

---

## Rapport de stage

---

# Utilisation du Deep Learning pour l'imagerie hyperspectrale en microscopie électronique.

---

Quentin Giton

### Résumé

Les avancées effectuées en microscopie électronique analytique permettent l'acquisition d'images hyperspectrales dans lesquelles chaque pixel représente un spectre de pertes d'énergie d'électrons. Le jeu de données acquis par le microscope possède donc une structure tridimensionnelle. Afin de déterminer les propriétés d'un échantillon observé, il est nécessaire de savoir de quels éléments chimiques ce dernier est constitué. Il est possible de décrire ces images hyperspectrales comme une combinaison convexe de spectres de perte d'énergie d'électrons – associés aux éléments chimiques présents dans l'échantillon. C'est ce qu'on appelle le démixage spectral.

Le démixage spectral est apparu pour la première fois dans le domaine de la télédétection par imagerie satellitaire. Un des outils utilisé en télédétection est un modèle de *deep learning* appelé Auto-encodeur. Au vu des similitudes entre les jeux de données traités par la télédétection et par la microscopie analytique, nous avons envie d'appliquer ces mêmes méthodes à nos données.

En télédétection, les méthodes de *deep learning* se révèlent être plus performantes pour effectuer le démixage. La question que nous nous posons alors est la suivante : ces méthodes sont-elles plus performantes que les algorithmes classiques pour traiter les données de spectroscopie électronique ?

**Mots clés :** Deep Learning, Auto-encodeurs, Démixage Spectral, Réseaux de neurones

## Remerciements

Je tiens, dans un premier temps, à remercier le Laboratoire de Physique des Solides et particulièrement Nathalie Brun, chercheuse dans ce laboratoire, pour sa confiance, sa disponibilité et toute l'autonomie qu'elle m'a offert durant ces quatre mois de stage. Je souhaite également remercier l'ensemble de l'équipe de microscopie électronique pour leur accueil qui m'a permis de m'épanouir lors de ce stage.

Je remercie Sarah Darmon, étudiante dans le M2 *Quantum, Light, Materials and Nano sciences* de l'Université Paris-Saclay, qui a réalisé son stage en parallèle du mien. Elle m'a accompagné pendant mes quatre mois de stage et m'a aidé à comprendre les notions physiques liées au sujet.

Enfin, je tiens à remercier Christine Keribin, une de mes professeur·e·s du M1 Mathématiques et Intelligence Artificielle, pour cette année et pour avoir transmis l'offre de stage de Nathalie Brun.

# Sommaire

Liste des symboles	1
Glossaire et abréviations	3
Table des figures	4
Introduction	5
<b>1 Contexte du stage</b>	<b>6</b>
1.1 Présentation du laboratoire	6
1.2 Spectroscopie EELS	6
1.3 Objectifs du stage	7
1.3.1 Démélange spectral	7
1.3.2 Soustraction du fond continu	7
1.3.3 Objectifs	8
<b>2 Modèle et outils</b>	<b>9</b>
2.1 Modèle de mélange linéaire	9
2.2 Les algorithmes classiques de démélange	10
2.2.1 NMF	10
2.2.2 VCA	10
2.3 Les auto-encodeurs	11
2.3.1 DAEU	11
2.3.2 EndNet	13
2.3.3 DeepTrans-HSU	14
2.4 Évaluation	15
2.4.1 RMSE	15
2.4.2 SAD	15
2.4.3 Évaluation qualitative	15

<b>3 Résultats</b>	<b>16</b>
3.1 Présentation des données . . . . .	16
3.2 NMF et VCA . . . . .	16
3.3 DAEU . . . . .	18
3.4 EndNet . . . . .	18
3.5 DeepTrans-HSU . . . . .	20
3.6 En résumé . . . . .	20
<b>Conclusion</b>	<b>22</b>
<b>Bibliographie</b>	<b>23</b>
<b>Appendices</b>	<b>24</b>
A. Cas pathologiques de la méthode de soustraction du fond continu . . . . .	24
B. Résultats sur un autre échantillon . . . . .	25

## Liste des symboles

Cette liste regroupe l'ensemble des symboles utilisés dans ce rapport de stage. Parmi les symboles en gras, les lettres minuscules représentent des vecteurs et les lettres en majuscules des matrices.

$\alpha$	Rang de factorisation.
$\alpha$	Vecteur de taille $c$ utilisé comme seuil dans la sixième couche de <a href="#">DAEU</a> .
$\varepsilon_{ij}$	Bruit additif poissonnien.
$\mu_{\mathcal{B}}, \sigma_{\mathcal{B}}$	Moyenne et variance empiriques au sein d'un <i>batch</i> $\mathcal{B}$
$\cdot_c$	Produit faisant office de produit scalaire dans le calcul des activations des neurones du réseau <a href="#">EndNet</a> .
$\gamma, \beta$	Paramètres de la transformation affine de la cinquième couche de <a href="#">DAEU</a> .
$\mathbf{A}$	Matrice de taille $c \times (n \cdot m)$ dont les colonnes sont les vecteurs d'abondance $\mathbf{a}_i$ associés aux pixels $\mathbf{x}_i$ .
$\mathbf{a}_{ij}$	Vecteur d'abondance de dimension $c$ associé au pixel $\mathbf{x}_{ij}$ observé.
$\mathbf{b}^{(l)}$	$l$ -ième couche du réseau <a href="#">DAEU</a> .
$\mathbf{h}$	Variable latente du réseau <a href="#">EndNet</a> .
$\mathbf{H}_1, \mathbf{H}_2$	Matrices de factorisation de tailles respectives $L \times \alpha$ et $\alpha \times (n \cdot m)$ .
$\mathbf{I}$	Image hyperspectrale d'entrée du réseau <a href="#">DeepTrans-HSU</a> .
$\mathbf{S}$	Matrice "source" de dimension $L \times c$ dont les colonnes sont les spectres $\mathbf{s}_k$ des éléments chimiques contenus dans l'échantillon.
$\mathbf{s}_k$	Spectre de dimension $L$ de l'élément chimique $k$ constituant l'échantillon.
$\mathbf{W}^{(e)}$	Poids de la partie encodeur du réseau <a href="#">EndNet</a> .
$\mathbf{X}$	Matrice de taille $L \times (n \cdot m)$ dont les colonnes sont les pixels observés $\mathbf{x}_i$ .
$\mathbf{x}_i$	$i$ -ième colonne de l'image hyperspectrale aplatie $X$ .
$\mathbf{x}_{ij}$	Pixel (spectre) de dimension $L$ de l'image hyperspectrale observé.
$\mathbf{y}$	Activations de la dernière couche cachée de <a href="#">EndNet</a> .
$\mathbf{z}$	Activation des neurones de la couche de <i>dropout</i> du réseau <a href="#">EndNet</a> .
$\mathbf{z}^*$	Couche du réseau <a href="#">EndNet</a> dont seules les deux plus hautes activations ont été sélectionnées.
$\mathcal{B}$	<i>Batch</i> de taille $B$ .
$\mathcal{F}$	Espace latent de l'auto-encodeur <a href="#">DAEU</a> .
$\mathcal{X}$	Espace des données.
$\phi$	Fonction de transition de l'espace des données $\mathcal{X}$ vers l'espace latent $\mathcal{F}$ .
$\psi$	Fonction de transition de l'espace latent $\mathcal{F}$ vers l'espace des données $\mathcal{X}$ .
$\star$	Produit terme à terme.
$\hat{\mathbf{I}}$	Image hyperspectrale estimée par le réseau <a href="#">DeepTrans-HSU</a> .
$a_{ij}^{(k)}$	$k$ -ième coordonnée du vecteur $\mathbf{a}_{ij}$ .

$B$	Taille de <i>batch</i> du réseau DAEU.
$C$	Constante d'échelle.
$c$	Nombre de composantes constituant l'échantillon.
$E$	Énergie en électronvolts (eV).
$J$	Fonction de perte d'énergie.
$L$	Nombre de canaux.
$m$	Largeur de l'échantillon en nombre de pixels.
$n$	Longueur de l'échantillon en nombre de pixels.
$r$	Exposant de la pente.
$^{(k)}$	$k$ -ième composante du vecteur considéré lors de la <i>batch normalisation</i> du réseau DAEU.
$W^{(l)}$	Poids de la $l$ -ième couche de DAEU.
$g$	Fonction d'activation LeakyReLU.

## Glossaire et abréviations

**ANC** Contrainte de positivité des vecteurs d’abondance - *Abundances Non-negativity Constraint*. [9](#)

**ASC** Contrainte de somme à un des vecteurs d’abondance - *Abundances Sum-to-one Constraint*. [9](#), [13](#), [14](#)

**CNRS** Centre National de la Recherche Scientifique. [6](#)

**DAEU** Réseau de neurones créé par les auteurs de l’article [1]. [1](#), [2](#), [4](#), [11](#), [12](#), [14](#), [18](#), [19](#)

**DeepTrans-HSU** Réseau de neurones créé par les auteurs de l’article [2]. [1](#), [4](#), [14](#), [20](#), [22](#)

**EELS** Spectroscopie par pertes d’énergie d’électrons - *Electron Energy Loss Spectroscopy*. [6](#), [7](#)

**EndNet** Réseau de neurones créé par les auteurs de l’article [3]. [1](#), [4](#), [13](#), [19](#)

**HAADF** *High-Angle Annular Dark-Field*. [22](#)

**LPS** Laboratoire de Physique des Solides. [5](#), [6](#)

**MLP** Perceptron multi-couches - *Multi Layer Perceptron*. [15](#)

**NMF** *Non-negative Matrix Factorisation*. [4](#), [5](#), [10](#), [16](#), [17](#), [20](#)

**PCA** Analyse en composantes principales - *Principal Component Analysis*. [10](#)

**RMSE** Racine de l’erreur quadratique moyenne - *Root Mean Square Error*. [15](#), [20](#)

**SAD** Distance angulaire spectrale - *Spectral Angle Distance*. [15](#), [20](#)

**STEM** Microscopie électronique en transmission à balayage - *Scanning Transmission Electron Microscopy*. [5–7](#), [9](#)

