

# Démixage en présence de pixels rares

Ce chapitre abordera le problème du démixage des images hyperspectrales sous la forme d'un problème d'optimisation. La méthode NMF sera présentée et nous proposerons de l'adapter au cas où une partie des *endmembers* sont connus *a priori*. Cette NMF adaptée sera utilisée conjointement à la méthode de détection présentée dans le chapitre précédent pour proposer une méthode d'estimation des *endmembers* des pixels rares.

## 4.1 Le démixage hyperspectral, un problème d'optimisation

Dans le cadre du démixage d'images hyperspectrales on cherche à estimer deux matrices  $\mathbf{A} \in \mathbb{R}^{P \times k}$  et  $\mathbf{S} \in \mathbb{R}^{k \times L}$  telles que :

$$\mathbf{Y} = \mathbf{AS} + \mathbf{N} \quad (4.1)$$

$$\mathbf{S} \geq 0, \mathbf{A} \geq 0, \mathbf{A}\mathbf{1}_K = \mathbf{1}_P$$

Ce problème d'estimation peut-être réécrit sous la forme d'un problème d'optimisation dont l'objectif est de minimiser une fonction de coût.

Une fonction de coût souvent utilisée est la norme de Frobenius, ou encore l'erreur quadratique de reconstruction (EQR) entre l'observation  $\mathbf{Y}$  et sa factorisation  $\mathbf{AS}$ .

$$J(\mathbf{A}, \mathbf{S}) = EQR(\mathbf{A}, \mathbf{S}) = \|\mathbf{Y} - \mathbf{AS}\|^2 \quad (4.2)$$

Lorsque l'on se place dans l'hypothèse qui est la notre d'un bruit gaussien, minimiser l'erreur quadratique de reconstruction entre  $\mathbf{Y}$  et  $\mathbf{AS}$  est optimal [88].

Nous cherchons alors à minimiser la fonction de coût (4.2) en fonction de  $\mathbf{A}$  et  $\mathbf{S}$ , sous contrainte de positivité et de somme à l'unité. Le problème d'optimisation contraint devient donc :

$$\begin{aligned} & \text{minimiser}_{\mathbf{A}, \mathbf{S}} J(\mathbf{A}, \mathbf{S}) & (4.3) \\ & \text{sous les contraintes } \mathbf{S} \geq 0, \mathbf{A} \geq 0, \mathbf{A}\mathbf{1}_K = \mathbf{1}_P \end{aligned}$$

Cependant la minimisation de  $J(\mathbf{A}, \mathbf{S})$  sous les seules contraintes de non-négativité des matrices  $\mathbf{A}$ , et  $\mathbf{S}$  et de somme à l'unité n'a pas de solution unique [44]. En effet, il existe des solutions dégénérées à ce problème. Par exemple si le couple  $\mathbf{A} \in \mathbb{R}_+^{P \times k}$  et  $\mathbf{S} \in \mathbb{R}_+^{k \times L}$  est solution de l'équation (4.3), alors toute matrice carrée inversible  $\mathbf{P} \in \mathbb{R}_+^{k \times k}$  telle que  $\mathbf{A}\mathbf{P}$  respecte la contrainte de somme à l'unité, permet d'obtenir une infinité de couples solution tels que :

$$\mathbf{Y} = \mathbf{A}\mathbf{S} + \mathbf{N} = \underbrace{\mathbf{A}\mathbf{P}}_{\mathbf{A}'} \underbrace{\mathbf{P}^{-1}\mathbf{S}}_{\mathbf{S}'} + \mathbf{N} \quad (4.4)$$

### 4.1.1 Contrainte de somme à l'unité relaxée

D'un point de vue théorique la somme à l'unité est une contrainte élégante et naturelle, puisqu'elle revient à dire que les abondances correspondent aux proportions de chaque matériau présent dans le pixel, et qu'en conséquent la somme de ces proportions doit être égale à 1.

Cependant, en pratique un matériau avec une réflexivité qui varie en fonction de l'angle d'incidence aura un unique *endmember* dont l'intensité dépendra de son orientation. De même, la mesure du spectre d'un matériau situé dans une zone ombragée différera du spectre mesuré pour le même matériau illuminé directement. Ce phénomène appelé variabilité d'*endmember* [28, 89] fait que l'hypothèse de somme à l'unité n'est pas toujours exacte. De ce fait on préférera appliquer cette contrainte de manière moins stricte en favorisant plutôt les solutions dont le résultat est proche de cette contrainte à l'aide d'un terme de régularisation :

$$J_2(\mathbf{A}, \mathbf{S}) = \|\mathbf{Y} - \mathbf{A}\mathbf{S}\|^2 + \alpha^2 \|\mathbf{A}\mathbf{1}_K - \mathbf{1}_P\|^2 \quad (4.5)$$

Le paramètre  $\alpha$  permet d'imposer de façon plus ou moins stricte la contrainte. Une valeur de  $\alpha$  élevée indique une contrainte imposée strictement alors qu'une valeur nulle indique que la contrainte n'est pas imposée. L'équation (4.5) se dé-

veloppe alors comme suit :

$$\begin{aligned}
 J_2(\mathbf{A}, \mathbf{S}) &= \|\mathbf{Y} - \mathbf{AS}\|^2 + \alpha^2 \|\mathbf{A}\mathbf{1}_K - \mathbf{1}_P\|^2 \\
 &= \text{tr} \left( \mathbf{Y}^T \mathbf{Y} - 2\mathbf{S}^T \mathbf{A}^T \mathbf{Y} + \mathbf{S}^T \mathbf{A}^T \mathbf{AS} \right) + \alpha^2 \text{tr} \left( \mathbf{1}_P^T \mathbf{1}_P - 2\mathbf{1}_K^T \mathbf{A}^T \mathbf{1}_P + \mathbf{1}_K^T \mathbf{A}^T \mathbf{A} \mathbf{1}_K \right) \\
 &= \text{tr} \left( (\mathbf{Y}^T \mathbf{Y} + \mathbf{1}_P^T \mathbf{1}_P) - 2(\mathbf{YS}^T + \alpha^2 \mathbf{1}_P \mathbf{1}_K^T) \mathbf{A}^T + (\mathbf{SS}^T + \alpha^2 \mathbf{1}_K \mathbf{1}_K^T) \mathbf{A}^T \mathbf{A} \right) \\
 &= \text{tr} \left( [\mathbf{Y} | \alpha \mathbf{1}_P]^T [\mathbf{Y} | \alpha \mathbf{1}_P] - 2[\mathbf{Y} | \alpha \mathbf{1}_P]^T [\mathbf{S} | \alpha \mathbf{1}_K] \mathbf{A}^T + [\mathbf{S} | \alpha \mathbf{1}_K]^T \mathbf{A}^T \mathbf{A} [\mathbf{S} | \alpha \mathbf{1}_K] \right) \\
 &= \text{tr} \left( \mathbf{Y}_c^T \mathbf{Y}_c - 2\mathbf{S}_c^T \mathbf{A}^T \mathbf{Y}_c + \mathbf{S}_c^T \mathbf{A}^T \mathbf{AS}_c \right) \\
 &= \|\mathbf{Y}_c - \mathbf{AS}_c\|^2
 \end{aligned}$$

où  $\text{tr}(\cdot)$  est l'opérateur trace et :

$$\mathbf{Y}_c = [\mathbf{Y} | \alpha \mathbf{1}_P] \quad \text{et} \quad \mathbf{S}_c = [\mathbf{S} | \alpha \mathbf{1}_K] \quad (4.6)$$

sont les matrices  $\mathbf{Y}$  et  $\mathbf{S}$  auxquelles ont été ajoutés une colonne de valeur constante  $\alpha$ .

On remarque que la fonction de coût  $J_2$  donnée par l'équation (4.5) est identique à la fonction de coût  $J$  de l'équation (4.2) dans laquelle on aurait remplacé les matrices  $\mathbf{Y}$  et  $\mathbf{S}$  par  $\mathbf{Y}_c$  et  $\mathbf{S}_c$ . En conséquence, les méthodes qui permettent de résoudre le problème d'optimisation donné en (4.3) sans contrainte de somme à l'unité, permettent également de résoudre le problème avec une contrainte de somme à l'unité relaxée.

### 4.1.2 Démixage par la NMF

La factorisation en matrices non-négatives (NMF) est une méthode de démixage qui consiste à factoriser une matrice non-négative  $\mathbf{Y} \in \mathbb{R}_+^{P \times L}$  en deux matrices facteurs  $\mathbf{A} \in \mathbb{R}^{P \times k}$  et  $\mathbf{S} \in \mathbb{R}^{k \times L}$  tels que  $\mathbf{AS} \approx \mathbf{Y}$  pour un rang de factorisation  $k$  donné, sous contrainte que tous les coefficients de  $\mathbf{A}$  et de  $\mathbf{S}$  soient positifs ou nuls.

La NMF vise à résoudre le problème d'optimisation suivant :

$$\begin{aligned}
 &\text{minimiser}_{\mathbf{A}, \mathbf{S}} \quad \|\mathbf{Y} - \mathbf{AS}\|^2 & (4.7) \\
 &\text{sous les contraintes} \quad \mathbf{S} \geq 0, \mathbf{A} \geq 0
 \end{aligned}$$

Ce problème d'optimisation est identique à celui du démixage évoqué dans l'équation (4.3) lorsque la contrainte de somme à l'unité n'est pas appliquée strictement.

Le problème ainsi posé est non-convexe, ce qui rend la recherche d'un minimum global difficile. Cependant si l'une des deux matrices facteur  $\mathbf{A}$  ou  $\mathbf{S}$  est connue, alors le problème devient convexe. Une méthode de résolution consiste alors à séparer le problème en deux sous-problèmes convexes. L'un consistant à estimer  $\mathbf{A}$  en supposant  $\mathbf{S}$  fixée et l'autre à estimer  $\mathbf{S}$  en supposant  $\mathbf{A}$  fixée. Ces deux sous problèmes s'expriment sous la forme :

$$\min_{\mathbf{S} \geq 0} \|\mathbf{Y} - \mathbf{AS}\|^2 \quad (4.8)$$

$$\min_{\mathbf{A} \geq 0} \|\mathbf{Y} - \mathbf{AS}\|^2 \quad (4.9)$$

Il s'agit là de deux problèmes des moindres carrés. De nombreux algorithmes ont été proposés pour les résoudre. Leur adaptation à la NMF est basée sur la méthode des moindres carrés alternés (ALS) qui consiste à résoudre itérativement chacun des deux sous-problèmes. Elle est donnée dans l'algorithme 1.

---

**Algorithme 1** NMF - ALS

---

```

/* Initialisation */
l ← 1
 $\mathbf{A}^{(1)} \geq 0$ 
 $\mathbf{S}^{(1)} \geq 0$ 
/* Jusqu'à convergence */
tant que  $EQR(\mathbf{A}^{(l)}, \mathbf{S}^{(l)})$  n'a pas convergé répéter
     $\mathbf{A}^{(l+1)} \leftarrow \min_{\mathbf{A} \geq 0} EQR(\mathbf{A}, \mathbf{S}^{(l)})$ 
     $\mathbf{S}^{(l+1)} \leftarrow \min_{\mathbf{S} \geq 0} EQR(\mathbf{A}^{(l+1)}, \mathbf{S})$ 
    l ← l + 1
fin tant que
Sortie :  $\mathbf{A}^{(l)}, \mathbf{S}^{(l)}$ 

```

---

#### 4.1.2.1 Problème des moindres carrés

Le problème des moindres carrés consiste à résoudre le problème d'optimisation suivant :

$$\text{minimiser}_{\mathbf{X}} \|\mathbf{Y} - \mathbf{AX}\|^2 \quad (4.10)$$

Où  $\mathbf{X} \in \mathbb{R}^{k \times L}$ . La méthode NMF nécessite cependant que les matrices soient positives. Une façon de garantir ce critère est de projeter dans  $\mathbb{R}_+^{k \times L}$  le résultat

obtenu par la méthode des moindres carrés en annulant les coefficients négatifs avec l'opérateur  $[\cdot]_+$ .

