

TABLE 3.4 – Taux de détection sur l'ensemble des patches de la base de test à partir du modèle de normalité défini à partir des curvelets

<i>NA</i>	3.31% ± 0.02%	<i>NA</i>	3.66% ± 0.02%	0.9% ± 0.1%	3.58% ± 0.01%
<i>NA</i>	1.94% ± 0.01%	<i>NA</i>	2.50% ± 0.04%	0.4% ± 0.1%	2.57% ± 0.01%
<i>NA</i>	2.91% ± 0.02%	<i>NA</i>	2.95% ± 0.02%	84.3% ± 0.4%	2.56% ± 0.01%
<i>NA</i>	1.33% ± 0.01%	<i>NA</i>	1.84% ± 0.01%	86.6% ± 1.4%	1.57% ± 0.02%
23.3% ± 0.3%	3.56% ± 0.03%	4.11% ± 0.2%	3.01% ± 0.03%	56.1% ± 0.6%	3.12% ± 0.01%
20.0% ± 0.3%	1.48% ± 0.01%	2.1% ± 1%	1.99% ± 0.03%	50.1% ± 1.1%	2.24% ± 0.02%
61.4% ± 0.1%	3.88% ± 0.03%	39.1% ± 0.3%	2.89% ± 0.02%	46.6% ± 0.4%	2.83% ± 0.01%
65.4% ± 0.6%	1.85% ± 0.02%	38.9% ± 0.9%	1.98% ± 0.02%	47.6% ± 0.9%	1.77% ± 0.01%
29.1% ± 0.3%	3.95% ± 0.03%	31.1% ± 0%	2.76% ± 0.04%	40.2% ± 0.6%	1.80% ± 0.01%
30.9% ± 0.7%	1.74% ± 0.03%	30.1% ± 0.2%	1.72% ± 0.02%	40.0% ± 0.3%	1.23% ± 0.02%
56% ± 0.3%	4.24% ± 0.01%	51.1% ± 0.3%	3.08% ± 0.02%	81.6% ± 0.2%	1.48% ± 0.04%
56.7% ± 0.7%	1.62% ± 0.02%	53.7% ± 0.4%	1.81% ± 0.05%	81.9% ± 0.2%	1.06% ± 0.03%
53.7% ± 0.2%	3.85% ± 0.01%	45.4% ± 0%	2.88% ± 0.01%	59.5% ± 0.3%	2.49% ± 0.02%
52.8% ± 0.6%	1.32% ± 0.02%	46.2% ± 0.5%	1.89% ± 0.02%	61.5% ± 0.2%	1.61% ± 0.04%
19.4% ± 0.7%	4.06% ± 0.01%	46.9% ± 0.5%	2.62% ± 0.01%	59.8% ± 0%	2.67% ± 0.02%
21.2% ± 1.2%	1.75% ± 0.03%	50.8% ± 0.8%	1.62% ± 0.01%	62.3% ± 0.5%	1.60% ± 0%
3.2% ± 0.4%	4.30% ± 0.03%	18.4% ± 0.4%	3.19% ± 0%	39.7% ± 0.4%	2.74% ± 0.01%
3.4% ± 0.3%	1.82% ± 0.02%	15.2% ± 0.1%	2.07% ± 0.02%	37.7% ± 1%	1.79% ± 0.01%
<i>NA</i>	4.23% ± 0%	<i>NA</i>	3.21% ± 0%	19.4% ± 1.1%	2.42% ± 0.01%
<i>NA</i>	1.82% ± 0.02%	<i>NA</i>	2.00% ± 0.01%	17.2 ± 1.7%	1.79% ± 0.01%
23.7% ± 0.7%	3.80% ± 0%	<i>NA</i>	2.91% ± 0.02%	31.3% ± 0.8%	2.04% ± 0%
36.6% ± 1.3%	1.45% ± 0.01%	<i>NA</i>	1.73% ± 0.04%	29.4% ± 0.7%	1.47% ± 0.02%
4.8% ± 0.4%	3.86% ± 0.02%	2.9% ± 0.8%	2.10% ± 0.01%	9.2% ± 0.3%	1.88% ± 0.03%
0.4% ± 0.3%	1.76% ± 0.02%	3.0% ± 1.4%	1.60% ± 0.01%	9.5% ± 0.4%	1.30% ± 0.01%
5.5% ± 0.2%	3.81% ± 0.02%	19.7% ± 0.4%	2.08% ± 0.02%	16% ± 1.2%	1.91% ± 0.01%
7.8% ± 0.4%	1.22% ± 0.02%	18.3% ± 1.2%	1.45% ± 0.04%	24% ± 1.5%	1.37% ± 0.04%
<i>NA</i>	3.71% ± 0.03%	0.5% ± 0.1%	2.37% ± 0.01%	71.2% ± 0.6%	2.22% ± 0%
<i>NA</i>	1.11% ± 0.01%	0%	1.54% ± 0.01%	61.3% ± 2.2%	1.47% ± 0.01%
<i>NA</i>	3.74% ± 0.02%	56% ± 0.5%	2.36% ± 0.03%	56.1% ± 0.4%	2.34% ± 0.01%
<i>NA</i>	1.41% ± 0.01%	57.5% ± 0.3%	1.54% ± 0.02%	57.8% ± 0.3%	1.68% ± 0.01%
16.2% ± 1%	4.01% ± 0.03%	46.8% ± 0.8%	2.41% ± 0.03%	53.5% ± 0.3%	1.76% ± 0.03%
11.2% ± 2.3%	1.35% ± 0.01%	48.7% ± 1.5%	1.53% ± 0.02%	51.9% ± 0.4%	1.37% ± 0.01%
87.9% ± 0.6%	3.81% ± 0.01%	47% ± 1.3%	2.48% ± 0.03%	15.6% ± 0.9%	1.49% ± 0.03%
95.1% ± 1.9%	1.50% ± 0	50.6% ± 2.2%	1.45% ± 0.01%	16.0% ± 0.2%	1.22% ± 0.03%
33.3% ± 0.9%	3.32% ± 0.01%	1.6% ± 0.4%	2.46% ± 0%	31.2% ± 0%	1.60% ± 0.01%
38.0% ± 0.7%	1.47% ± 0.02%	0.5% ± 0.5%	1.48% ± 0.02%	28.7% ± 4.7%	1.30% ± 0.02%