**VCA** *Vertex Component Analysis*. [4](#), [5](#), [10](#), [16–18](#), [20](#)

## Table des figures

1	Schéma d'une image hyperspectrale. . . . .	6
2	Illustration de la méthode de soustraction du fond continu. . . . .	8
3	Illustration de la disposition de spectres selon un simplexe. . . . .	11
4	Schéma du réseau DAEU. . . . .	12
5	Schéma du réseau DeepTrans-HSU [2]. . . . .	14
6	Abondances et composantes de l'échantillon étudié. . . . .	16
7	Résultat du démixage avec VCA. . . . .	17
8	Résultat du démixage avec NMF. . . . .	17
9	Résultat du démixage avec DAEU. . . . .	18
10	Résultat du démixage avec EndNet. . . . .	19
11	Résultat du démixage avec DeepTrans-HSU. . . . .	20
12	Estimation des cartes d'abondance de la lame mince de platine, cobalt et ruthénium par les différents algorithmes présentés. . . . .	21
13	<i>Endmembers</i> estimés par les différents algorithmes présentés. . . . .	21
14	Evaluation quantitative des performances des divers algorithmes de démixage. . . . .	21
15	Illustration de la variabilité des fonds continus ajustés : le choix arbitraire de la fenêtre de calcul a un grand impact sur l'estimation du fond continu. . . . .	24
16	Illustration de l'intersection du modèle de fond continu avec le signal d'intérêt : la partie de la composante est négative à partir de l'intersection entre le fond continu estimé et le spectre. . . . .	24
17	Illustration des problèmes causés par deux seuils proches : la faible distance séparant les seuils de Ni L et de Cu L force la fenêtre de calcul à se superposer à la queue du seuil Ni L et le modèle puissance ne permet pas de modéliser le fond continu. . . . .	25
18	Estimation des cartes d'abondance de la lame mince de titane, baryum et néodyme par les différents algorithmes présentés. . . . .	25
19	<i>Endmembers</i> estimés par les différents algorithmes présentés. . . . .	26



## Introduction

Dans le cadre du M1 Mathématiques et Intelligence Artificielle de l'Université Paris-Saclay, j'ai réalisé un stage de quatre mois dans le [Laboratoire de Physique des Solides \(LPS\)](#) d'Orsay, du 20 mars 2023 au 21 juillet 2023. J'ai été encadré par Nathalie Brun, chercheuse de l'équipe [Microscopie électronique en transmission à balayage - Scanning Transmission Electron Microscopy \(STEM\)](#).

Les membres de cette équipe utilisent divers microscopes électroniques afin d'étudier la nature et la composition de différents échantillons. Ces microscopes fournissent des données, sous forme de cubes, appelées images hyperspectrales, où chaque pixel correspond à un spectre comprenant plusieurs centaines de canaux. Il s'avère que certains satellites sont également en mesure d'acquérir de telles images. Or, en télédétection, le traitement des images hyperspectrales a connu récemment de nombreuses avancées. Nous pouvons donc appliquer les techniques mises au point dans ce dernier domaine aux données issues de microscopie électronique.

En particulier, retrouver les éléments chimiques, leurs proportions et leur répartition sont des informations essentielles à l'étude d'un échantillon. Ce procédé est appelé démixage spectral. Pour l'effectuer, la méthode de soustraction du fond continu est couramment utilisée. Il existe de nombreux autres algorithmes permettant d'effectuer du démixage spectral et qui, par chance, permettent de pallier les nombreux inconvénients de la méthode de soustraction du fond continu. Certains de ces algorithmes sont fondés sur le *deep learning* [1]-[4]. Bien que cela semble prometteur, il s'avère que des méthodes classiques et proposant de bonnes performances existent, notamment les algorithmes *Non-negative Matrix Factorisation* (NMF) [5] et *Vertex Component Analysis* (VCA) [6]. L'objectif de ce stage d'étudier différentes méthodes de *deep learning* et de déterminer s'il en existe une qui performerait mieux que les algorithmes classiques.

Le rapport est constitué de trois grandes parties. La première introduit le contexte du stage avec une présentation du laboratoire, les notions physiques impliquées dans le sujet et les objectifs du stage. La deuxième présente le modèle mathématique à la base de toute l'étude, ainsi que les divers modèles étudiés. Enfin, la troisième décrit les résultats obtenus au cours de l'étude.

# 1 Contexte du stage

## 1.1 Présentation du laboratoire

Le [Laboratoire de Physique des Solides \(LPS\)](#) est un laboratoire de la faculté des sciences de l'Université Paris-Saclay, associé au [Centre National de la Recherche Scientifique \(CNRS\)](#) en tant qu'unité mixte de recherche. Il est spécialisé dans l'étude de la physique de la matière condensée et est situé à Orsay. Fondé en 1959 par André Guinier, Jacques Friedel et Raimond Castaing, le [LPS](#) est actuellement dirigé par Pascale Foury-Leylekian.

Les trois axes de recherche du [LPS](#) sont les nouveaux états électroniques de la matière, la matière molle et interface physique-biologie et les phénomènes physiques aux dimensions réduites. C'est dans ce dernier axe que travaille l'équipe [STEM](#).

## 1.2 Spectroscopie EELS

La [Spectroscopie par pertes d'énergie d'électrons - Electron Energy Loss Spectroscopy \(EELS\)](#) consiste à exposer un échantillon mince à un faisceau d'électrons, puis à mesurer à l'aide d'un spectromètre magnétique la perte en énergie des électrons ayant interagi inélastiquement avec l'échantillon [7], [8]. On obtient ainsi un spectre de perte d'énergie d'électrons, ou spectre [EELS](#), comportant des seuils caractéristiques des éléments chimiques présents dans l'échantillon. Dans un microscope [STEM](#), le faisceau d'électrons est focalisé à la surface de l'échantillon sous forme d'une sonde de taille subnanométrique. Une image est obtenue en balayant cette sonde sur une zone d'intérêt à la surface de l'échantillon et en acquérant un spectre [EELS](#) pour chaque position de cette sonde. Cette image est appelée *image hyperspectrale* (parfois "image", quand il n'y a pas d'ambiguïté). Chaque pixel de l'image hyperspectrale correspond donc à un spectre [EELS](#) comportant  $L$  canaux. À la fin de l'acquisition, on obtient un cube de dimension  $n \times m \times L$  dont les deux premières dimensions ( $n$  et  $m$ ) sont spatiales tandis que la dernière ( $L$ ) est spectrale (Figure 1). En microscopie électronique, les dimensions spatiales sont de l'ordre de la centaine de pixels et la dimension spectrale est de l'ordre de 200 à 500 canaux.

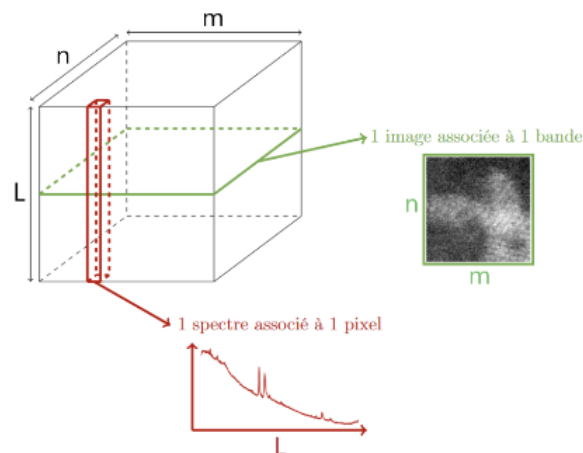


FIGURE 1 – Schéma d'une image hyperspectrale.

## 1.3 Objectifs du stage

### 1.3.1 Démélange spectral

Lorsqu'un échantillon est placé dans un microscope [STEM](#), nous connaissons en général les composants chimiques qui le composent, mais nous ne connaissons pas leur répartition spatiale à l'échelle nanométrique. La tâche consistant à retrouver ces informations à partir de l'image hyperspectrale seule s'appelle le *démélange spectral*.

L'objectif du démélange est d'obtenir les spectres [EELS](#) des éléments chimiques "purs" présents dans l'échantillon, ainsi que leurs proportions respectives au sein du matériau. Ces spectres sont appelés "composantes" (ou "*endmembers*") et les proportions respectives de ces composantes sont appelées "cartes" (mais aussi "abondances" ou "*abundances*").

### 1.3.2 Soustraction du fond continu

La première méthode de démélange dont nous allons parler est celle principalement utilisée par les chercheurs et chercheuses travaillant en microscopie électronique.

Le seuil caractéristique de perte d'énergie d'électrons d'un élément chimique est toujours superposé à un *fond continu* provenant de différents phénomènes physiques. La première étape afin de retrouver le seuil de l'élément considéré est donc de soustraire ce fond continu en ajustant un modèle empirique sur un intervalle d'énergie précédant le seuil d'intérêt, défini manuellement par l'utilisateur·rice (voir la fenêtre rouge "bkgd" de la Figure 2).

Le modèle utilisé pour l'ajustement est un modèle de loi puissance (1), couramment utilisé en physique pour modéliser les résidus de seuils caractéristiques.

$$J(E) = CE^{-r}, \quad (1)$$

où

- $J : \mathbb{R}_+^* \rightarrow \mathbb{R}_+$  est la fonction de perte d'énergie ;
- $E \in \mathbb{R}_+^*$  est une énergie en électronvolts (eV) ;
- $C \in \mathbb{R}_+$  est appelée la *constante d'échelle* ;
- $r \in \mathbb{R}_+$  est appelé l'*exposant de la pente*.

Une fois le fond continu ajusté sur le spectre et soustrait au signal total, il devient possible de mesurer la quantité de signal restant sur un intervalle d'énergie défini manuellement – lui aussi – par l'utilisateur·rice (voir la fenêtre verte "signal" de la Figure 2). Cette opération est ensuite répétée automatiquement par le logiciel *Digital Micrograph* [9] afin d'obtenir la carte de l'élément chimique considéré.

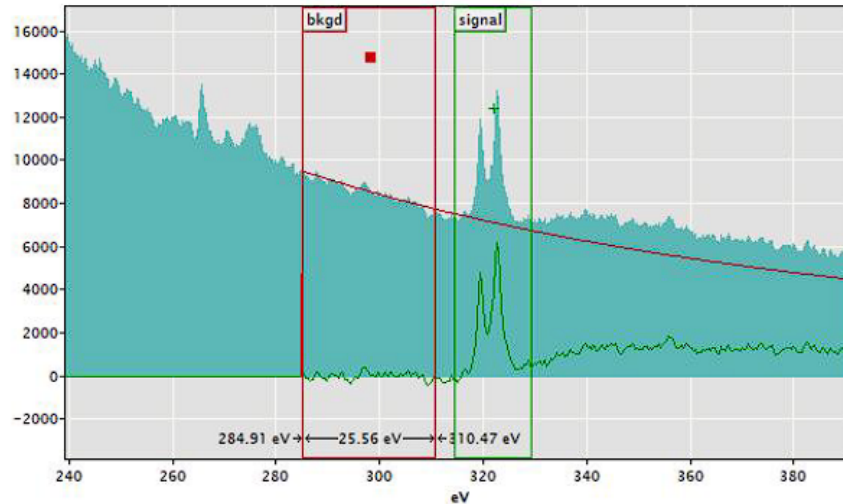


FIGURE 2 – Illustration de la méthode de soustraction du fond continu.