Plusieurs méthodes utilisent cette approche :

**Algorithme multiplicatif :**

Lee et Seung furent parmi les premiers à populariser la méthode NMF [43]. Ils proposent un algorithme de NMF basé sur un schéma d'algorithme des moindres carrés alternés qui utilise des règles multiplicatives pour mettre à jour les matrices  $\mathbf{A}$  et  $\mathbf{S}$  à chaque itération [63].

$$\mathbf{A} \leftarrow [\mathbf{Y}\mathbf{S}^T(\mathbf{S}\mathbf{S}^T)^{-1}]_+ \quad (4.11)$$

$$\mathbf{S} \leftarrow [(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{Y}]_+ \quad (4.12)$$

La démonstration permettant d'aboutir aux équations (4.11) et (4.12) est détaillée dans [90]. La convergence de cet algorithme n'est cependant pas garantie [91, 92].

**Algorithme du gradient projeté :**

L'algorithme du gradient projeté, PG (pour Projected Gradient) [62], utilise un processus de mise à jour additif des matrices  $\mathbf{A}$  et  $\mathbf{S}$ . A partir de la fonction de régularisation  $\text{EQR}(\mathbf{A}, \mathbf{S}) = \|\mathbf{Y} - \mathbf{A}\mathbf{S}\|^2$ , chaque itération de l'algorithme du gradient projeté peut être résumée ainsi :

$$\mathbf{A} \leftarrow \left[ \mathbf{A} - \mu_{\mathbf{A}} \frac{\partial}{\partial \mathbf{A}} \text{EQR}(\mathbf{A}, \mathbf{S}) \right]_+ \quad (4.13)$$

$$\mathbf{S} \leftarrow \left[ \mathbf{S} - \mu_{\mathbf{S}} \frac{\partial}{\partial \mathbf{S}} \text{EQR}(\mathbf{A}, \mathbf{S}) \right]_+ \quad (4.14)$$

La démonstration permettant d'aboutir aux équations (4.13) et (4.14) est présentée dans [90].  $\mu_{\mathbf{A}}$  et  $\mu_{\mathbf{S}}$  sont les *pas* du gradient. Ces pas peuvent être fixes ou variables en fonction des algorithmes dérivant du gradient projeté et auront une influence sur la précision ou la rapidité de convergence de l'algorithme. Enfin les gradients de la fonction de régularisation  $\text{EQR}(\mathbf{A}, \mathbf{S})$  par rapport à  $\mathbf{A}$  et  $\mathbf{S}$  sont donnés dans [90] :

$$\frac{\partial}{\partial \mathbf{A}} \text{EQR}(\mathbf{A}, \mathbf{S}) = 2(\mathbf{A}\mathbf{S} - \mathbf{Y})\mathbf{S}^T \quad (4.15)$$

$$\frac{\partial}{\partial \mathbf{S}} \text{EQR}(\mathbf{A}, \mathbf{S}) = 2\mathbf{A}^T(\mathbf{AS} - \mathbf{Y}) \quad (4.16)$$

D'autres méthodes de résolution du problème des moindres carrés ont également été proposées, en utilisant par exemple une approche quasi-newtonienne [93, 94]. Pour garantir les contraintes de positivité de la NMF, certaines méthodes utilisent le problème de moindres carrés contraints.

#### 4.1.2.2 Problème des moindres carrés contraint

Le problème des moindres carrés contraint vise à minimiser la même fonction de coût que la version non-contrainte (4.10), en ajoutant une contrainte de positivité :

$$\begin{aligned} & \text{minimiser}_{\mathbf{X}} \|\mathbf{Y} - \mathbf{AX}\|^2 & (4.17) \\ & \text{sous la contrainte } \mathbf{X} \geq 0 \end{aligned}$$

L'utilisation d'une méthode de résolution des moindres carrés contraint pour résoudre les équations (4.8) et (4.9) permet d'obtenir à chaque itération la solution optimale.

Plusieurs méthodes ont été développées pour résoudre ce problème [95, 96, 97]. Dans cette thèse nous avons principalement utilisé la méthode BPP (Block Principal Pivoting) [98].

**Remarque :** L'équation (4.17) peut être décomposée colonne par colonne :

$$\min_{\mathbf{X} \geq 0} \|\mathbf{Y} - \mathbf{AX}\|_F^2 = \sum_j \min_{\mathbf{x}_{\cdot j} \geq 0} \|\mathbf{y}_{\cdot j} - \mathbf{Ax}_{\cdot j}\|_F^2 \quad (4.18)$$

où  $\mathbf{y}_{\cdot j}$  et  $\mathbf{x}_{\cdot j}$  sont respectivement la  $j^{\text{e}}$  colonne de  $\mathbf{Y}$  et  $\mathbf{X}$ . Le problème des moindres carrés contraint revient donc à résoudre l'équation suivante :

$$\begin{aligned} & \text{minimiser}_{\mathbf{x}} \|\mathbf{y} - \mathbf{Ax}\|^2 & (4.19) \\ & \text{sous la contrainte } \mathbf{x} \geq 0 \end{aligned}$$

#### Algorithme Bloc Principal Pivoting :

Résoudre le problème de moindre carrés contraint vecteur par vecteur n'est pas efficace d'un point de vu calculatoire, la résolution directe sous forme matricielle (4.17) est plus efficace. Cependant nous nous contenterons de présenter la méthode BPP pour résoudre le cas restreint de l'équation (4.19). Le lecteur

intéressé par les améliorations que permet la résolution sous forme matricielle peut se référer à [64].

Les conditions d'optimalités de Karush-Kuhn-Tucker (KKT) pour l'équation (4.19) sont les suivantes :

$$\mathbf{z} = \mathbf{A}^T \mathbf{A} \mathbf{x} - \mathbf{A}^T \mathbf{y} \quad (4.20)$$

$$\mathbf{z} \geq 0 \quad (4.21)$$

$$\mathbf{x} \geq 0 \quad (4.22)$$

$$x_i z_i = 0 \quad i = 1, \dots, k \quad (4.23)$$

Les conditions KKT sont des conditions nécessaires d'optimalité. Elles sont suffisantes lorsque la matrice  $\mathbf{A}$  est de rang plein, ce qui est le cas pour la NMF. Toute solution qui satisfait les conditions KKT est donc une solution optimale de l'équation (4.19).

La méthode BBP s'inspire des méthodes dites Active-Set [94], les variables sont divisées en deux groupes  $F$  et  $G$  complémentaires avec  $F \cup G = \{1, \dots, k\}$  et  $F \cap G = \emptyset$ . On note  $\mathbf{x}_F, \mathbf{z}_F$  et  $\mathbf{x}_G, \mathbf{z}_G$  les sous vecteurs correspondant aux indices dans  $F$  et  $G$  respectivement (c'est à dire,  $\mathbf{x}_F = \{x_i | i \in F\}$ ). De même on note  $\mathbf{A}_F$  et  $\mathbf{A}_G$  les sous matrices de  $\mathbf{A}$  dont seules les colonnes d'indice dans  $F$  (respectivement dans  $G$ ) ont été conservées.

Les vecteur  $\mathbf{x}_F$  et  $\mathbf{z}_G$  sont initialisés par un vecteur nul et seront mis à jour successivement par les formules :

$$\mathbf{x}_F = \min_{\mathbf{x}_F} \|\mathbf{A}_F \mathbf{x}_F - \mathbf{y}\|^2 \quad (4.24)$$

$$\mathbf{z}_G = \mathbf{A}_G (\mathbf{A}_F \mathbf{x}_F - \mathbf{y}) \quad (4.25)$$

Si  $\mathbf{x}_F \geq 0$  et  $\mathbf{z}_G \geq 0$  alors le vecteur  $\mathbf{x} = (\mathbf{x}_F, 0)$  est une solution optimale de l'équation (4.19). Dans le cas contraire, les groupes  $F$  et  $G$  vont être mis à jour pour échanger les variables qui ne respectent pas les conditions KKT (4.21) et (4.22). Les règles de mises à jour sont les suivantes :

$$F = (F - H_1) \cup H_2 \quad (4.26)$$

$$G = (G - H_2) \cup H_1 \quad (4.27)$$

$$\text{avec } H_1 = \{i \in F | x_i < 0\} \text{ et } H_2 = \{i \in G | z_i < 0\}$$

Les vecteur  $\mathbf{x}_F$  et  $\mathbf{z}_G$  sont alors mis à jour avec les nouveaux groupes et le

processus est répété jusqu'à ce que le critère d'arrêt  $\mathbf{x}_F \geq 0$  et  $\mathbf{z}_G \geq 0$  soit rempli.

#### 4.1.2.3 Contraintes supplémentaires pour la NMF

Afin de limiter le nombre de solutions de la NMF pour les images hyperspectrales, plusieurs autres termes de régularisation associés à  $EQR(\mathbf{A}, \mathbf{S})$  ont été proposés dans la littérature [44, 99, 29, 100] sous la forme d'une nouvelle fonction de coût :

$$C(\mathbf{A}, \mathbf{S}) = EQR(\mathbf{A}, \mathbf{S}) + \lambda_1 R_1(\mathbf{A}) + \lambda_2 R_2(\mathbf{S}) + \lambda_3(\mathbf{A}, \mathbf{S}) \dots \quad (4.28)$$

où les  $\lambda_i R_i$  définissent des termes de régularisation pondérés. La fonction de coût que la NMF doit minimiser devient alors la fonction  $C(\mathbf{A}, \mathbf{S})$ , et non plus seulement  $EQR(\mathbf{A}, \mathbf{S})$ .

Le choix des termes de régularisation requiert une connaissance *a priori* sur la forme des solutions. Par exemple Kim et Park [101] proposent un terme de régularisation pour favoriser la "sparsité" (rareté) des lignes de  $\mathbf{A}$  (c'est à dire favoriser l'apparition de valeur nulle) :

$$R(\mathbf{A}) = \sum_{i=1}^I \|\mathbf{a}_i\|_1^2 \quad (4.29)$$

Où  $\mathbf{a}_i$  est le  $i^e$  vecteur ligne de  $\mathbf{A}$  et  $\|\cdot\|_1$  est la norme  $L_1$ . Ou encore une contrainte qui encourage les solutions dont la somme des coefficients des lignes de  $\mathbf{A}$  est proche de 1 [102] :

$$R(\mathbf{A}) = \|\mathbf{A}\mathbf{1}_K - \mathbf{1}_I\|^2 \quad (4.30)$$

Où  $\mathbf{1}_I = [1, \dots, 1]^T \in \mathbb{R}^I$  est un vecteur de taille I contenant uniquement des 1.

L'ajout de termes de régularisation à la NMF permet de rendre compte d'une certaine réalité physique, en effet certains matériaux ne sont présents que dans certaines portions de la scène, une contrainte de sparsité permet de rendre compte de la dispersion des *endmembers*.

**Remarque :** Si l'ajout de termes de régularisation permet de réduire le risque d'obtenir des solutions dégénérées, les estimations des matrices  $\mathbf{A}$  et  $\mathbf{S}$  sont données à une permutation près. En effet pour une matrice de permutation  $\mathbf{P}$  on a :  $\mathbf{AS} = \mathbf{AP}^T\mathbf{PS}$ . En pratique, les matrices  $\mathbf{A}$  et  $\mathbf{S}$  sont inconnues, ce n'est pas le cas lorsque l'on travaille avec des données simulées, retrouver cette permutation est alors nécessaire pour pouvoir comparer les matrices  $\mathbf{A}$  et  $\mathbf{S}$  à leurs estimées. Dans cette situation la matrice de permutation  $\mathbf{P}$  peut être retrouvée

en cherchant la meilleure correspondance au sens de la distance angulaire entre les spectres de  $\mathbf{S}$  et ceux de la matrice estimée  $\hat{\mathbf{S}}$ . Cette opération consiste à résoudre un problème d'affectation (voir annexe B) qui ne sera pas détaillé ici. Dans la suite de cette thèse, nous considérerons que les matrices estimées ont été corrigées pour annuler cette permutation lorsque c'était nécessaire sans en faire mention.