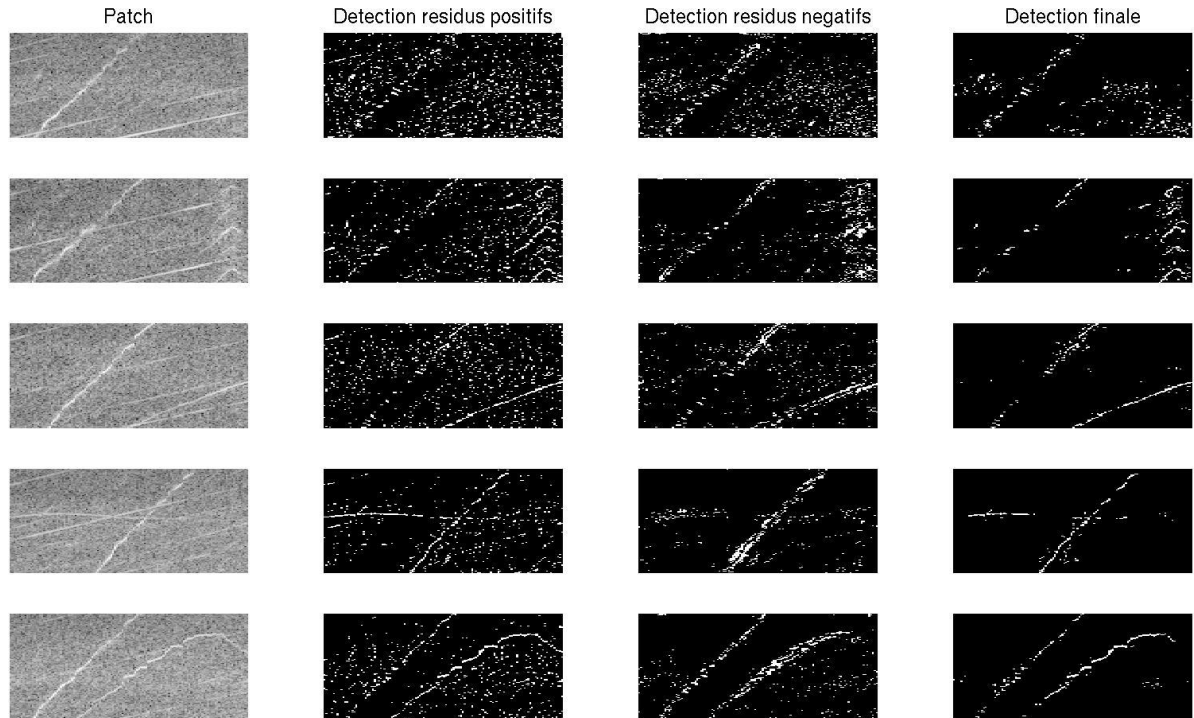


FIGURE 3.19 – Détection des points inusuels sur différents patches. Chaque ligne correspond à un patch, les points détectés à partir des résidus positifs, les points inusuels détectés à partir des résidus négatifs et les points inusuels détectés sur les résidus positifs avec filtrage par les résidus négatifs. La première ligne correspond à un patch normal, les autres sont des patches contenant des signatures inusuelles. Les signatures atypiques se trouvent dans la détection finale établie à partir des résidus positifs et des résidus négatifs.

différentes. L'impact des résidus négatifs est visible, les points détectés au niveau de ces résidus se trouvent bien autour des signatures inusuelles et de la raie N_1 . Le filtrage des détections sur les résidus positifs par les résidus négatifs permet bien d'éliminer de nombreuses fausses détections. Pour les patches contenant les signatures inusuelles, nous voyons apparaître principalement les signatures inusuelles et la raie N_1 dans la détection finale. L'approche basée sur la définition d'un modèle de normalité à partir du dictionnaire des curvelets est donc pertinente pour la détection des signatures inusuelles sur les spectrogrammes. Pour le patch normal, nous détectons principalement la raie N_1 qui reste détectée du fait de son décalage sur les différents spectrogrammes. Cependant de nombreux points restent détectés en dehors de cette raie. Cela provient de la définition du seuil de détection, ce dernier est déterminé intra-patch comme un quantile de la distribution des résidus. La définition de ce seuil ne tient pas compte de la présence ou non de signatures inusuelles sur le patch et entraîne une détection automatique de plusieurs points, y compris lorsque le patch ne contient pas de signature inusuelle. La définition du modèle de normalité à partir du dictionnaire des curvelets avec les résidus de reconstruction pour détecter les signatures atypiques permettent la mise en évidence des signatures inusuelles sur les patches.

3.7 Conclusions