Comme énoncé plus haut, c'est une méthode qui est très largement utilisée dans la communauté. Cependant, elle possède de nombreux inconvénients.

Premièrement, c'est une méthode très arbitraire et utilisateur-riche - dépendante. En effet, pour chaque composante, il y a quatre paramètres à fixer à la main (les bornes des intervalles d'ajustement du fond continu et de calcul du signal restant). Bien que certaines règles empiriques soient communément admises, il n'existe pas de méthode optimale pour fixer ces quatre paramètres. De plus, l'ajustement du fond continu dépend très fortement des bornes choisies (voir Figure 15).

Ensuite, il existe des situations dans lesquelles le fond continu s'intersecte avec le signal que l'on cherche à étudier. Cette situation conduit à un signal négatif après soustraction du fond, ce qui n'est pas quelque chose interprétable physiquement (voir Figure 16).

Ce genre d'intersections parasites interviennent également lorsque les seuils de deux composants chimiques différents sont très proches. Cela force l'utilisateur-riche à superposer la fenêtre "bkgd" avec la queue résiduelle du premier seuil et le modèle de puissance (1) ne parvient plus à modéliser le fond continu correctement (voir Figure 17).

Enfin, c'est une méthode qui prend du temps. En effet, il est nécessaire de tester de nombreuses dispositions des fenêtres de calcul pour cartographier un composant chimique, puis de répéter l'opération pour chacun des éléments chimiques présents dans l'échantillon, puis de répéter ces opérations s'il s'avérait que plusieurs images hyperspectrales devaient être étudiées.

Fort heureusement, il existe d'autres méthodes de démixage spectral qui sont, elles, plus "déterministes" (au sens où leurs résultats ne varient pas d'un-e utilisateur-riche à un-e autre).

### 1.3.3 Objectifs

En télédétection satellitaire, beaucoup d'articles fleurissent à propos de l'utilisation du *deep learning* à des fins de démixage [1]-[4]. Bien que les données issues de télédétection et celle issues de microscopie électronique aient la même structure, il est nécessaire de vérifier la pertinence de l'utilisation de ces derniers modèles pour le démixage de nos données.

C'est de ce constat que vient l'objectif du stage : évaluer différents réseaux de neurones sur une tâche de démixage spectral et comparer leurs résultats à ceux de méthodes classiques. Il s'agit d'un travail qui a déjà été commencé dans l'équipe [STEM](#) [10].

## 2 Modèle et outils

### 2.1 Modèle de mélange linéaire

Durant la totalité de cette étude, nous supposons que les *endmembers* que nous recherchons sont mélangés linéairement afin de former les spectres de l'image hyperspectrale que nous observons (2). Autrement dit, chaque pixel de l'image (donc chaque spectre de l'image, voir Figure 1), est supposé être une combinaison linéaire des *endmembers*.

$$\forall (i, j) \in \llbracket 1, n \rrbracket \times \llbracket 1, m \rrbracket, \quad \mathbf{x}_{ij} = \mathbf{S}\mathbf{a}_{ij} + \boldsymbol{\varepsilon}_{ij} \quad (2)$$

Dans l'équation (2), on désigne par

- $\mathbf{x}_{ij} \in \mathbb{R}^L$  le pixel (le spectre) observé ;
- $\mathbf{a}_{ij} \in \mathbb{R}^c$  le vecteur d'abondance associé au pixel observé et  $c$  le nombre de composantes constituant l'échantillon ;
- $\mathbf{S} \in \mathcal{M}_{L,c}(\mathbb{R})$  la matrice "source" dont les colonnes  $\mathbf{s}_k \in \mathbb{R}^L$  sont les signatures spectrales des composantes ;
- $\boldsymbol{\varepsilon}_{ij} \in \mathbb{R}^L$  le vecteur représentant les erreurs de mesure et le bruit.

Certaines des méthodes fondées sur ce modèle et que nous allons présenter font l'hypothèse d'un mélange *convexe*, et non plus linéaire, sous la forme de deux nouvelles contraintes :

- [Contrainte de positivité des vecteurs d'abondance - Abundances Non-negativity Constraint \(ANC\)](#)

$$\forall (i, j, k) \in \llbracket 1, n \rrbracket \times \llbracket 1, m \rrbracket \times \llbracket 1, c \rrbracket, \quad a_{ij}^{(k)} \geq 0 ;$$

- [Contrainte de somme à un des vecteurs d'abondance - Abundances Sum-to-one Constraint \(ASC\)](#)

$$\forall (i, j) \in \llbracket 1, n \rrbracket \times \llbracket 1, m \rrbracket, \quad \sum_{k=1}^c a_{ij}^{(k)} = 1.$$

En aplatissant l'image hyperspectrale en une matrice dimension  $L \times (n \cdot m)$ , et en négligeant le bruit et les erreurs de mesure, il est possible de réécrire le modèle linéaire (2) sous la forme (3) :

$$\mathbf{X} = \mathbf{S}\mathbf{A}, \quad (3)$$

où

- $\mathbf{X}$  est une matrice de taille  $L \times (n \cdot m)$  dont les colonnes sont les pixels observés  $\mathbf{x}_i$  ;
- $\mathbf{A}$  est une matrice de taille  $c \times (n \cdot m)$  dont les colonnes sont les vecteurs d'abondance  $\mathbf{a}_i$  associés aux pixels  $\mathbf{x}_i$ .

Cet aplatissement se fait selon la bijection

$$\begin{array}{ccc} \llbracket 1, n \rrbracket \times \llbracket 1, m \rrbracket & \xrightarrow{\sim} & \llbracket 1, nm \rrbracket \\ (i, j) & \mapsto & (i-1)m + j \end{array}$$

Il est important de noter qu'en aplatisant l'image ainsi, nous perdons toutes les informations spatiales que l'image contenait initialement. Cependant, cet aplatissement n'est pas systématique, certaines méthodes avancées tiennent compte de la structure spatiale (2.3.3).

## 2.2 Les algorithmes classiques de démixage

Maintenant que le modèle est posé, voyons les deux principaux algorithmes classiques de démixage faisant appel à ce modèle et auxquels nous comparerons les résultats de nos réseaux de neurones.

### 2.2.1 NMF

La *Non-negative Matrix Factorisation* (NMF) est une décomposition matricielle [5] similaire à l'*Analyse en composantes principales - Principal Component Analysis* (PCA), mais sans contrainte d'orthogonalité des composantes. La contrainte à respecter, ici, est que les composantes doivent être positives.

L'algorithme NMF consiste en la réécriture de  $\mathbf{X}$  comme le produit  $\mathbf{X} \simeq \mathbf{H}_1 \mathbf{H}_2$ , où  $\mathbf{H}_1, \mathbf{H}_2$  sont deux matrices de dimensions respectives  $L \times \alpha$  et  $\alpha \times (n \cdot m)$  et dont leurs coefficients sont positifs. Plus le *rang de factorisation*  $\alpha$  est petit devant  $c$  et  $n \cdot m$ , plus la réduction de dimension opérée par NMF est importante. Les matrices  $\mathbf{H}_1$  et  $\mathbf{H}_2$  sont déterminées par résolution du problème d'optimisation sous contraintes suivant :

$$\min_{\mathbf{H}_1, \mathbf{H}_2 \geq 0} \|\mathbf{X} - \mathbf{H}_1 \mathbf{H}_2\|_2.$$

L'algorithme NMF décrit ici n'est qu'un parmi tant d'autres [11], il est possible de changer de norme, voire de minimiser autre chose qu'une norme, il est possible d'y ajouter nombre de contraintes, etc. Nous avons utilisé uniquement la version présentée ici dans nos comparaisons. Nous avons utilisé la version implémentée dans [12].

### 2.2.2 VCA

La *Vertex Component Analysis* (VCA) est une méthode non supervisée d'extraction des spectres des composantes de données hyperspectrales [6]. Elle a été développée spécialement pour cette utilisation. Cette méthode fait l'hypothèse de mélange convexe que nous avons énoncé plus haut. De ce fait, tous les spectres de l'image sont contenus dans un simplexe de dimension  $c - 1$  dont les  $c$  sommets sont les *endmembers* que nous cherchons.

L'algorithme VCA consiste en une procédure itérative de projection du simplexe dans la direction orthogonale au sous-espace engendré par les *endmembers* déjà déterminés. À chaque itération, le nouvel *endmember* est le point dont la projection est la plus "grande".

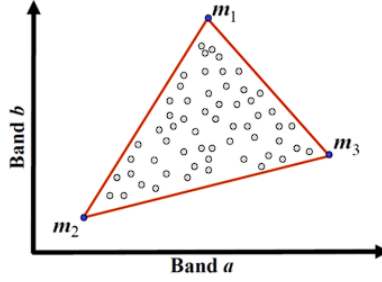


FIGURE 3 – Illustration de la disposition de spectres selon un simplexe.

Le code de cet algorithme que nous avons utilisé est celui fourni par les auteurs de [1].

## 2.3 Les auto-encodeurs

Puisque l'objectif du démixage spectral est d'extraire des spectres caractéristiques de l'image et, ce, de façon non supervisée, l'utilisation d'auto-encodeurs semble pertinente. En effet, l'objectif d'un auto-encodeur est d'apprendre une représentation de plus basse dimension d'un ensemble de données.

Plus formellement, un auto-encodeur se compose toujours de deux parties, l'*encodeur* et le *décodeur*, qui peuvent être définies comme des transitions  $\phi$  et  $\psi$ , telles que :

$$\begin{cases} \phi : \mathcal{X} \longrightarrow \mathcal{F} \\ \psi : \mathcal{F} \longrightarrow \mathcal{X} \\ \phi, \psi = \arg \min_{\phi, \psi} \|\mathbf{x} - (\psi \circ \phi)\mathbf{x}\|^2 \end{cases}$$

où  $\mathcal{X} = \mathbb{R}^L$  est l'espace dans lequel vit  $\mathbf{x}$ , et où  $\mathcal{F} = \mathbb{R}^c$  est l'espace dans lequel est stocké la représentation latente de  $\mathbf{x}$ .