## 4.2 Démixage en présence de pixels rares

La méthode NMF permet le démixage des *endmembers* dominants mais échoue à démixer les pixels rares [44]. Les critères de détection présentés dans le chapitre précédent permettent de séparer les pixels rares des autres pixels. Nous pouvons tirer profit de cette séparation pour améliorer les performances du démixage en procédant par deux étapes différentes pour les dominants et les rares. En effet, en utilisant la méthode par seuillage proposé en 3.2, nous disposons d'une part des pixels dominants qui sont correctement démixés, et d'autre part des pixels rares dont le démixage est incorrect.

La méthode que nous proposons procède donc en plusieurs étapes :

- Premièrement, la méthode NMF est utilisée pour estimer les *endmembers* dominants  $\hat{\mathbf{S}}_d$  et leurs abondances  $\hat{\mathbf{A}}_d$ . Nous obtenons ainsi une estimation de la matrice  $\mathbf{Y}_d = \hat{\mathbf{A}}_d \hat{\mathbf{S}}_d$ .
- À partir de  $\mathbf{Y}_d$  et de l'image d'origine  $\mathbf{Y}$ , nous calculons l'erreur de reconstruction pixel par pixel. Avec le critère de seuil défini par l'équation (3.18) il est alors possible de séparer les pixels en deux groupes : le groupe  $P_d$  des pixels dominants qui ont été correctement démixés par  $\hat{\mathbf{A}}_d$  et  $\hat{\mathbf{S}}_d$ , et le groupe des pixels rares  $P_r$  qui restent à démixer.
- La dernière étape consiste à démixer séparément les pixels rares  $P_r$ . Cette séparation s'apparente à une sélection de pixels d'intérêt. La majorité des pixels dominants ayant été retirés, au sein du groupe  $P_r$  la majorité des pixels sont des pixels rares, ils ne sont plus sous-représentés, ce qui permet de les démixer.

Une question subsiste : comment démixer les pixels du groupe  $P_r$  ? Ces pixels contiennent des *endmembers* rares, mais rien ne permet de garantir qu'ils sont composés uniquement d'*endmembers* rares. En effet, un pixel rare peut-être un mélange d'*endmembers* rares et dominants. Nous proposons donc dans la partie ci-dessous une méthode NMF modifiée pour démixer les pixels rares lorsque l'on dispose a priori d'une partie des *endmembers*.

### 4.2.1 NMF avec des endmembers connus a priori

Dans cette partie nous supposons que l'on dispose d'un ensemble de pixels  $\mathbf{Y} \in \mathbb{R}^{P \times L}$  qui respectent le modèle de mélange linéaire :

$$\mathbf{Y} = \mathbf{A}\mathbf{S} + \mathbf{N} \quad (4.31)$$

Avec  $\mathbf{A} \in \mathbb{R}^{P \times k}$  la matrice de mélange et  $\mathbf{S} \in \mathbb{R}^{k \times L}$  la matrice des *endmembers*.

Nous supposons aussi que l'on connaît  $\mathbf{S}_d \in \mathbb{R}^{k_d \times L}$ ,  $k_d < k$ , une partie des endmembers (les dominants). Le modèle de mélange linéaire se réécrit alors :

$$\mathbf{Y} = [\mathbf{A}_d | \mathbf{A}_r] \begin{bmatrix} \mathbf{S}_d \\ \mathbf{S}_r \end{bmatrix} + \mathbf{N} = \mathbf{A}_d \mathbf{S}_d + \mathbf{A}_r \mathbf{S}_r + \mathbf{N} \quad (4.32)$$

Avec  $\mathbf{S}_r \in \mathbb{R}^{k_r \times L}$ ,  $k_r = k - k_d$ , la matrices des *endmembers* non connus.

La méthode proposée s'inspire de la méthode ALS présentée dans l'algorithme 1. Dans un premier temps on suppose  $\mathbf{S}$  connue et l'on cherche à estimer  $\mathbf{A}$  en minimisant l'erreur quadratique de reconstruction :

$$\min_{\mathbf{A} > 0} \|\mathbf{Y} - \mathbf{A}\mathbf{S}\|^2 \quad (4.33)$$

Ensuite on calcule le résidu entre les pixels observés  $\mathbf{Y}$  et la contribution due aux *endmembers* connus  $\mathbf{A}_d \mathbf{S}_d$  en forçant la positivité du résidu.

$$\mathbf{Y}_{res} = \max(\mathbf{Y} - \mathbf{A}_d \mathbf{S}_d, 0) \quad (4.34)$$

On estime alors la contribution en minimisant l'erreur quadratique entre le résidu, et la contribution due aux *endmembers* à estimer.

$$\min_{\mathbf{S}_d > 0} \|\mathbf{Y}_{res} - \mathbf{A}_r \mathbf{S}_r\|^2 \quad (4.35)$$

On itère ensuite ces trois étapes jusqu'à convergence. La méthode est résumée par l'algorithme 2.

### 4.2.2 Résultats expérimentaux

Nous générons 10 000 pixels, composés de 8 *endmembers* sur 581 bandes spectrales. Parmi ces *endmembers* 2 sont considérés comme rares et sont présents dans 1% et 0.5% des pixels générés. Un bruit blanc gaussien est ajouté aux pixels pour obtenir un SNR de 25db. Les *endmembers* sélectionnés sont présentés dans

---

**Algorithme 2** NMF - a priori
 

---

```

/* Initialisation */
l ← 1
 $\mathbf{A}^{(1)} = [\mathbf{A}_d^{(1)} | \mathbf{A}_r^{(1)}]$ 
 $\mathbf{S}^{(1)} = \begin{bmatrix} \mathbf{S}_d^{(1)} \\ \mathbf{S}_r^{(1)} \end{bmatrix}$ 
/* Jusqu'à convergence */
tant que  $EQR(\mathbf{A}^{(l)}, \mathbf{S}^{(l)})$  n'a pas convergé répéter
     $\mathbf{A}^{(l+1)} \leftarrow \min_{\mathbf{A} \geq 0} \|\mathbf{Y} - \mathbf{A}\mathbf{S}^{(l)}\|^2$ 
     $\mathbf{Y}_{res} = \max(\mathbf{Y} - \mathbf{A}_d^{(l+1)}\mathbf{S}_d^{(l)}, 0)$ 
     $\mathbf{S}_r^{(l+1)} \leftarrow \min_{\mathbf{S}_r \geq 0} \|\mathbf{Y}_{res} - \mathbf{A}_r^{(l)}\mathbf{S}_r\|^2$ 
     $\mathbf{S}^{(l+1)} = \begin{bmatrix} \mathbf{S}_d^{(l+1)} \\ \mathbf{S}_r^{(l+1)} \end{bmatrix}$ 
    l ← l + 1
fin tant que
Sortie :  $\mathbf{A}^{(l)}, \mathbf{S}^{(l)}$ 
    
```

---

la figure 4.1.

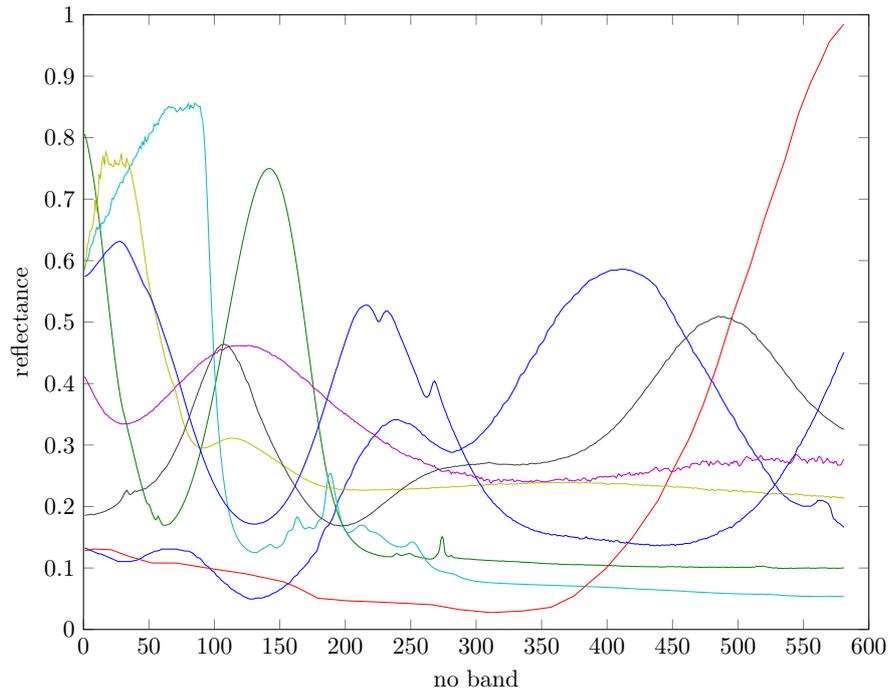


FIGURE 4.1 – Endmembers utilisés pour générer les pixels observés

Nous comparons les *endmembers* estimés par la méthode NMF à ceux obtenus par la méthode proposée. La figure 4.2 présente les *endmembers* estimés par la méthode proposée. On constate que les *endmembers* rares sont plus bruités que les dominants. Leur estimation étant effectuée à partir d'un faible nombre de pixels (les pixels rares), elle est plus sensible au bruit.

Le SAD (Spectral Angle Distance) permet de comparer deux spectres en me-

## 4.2 DÉMIXAGE EN PRÉSENCE DE PIXELS RARES

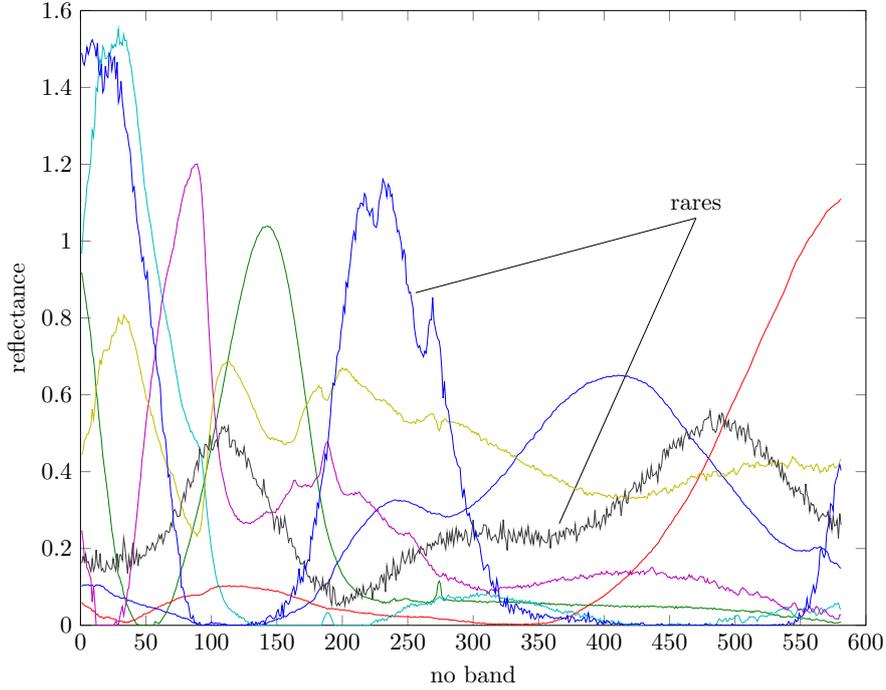


FIGURE 4.2 – Endmembers estimés par la méthode proposée

surant leur distance angulaire. Le MSAD (Mean SAD) permet de comparer les *endmembers* estimés aux valeurs attendues en calculant la moyenne des SAD pour chaque spectre. Ils sont définis comme suit :

$$SAD(\mathbf{s}, \hat{\mathbf{s}}) = \cos^{-1}\left(\frac{\mathbf{s}^T \hat{\mathbf{s}}}{\|\mathbf{s}\| \|\hat{\mathbf{s}}\|}\right) \quad (4.36)$$

$$MSAD(\mathbf{S}, \hat{\mathbf{S}}) = \frac{1}{N} \sum_{k=1}^N \cos^{-1}\left(\frac{\mathbf{s}_k^T \hat{\mathbf{s}}_k}{\|\mathbf{s}_k\| \|\hat{\mathbf{s}}_k\|}\right) \quad (4.37)$$

Le tableau 4.1 présente les résultats moyens obtenus après 1000 répétitions. La figure 4.2 permet d'observer les *endmembers* obtenus par la méthode proposée. Les matrices d'abondance, des *endmembers* et la matrice reconstruite à partir des estimations de la méthode proposée sont plus proches des résultats attendus que pour la NMF.