Nous avons défini un modèle de normalité à partir de la transformée en curvelets pour les différents patchs de la subdivision. Ce modèle est issu de l'apprentissage du support normal des atomes des curvelets sur les transformées en curvelet de patchs sans signature inusuelle. Le support normal correspond aux atomes activés redondants dans ces représentations. La reconstruction à partir de ce dictionnaire donne une estimation normale du patch, des artéfacts liés aux signatures inusuelles et aux basses fréquences de la transformée restent présents sur cette reconstruction.

Malgré la représentation dans le dictionnaire des curvelets, cette représentation ne permet pas de détecter les signatures inusuelles en fixant un score de détection sur l'intégralité du patch. Les signatures atypiques restent noyées du fait de leur faible dimension et intensité.

Une étude ponctuelle intrinsèque des résidus positifs et négatifs donne de bons résultats de détection des différents points provenant de signatures atypiques. Les approches permettent de fixer automatiquement des seuils de détection à partir de tests statistiques sur la queue de distribution des résidus ou à partir de valeurs extrêmes. Ces deux méthodes donnent des résultats assez proches. L'algorithme détecte bien les signatures inusuelles mais également les points issus de vibrations liées au N_1 qui sont décalées sur les différents spectrogrammes. La valeur du seuil fixée par les résidus intrinsèques (sans tenir compte des autres données) entraîne nécessairement des points détectés indépendamment du fait que le patch contienne des signatures inusuelles ou non. Cette approche engendre donc de fausses détections, particulièrement pour les patchs normaux.

La définition du modèle de normalité à partir du dictionnaire des curvelets avec les résidus de reconstruction pour détecter les signatures atypiques permettent la mise en évidence automatique des signatures inusuelles sur les patchs sans a priori sur ces dernières.

Chapitre 4

Représentation par dictionnaire data-driven - Non-Negative Matrix Factorization (NMF)

4.1 Les dictionnaires adaptatifs/data-driven

4.1.1 Définition mathématique du problème

Dans cette partie, nous présentons deux types de dictionnaires.