### 2.3.1 DAEU

Le premier auto-encodeur auquel nous nous sommes intéressés se nomme **DAEU**. Le code associé à ce réseau a été communiqué par ses auteurs [1]. La partie encodeur de ce réseau est composée d'un total de neuf couches dont huit sont cachées. Le décodeur, quant à lui, n'est composé que d'une couche : la couche de sortie.

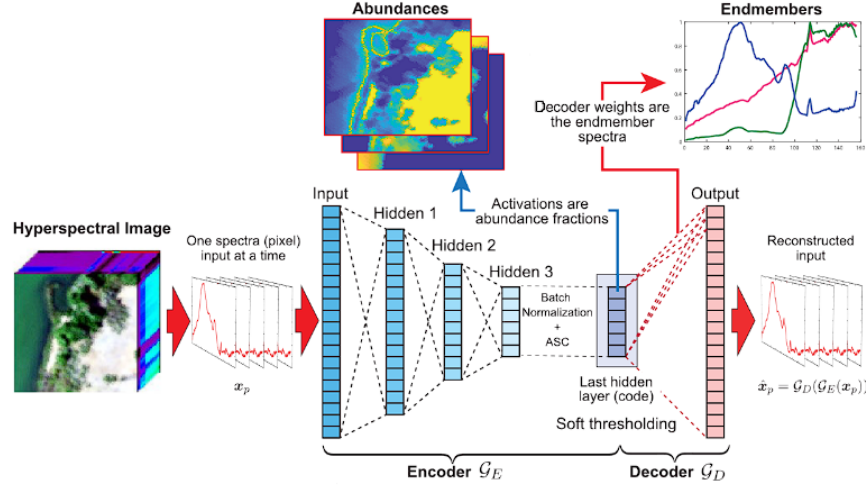


FIGURE 4 – Schéma du réseau DAEU.

### Encodage

L'encodeur de ce réseau est d'abord composé de cinq couches qui compressent les données d'entrée : la couche d'entrée  $\mathbf{b}^{(0)}$  de taille  $L$ , ainsi que de quatre couches cachées  $\mathbf{b}^{(1)}$ ,  $\mathbf{b}^{(2)}$ ,  $\mathbf{b}^{(3)}$  et  $\mathbf{b}^{(4)}$ , de tailles respectives  $9c$ ,  $6c$ ,  $3c$  et  $c$ . Ces quatre couches possèdent la même fonction d'activation  $g$  appelée LeakyReLU, qui est définie par

$$x \mapsto \begin{cases} x & \text{si } x > 0 \\ 0.1x & \text{sinon} \end{cases} \quad (4)$$

Pour  $l \in \{1, 2, 3, 4\}$ , on a

$$\mathbf{b}^{(l)} = g(\mathbf{W}^{(l)}\mathbf{b}^{(l-1)}). \quad (5)$$

L'encodeur de DAEU réduit donc la dimension d'entrée  $L$  à  $c$ , qui est le nombre d'*endmembers* que l'on cherche.

La cinquième couche cachée est une couche de *batch normalisation* : elle normalise la sortie de la couche cachée précédente  $\mathbf{b}^{(4)}$  puis applique la transformation affine  $[x \mapsto \gamma x + \beta]$ . Pour un *batch*  $\mathcal{B}$  donné de taille  $B \leq nm$  et composé des vecteurs  $\{\mathbf{b}_1^{(4)}, \dots, \mathbf{b}_B^{(4)}\}$ , on définit la couche de *batch normalisation* par

$$\forall i \in \llbracket 1, B \rrbracket, \quad \mathbf{b}_i^{(5)} = \text{BN}_{\gamma, \beta}(\mathbf{b}_i^{(4)}) := \gamma \bar{\mathbf{b}}_i^{(4)} + \beta \quad (6)$$

avec, pour tout  $i \in \llbracket 1, B \rrbracket$  et tout  $k \in \llbracket 1, c \rrbracket$ ,

$${}^{(k)}\bar{\mathbf{b}}_i^{(4)} = \frac{{}^{(k)}\mathbf{b}_i^{(4)} - {}^{(k)}\boldsymbol{\mu}_{\mathcal{B}}}{\sqrt{{}^{(k)}\boldsymbol{\sigma}_{\mathcal{B}}^2 + \varepsilon}}, \quad (7)$$

$${}^{(k)}\boldsymbol{\mu}_{\mathcal{B}} = \frac{1}{B} \sum_{i=1}^B {}^{(k)}\mathbf{b}_i^{(4)}, \quad (8)$$

$${}^{(k)}\boldsymbol{\sigma}_{\mathcal{B}}^2 = \frac{1}{B} \sum_{i=1}^B \left( {}^{(k)}\mathbf{b}_i^{(4)} - {}^{(k)}\boldsymbol{\mu}_{\mathcal{B}} \right)^2, \quad (9)$$



où  $\cdot^{(k)}$  est la notation de la  $k$ -ième composante du vecteur considéré.

Ici, la grandeur  $\varepsilon$  est un réel très petit et arbitraire. Elle est utilisée pour stabiliser le calcul numérique de la fraction. Les paramètres  $\gamma$  et  $\beta$  sont, quant à eux, appris par le réseau de neurones. La couche de *batch normalisation* possède quelques avantages : elle permet d'accélérer l'apprentissage du réseau et réduit la présence de perturbations aléatoires dans les distributions des données à travers les couches du réseau.

La sixième couche cachée du réseau exploite le caractère parcimonieux des cartes d'abondance et mettant à zéro les noeuds dont les valeurs sont sous un certain seuil  $\alpha \in \mathbb{R}^c$  :

$$\mathbf{b}^{(6)} := \max \{0; \mathbf{b}^{(5)} - \alpha\} \quad (10)$$

Tout comme les coefficients  $\gamma$  et  $\beta$ ,  $\alpha$  est un paramètre appris par le réseau.

La septième couche cachée assure l'ASC via la transformation

$$\forall i \in \llbracket 1, c \rrbracket, \quad \mathbf{b}_i^{(7)} := \frac{\mathbf{b}_i^{(6)}}{\sum_{j=1}^c \mathbf{b}_j^{(6)}}. \quad (11)$$

Enfin, la huitième et dernière couche cachée de l'encodeur est une couche de régularisation. Elle applique un bruit multiplicatif gaussien afin de dégrader les entrées et prévenir le sur-apprentissage du réseau.

### Décodage

Le décodeur n'est constitué que de la couche de sortie du réseau. Elle est définie simplement de la façon suivante :

$$\hat{\mathbf{x}} := \mathbf{W}^{(9)} \mathbf{b}^{(8)} \quad (12)$$

En comparant cette dernière équation avec l'équation (2), il vient que l'on doit avoir  $\mathbf{W}^{(9)} = \mathbf{S}$  et  $\mathbf{b}^{(8)} = \mathbf{a}$ . Autrement dit, à la fin de l'entraînement du réseau, les *endmembers* recherchés seront les poids de la dernière couche et les cartes pourront être retrouvées dans les activations de la dernière couche *cachée*.

### 2.3.2 EndNet

Le deuxième auto-encodeur qui nous a intéressé est appelé **EndNet** et a été créé par les auteurs de [3]. L'implémentation de ce modèle dont on dispose a été réalisée par les auteurs de [1]. C'est un auto-encodeur dont l'encodeur est constitué de sept couches et dont le décodeur n'est constitué que d'une couche.

#### Encodage

Après la couche d'entrée de taille  $L$ , la première couche cachée est une couche où les données sont bruitées afin d'améliorer les capacité de généralisation de l'auto-encodeur. Le bruit est limité à n'affecter au maximum que 40% des données

La deuxième couche cachée s'avère être déjà de taille  $c$ . La variable latente est  $\mathbf{h} = \mathbf{W}^{(e)} \cdot_c \tilde{\mathbf{x}}$ , où  $\cdot_c$  est défini par

$$\mathbf{x} \cdot_c \mathbf{y} := C(\mathbf{x}, \mathbf{y}), \quad C(\mathbf{x}, \mathbf{y}) := 1 - \frac{\text{SAD}(\mathbf{x}, \mathbf{y})}{\pi}, \quad \text{SAD}(\mathbf{x}, \mathbf{y}) := \arccos \left( \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\|_2 \|\mathbf{y}\|_2} \right). \quad (13)$$

Ensuite, vient une couche de *batch normalisation* similaire à (6) suivie d'une activation ReLU et d'une couche de *dropout* :

$$\mathbf{z} := r \star \text{ReLU}(\text{BN}(\mathbf{h})), \quad r \sim \mathcal{B}(p), \quad p \in ]0, 1]^c, \quad (14)$$

où  $\star$  désigne le produit terme à terme.

Enfin, seules les deux plus hautes activations sont sélectionnées (les autres étant mises à zéro) et une régularisation  $\ell^1$  est appliquée pour garantir **ASC** :

$$\mathbf{y} := \frac{\mathbf{z}^*}{\|\mathbf{z}^*\|_1 + \varepsilon}, \quad (15)$$

où  $\mathbf{z}^*$  désigne le vecteur dont seules les deux plus hautes activations ont été sélectionnées.

### Décodage

Comme pour **DAEU**, le décodeur n'est constitué que de la couche de sortie du réseau. On retrouve également les *endmembers* comme étant les poids de la dernière couche et les cartes peuvent être obtenues grâce aux activations de l'avant-dernière couche.

### 2.3.3 DeepTrans-HSU

Dans les réseaux de neurones précédents, les informations spatiales contenues dans l'image n'étaient pas analysées. Ce nouveau, et dernier, réseau prend en compte ces informations grâce à la présence de couches convolutionnelles. De plus, il s'agit d'un réseau de type *transformer*, que nous pensons être pertinent quant à l'amélioration de la qualité du démixage. Ce dernier auto-encodeur est sensiblement plus compliqué que les deux précédents, comme la figure 5 le suggère. Il a été mis au point par les auteurs de [2] et le code a été rendu disponible par ces mêmes auteurs.