	MSAD	$\ \mathbf{S} - \hat{\mathbf{S}}\ _F$	$\ \mathbf{A} - \hat{\mathbf{A}}\ _F$	$\ \mathbf{Y} - \hat{\mathbf{A}}\hat{\mathbf{S}}\ _F$
NMF	$0.425 \pm 0.015$	$18.38 \pm 1.42$	$30.43 \pm 1.43$	$30.52 \pm 6.37$
Méthode proposé	$0.335 \pm 0.01$	$13.77 \pm 0.73$	$27.40 \pm 0.67$	$9.45 \pm 0.83$

TABLE 4.1 – Comparaison entre la NMF et la méthode proposée

Si les résultats expérimentaux montrent de meilleures performances pour la

méthode proposée que pour la NMF simple, ils mettent aussi en évidence les lacunes de la méthode proposée. En effet, l'estimation des *endmembers* rares reposant sur la détection de pixels rares la méthode est très sensible au résultat de cette détection. Lorsque qu'un très faible nombre de pixels sont détectés comme rares, il arrive que la méthode échoue à estimer les *endmembers* rares à cause de la trop faible quantité de pixels. De même, quand le nombre de faux positifs est trop élevé, les *endmembers* estimés s'éloignent des résultats attendus.

### 4.3 Conclusion

Dans ce chapitre, nous avons présenté la problématique du démixage en imagerie hyperspectrale lorsque la scène contient des objets de faibles dimensions.

Partant du principe que nous disposons d'une méthode permettant de connaître ou estimer d'une part les pixels rares qui contiennent les *endmembers* rares, et d'autre part d'une estimation des *endmembers* dominants, nous avons proposé un algorithme pour démixer ces pixels rares en utilisant ces connaissances *a priori*.

Les résultats de la méthode retenue sont encourageants, cependant les expérimentations ont mis en évidence les limites de l'algorithme lorsque le nombre de pixels rares est très faible ou au contraire, le nombre de faux positifs est trop élevé. Dans le chapitre suivant, nous proposerons l'utilisation d'une méthode de rééchantillonnage pour contourner ces difficultés.



# Chapitre 5

## Méthode rééchantillonnage Bootstrap

La méthode Bootstrap [103] est une méthode de rééchantillonnage qui permet la création de nouveaux échantillons à partir des échantillons observés. Dans ce chapitre nous présentons la méthode bootstrap et son application en imagerie hyperspectrale. Nous proposons une étude sur l'utilisation de la méthode bootstrap pour résoudre la problématique de démixage. Nous proposons deux nouvelles méthodes de démixage d'image hyperspectrale en présence d'*endmembers* rares qui tirent profit du rééchantillonnage Bootstrap [104].

### 5.1 Histoire du bootstrap

La technique dite du bootstrap [105, 2] est une méthode d'inférence statistique introduite par Bradley Efron en 1979 [103] qui a été utilisée initialement pour l'estimation de paramètres statistiques tels que la variance, la moyenne ou l'écart type d'un échantillon de population. Le nom de la méthode provient de l'expression anglaise "to pull oneself up by one's bootstraps" qui se traduit littéralement par "se hisser en tirant sur les languettes de ses propres bottes" en référence au baron von Münchhausen qui se serait sorti du marécage dans lequel il était embourbé en se tirant par les bottes. En effet, par bootstrap, on se réfère généralement à la technique de ré-échantillonnage [106], qui consiste à créer de nouveaux échantillons à partir des observations initiales.

Le ré-échantillonnage bootstrap est utilisé principalement pour accroître artificiellement les échantillons limités en données. À partir d'un échantillon initial issu d'une population dont on veut estimer certaines informations, on tire au sort, avec remise,  $n$  observations pour obtenir un nouvel échantillon. On répète l'opération plusieurs fois pour obtenir un grand nombre de nouveaux échantillons. Les nouvelles observations ainsi obtenues sont alors utilisées afin d'affiner l'estimation des paramètres faite sur les observations initiales. La figure 5.1 illustre la

création de nouveaux échantillons bootstrap par tirage avec remise à partir d'un échantillon initial.

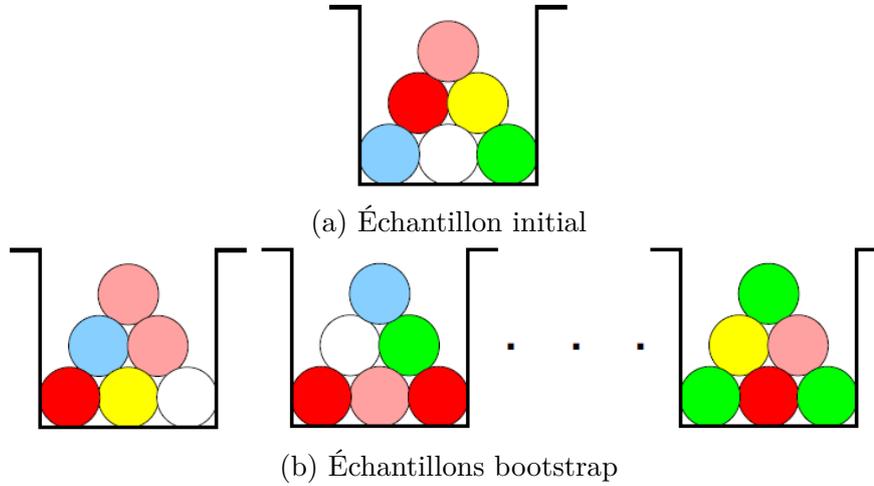


FIGURE 5.1 – Illustration du ré-échantillonnage bootstrap (Illustration issue [2])

Les méthodes de ré-échantillonnage bootstrap ont ensuite été utilisées en traitement du signal et de l'image [107, 108, 109]. Plus récemment l'utilisation de méthodes bootstrap a fait son apparition dans le traitement d'images hyperspectrales, tout d'abord pour traiter des problématiques de classifications [10, 12] et enfin dans des méthodes de démixage [110, 111].

En classification la méthode bootstrap est utilisée pour résoudre le problème de déséquilibre de classe [12]. Les problématiques de déséquilibre de classe apparaissent lorsqu'une classe est sous-représentée dans le jeu de données d'apprentissage. Le rééchantillonnage bootstrap permet de résoudre ce problème en générant de nouveaux ensembles d'apprentissage à partir de l'ensemble de données d'origine. Dans ce cas le bootstrap consiste à échantillonner chaque classe de façon uniforme, avec remise, pour obtenir un nouvel ensemble de données dans lequel les classes sont équilibrées. Dans la suite nous présentons une méthode de rééchantillonnage, inspirée de bootstrap, adaptée au démixage hyperspectral.

## 5.2 Application du bootstrap au démixage hyperspectral

Dans le cas d'un démixage des images hyperspectrales le paramètre que nous cherchons principalement à affiner est l'estimation des *endmembers* de la matrice  $\mathbf{S}$  et la matrice de mélange  $\mathbf{A}$  associée.

Le ré-échantillonnage bootstrap consiste à générer de "nouvelles" observations artificielles à partir d'un échantillon initial. Pour adapter la méthode bootstrap au démixage d'images hyperspectrales dans le cas d'un modèle de mélange linéaire

nous proposons de générer de nouveaux pixels  $\mathbf{y}_j^*$  à partir de la matrice des pixels observés  $\mathbf{Y}$ . Pour cela,  $q$  pixels sont choisis aléatoirement dans  $\mathbf{Y}$  et  $q$  coefficients  $b_{1,j} \dots b_{q,j}$  sont tirés uniformément dans  $[0, 1]$  puis normalisés de sorte que  $b_{i,j} \geq 0$  et  $\sum_i b_{i,j} = 1$ . Le nouveau pixel est alors décrit par l'équation suivante :

$$\mathbf{y}_j^* = \sum_{i=1}^q b_{i,j} \mathbf{y}_{p_{i,j}} \quad (5.1)$$

$$\sum_{i=1}^q b_{i,j} = 1 \text{ et } b_{i,j} \geq 0$$

Où les  $p_{i,j}$  sont les indices pixels choisis aléatoirement dans  $\mathbf{Y}$ . Si l'on remplace les pixels  $\mathbf{y}_{p_{i,j}}$  dans l'équation (5.1) par leur expression selon le modèle de mélange linéaire présenté par l'équation (2.1), nous obtenons l'expressions suivante :

$$\mathbf{y}_j^* = \sum_{i=1}^q b_{i,j} \left( \sum_{k=1}^K \alpha_{p_{i,j},k} \mathbf{s}_k + \mathbf{n}_{p_{i,j}} \right) \quad (5.2)$$

$$= \sum_{k=1}^K \left( \sum_{i=1}^q b_{i,j} \alpha_{p_{i,j},k} \right) \mathbf{s}_k + \sum_{i=1}^q b_{i,j} \mathbf{n}_{p_{i,j}} \quad (5.3)$$

En posant  $a_{j,k}^* = \sum_{i=1}^q b_{i,j} \alpha_{p_{i,j},k}$  et  $\mathbf{n}_j^* = \sum_{i=1}^q b_{i,j} \mathbf{n}_{p_{i,j}}$ , l'équation (5.3) devient :

$$\mathbf{y}_j^* = \sum_{k=1}^K a_{j,k}^* \mathbf{s}_k + \mathbf{n}_j^* \quad (5.4)$$

Comme d'une part, l'équation (2.5) du modèle de mélange linéaire impose que la somme sur  $k$  des  $\alpha_{p_{i,j},k}$  soit unitaire, et d'autre part, la somme sur  $i$  des  $b_{i,j}$  est elle aussi égale à 1 par construction (voir équation (5.1)), nous pouvons en déduire que la somme sur  $k$  des  $a_{j,k}^*$  est elle aussi unitaire :

$$\sum_{k=1}^K a_{j,k}^* = 1 \quad (5.5)$$

De plus le terme  $\mathbf{n}_j^*$  est une somme pondérée des vecteurs de bruit gaussiens centrés  $\mathbf{n}_p$ . Par conséquent  $\mathbf{n}_j^*$  suit lui aussi une loi gaussienne centrée.

Nous pouvons en déduire que les pixels générés par bootstrap respectent le modèle de mélange linéaire et ses contraintes, avec les même *endmembers*  $\mathbf{S}$  que les pixels observés. En générant  $P_b$  pixels par bootstrap nous pouvons produire

un nouvel échantillon  $\mathbf{Y}^*$  qui peut s'écrire sous forme matricielle :

$$\mathbf{Y}^* = \mathbf{A}^* \mathbf{S} + \mathbf{N}^* \quad (5.6)$$

Avec  $\mathbf{Y}^*$  dont les lignes correspondent aux pixels bootstraps  $\mathbf{y}_j^*$ , la nouvelle matrice de bruit  $\mathbf{N}^*$  dont les lignes sont les vecteurs  $\mathbf{n}_j^*$ , la matrice  $\mathbf{A}^*$  correspond aux abondances  $a_{j,k}^*$  des pixels bootstraps et  $\mathbf{S}$  est la matrice des *endmembers* initiale.