- Les dictionnaires prédéfinis à partir de fonctions génériques dont les atomes ne sont pas modifiés suivant les données étudiées. On parle alors de dictionnaire non-adaptatif comme ses atomes restent les mêmes. Le dictionnaire est choisi de manière à ce que ses atomes soient par nature adaptés à la structure des données. La complémentarité entre le dictionnaire et les données provient de l'avis d'un expert. Ils forment en général des bases.
- Les dictionnaires data-driven définis par apprentissage sur les données. Les atomes de ces dictionnaires correspondent à des fonctions calculées et apprises à partir d'un échantillon des données. On parle de dictionnaire adaptatif car les atomes sont modifiés en fonction des données. Ils sont donc spécifiques aux données. L'adaptation du dictionnaire aux données est établie à partir de la procédure d'apprentissage. Ces dictionnaires font partie du domaine de dictionary learning [87, 79].

Dans ce chapitre, nous nous intéressons au second type de dictionnaire. Les atomes de ce dernier sont appris de manière à caractériser le mieux possible les différents éléments présents dans les données. Ces dictionnaires sont appris sur un jeu de données $X \in \mathbb{R}^{p \times n}$ contenant n données de dimension p en résolvant une fonctionnelle de la forme :

$$[D^*, C^*] = \underset{D, C}{\operatorname{argmin}} \{g(X, DC)\} \text{ avec } X \approx DC \quad (4.1)$$

$D \in \mathbb{R}^{p \times r}$ et $C \in \mathbb{R}^{r \times n}$ représentent respectivement le dictionnaire et les coefficients dans ce dictionnaire et g la fonction de coût mesurant la distance entre les données $X \in \mathbb{R}^{n \times p}$ et leurs estimations selon leurs représentations à partir du dictionnaire D . La distance l_2 est la plus courante pour apprendre les dictionnaires, il est également possible d'utiliser la divergence de Kullback-Leibler. Il s'agit d'une forme de décomposition matricielle, la dimension r du dictionnaire (son nombre d'atomes) doit être définie également.

- Si $r < p$ on parle de réduction de dimension, les données sont projetées dans un espace de dimension réduite pour être étudiées ou représentées. Cette approche permet de caractériser les éléments redondants (c'est-à-dire observés plusieurs fois) dans les données.
- Si $r > p$ on parle alors de dictionnaire surcomplet, les données sont projetées dans un espace de plus grande dimension permettant une décomposition plus fine des données. Ce type de dictionnaire peut permettre une représentation parcimonieuse des données en seillant les coefficients non significatifs. Le dictionnaire ainsi défini est alors considéré comme redondant (plusieurs atomes peuvent caractériser les mêmes éléments).
- Si $r = p$ les données sont projetées dans un espace de même dimension définie par une matrice de passage.

La fonctionnelle définie par l'équation 4.1 ne possède pas de solution unique. Afin de trouver une solution robuste et intéressante, des contraintes ϕ doivent être ajoutées (4.2). Ces contraintes peuvent être sur les coefficients C et/ou le dictionnaire D . Il peut s'agir de contraintes de parcimonie, de régulation l_2 , de positivité, d'invariance par translation ou par rotation,...

$$[D^*, C^*] = \underset{D, C}{\operatorname{argmin}} \{g(X, DC) + \lambda_C \phi_C(C) + \lambda_D \phi_D(D)\} \quad (4.2)$$

Dans la littérature récente [98, 93], les problèmes étudiés attribuent des contraintes de parcimonie sur les coefficients C (4.3). Cette parcimonie provient d'une pénalisation l_0 des coefficients ajoutée à la fonction de coût. Cette formulation n'est cependant pas convexe, elle est approximée en remplaçant la pseudo-norme l_0 par la norme l_1 (4.4) qui peut être vue comme la norme "la plus proche" de l_0 [87, 79, 4]. Ces formulations sont donc composées d'un terme d'attache aux données et d'un terme de parcimonie sur les coefficients.

