b montrent bien l'échec de l'algorithme :

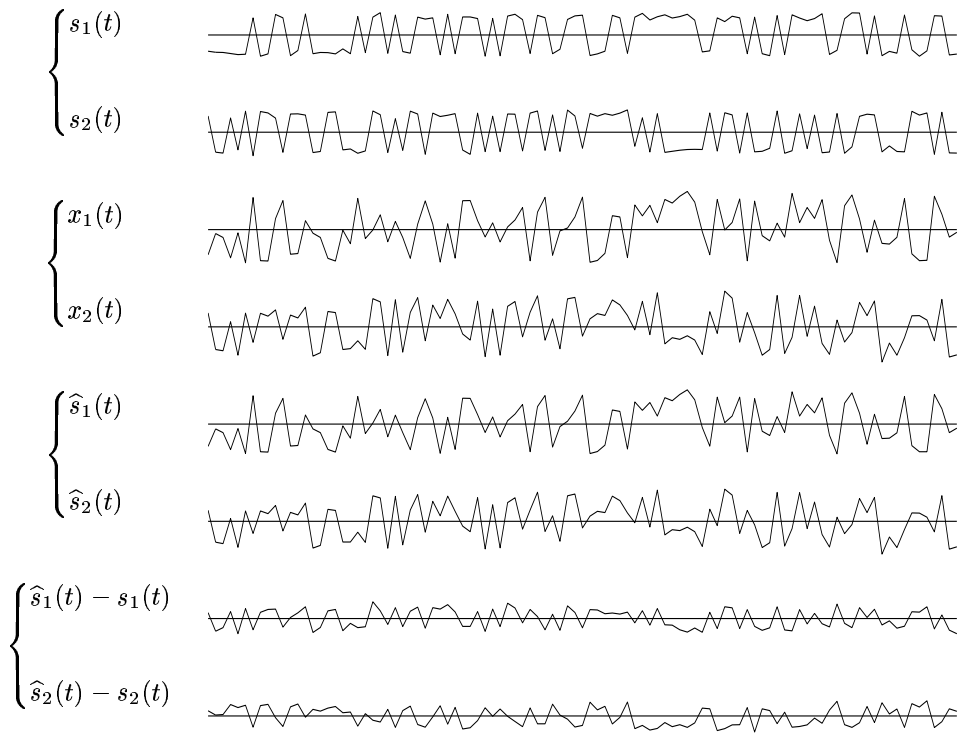


Fig. 1: Résultats de la séparation avec  $SNR = 15 \text{ dB}$  et  $m^{(0)} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$



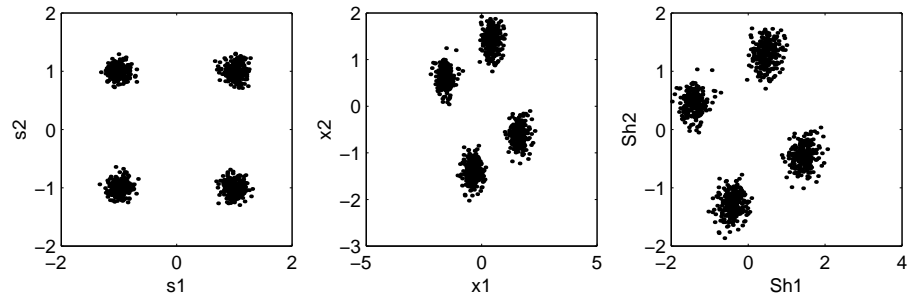


Fig. 3: Résultats de la séparation avec  $SNR = 15 \text{ dB}$  et  $m^{(0)} = \begin{pmatrix} 0.5 & -0.5 \\ 0.5 & -0.5 \end{pmatrix}$ : distribution dans l'espace des phases des sources, des signaux mélangés et des sources séparées

Nous avons ensuite modifié la condition initiale des moyennes du mélange gaussien :  $m^{(0)} = \begin{pmatrix} 0.5 & -0.5 \\ 0.5 & -0.5 \end{pmatrix}$ . Autrement dit, nous allons partir cette fois d'une bonne condition initiale. Nous notons, dans ce cas, la convergence de l'algorithme vers la bonne solution. Les figures suivantes illustrent bien ces résultats :