## 5.3 Étude de la méthode bootstrap

Dans cette partie, après avoir mis en évidence l'influence de l'utilisation du bootstrap sur le bruit d'un jeu de données, nous étudierons dans quelles mesures la méthode bootstrap permet d'améliorer le démixage dans un cas général. Le cas particulier des pixels rares sera traité dans la partie suivante.

### 5.3.1 Influence de l'algorithme bootstrap sur le bruit blanc gaussien

De manière générale, les méthodes de démixage sont sensibles au bruit, plus le SNR est élevé, meilleure est l'estimation des *endmembers*. La méthode bootstrap telle que présentée dans la partie précédente consiste à effectuer des combinaisons linéaires de pixels observés. Nous étudions ici l'impact qu'elle peut avoir sur les vecteurs de bruits contenus dans ces pixels.

D'après l'équation de mélange linéaire (2.3) le pixel  $y_p$  s'écrit :

$$\mathbf{y}_p = \boldsymbol{\alpha}_p \mathbf{S} + \mathbf{n}_p \quad (5.7)$$

Avec  $\mathbf{n}_p$  le vecteur de bruit gaussien, identiquement, indépendamment distribué. Le vecteur bruit suit une loi normale multidimensionnelle  $\mathbf{n}_p \sim \mathcal{N}(0, \sigma^2 I)$ , où  $\sigma$  influence directement le SNR.

Si l'on choisi aléatoirement, avec une probabilité uniforme,  $q$  pixels de l'image notés  $(\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_q)$  et  $b_1, b_2, \dots, b_q$  les coefficients de bootstrap correspondants,

le pixel bootstrap  $\mathbf{y}^*$  s'écrit d'après les équation (5.3) et (5.4) :

$$\mathbf{y}^* = \boldsymbol{\alpha}^* \mathbf{S} + \mathbf{n}^* \quad (5.8)$$

$$\boldsymbol{\alpha}^* = \sum_{i=1}^q b_i \boldsymbol{\alpha}_i \quad (5.9)$$

$$\mathbf{n}^* = \sum_{i=1}^q b_i \mathbf{n}_i \quad (5.10)$$

D'après les hypothèses du modèle de mélange linéaire, les vecteurs bruits  $\mathbf{n}_i$  sont indépendants. On en déduit la loi de probabilité d'un vecteur de bruit pour un pixel obtenu par bootstrap (ou "pixel bootstrap") :

$$\mathbf{n}^* \sim \mathcal{N}(0, \sum_{i=1}^q b_i^2 \sigma^2 I) \quad (5.11)$$

Par définition (5.1) les coefficients bootstrap sont positifs  $b_i \geq 0$  et leur somme est unitaire  $\sum_{i=1}^q b_i = 1$ . On en déduit immédiatement que la somme des coefficients au carré est inférieure à 1 :  $\sum_{i=1}^q b_i^2 \leq 1$ . Le cas d'égalité est un cas dégénéré atteint uniquement lorsque l'un des coefficient  $b_i$  vaut 1, les autres étant tous nul. La loi de probabilité de  $\mathbf{n}^*$ , équation (5.11), peut se réécrire :

$$\mathbf{n}^* \sim \mathcal{N}(0, \sigma^{*2} I) \quad (5.12)$$

$$\sigma^* < \sigma$$

La variance des pixels bootstraps étant inférieure à celle des pixels d'origines et le vecteur d'abondance  $\boldsymbol{\alpha}^*$  ayant des propriétés similaires au vecteurs d'abondance d'origine  $\boldsymbol{\alpha}_p$ , le SNR d'une image  $\mathbf{Y}^*$  constituée de pixels bootstraps doit être supérieur à celui de l'image  $\mathbf{Y}$  d'origine.

Cette propriété peut se vérifier expérimentalement, en générant une image  $\mathbf{Y}$  contenant 100 pixels avec un SNR fixé entre 10 et 30db, puis 100 pixels bootstraps  $\mathbf{Y}^*$  obtenus en mélangeant  $q$  pixels de  $\mathbf{Y}$ , pour  $q$  allant de 2 jusqu'à 25. L'opération est répétée pour plusieurs générations de  $\mathbf{Y}$  et différentes réalisations du bruit. Nous observons à chaque fois une augmentation du SNR de  $\mathbf{Y}^*$  par rapport à celui de  $\mathbf{Y}$ , cette augmentation dépend de façon logarithmique du nombre de pixels  $q$  utilisés. La figure 5.2 montre le résultat obtenu dans le cas d'un SNR initial de 20db.

Les résultats confirment notre approche théorique. L'utilisation du bootstrap permet d'augmenter le SNR. Nous nous attendons donc à de possibles améliorations de performances de la NMF en utilisant le bootstrap, notamment en cas de

bruit important.

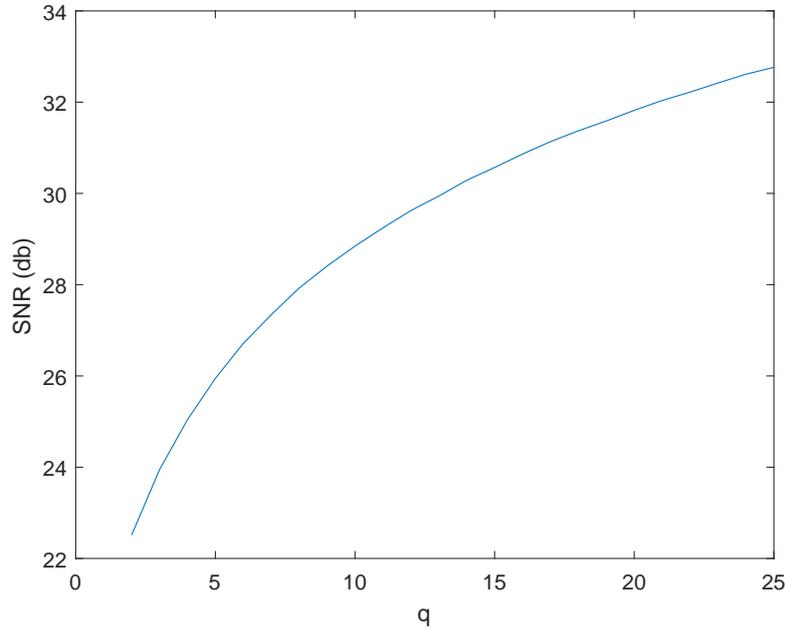


FIGURE 5.2 – SNR de  $\mathbf{Y}^*$  en fonction du nombre de pixels  $q$  utilisés pour le bootstrap

#### 5.3.2 Effet du bootstrap sur le démixage

Étudions à présent l'effet d'un rééchantillonnage par bootstrap sur les méthodes de démixage basées sur la NMF.

Pour les tests effectués des images  $\mathbf{Y}$  de 100 pixels ont été créés avec 6 endmembers de 166 bandes spectrales. Pour chaque image une image bootstrap  $\mathbf{Y}^*$  contenant 100 pixels bootstraps est générée à partir de  $q$  pixels de l'image d'origine. Les *endmembers* sont estimés en appliquant une méthode NMF aux pixels bootstraps  $\mathbf{Y}^*$ .

La qualité de l'estimation des *endmembers* est évaluée en mesurant le MSAD (voir équation (4.37)) après démixage sur l'image  $\mathbf{Y}^*$ .

La figure 5.3 présente les résultats obtenus en fonction du nombre  $q$  de pixels d'origine utilisés pour créer les pixels bootstraps de l'image  $\mathbf{Y}^*$  pour un SNR de 15db dans l'image d'origine. Le cas  $q = 1$  correspond à l'estimation obtenue en l'absence de bootstrap.

De même, la figure 5.4 présente les résultats obtenus lorsque le SNR de l'image d'origine est de 30db.

On observe des résultats différents en fonction de la méthode NMF utilisée :

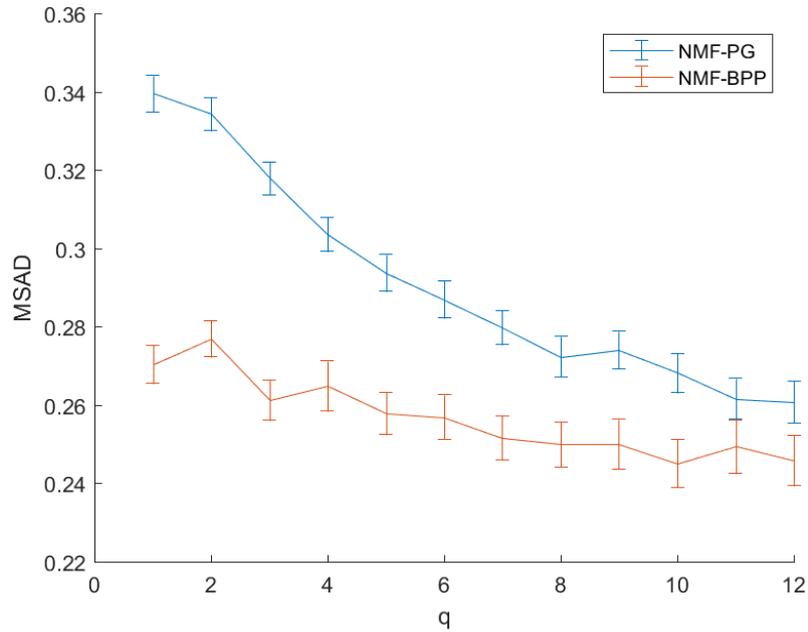


FIGURE 5.3 – Qualité de l'estimation des endmembers en fonction du nombre de pixels  $q$  utilisés pour le bootstrap pour un SNR initial de 15dB en utilisant la NMF-PG et la NMF-BPP pour démixer  $\mathbf{Y}^*$

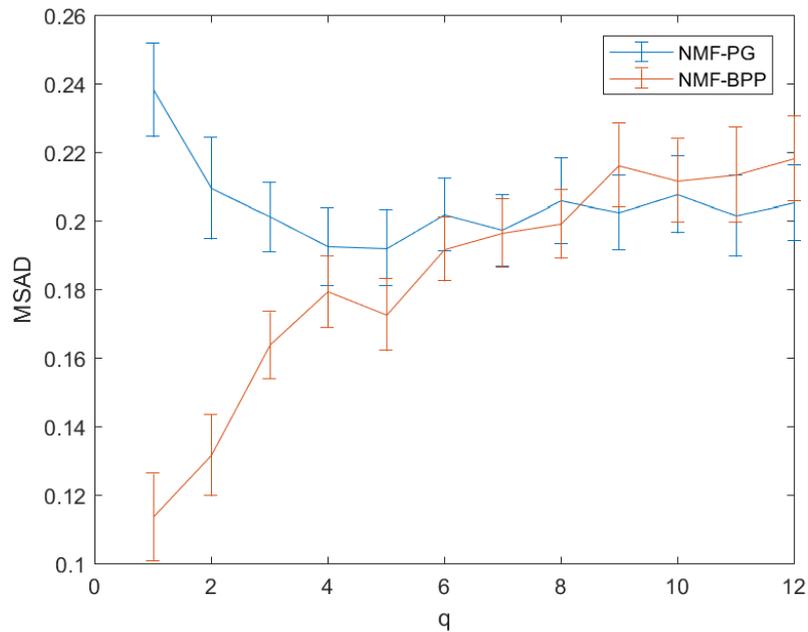


FIGURE 5.4 – Qualité de l'estimation des endmembers en fonction du nombre de pixels  $q$  utilisés pour le bootstrap pour un SNR initial de 30dB en utilisant la NMF-PG et la NMF-BPP pour démixer  $\mathbf{Y}^*$

Dans le cas de la NMF-PG, le pré-traitement par bootstrap permet d'améliorer l'estimation des *endmembers*. Dans les cas où le SNR est faible, plus le nombre  $q$  de pixels utilisés pour réaliser le bootstrap est élevé, meilleure est l'estimation. Dans le cas d'un SNR plus élevé, le bootstrap permet une meilleure estimation, même si les performances stagnent lorsque le nombre  $q$  de pixels utilisés est identique ou supérieur au nombre d'endmembers présents dans l'image d'origine.