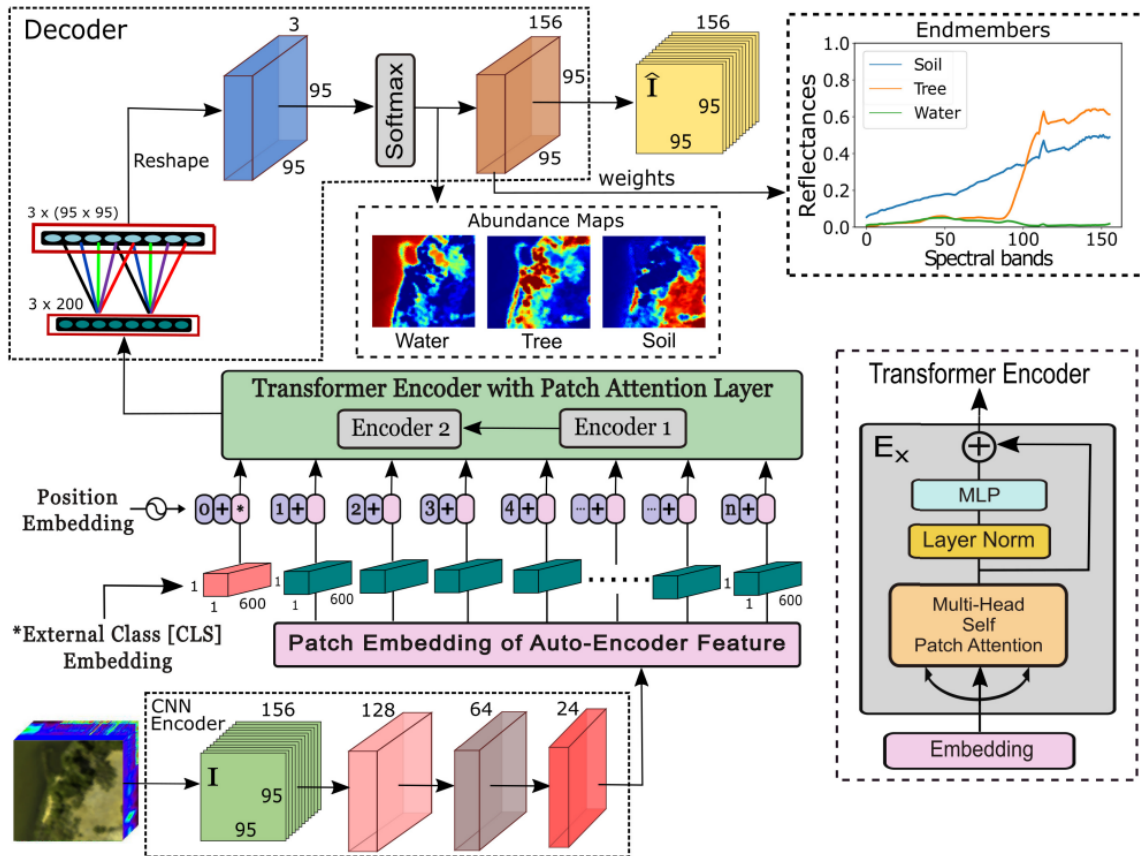


FIGURE 5 – Schéma du réseau DeepTrans-HSU [2].

L'entrée de ce réseau de neurones est une image hyperspectrale notée  $\mathbf{I}$ . Cette image hyperspectrale traverse trois couches de convolution pour que leurs caractéristiques principales soient représentées avec un plus petit nombre de canaux. La sortie de cette succession de couches de convolution est divisée en *patches*. Ces *patches* sont redimensionnés en vecteurs et passé à travers le transformeur, qui est constitué d'une attention multi-têtes et d'un [Perceptron multi-couches - Multi Layer Perceptron \(MLP\)](#).

La sortie de ce transformeur est agrandie et redimensionnée jusqu'à correspondre à la taille des cartes d'abondances. Une couche de convolution est utilisée pour réduire le bruit généré par cet agrandissement. La fonction d'activation *softmax* est alors utilisée pour obtenir les cartes d'abondances finales. Enfin, le décodeur reconstruit l'image hyperspectrale  $\hat{\mathbf{I}}$  en utilisant une unique et dernière couche de convolution, dont les poids sont les *endmembers*.

## 2.4 Évaluation

Nous avons enfin passé en revue toutes les méthodes de démixage qui nous ont intéressées pendant ce stage. Nous souhaitons désormais mesurer les performances de ces algorithmes afin de les comparer. Il existe deux grandes métriques permettant d'évaluer les performances d'un algorithme de démixage : la [Racine de l'erreur quadratique moyenne - Root Mean Square Error \(RMSE\)](#) et la [Distance angulaire spectrale - Spectral Angle Distance \(SAD\)](#).

### 2.4.1 RMSE

Rappelons sa définition :

$$\forall \mathbf{P}, \mathbf{Q} \in \mathcal{M}_{n,m}(\mathbb{R}), \quad \text{RMSE}(\mathbf{P}, \mathbf{Q}) := \sqrt{\frac{1}{n} \sum_{i=1}^n \sum_{j=1}^m (p_{ij} - q_{ij})^2}. \quad (16)$$

En pratique, cette métrique est utilisée pour comparer les cartes d'abondances estimées et réelles entre elles. Pour comparer les *endmembers* entre eux, on utilise plutôt la [SAD](#).

### 2.4.2 SAD

Cette métrique calcule l'angle séparant deux vecteurs.

$$\forall \mathbf{y}, \hat{\mathbf{y}} \in \mathbb{R}^n, \quad \text{SAD}(\mathbf{y}, \hat{\mathbf{y}}) := \arccos \left( \frac{\mathbf{y}^T \hat{\mathbf{y}}}{\|\mathbf{y}\|_2 \|\hat{\mathbf{y}}\|_2} \right). \quad (17)$$

### 2.4.3 Évaluation qualitative

Bien que nous ayons défini des métriques pour l'évaluation de la qualité du démixage, la principale évaluation est visuelle et qualitative. C'est en un rapide coup d'oeil que la-le physicien-ne est capable d'analyser le résultat d'un démixage. La principale comparaison des algorithmes de démixage se fera donc de manière qualitative.

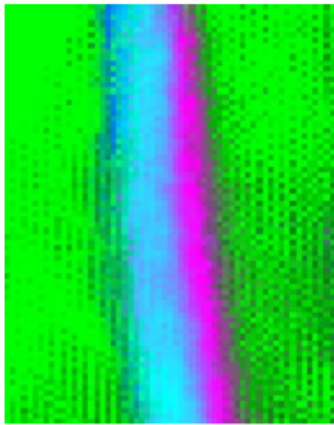
La qualité d'un démixage se faisant en comparant les spectres et abondances estimés à une "vérité terrain", cela implique que nous ayons sous la main de telles informations. Cependant, puisque nous étudions un

échantillon réel, nous n'avons pas de telle "vérité terrain". Nous avons donc appliqué VCA à nos données et avons considéré le résultat comme la "vérité terrain" que nous essaierons d'approcher via les autres méthodes.

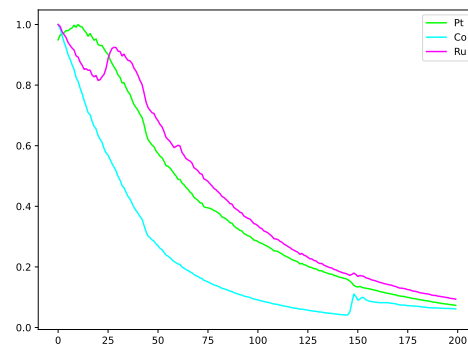
### 3 Résultats

#### 3.1 Présentation des données

Nous ne présenterons ici que les résultats obtenus sur une seule image hyperspectrale. Cet échantillon est une multi-couche métallique de platine (Pt), de cobalt (Co) et de ruthénium (Ru), étudiée dans [13].



(a) Représentation en fausses couleurs (Vert : Pt, Bleu : Co, Rose : Ru) de l'image hyperspectrale. L'échantillon observé est de dimension  $9 \times 7,2$  nanomètres.



(b) Composantes de platine, cobalt et ruthénium présentes dans l'échantillon (a).

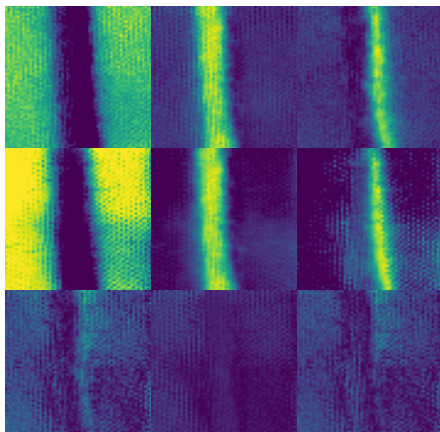
FIGURE 6 – Abondances et composantes de l'échantillon étudié.

#### 3.2 NMF et VCA

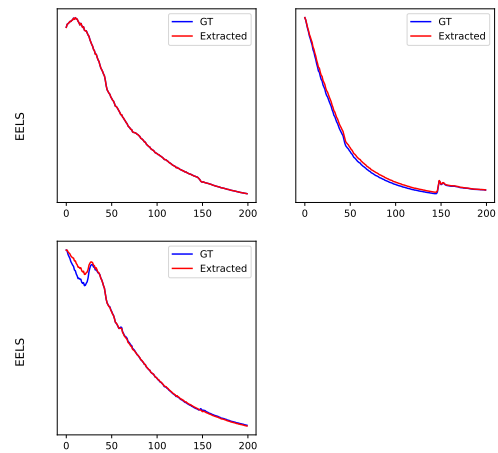
Commençons cette section par la présentation des résultats obtenus par nos méthodes classiques, NMF et VCA.

On remarque que les résultats de VCA (Figure 7) ne correspondent pas exactement aux résultats choisis pour représenter la vérité terrain, c'est parce que l'algorithme VCA comporte une part d'aléatoire.

Les résultats de NMF (Figure 8), bien que différents, sont intéressants. En effet, on sait par un autre type d'analyse qu'il y a effectivement une présence de platine résiduel dans la couche de cobalt. D'autre part, la composante extraite de ruthénium ressemble à un spectre de référence [14]. Le démixage effectué par NMF est donc satisfaisant. Il s'avère d'ailleurs que c'est une méthode fréquemment utilisée en spectroscopie.

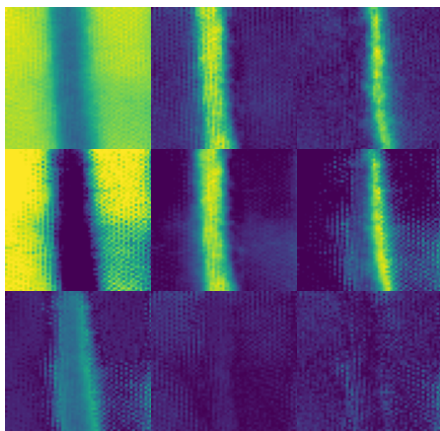


(a) Première ligne : cartes estimées par VCA. Deuxième ligne : cartes "vérité terrain". Troisième ligne : valeur absolue de la différence des deux lignes précédentes.