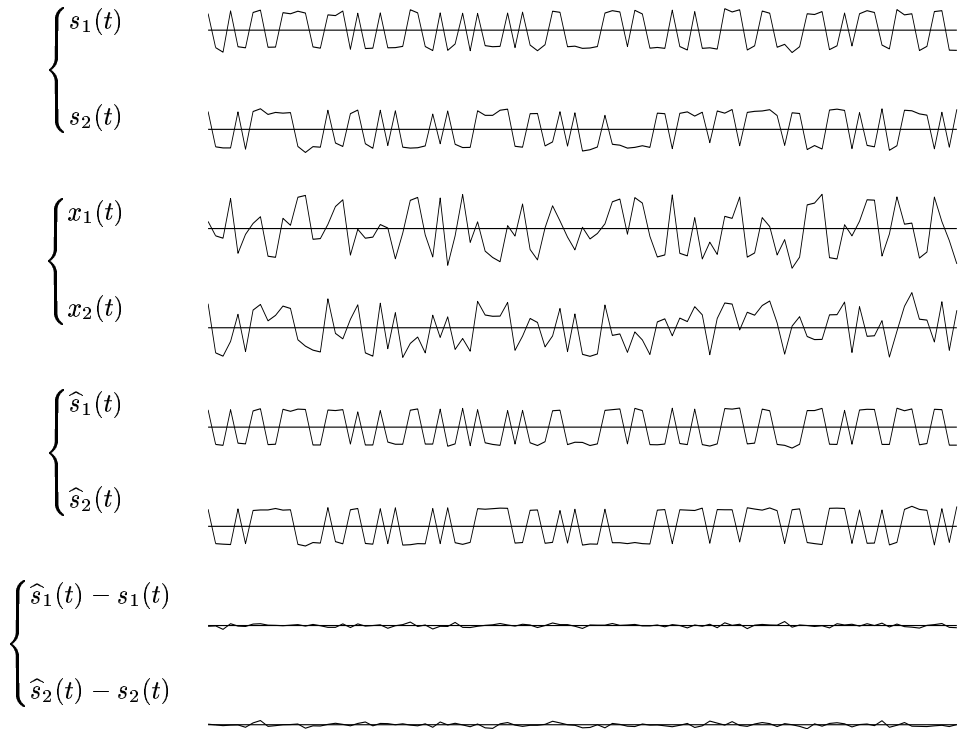


Fig. 2: Résultats de la séparation avec  $SNR = 15 \text{ dB}$  et  $m^{(0)} = \begin{pmatrix} 0.5 & -0.5 \\ 0.5 & -0.5 \end{pmatrix}$

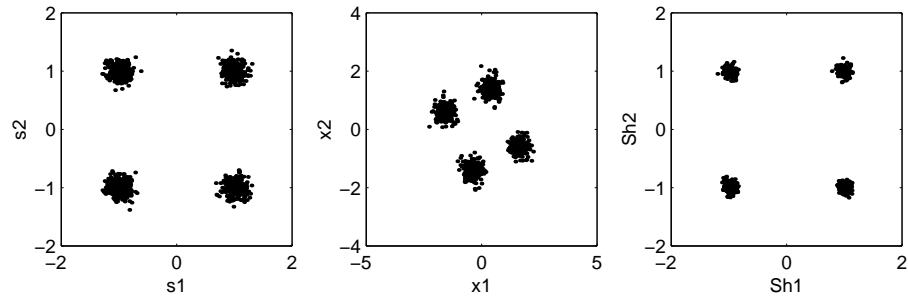


Fig. 3: Résultats de la séparation avec  $SNR = 15 \text{ dB}$  et  $m^{(0)} = \begin{pmatrix} 0.5 & -0.5 \\ 0.5 & -0.5 \end{pmatrix}$ : distribution dans l'espace des phases des sources, des signaux mélangés et des sources séparées