Dans le cas de la NMF-BPP, le bootstrap permet d'améliorer l'estimation des *endmembers* lorsque le SNR est faible, mais il a un effet contre productif dans le cas d'un SNR élevé. Dans la partie précédente nous avons vu que la méthode bootstrap permettait de réduire la variance de bruit des pixels bootstraps par rapport à celle des pixels d'origine, et donc d'augmenter le SNR. Les pixels doivent subir un phénomène similaire en "diluant" les *endmembers*. Cette dilution pourrait réduire les performances de la NMF-BPP dans le cas d'un SNR élevé. À SNR faible, le gain de SNR qu'apporte la méthode bootstrap permettrait de contrebalancer ce phénomène.

Les estimations obtenues par la méthode NMF-BPP sont presque toujours meilleures que celles obtenues par la NMF-PG. Dans la suite de ce manuscrit, nous continuerons nos expérimentations en conservant uniquement la NMF-BPP.

#### 5.3.3 Influence de la présence des pixels originaux dans une image "bootstrapée"

Il peut y avoir deux façons différentes d'utiliser le bootstrap :

- Soit en générant une image composée uniquement de pixels bootstraps (comme au paragraphe précédent).
- Soit en générant des pixels bootstraps pour enrichir l'image d'origine.

Nous utiliserons ici des images  $\mathbf{Y}$  de 100 pixels créées à partir de 6 endmembers de 166 bandes spectrales.

Dans le premier cas, nous générons une image bootstrap  $\mathbf{Y}^*$  contenant 200 pixels bootstraps à partir de  $q$  pixels de l'image d'origine. Les méthodes de démixage basées sur la NMF sont ensuite appliquées à  $\mathbf{Y}^*$ .

Dans le second cas, nous générons une image bootstrap  $\mathbf{Y}^*$  contenant 100 pixels bootstraps à partir de  $q$  pixels de l'image d'origine. Les pixels bootstraps sont ensuite ajoutés à ceux de l'image d'origine, et les méthodes de démixage par NMF sont utilisées sur cette image augmentée  $\begin{bmatrix} \mathbf{Y} \\ \mathbf{Y}^* \end{bmatrix}$ .

À SNR faible, on peut observer l'apport du bootstrap en terme de débruitage. La figure 5.5 montre que les performances de la NMF-BPP sont meilleures pour image composée uniquement de pixels bootstraps que pour l'image augmentée. En effet, la présence des pixels originaux, dont le bruit n'a pas été réduit, dans

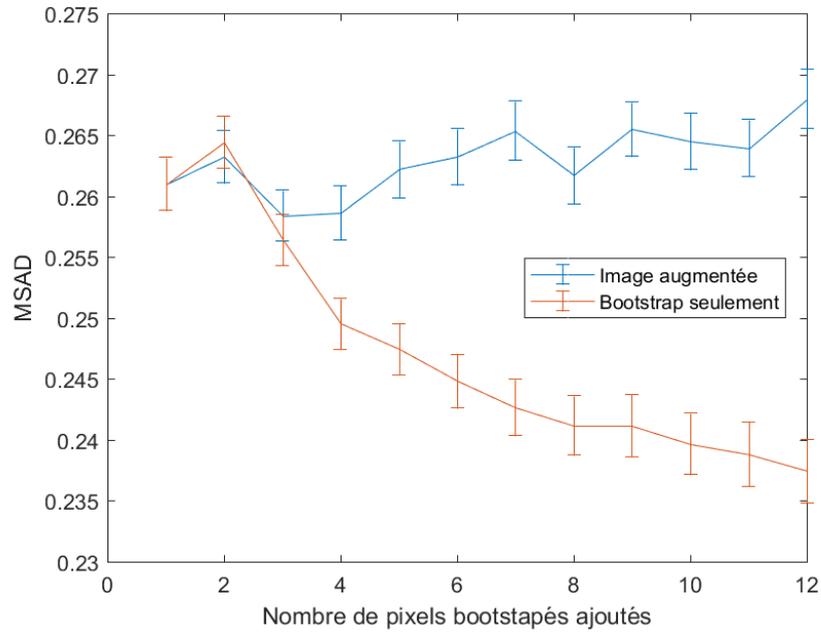


FIGURE 5.5 – Qualité de l’estimation des endmembers en fonction de nombre de pixels  $q$  utilisés pour le bootstrap pour un SNR initial de 15dB pour différentes façons d’utiliser le bootstrap

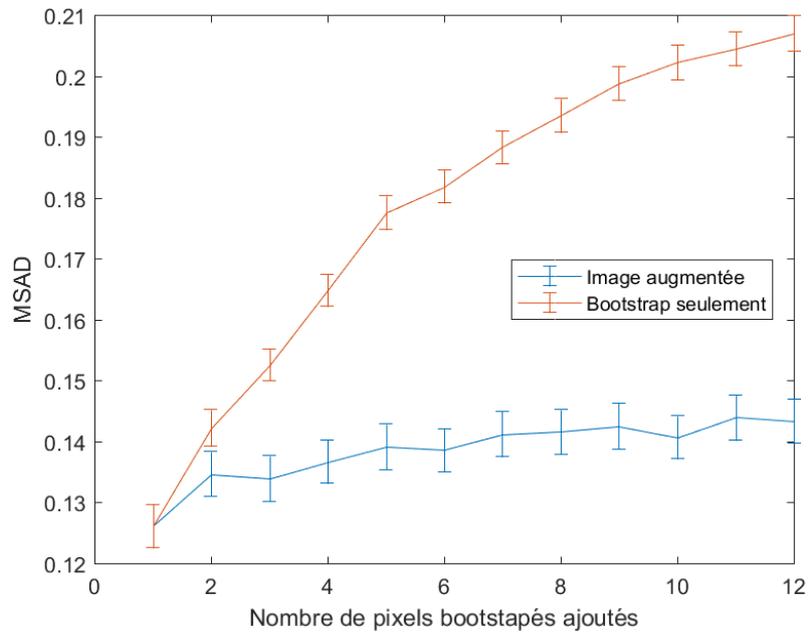


FIGURE 5.6 – Qualité de l’estimation des endmembers en fonction de nombre de pixels  $q$  utilisés pour le bootstrap pour un SNR initial de 30dB pour différentes façons d’utiliser le bootstrap

l'image augmentée ne facilite pas le démixage par la NMF.

À SNR élevé les résultats sont de meilleure qualité dans le cas de l'image augmentée comme on peut l'observer sur la figure 5.6. Dans la section précédente nous avons émis l'hypothèse que le gain de SNR apporté par bootstrap était négligeable face au phénomène de dilution des *endmembers* qui réduisait les performances de la NMF. En effet, le fait de conserver les pixels originaux réduit ici l'impact de ce phénomène de dilution.

Les résultats expérimentaux dans le cas général ont permis d'étudier le comportement des méthodes de démixage basées sur la NMF. Nous avons montré que ces résultats variaient en fonction de la méthode NMF utilisée, de la façon dont était utilisé le bootstrap et du SNR de l'image d'origine sans toutefois pouvoir mettre en évidence un avantage significatif à l'utilisation du bootstrap dans un cas général.

## 5.4 Étude en présence de pixels rares

Dans la partie précédente, nous avons étudié l'utilisation du bootstrap dans un cas général. Ici, nous l'étudierons sur le démixage en présence de pixel rare, pour déterminer s'il est possible d'améliorer les résultats de démixage à l'aide de la méthode bootstrap.

### 5.4.1 Enrichissement de l'image originale par Bootstrap

En supposant que l'on ait pu déterminer la ou les régions contenant les *endmembers* rares, par exemple en utilisant une des méthodes de détection présentée dans le chapitre 3, nous disposons de deux sous-matrices  $\mathbf{Y}_d$  et  $\mathbf{Y}_r$  contenant les pixels dominants pour l'une et les pixels rares pour l'autre. L'idée ici est d'utiliser l'algorithme de bootstrap sur  $\mathbf{Y}_r$  pour augmenter le nombre de pixels contenant les *endmembers* rares, puis les ajouter à l'image originale.

L'algorithme se décompose de la façon suivantes :

- Entrées : image hyperspectrale  $\mathbf{Y}$ , nombre  $k$  d'*endmembers*
- Repérage des zones d'intérêt contenant les pixels rares. Ces zones d'intérêt sont regroupées en une imagerie  $\mathbf{Y}_r$ .
- Utilisation du bootstrap sur l'imagerie  $\mathbf{Y}_r$  pour augmenter le nombre de pixels d'intérêt. On stocke ces pixels dans  $\mathbf{Y}_r^* = (\mathbf{Y}_r, \mathbf{Y}^*)$ .
- Concaténation de l'image de départ avec les pixels bootstraps :  $\mathbf{Y}_{augmented} = (\mathbf{Y}, \mathbf{Y}_r^*)^T$
- Démixage de tous les *endmembers* par NMF-BPP sur  $\mathbf{Y}_{augmented}$  :  $\mathbf{Y}_{augmented} = \mathbf{A}_{augmented} \tilde{\mathbf{S}}$
- On réutilise  $\tilde{\mathbf{S}}$  pour réaliser la factorisation de l'image  $\mathbf{Y} = \tilde{\mathbf{A}} \tilde{\mathbf{S}}$

Pour tester la méthode nous générons des images des 1600 pixels, et 166 bandes, comportant 5 *endmembers* dominants, et 1 *endmember* rare localisé sur une cible de 4 pixels. L'abondance de cet *endmember* est entre 0.66 et 0.8 sur la cible. Le SNR est fixé à 30dB. La région d'intérêt (contenant les pixels rares) a été détectée au préalable, elle est constituée de 16 pixels (couronne de largeur 1 pixels autour de la cible). Nous avons fait varier le nombre de pixels ajoutés par bootstrap dans l'image de départ.

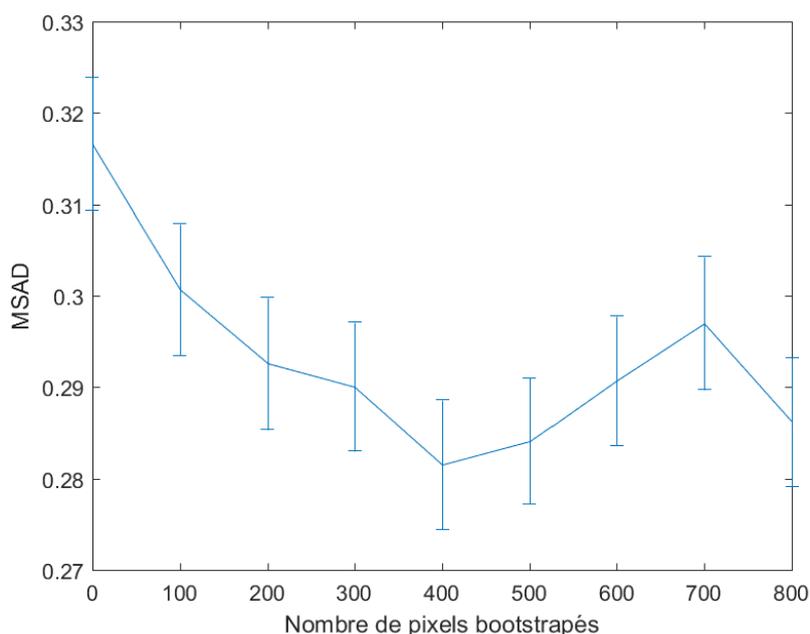


FIGURE 5.7 – Qualité de l'estimation des *endmembers* démixés par NMF-BPP en fonction du nombre de pixels bootstraps ajoutés à l'image d'origine

La figure 5.7 montre l'évolution de l'estimation de l'ensemble des *endmembers* en fonction du nombre de pixels bootstrap ajoutés. La figure 5.8 montre quand à elle l'évolution de l'estimation de l'*endmember* rare seul. Lorsque le nombre de pixels ajoutés est nul, les résultats obtenus sont ceux de la NMF sans bootstrap.