$$[D^*, C^*] = \underset{D, C}{\operatorname{argmin}} \left\{ \frac{1}{2} \|X - DC\|_2^2 + \lambda \|C\|_0 \right\} \quad (4.3)$$

$$[D^*, C^*] = \underset{D, C}{\operatorname{argmin}} \left\{ \frac{1}{2} \|X - DC\|_2^2 + \lambda \|C\|_1 \right\} \quad (4.4)$$

avec $\|C\|_0 = \operatorname{card}(C \neq 0)$

4.1.2 Les méthodes de résolution

Les problématiques (4.3) et (4.4) se résolvent généralement par des optimisations alternées itératives des paramètres C et D en supposant l'un ou l'autre fixé. L'optimisation du paramètre

C avec les contraintes de parcimonie (en supposant D fixé) est effectuée à partir de méthodes de sparse coding [87, 41]. Ce problème est similaire aux problématiques de type LASSO [110].

$$C^k = \underset{C}{\operatorname{argmin}} \left\{ \frac{1}{2} \|X - D^k C\|_2^2 + \lambda \|C\|_1 \right\} \quad (4.5)$$

avec D^k l'estimation du dictionnaire à l'itération k . Les algorithmes de matching pursuit (MP) [82] permettent de donner une solution approximative de (4.5) en considérant la norme l_0 . Il s'agit d'une méthode itérative consistant à déterminer l'atome du dictionnaire ayant la plus forte corrélation avec les signaux et de retrancher la contribution de cet atome des signaux. Le processus est répété à l'itération suivante avec le signaux résiduels. L'algorithme s'arrête à un nombre d'atomes prédéterminé par l'utilisateur ou à un niveau d'erreur de reconstruction. L'orthogonal matching pursuit (OMP) [43] est une extension du MP où une nouvelle représentation est calculée à chaque itération à partir des atomes sélectionnés et retranchés des signaux. Les algorithmes d'optimisation de basis pursuit (BP) [28] prennent en compte la norme l_1 et utilisent la programmation linéaire pour résoudre (4.5). Dans le même cadre, l'algorithme FOCUSS (FOCal Underdetermined System Solver) [51] résout le problème (4.5) en remplaçant la norme l_1 par une pseudo-norme l_p avec $0 < p \leq 1$. Ces pseudo-normes sont plus proches de la pseudo-norme l_0 que la norme l_1 . Les algorithmes proximaux [88] permettent également de résoudre cette problématique.

La mise à jour du dictionnaire D peut s'effectuer par une méthode similaire aux moindres carrés, la méthode des directions optimales (MOD) [46], si aucune contrainte n'est donnée à D . Généralement, pour éviter de trop grandes valeurs du dictionnaire, des contraintes limitant la norme 2 des atomes du dictionnaire sont ajoutées.

4.1.3 Quelques exemples de dictionnaires data-driven (adaptatifs)

L'ACP peut être considérée comme le dictionnaire appris le plus répandu. Les atomes du dictionnaire correspondent aux composantes principales maximisant la variance au sein des données.

Les k-means [49] sont une méthode permettant de construire un dictionnaire data-driven. Les atomes du dictionnaire sont définis par les centroïdes des clusters. Les données sont alors représentées par un coefficient dont la dimension est égale au nombre de clusters et prenant la valeur 1 au niveau du cluster attribué et 0 ailleurs. Il s'agit de la plus parcimonieuse des représentations. Les K-SVD [4] sont présentés comme une généralisation des k-means avec une représentation des données par plus d'un atome et dont les coefficients peuvent prendre des valeurs différentes de 0 ou 1. La particularité de ce dictionnaire l'apparentant au k-means provient de la mise à jour itérative du dictionnaire atome par atome (en considérant tous les autres fixés) à partir d'une décomposition en valeur singulière (SVD).

Les méthodes précédentes permettent de trouver une bonne représentation des données X

dans les atomes appris de D en imposant une parcimonie sur les coefficients C . Les dictionnaires ne sont pas définis uniquement à partir des contraintes de parcimonie. La Non-Negative Matrix Factorization (NMF) [70] permet de proposer un dictionnaire en imposant la positivité au dictionnaire et aux coefficients. Cette représentation impose donc uniquement des combinaisons linéaires positives. Une description plus détaillée de la méthode est donnée par la suite. Il est également possible de combiner les contraintes. Dans [5], l'algorithme K-SVD est appliqué avec une contrainte supplémentaire de positivité et d'invariance par translation dans [78]. Des contraintes de parcimonie sont également ajoutées à la NMF dans [93]. Les réseaux de neurones [66, 68] tels que les autoencodeurs [109] peuvent également correspondre à des dictionnaires appris à partir des données. Les éléments des différentes couches donnent une représentation de la donnée, les poids au niveau de ces couches correspondent au dictionnaire. Ce type de décomposition possède la particularité de ne pas être linéaire.