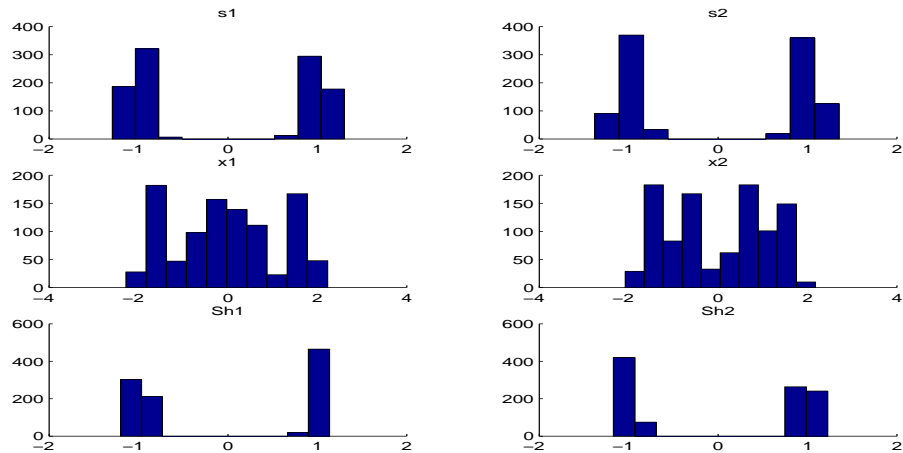


Fig. 4: Résultats de la séparation avec  $SNR = 15 \text{ dB}$  et  $m^{(0)} = \begin{pmatrix} 0.5 & -0.5 \\ 0.5 & -0.5 \end{pmatrix}$ : histogrammes des sources, signaux mélangés et des sources séparées

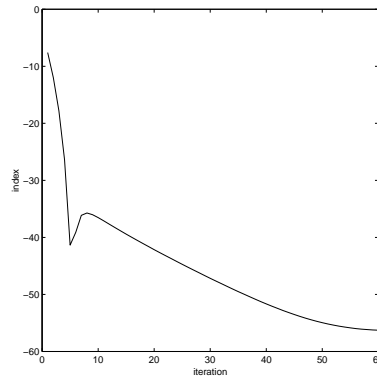


Fig. 5: Résultats de l'identification de la matrice de mélange pour 1000 échantillons avec  $SNR = 15 \text{ dB}$  et  $m^{(0)} = \begin{pmatrix} 0.5 & -0.5 \\ 0.5 & -0.5 \end{pmatrix}$

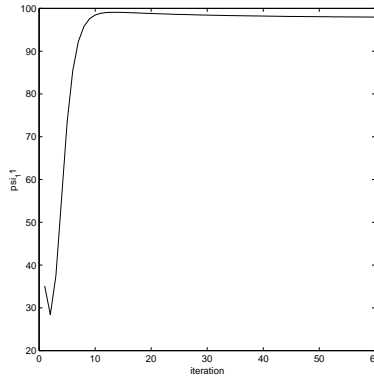


Fig. 6: Résultats de l'estimation de l'hyperparamètre  $\psi(1,1)$  pour 1000 échantillons avec  $SNR = 15 dB$  et

$$m^{(0)} = \begin{pmatrix} 0.5 & -0.5 \\ 0.5 & -0.5 \end{pmatrix}$$

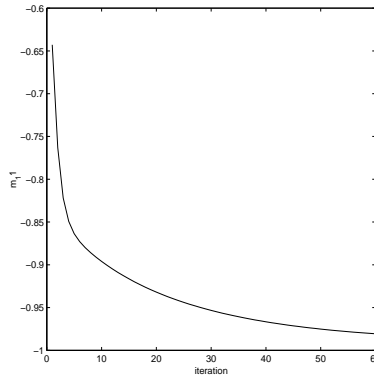


Fig. 7: Résultats de l'estimation de l'hyperparamètre  $m(1,1)$  pour 1000 échantillons avec  $SNR = 15 dB$  et