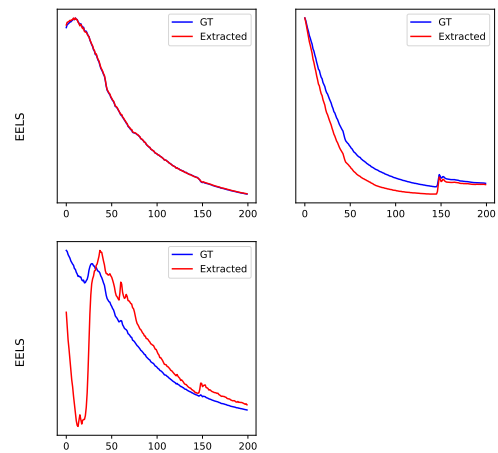


(b) En rouge, les spectres estimés par VCA. En bleu, les spectres "vérité terrain".

FIGURE 7 – Résultat du démixage avec VCA.



(c) Première ligne : cartes estimées par NMF. Deuxième ligne : cartes "vérité terrain". Troisième ligne : valeur absolue de la différence des deux lignes précédentes.



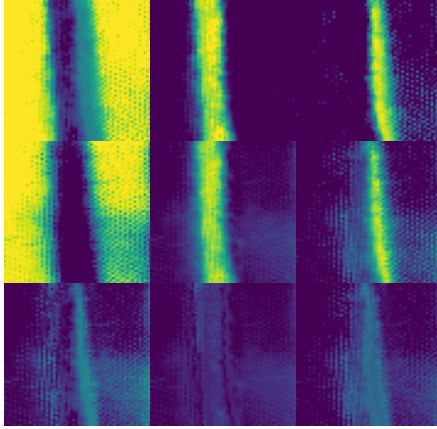
(d) En rouge, les spectres estimés par NMF. En bleu, les spectres "vérité terrain".

FIGURE 8 – Résultat du démixage avec NMF.

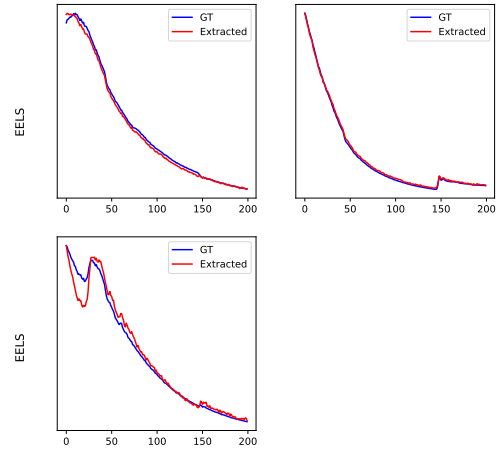
### 3.3 DAEU

Bien que le code associé à cette méthode ait été fourni par les auteurs de [1], il différait de l'article. Après avoir retrouvé les résultats annoncés dans l'article (pour de valeurs d'hyperparamètres différentes), nous avons entraîné ce réseau de neurones sur notre échantillon. La fonction de perte de ce réseau est :

$$\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}}) = \text{SAD}(\mathbf{x}, \hat{\mathbf{x}}).$$



(a) Première ligne : cartes estimées par DAEU. Deuxième ligne : cartes "vérité terrain". Troisième ligne : valeur absolue de la différence des deux lignes précédentes.



(b) En rouge, les spectres estimés par DAEU. En bleu, les spectres "vérité terrain".

FIGURE 9 – Résultat du démixage avec DAEU.

Bien que nous n'avions pas de réelles attentes à propos de ce réseau, nous sommes satisfait de ses résultats (voir Figure 9). Cependant, nous pouvons constater qu'il ne performe pas mieux que VCA, que ce soit en estimation des cartes ou en extraction des *endmembers*.

### 3.4 EndNet

Nous avons choisi de tester ce réseau car sa fonction de perte semblait intéressante :

$$\begin{aligned} \mathcal{L} = & \frac{\lambda_0}{2} \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 + \lambda_1 D_{\text{KL}}(1.0 \| C(\mathbf{x}, \hat{\mathbf{x}})) + \lambda_2 \|\mathbf{z}\|_1 \\ & + \lambda_3 \|\mathbf{W}^{(\mathbf{e})}\|_2 + \lambda_4 \|\mathbf{W}^{(\mathbf{d})}\|_2 + \lambda_5 \|\boldsymbol{\rho}\|_2, \end{aligned} \quad (18)$$

où

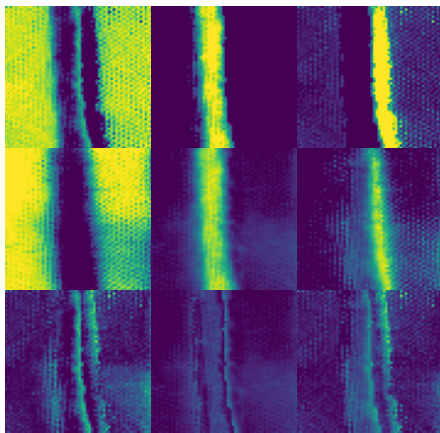
- $D_{KL}$  est la divergence de Kullback-Leibler. Ce terme a pour objectif de minimiser la dissimilarité entre les distributions des scores SAD des échantillons d'entrée et celles des échantillons reconstruits.
- $\mathbf{z}$  est le vecteur que l'on trouve en sortie de l'activation ReLU. Ce terme permet de pénaliser les couches qui génèrent une réponse constante pour des valeurs d'entrée différentes.
- $\mathbf{W}^{(e)}$  et  $\mathbf{W}^{(d)}$  sont respectivement les poids de l'encodeur et du décodeur.
- $\rho$  est un paramètre de translation qui apparaît au moment de la batch-normalisation : les données sont centrées-réduites, puis translatées de  $\rho$ .

Cependant, nous avons dû faire face à de nombreux problèmes vis-à-vis de l'article [3]. En effet, ce dernier manquait de précisions : plein de termes n'étaient pas expliqués, et d'autres pas explicités. Par exemple, la valeur de nombreux hyperparamètres nécessaire à l'obtention des résultats présentés n'était pas donnée.

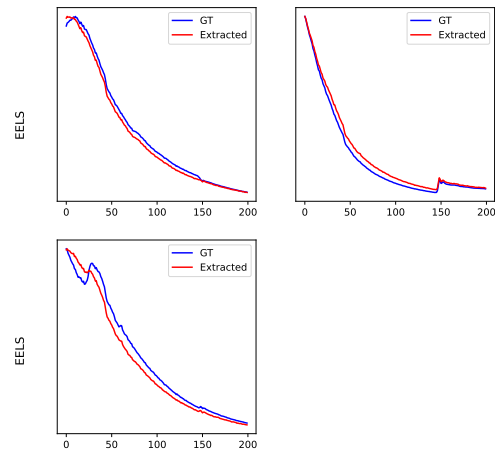
De plus, le code fourni n'a pas été implémenté par les auteurs de [3], mais par ceux de [1]. Si bien que, nous qui pensions travailler avec la perte (18), travaillions en réalité avec

$$\mathcal{L} = \frac{\lambda_0}{2} \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 + \lambda_1 D_{KL}(1.0 \| C(\mathbf{x}, \hat{\mathbf{x}})).$$

Après avoir, tant bien que mal, retrouvé les résultats présentés dans [3], nous avons essayé ce dernier réseau sur nos données.



(a) Première ligne : cartes estimées par [EndNet](#).  
Deuxième ligne : cartes "vérité terrain". Troisième ligne : valeur absolue de la différence des deux lignes précédentes.



(b) En rouge, les spectres estimés par [EndNet](#).  
En bleu, les spectres "vérité terrain".

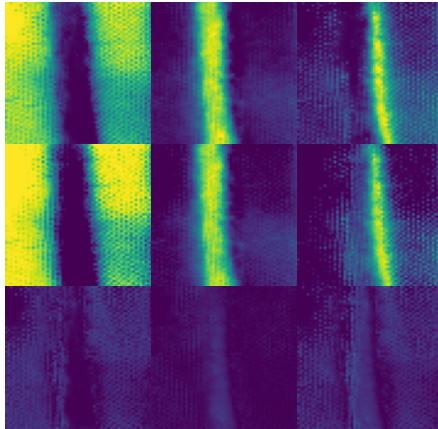
FIGURE 10 – Résultat du démixage avec [EndNet](#).

Nos attentes étaient assez élevées, nous pensions que [EndNet](#) performerai mieux que [DAEU](#) au vu de sa fonction de perte. Ce n'est que bien plus tard que nous nous sommes rendu compte que la fonction de perte n'était pas celle que nous pensions, ce qui pourrait expliquer la mauvaise qualité des résultats (Figure 10) produits par [EndNet](#). Les hyperparamètres  $\lambda_0$  et  $\lambda_1$  ont été respectivement fixés à  $10^{-2}$  et  $10^2$ .

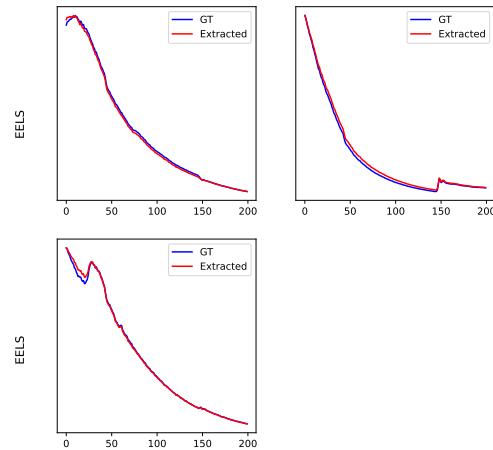


### 3.5 DeepTrans-HSU

Jusqu'à présent, les résultats de démixage obtenus par *deep learning* étaient loin d'être convaincants. Nous plaçons nos dernier espoirs en **DeepTrans-HSU** et sa structure de *transformer*. En effet, certains seuils de perte d'énergie d'électrons sont très peu discernables du fond continu (comme le platine), si bien que nous pensions qu'un peu d'attention serait tout ce dont nous aurions besoin pour un démixage réussi.



(a) Première ligne : cartes estimées par **DeepTrans-HSU**. Deuxième ligne : cartes "vérité terrain". Troisième ligne : valeur absolue de la différence des deux lignes précédentes.



(b) En rouge, les spectres estimés par **DeepTrans-HSU**. En bleu, les spectres "vérité terrain".