Les résultats montrent qu'augmenter artificiellement le nombre de pixels permet d'améliorer l'estimation des *endmembers* rares en conservant la bonne estimation des dominants. En revanche, il semble qu'il n'est pas nécessaire d'ajouter beaucoup de pixels et qu'au contraire, un ajout de pixels trop important pourrait être contre productif.

### 5.4.2 NMF-BDR

La méthode que NMF-BDR que nous proposons [104] dans cette partie consiste à Bootstraper les pixels Dominant et les pixels Rare ensemble, d'où son nom BDR.

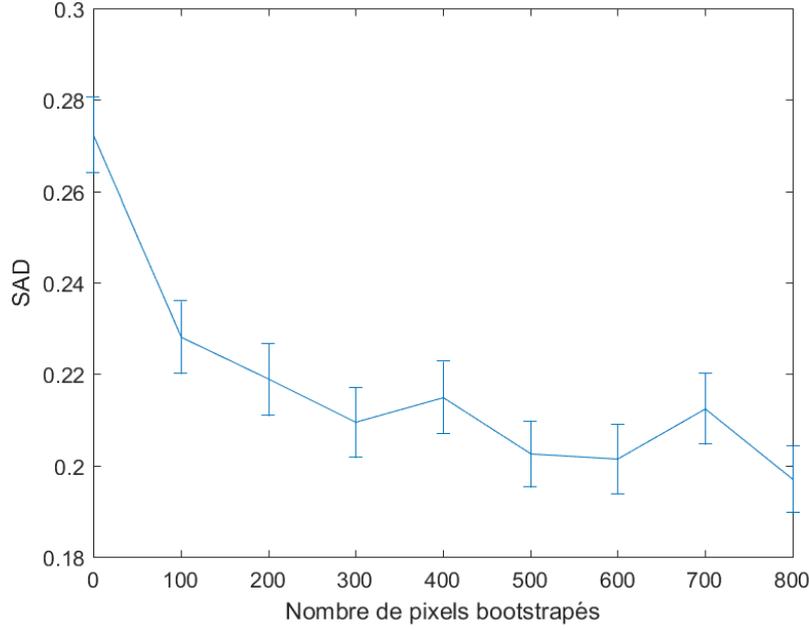


FIGURE 5.8 – Qualité de l'estimation de l'*endmember* rare démixé par NMF-BPP en fonction du nombre de pixels bootstraps ajoutés à l'image d'origine

La méthode NMF-BDR s'inspire des méthodes bootstrap utilisées pour la classification en imagerie hyperspectrale. Les deux sous-matrices  $\mathbf{Y}_d$  et  $\mathbf{Y}_r$  peuvent être vues comme une classification des pixels observés en deux classes : les pixels dominants et les pixels rares. Par définition, la plupart des pixels de l'image sont des pixels dominants et très peu d'entre eux sont des pixels rares. L'idée clé est de faire en sorte que les pixels rares perdent leur caractéristique "rare" en utilisant le rééchantillonnage bootstrap.

Nous générons une nouvelle image de  $P$  pixels bootstrap. Les pixels bootstrap sont générés en combinant deux pixels observés. Pour s'assurer que les *endmembers* rares ne le soient plus dans la nouvelle image, nous choisissons aléatoirement un pixel dans chaque classe, de sorte que :

$$y_j^* = \alpha_j y_{d,j} + \beta_j y_{r,j} \quad (5.13)$$

Avec  $y_{d,j}$  un pixel choisi aléatoirement dans  $\mathbf{Y}_d$  et  $y_{r,j}$  pris aléatoirement dans  $\mathbf{Y}_r$ .  $\alpha_j$  et  $\beta_j$  sont positifs et de somme unitaire  $\alpha_j + \beta_j = 1$ .

Le nombre de pixels dans  $\mathbf{Y}_r$  est très inférieur au nombre de pixels  $P$  présents dans l'image, en conséquent chaque pixel rare est présent dans plusieurs pixels de  $\mathbf{Y}^*$ . De plus comme les *endmembers* dominants sont présents dans de nombreux pixels, en sélectionnant un pixel de  $\mathbf{Y}_d$  pour chaque pixel bootstrap, on s'assure

qu'ils sont présents dans plusieurs pixels de  $\mathbf{Y}^*$ . Par conséquent les *endmembers* rares et dominants sont présents dans un grand nombre de pixels de  $\mathbf{Y}^*$  et il est donc possible des les estimer avec une méthode NMF en minimisant la fonction de cout suivante :

$$C(\mathbf{A}^*, \mathbf{S}) = \|\mathbf{Y}^* - \mathbf{A}^* \mathbf{S}\|^2 \quad (5.14)$$

La méthode NMF nous permet d'estimer la matrice des endmembers  $\mathbf{S}$  et la matrice des abondances bootstraps  $\mathbf{A}^*$ .

La matrice des abondances bootstraps  $\mathbf{A}^*$  n'est pas pertinente, nous devons estimer la matrice d'abondance  $\mathbf{A}$  correspondant à l'image d'origine  $Y$ . Pour cela nous proposons d'utiliser une méthode d'estimation des moindre carré non-négative pour résoudre l'équation suivante :

$$\mathbf{A} = \min_{\mathbf{A}} \|\mathbf{Y} - \mathbf{A} \mathbf{S}\|^2 \quad (5.15)$$

Nous obtenons alors une estimation  $\mathbf{S}$  de tous les *endmembers* dominants et rares ainsi que leur matrice d'abondances  $\mathbf{A}$ .

### 5.4.3 NMF-BR

Dans cette section, nous supposons avoir une estimation des *endmembers* dominants  $\mathbf{S}_d$  qui nous a permis d'obtenir  $\mathbf{Y}_d$  et  $\mathbf{Y}_r$ . Nous nous intéressons uniquement aux pixels observés présents dans  $\mathbf{Y}_r$ . Cette matrice contient peu de pixels, la plupart sont des pixels rares, ou des pixels avec une réalisation du bruit anormale. Le nombre de pixels présents dans  $\mathbf{Y}_r$  peut être trop faible pour obtenir une estimation fiable des *endmembers* rares avec la méthode NMF [12]. Pour résoudre ce problème, nous utilisons la méthode de rééchantillonnage bootstrap pour augmenter le nombre de pixels utilisables.

Notons  $\mathbf{Y}^*$  la matrice contenant  $P_b$  pixels bootstraps, rééchantillonnés à partir de  $\mathbf{Y}_r$  :

$$\mathbf{Y}^* = \mathbf{B} \mathbf{Y}_r \quad (5.16)$$

Avec  $\mathbf{B} \in \mathbb{R}^{P_b \times P_r}$  la matrice des coefficients bootstrap de l'équation (5.1). La matrice  $\mathbf{Y}^*$  peut, de façon similaire à l'équation (4.32), s'exprimer sous la forme :

$$\mathbf{Y}^* = \mathbf{A}_d^* \mathbf{S}_d + \mathbf{A}_r^* \mathbf{S}_r + \mathbf{N}_r \quad (5.17)$$

Où  $\mathbf{A}^*$  est la matrice d'abondance artificielle correspondant aux pixels boots-

traps, et  $\mathbf{S}$  la matrice des *endmembers* initiaux.

Nous utilisons la méthode NMF *a priori*, présentée dans la partie 4.2.1 pour démixer l'image  $\mathbf{Y}^*$  avec les *endmembers* dominants  $\mathbf{S}_d$  connus, pour minimiser la fonction de coût :

$$C(\mathbf{A}^*, \mathbf{S}_r) = \|\mathbf{Y}^* - \mathbf{A}^* \mathbf{S}_r\|^2 \quad (5.18)$$

$$s.t. S_r \geq 0, \mathbf{A}^* \geq 0, \mathbf{A}^* \mathbf{1}_{P_r} = \mathbf{1}_{P_r} \quad (5.19)$$

Nous obtenons une estimation des *endmembers* rares  $\mathbf{S}_r$  et les abondances bootstraps  $\mathbf{A}^*$ . Nous disposons donc d'une estimation de tous les *endmembers*  $\mathbf{S}$ , cependant nous devons encore estimer les abondances. L'estimation des abondances se fait avec un méthode des moindres carrés non-négatifs telle que :

$$\mathbf{A} = \min_{\mathbf{A}} \|\mathbf{Y} - \mathbf{A} \mathbf{S}\|^2 \quad (5.20)$$

La méthode complète est présentée par l'algorithme 3.

---

**Algorithme 3** NMF-BR
 

---

**Entrée :**  $\mathbf{Y}$

```

/* Estimation des endmembers dominants */
 $\hat{\mathbf{A}}_d, \hat{\mathbf{S}}_d \leftarrow \text{NMF}(\mathbf{Y})$ 
/* Detection des pixels rares */
 $\hat{\sigma}^2 \leftarrow \text{estimationVarianceDuBruit}(\mathbf{Y})$ 
 $\tau \leftarrow \hat{\sigma}^2 + 3\sqrt{\frac{2\hat{\sigma}^4}{L}}$  /* Estimation de la valeur du seuil */
 $r_p \leftarrow \|\mathbf{y}_p - \hat{\mathbf{a}}_{pd} \hat{\mathbf{S}}_d\|^2$ 
 $\mathbf{Y}_r \leftarrow \{\mathbf{y}_p \mid r_p \geq \tau\}$ 
/* Estimation des endmembers rares */
 $\mathbf{Y}^* \leftarrow \text{bootstrap}(\mathbf{Y}_r)$ 
 $\hat{\mathbf{A}}^*, \hat{\mathbf{S}}_r \leftarrow \text{NMF}_{\text{apriori}}(\mathbf{Y}^* | \hat{\mathbf{S}}_d)$ 
 $\hat{\mathbf{A}} \leftarrow \min_{\mathbf{A}} \|\mathbf{Y} - \mathbf{A} \hat{\mathbf{S}}\|$ 
    
```

**Sortie :**  $\hat{\mathbf{A}}, \hat{\mathbf{S}}$

---

#### 5.4.4 Résultats comparatifs

Nous générons 1000 images de 1600 pixels, 166 bandes, avec 5 *endmembers* dominants répartis sur l'image et 2 rares : un sur quatre cibles de 4 pixels, et un sur une cible de 9 pixels. Les abondances choisies sont entre 0.2 et 0.33 : pour une cible de 4 pixels, cela correspondrait à 1 pixel pur détecté à cheval sur 4 pixels. La figure 5.9 présente les *endmembers* utilisés et les cartes d'abondances pour chacun des deux *endmembers* rares pour une des réalisations.