$$m^{(0)} = \begin{pmatrix} 0.5 & -0.5 \\ 0.5 & -0.5 \end{pmatrix}$$

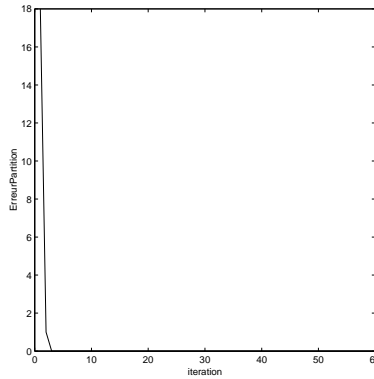


Fig. 8: Evolution des erreurs cumulées au cours des itérations pour 1000 échantillons et  $SNR = 15 dB$  et

$$m^{(0)} = \begin{pmatrix} 0.5 & -0.5 \\ 0.5 & -0.5 \end{pmatrix}$$

### 5.3 Conclusion

Nous avons proposé, dans ce chapitre, un algorithme EM pénalisé. La pénalisation consiste à pondérer la vraisemblance, lors du calcul de la fonctionnelle, par d'une part l'*a priori* gaussien pour la matrice de mélange  $\mathbf{A}$  et d'autre part l'*a priori* gamma inverse pour les variances du mélange gaussien modélisant l'*a priori* des sources. Cette modification de l'algorithme EM n'affecte pas son efficacité asymptotique [7]. Nous avons montré, lors des simulations, la sensibilité de cet algorithme aux conditions initiales, ce qui était prévisible vu la dépendance des équations de ré-estimation des valeurs estimées à l'itération précédente.

## Chapitre 6

# Conclusion

Nous avons proposé des solutions au problème de séparation de sources avec une approche bayésienne. Parmi les spécificités de cette approche, on note essentiellement:

- La prise en compte du bruit et l'incertitude sur le modèle de mélange.
- L'introduction des *a priori* sur les éléments de la matrice de mélange, sur les sources et sur les hyperparamètres.

Dans ce travail, nous avons considéré un *a priori* gaussien pour les éléments de la matrice. Le choix de cet *a priori* a été motivé par les points suivants:

- La facilité de son implantation.
- Il constitue un *a priori* conjugué pour la vraisemblance.
- La facilité de son interprétation: On connaît les éléments de la matrice de mélange avec une certaine incertitude. Nous envisageons de traiter d'autres classes d'*a priori* pour  $\mathbf{A}$  afin de couvrir un plus grand nombre d'applications.

L'*a priori* mélange de gaussiennes pour les sources constitue une classe de distributions assez large et aussi importante du fait de son application surtout en communications numériques et en traitement d'images.

L'*a priori* gamma inverse pour les variances du mélange gaussien était nécessaire pour régulariser la solution et éviter la dégénérescence de la vraisemblance vers l'infini (variance vers 0).

Nous envisageons aussi de modéliser la corrélation spatiale et temporelle des sources. Ce qui donne à l'approche bayésienne son vrai apport au problème de séparation de sources.

# Bibliographie

- [1] A. Mohammad-Djafari, “A Bayesian approach to source separation,” in *MaxEnt99 Proceedings*, Kluwer, 1999.
- [2] E. Moulines, J. Cardoso, and E. Gassiat, “Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models,” in *ICASSP-97*, (Munich, Germany), Apr. 1997.
- [3] A. Mohammad-Djafari, “Model selection for inverse problems: Best choice of basis function and model order selection,” in *MaxEnt99 Proceedings*, Kluwer, 1999.
- [4] R. A. Redner and H. F. Walker, “Mixture densities, maximum likelihood and the EM algorithm,” *SIAM Rev.*, vol. 26, pp. 195–239, Apr. 1984.
- [5] O. Bermond, *Méthodes statistiques pour la séparation de sources*. PhD thesis, Ecole Nationale Supérieure des Télécommunications, Jan. 2000.
- [6] A. Ridolfi and J. Idier, “Penalized maximum likelihood estimation for univariate normal mixture distributions,” in *17*, (Vannes, France), pp. 259–262, Sept. 1999.
- [7] R. A. Redner, “Maximum likelihood estimation for mixture models,” technical memorandum, NASA, Oct. 1980.