FIGURE 11 – Résultat du démixage avec **DeepTrans-HSU**.

Cette fois, et conformément à nos attentes, le réseau a une performance supérieure à celle des deux précédents (Figure 11). De plus, il performe également mieux que **VCA**, aussi bien en estimation des abondances qu'en extraction des composantes. Cependant, son entraînement dure cinq minutes, alors que **VCA** nous renvoie de bons résultats en un temps de l'ordre de la seconde.

Ces résultats sont encourageants, mais des tests complémentaires seraient nécessaires pour confirmer l'intérêt de ce réseau.

### 3.6 En résumé

Dans cette sous-section, nous présentons un récapitulatif des résultats de démixage (Figure 12 et 13), ainsi qu'un tableau indiquant les scores **RMSE** et **SAD** obtenus par les différents algorithmes (Figure 14).

Le tableau des scores semble indiquer que **NMF** est la méthode la plus performante lorsqu'il s'agit de retrouver les abondances, alors que **DeepTrans-HSU** performe mieux en extraction des *endmembers*.



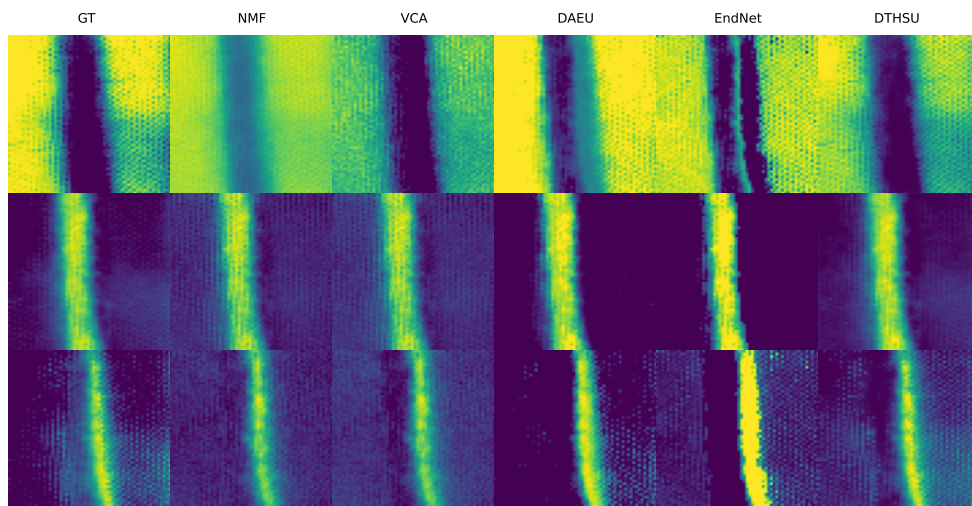


FIGURE 12 – Estimation des cartes d’abondance de la lame mince de platine, cobalt et ruthénium par les différents algorithmes présentés.

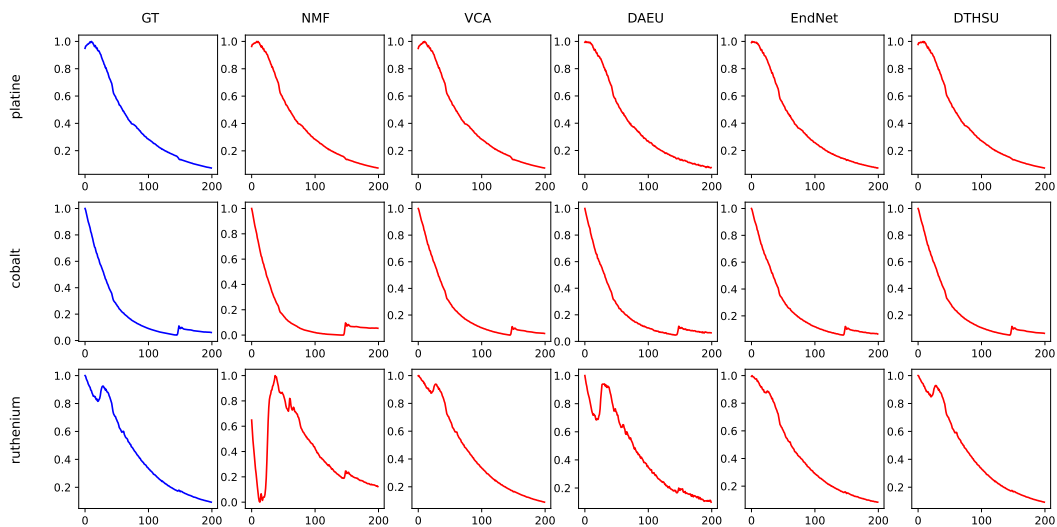


FIGURE 13 – *Endmembers* estimés par les différents algorithmes présentés.

		Algorithmes				
Métriques		NMF	VCA	DAEU	EndNet	DTHSU
	RMSE	0,138497 ± 0,000002	0,131961 ± 0,005846	0,179781 ± 0,040602	0,184659 ± 0,020384	0,142637 ± 0,011953
	SAD	0,174623 ± 0,000002	0,241456 ± 0,016819	0,169001 ± 0,026765	0,254404 ± 0,001012	0,028211 ± 0,008790

FIGURE 14 – Evaluation quantitative des performances des divers algorithmes de démixage.

## Conclusion

Durant ce stage, nous avons testé diverses méthodes de démixage spectral faisant appel au *deep learning*. De nombreux soucis de reproductibilité des divers articles nous ont énormément ralenti dans notre démarche.

De ce que nous avons pu voir, nous ne pouvons pas conclure quant à la pertinence de l'utilisation de réseaux de neurones pour réaliser des tâches de démixage spectral. En effet, bien que [DeepTrans-HSU](#) offre de bons résultats parfois, cela n'est pas toujours le cas (Figures 18 et 19). De plus, ce dernier est bien plus lent que les méthodes classiques et il est nécessaire de l'entraîner à chaque nouvelle image hyperspectrale.

Cependant, nous pensons qu'il existe des solutions à ces problèmes que nous n'avons pas eu le temps de mettre en place. Par exemple, il est possible d'envisager d'utiliser un auto-encodeur dont l'encodeur aurait des poids fixés et dont seuls les poids du décodeur seraient à optimiser. Cela permettrait, par exemple, de limiter grandement les temps de calcul. Il est également possible d'envisager la création d'un auto-encodeur entraîné de telle sorte à être bon – en moyenne – sur les images hyperspectrales issues de microscopie électronique, ou encore de coupler les informations apportées par une image hyperspectrale avec celles apportée par de l'imagerie [High-Angle Annular Dark-Field \(HAADF\)](#) [15] de cette même image. Tout espoir n'est donc pas perdu, et il s'agit alors d'utiliser ce que nous a appris cette série d'essais sur des réseaux existants pour construire notre propre réseau.

Les travaux effectués pendant ce stage ont donné lieu à deux présentations orales : la première lors de la *Première Journée du Deep Learning pour la Science* et la seconde lors du *Congrès Junior Pluridisciplinaire* de l'Université Paris-Saclay.

## Bibliographie

- [1] B. PALSSON, J. SIGURDSSON, J. R. SVEINSSON et M. O. ULFARSSON, « Hyperspectral Unmixing Using a Neural Network Autoencoder », *IEEE Access*, t. 6, p. 25 646-25 656, 2018. doi : [10.1109/ACCESS.2018.2818280](https://doi.org/10.1109/ACCESS.2018.2818280).
- [2] P. GHOSH, S. K. ROY, B. KOIRALA, B. RASTI et P. SCHEUNDERS, « Hyperspectral Unmixing using Transformer Network », *IEEE Transaction on Geoscience and Remote Sensing*, t. 60, 2022.
- [3] S. OZKAN, B. KAYA et G. B. AKAR, « EndNet : Sparse AutoEncoder Network for Endmember Extraction and Hyperspectral Unmixing », *IEEE Transactions on Geoscience and Remote Sensing*, t. 57, n° 1, p. 482-496, 2019. doi : [10.1109/TGRS.2018.2856929](https://doi.org/10.1109/TGRS.2018.2856929).
- [4] B. PALSSON, J. R. SVEINSSON et M. O. ULFARSSON, « Blind Hyperspectral Unmixing Using Autoencoders : A Critical Comparison », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, t. 15, p. 1340-1372, 2022. doi : [10.1109/JSTARS.2021.3140154](https://doi.org/10.1109/JSTARS.2021.3140154).
- [5] L. MIAO et H. QI, « Endmember Extraction From Highly Mixed Data Using Minimum Volume Constrained Nonnegative Matrix Factorization », *IEEE Transactions on Geoscience and Remote Sensing*, t. 45, n° 3, p. 765-777, 2007. doi : [10.1109/TGRS.2006.888466](https://doi.org/10.1109/TGRS.2006.888466).
- [6] J. NASCIMENTO et J. DIAS, « Vertex component analysis : a fast algorithm to unmix hyperspectral data », *IEEE Transactions on Geoscience and Remote Sensing*, t. 43, n° 4, p. 898-910, 2005. doi : [10.1109/TGRS.2005.844293](https://doi.org/10.1109/TGRS.2005.844293).
- [7] M. KOCIAK, O. STÉPHAN, M. G. WALLS, M. TENCÉ et C. COLLIEX, « Spatially Resolved EELS : The Spectrum-Imaging Technique and Its Applications », in *Scanning Transmission Electron Microscopy : Imaging and Analysis*, S. J. PENNYCOOK et P. D. NELLIST, éd. New York, NY : Springer New York, 2011, p. 163-205, ISBN : 978-1-4419-7200-2. doi : [10.1007/978-1-4419-7200-2\\_4](https://doi.org/10.1007/978-1-4419-7200-2_4). adresse : [https://doi.org/10.1007/978-1-4419-7200-2\\_4](https://doi.org/10.1007/978-1-4419-7200-2_4).
- [8] GATAN. « What is EELS ? » (2016), adresse : <https://eels.info/about/overview>.
- [9] GATAN, *Digital Micrograph*, version 3.5.3. adresse : <https://www.gatan.com/products/tem-analysis/gatan-microscopy-suite-software>.
- [10] N. BRUN, G. LAMBERT et L. BOCHER, « Deep Learning for EELS hyperspectral images unmixing using Autoencoders », *Journal of Spectral Imaging*, 2023.
- [11] X.-R. FENG, H.-C. LI, R. WANG, Q. DU, X. JIA et A. PLAZA, « Hyperspectral Unmixing Based on Nonnegative Matrix Factorization : A Comprehensive Review », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, t. 15, p. 4414-4436, 2022. doi : [10.1109/JSTARS.2022.3175257](https://doi.org/10.1109/JSTARS.2022.3175257).
- [12] SCIKIT-LEARN. « Non-negative Matrix Factorization ». (2007 - 2023), adresse : <https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.NMF.html>.
- [13] W. LEGRAND, D. MACCARIELLO, F. AJEJAS et al., « Room-temperature stabilization of antiferromagnetic skyrmions in synthetic antiferromagnets », *Nature Materials*, t. 19, p. 1-9, jan. 2020. doi : [10.1038/s41563-019-0468-3](https://doi.org/10.1038/s41563-019-0468-3).
- [14] C. UNIVERSITY. « Ruthenium Bulk  $M_{2,3}$  and  $M_{4,5}$  ». (2023), adresse : <https://muller.research.engineering.cornell.edu/spectra/ruthenium-bulk-m23-and-m45/>.
- [15] WIKIPEDIA. « Annular dark-field imaging ». (2023), adresse : [https://en.wikipedia.org/wiki/Annular\\_dark-field\\_imaging#HAADF](https://en.wikipedia.org/wiki/Annular_dark-field_imaging#HAADF).