Nous avons comparé les performances de démixage pour la NMF sur l'image

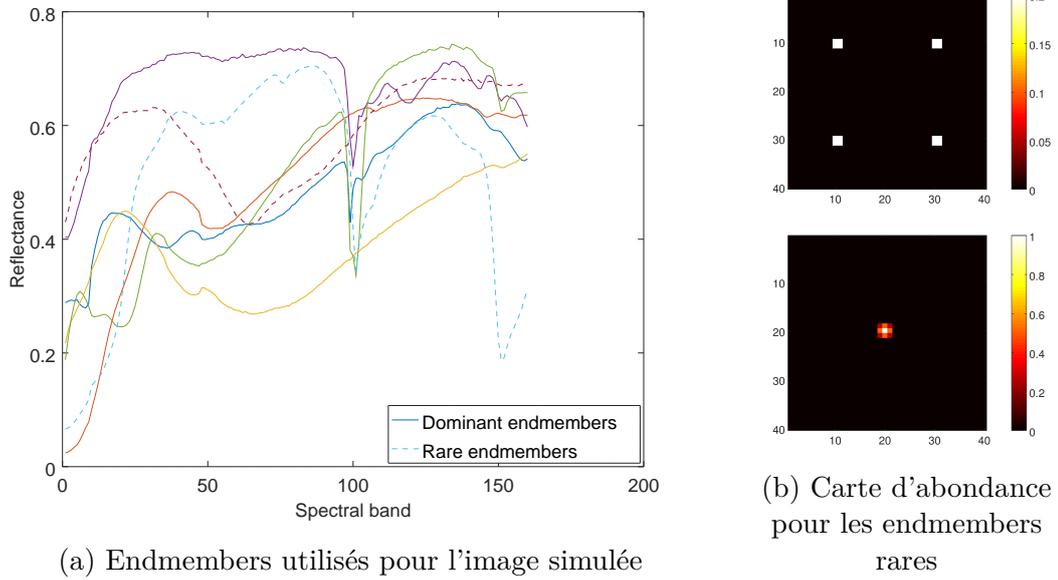


FIGURE 5.9 – Spectres de réflexion des *endmembers* utilisés pour réaliser l'image simulée (a) et cartes d'abondance des spectres rares (b)

de départ, les méthodes proposées NMF-BR et NMF-BDR pour des SNR variant entre 24 et 30dB.

La qualité d'estimation est évaluée à l'aide du MSAD pour mesurer la distance entre les valeurs estimées et les valeurs réelles. Les résultats numériques sont présentés dans le tableau 5.1.

SNR (db)	24	26	28	30
NMF	0.2656	0.2365	0.2239	0.2033
NMF-BDR	0.2082	0.1961	0.1859	0.1805
NMF-BR	0.1958	0.1808	0.1762	0.1645

TABLE 5.1 – MSAD pour la NMF et les méthodes proposées avec bootstrap des rares et dominants NMF-BDR ou des rares uniquement NMF-BR

L'estimation des abondance est évaluée en utilisant l'erreur quadratique moyenne normalisée (NMSE) entre les abondances estimées et la vérité terrain :

$$NMSE = \frac{\|\mathbf{A} - \hat{\mathbf{A}}\|^2}{\|\mathbf{A}\|^2} \quad (5.21)$$

Les résultats numériques sont présentés dans le tableau 5.2.

SNR (db)	24	26	28	30
NMF	0.0194	0.0195	0.0195	0.0193
NMF-BDR	0.0183	0.0186	0.0184	0.0181
NMF-BR	0.0178	0.0178	0.0177	0.0178

TABLE 5.2 – NMSE pour les méthodes NMF, NMF-BDR et NMF-BR.

On observe que les deux méthodes proposées permettent d'améliorer l'estimation des endmembers par rapport à une NMF simple sur l'image de départ. La NMF-BR permet d'obtenir de meilleurs résultats que la NMF-BDR en termes d'estimation des *endmembers* et des abondances.

La figure 5.10 permet d'observer les estimations obtenus par la méthode NMF-BR.

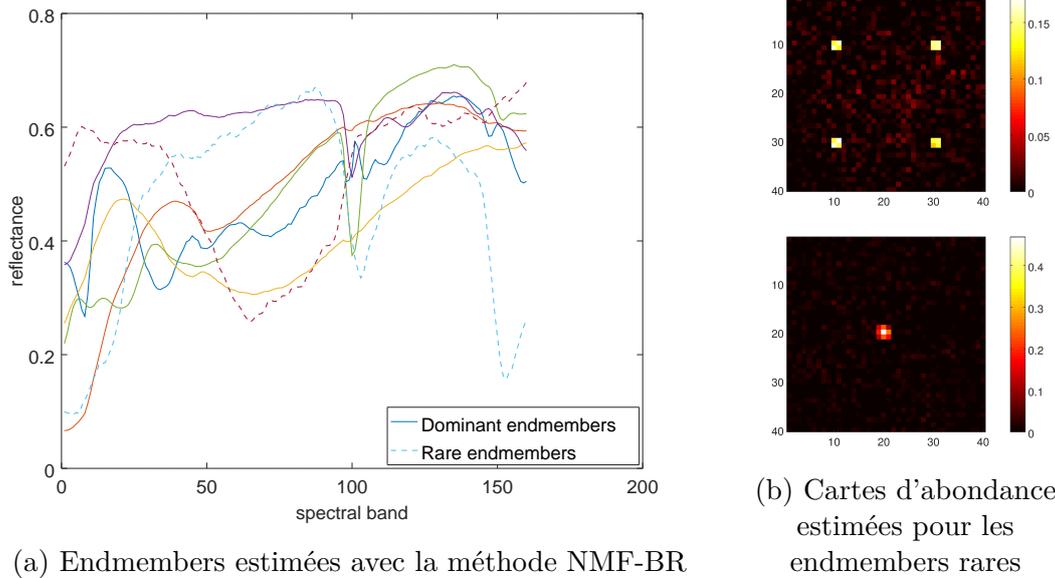


FIGURE 5.10 – Endmembers et abondances estimés avec la méthode NMF-BR

## 5.5 Application sur des données réelles

Nous avons appliqué la méthode proposée à une image issue du capteur HY-DICE (voir annexe A). L'image utilisée est de taille 100x150 pixels et contient 169 bandes spectrales, elle est présentée dans la figure 5.11a. La scène représente un champ d'herbes, bordé par une route à droite et une forêt à gauche. Quinze panneaux sont disposés dans le champ et organisés en cinq lignes et trois colonnes. Les panneaux sont rangés par taille de 3x3m dans la colonne de gauche puis 2x2m et 1x1m pour les colonnes centrale et droite. Les panneaux sur une même ligne sont faits dans le même matériau. La figure 5.11b montre la position des panneaux. Ces panneaux sont considérés comme des cibles de petites tailles dont

nous désirons estimer les *endmembers*. La figure 5.12 présente les *endmembers* contenus dans l'image.

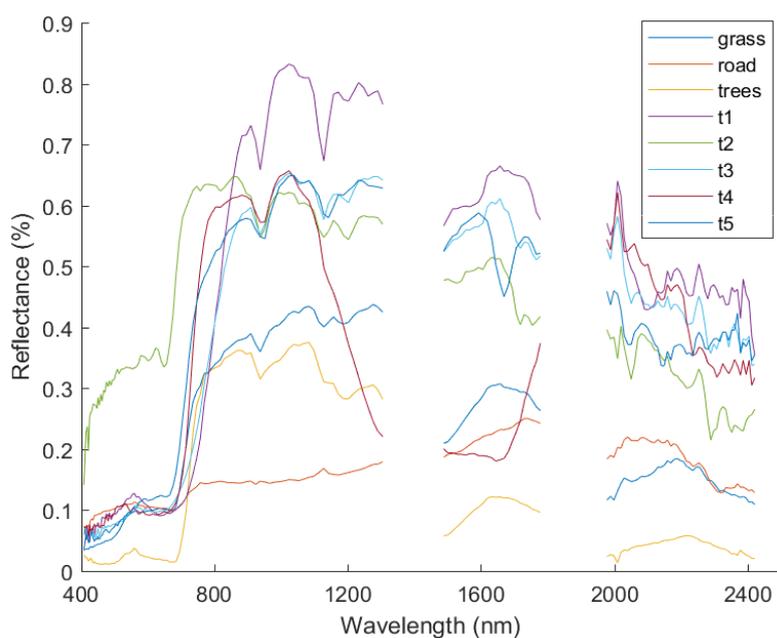
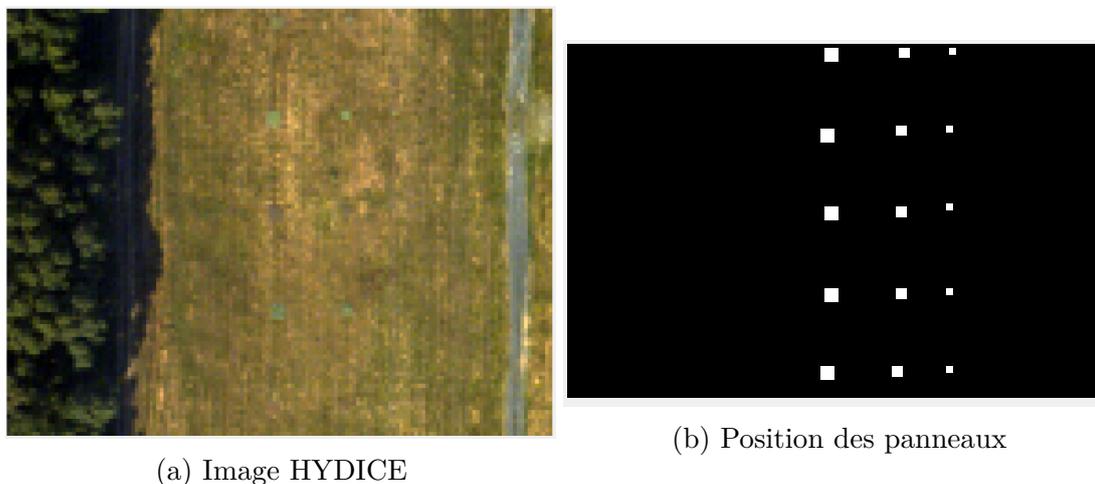


FIGURE 5.12 – Endmembers présent dans l'image

Nous utilisons les méthodes proposées NMF-BDR et NMF-BR pour démixer l'image et comparons les *endmembers* estimés à la vérité terrain. Les résultats numériques sont présentés dans le tableau 5.3.

Les méthodes proposées NMF-BDR et NMF-BR permettent toutes les deux d'améliorer l'estimation des *endmembers* pour l'image HYDICE par rapport à la NMF-BPP. Les performances de la NMF-BR sont nettement meilleures que celles de la NMF-BDR. Après démixage par la méthode NMF-BR, chaque pixel à été affecté à la classe correspondant à son abondance majoritaire. Le résultat

	NMF-BPP	NMF-BFR	NMF-BR
MSAD	0.5179	0.4184	0.2715

TABLE 5.3 – Comparaison des méthodes NMF-BPP, NMF-BDR et NMF-BR selon le critère MSAD

est présenté par la figure 5.13. Les labels 1, 2, et 3 sont associés respectivement au champ, à la route et à la zone boisée. Les classes 4 à 8 correspondent à chaque matériau présent dans les cibles. Nous observons que le démixage a permis de retrouver toutes les cibles. Certains pixels sont classés incorrectement, notamment en bordure de route ou dans la forêt. Ces imperfections sont dues à des erreurs d'estimation d'abondances qui peuvent s'expliquer par la forte variabilité des *endmembers* dans ces régions.



FIGURE 5.13 – Carte des abondances après démixage

## 5.6 Conclusion

Dans ce chapitre nous avons étudié la méthode de rééchantillonnage Bootstrap. Nous avons montré qu'elle permettait de rééchantillonner les pixels observés en conservant les propriétés de mélange linéaire. La création de pixels bootstrap permet de créer de nouveaux pixels moins bruités que les pixels observés, dans le cas de SNR faible, ce débruitage permet d'améliorer les résultats d'un démixage. Nous avons montré qu'utiliser une méthode de démixage sur des pixels bootstrap pouvait améliorer la qualité des estimations, c'est le cas notamment pour la méthode NMF-PG, cependant cela peut également conduire à une dégradation des performances comme pour la méthode NMF-BPP.

Nous avons ensuite étudié plusieurs façons d'utiliser la méthode de rééchantillonnage en présence d'*endmembers* rares. Les méthodes proposées ont pour prérequis qu'il est possible de classer les pixels suivant qu'ils contiennent ou non un *endmember* rare. Nous avons montré que les méthodes proposées NMF-BDR et NMF-BR permettaient toutes les deux d'améliorer les résultats de démixage en présence de pixels rares, la méthode NMF-BR étant la mieux adaptée à la problématique. L'utilisation de la méthode du bootstrap permet donc d'améliorer le démixage en présence d'objets de petites tailles.