Les dictionnaires data-driven vus ici sont donc définis comme une décomposition matricielle sous contraintes où les atomes du dictionnaire et les coefficients peuvent être mis à jour alternativement. Les différentes méthodes d'optimisation définissent différents dictionnaires.

4.1.4 Les dictionnaires appris pour caractériser les spectrogrammes

Dans le chapitre précédent, nous avons sélectionné le dictionnaire non-adaptatif des curvelets pour caractériser le modèle de normalité car la structure de ces atomes permet de représenter les formes courbes que sont les signatures vibratoires de manière parcimonieuse (avec un seuillage des coefficients). Cependant, les atomes basses fréquences de ce dictionnaire restent activés et entraînent une reconstruction grossière des signatures atypiques. L'utilisation de ce dictionnaire permet de réduire les signatures inusuelles sur la reconstruction mais pas de les supprimer. Les dictionnaires data-driven s'adaptent à la donnée. Ils peuvent apprendre les éléments normaux spécifiques et redondants sur les différents patches normaux des spectrogrammes afin de définir le modèle de normalité et permettre la reconstruction de ces éléments uniquement.

La dimension de nos patches (128×128) est nettement plus grande que le nombre de données dont nous disposons (de l'ordre de 500). Il nous est donc impossible de définir un dictionnaire surcomplet (comme pour les curvelets) car les atomes correspondraient alors aux données utilisées pour l'apprentissage du dictionnaire. Nous nous plaçons donc dans un cadre de réduction de dimension avec une nouvelle dimension r très inférieure à la dimension des patches et au nombre de données utilisées pour l'apprentissage. Cette réduction de dimension permet de considérer principalement les éléments redondants au sein des données qui correspondent aux caractères normaux de nos patches. Nous ne cherchons donc pas à trouver une représentation parcimonieuse à l'instar de la transformée en curvelets mais cherchons une représentation des vibrations normales en dimension réduite. Cela est possible car le nombre de raies présentes sur les différents patches est limité. Cependant, un des désavantages des dictionnaires data-driven par rapport aux dictionnaires génériques est la non-unicité de la représentation. En fonction des données d'apprentissage, les dictionnaires définis ne sont pas les mêmes.

Dans ce chapitre nous utilisons la Non-Negative Matrix Factorization (NMF) pour caractériser le modèle de normalité des spectrogrammes. Nous considérons ainsi les spectrogrammes comme une combinaison positive des différents atomes/sources du dictionnaire. La NMF a principalement été utilisée pour effectuer de la séparation de sources aveugle [30] et de la décomposition d'images [69]. Nous utilisons une nouvelle fois un dictionnaire linéaire car les spectrogrammes vibratoires peuvent être considérés comme une combinaison de plusieurs sources, chaque source étant caractérisée par une raie vibratoire. Cette approche est pertinente car les spectrogrammes définissent des données positives ou chaque nouvelle source s'ajoute aux précédentes (ou du moins contribue positivement à l'ensemble). La NMF semble donc être en adéquation avec la structure des spectrogrammes. Nous définissons donc un modèle de normalité basé sur le dictionnaire appris par la NMF et donnons des résultats de détection d'anomalies sur le patch et ponctuellement.

4.2 Non-Negative Matrix Factorization (NMF)

4.2.1 Formulation mathématique

La NMF, définie dans [70], consiste à décomposer une matrice positive $X \in \mathbb{R}_+^{p \times n}$ en un produit de deux matrices positives $D \in \mathbb{R}_+^{p \times r}$ et $C \in \mathbb{R}_+^{r \times n}$ (4.6). r définit la nouvelle dimension de représentation dans laquelle les données X sont décomposées. La valeur de r doit être renseignée au préalable.

$$[D^*, C^*] = \underset{D>0, C>0}{\operatorname{argmin}} \|X - DC\|_2^2 \quad (4.6)$$

La matrice D correspond au dictionnaire appris sur les données X et la matrice C aux coefficients de ces mêmes données dans le dictionnaire. Il est également possible de définir comme fonction de perte une distance proche de la divergence de Kullback-Leibler.