## Appendices

### A. Cas pathologiques de la méthode de soustraction du fond continu

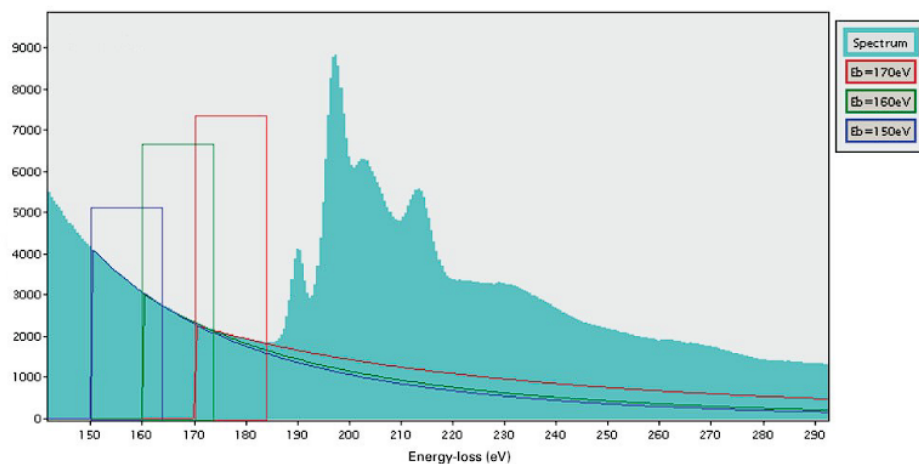


FIGURE 15 – Illustration de la variabilité des fonds continus ajustés : le choix arbitraire de la fenêtre de calcul a un grand impact sur l'estimation du fond continu.

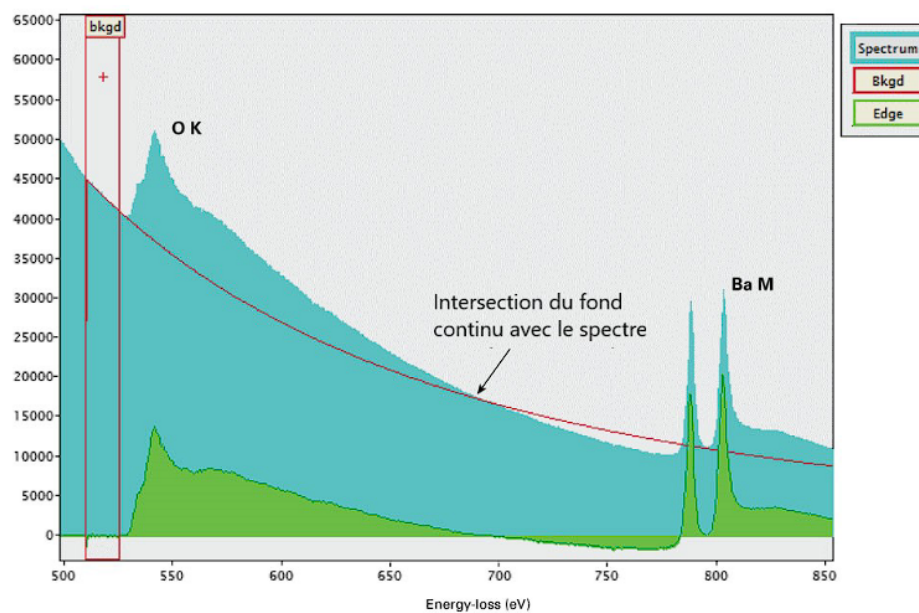


FIGURE 16 – Illustration de l'intersection du modèle de fond continu avec le signal d'intérêt : la partie de la composante est négative à partir de l'intersection entre le fond continu estimé et le spectre.

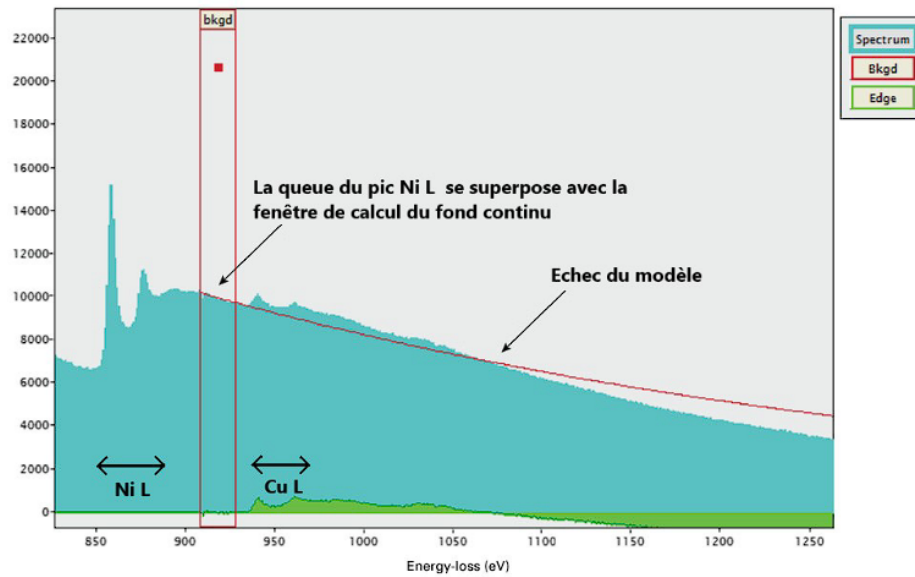


FIGURE 17 – Illustration des problèmes causés par deux seuils proches : la faible distance séparant les seuils de Ni L et de Cu L force la fenêtre de calcul à se superposer à la queue du seuil Ni L et le modèle puissance ne permet pas de modéliser le fond continu.

## B. Résultats sur un autre échantillon

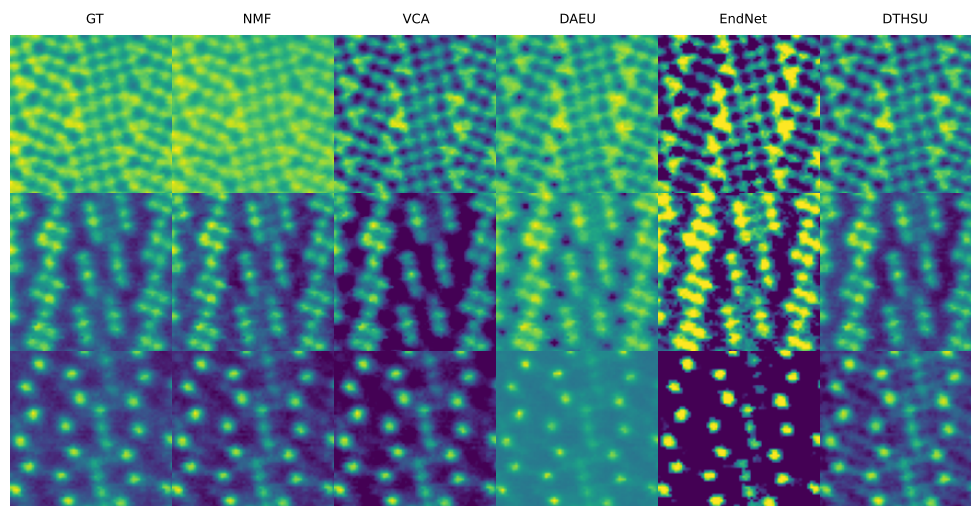


FIGURE 18 – Estimation des cartes d'abondance de la lame mince de titane, baryum et néodyme par les différents algorithmes présentés.

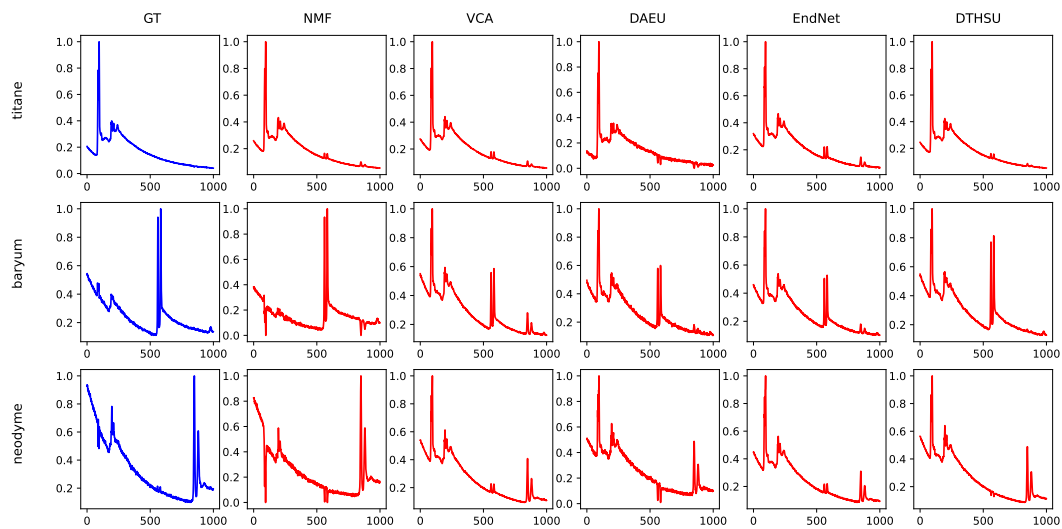


FIGURE 19 – *Endmembers* estimés par les différents algorithmes présentés.