4.2.2 Résolution de la problématique

Ce problème est bi-convexe, c'est-à-dire qu'il est convexe selon chacune des variables mais n'est pas convexe en fonction des 2 variables simultanément. Comme pour les apprentissages de dictionnaire, ce problème est résolu par une optimisation alternée en D et C . A partir des mises à jour définies dans [70] avec i et j les indices des matrices :

$$\begin{aligned} C_{ij}^{k+1} &= C_{ij}^k \frac{(D^{kT} X)_{ij}}{(D^{kT} D^k C^k)_{ij}} \\ D_{ij}^{k+1} &= D_{ij}^k \frac{(X C^{k+1T})_{ij}}{(D^k C^{k+1} C^{k+1T})_{ij}} \end{aligned} \quad (4.7)$$

la fonction $k \mapsto \|X - D^k C^k\|_2^2$ est décroissante. C^k et D^k sont les matrices de coefficients et de dictionnaire à l'itération k .

Ces mises à jour permettent donc la décroissance de la fonction de coût minorée par 0, l'algorithme converge donc bien vers une solution de (4.6). De plus, tant que l'initialisation (définie aléatoirement en général), est positive, la série $(C^k, D^k)_{k \in \mathbb{N}}$ reste positive.

4.3 Le modèle de normalité défini à partir de la NMF

4.3.1 L'apprentissage du dictionnaire

Le modèle de normalité est comme précédemment défini à partir du dictionnaire dans lequel les données sont projetées. Le dictionnaire défini par la NMF permet de caractériser les éléments redondants présents au sein des données. Dans le cas de nos spectrogrammes, il s'agit des raies présentes sur la majorité des patchs. De plus, la positivité impose des combinaisons linéaires positives empêchant les atomes du dictionnaire de reconstruire des raies inusuelles. Tous les atomes du dictionnaire sont utilisés pour représenter la donnée en dimension réduite, ainsi une signature atypique présente sur un des atomes sera retrouvée sur l'ensemble des reconstructions par la combinaison additive. De même, une signature présente sur un patch ne peut pas être reconstruite si elle n'est pas caractérisée par les atomes. Représenter cette signature reviendrait à modifier les coefficients de représentation pour différents atomes, ce qui entraînerait une modification d'une plus grande ampleur sur la reconstruction due aux aspects linéaire et positif. Ce dictionnaire possède donc la particularité de ne pas pouvoir supprimer ce qui est ajouté par les atomes. Cette contrainte de positivité liée à la relation linéaire est essentielle pour définir le modèle de normalité. Seules les signatures redondantes dans les données utilisées dans l'apprentissage sont représentées dans les atomes et seules ces signatures sont reconstruites. Ce dictionnaire est donc établi pour apprendre et reconstruire uniquement les éléments normaux de nos spectrogrammes.

Nous utilisons des patchs sans signature inusuelle de Ω_{App}^j pour définir la matrice des données d'apprentissage. Nous notons $n_{App}^j = \text{card}(\Omega_{App}^j)$ le nombre de données utilisées pour l'apprentissage. Nous notons $\vec{Z}_{\mathcal{K}_j^{128}}^i \in \mathbb{R}_+^{128^2}$ le patch vectorisé permettant de créer la matrice d'apprentissage $\vec{Z}_{\Omega_{App}^j} \in \mathbb{R}_+^{128^2 \times n_{App}^j}$ par concaténation des patchs vectorisés d'apprentissage. À l'inverse des matrices en analyse de données, les lignes de la matrice $\vec{Z}_{\Omega_{App}^j}$ caractérisent les différentes variables correspondant aux points du patch tandis que les colonnes caractérisent les différentes observations de ce même patch. Le dictionnaire issu de la NMF est alors obtenu en décomposant la matrice $\vec{Z}_{\Omega_{App}^j}$ en produit des matrices $\mathcal{D}_j^{NMF} \in \mathbb{R}_+^{128^2 \times r_j}$ correspondant au dictionnaire appris et $C_{\Omega_{App}^j} \in \mathbb{R}_+^{r_j \times n_{App}^j}$ correspondant aux coefficients dans \mathcal{D}_j^{NMF} des patchs de Ω_{App}^j (4.8). r_j représente le rang de la représentation. Un dictionnaire est donc calculé pour chaque patch j . Il