Ircam AudioPrint : Réduction de Dimension des Empreintes Sonores par Analyse Discriminante

Rémi Mignot

Ircam AudioPrint:

Réduction de Dimension des Empreintes Sonores par Analyse Discriminante

Rémi Mignot

IRCAM -CNRS, rapport interne, projet BeeMusic

16 mars 2016

Résumé

Dans le cadre du travail mis en œuvre pour l'amélioration de l'indexation audio du projet BeeMusic, nous avons développé de nouvelles empreintes sonores intrinsèquement plus robustes à certains types de dégradations audio : changement d'échelles (temps et fréquences), égalisation, bruit additif. Cependant, ces données obtenues, qui décrivent une portion de signal musical d'environ deux secondes, sont de très grande dimension : chacune de ces empreintes initiales est de dimension environ mille. Dans ce document, nous expliquons comment tirer partie de cette grande dimensionnalité pour accroître une fois de plus la robustesse. Ici, nous obtenons une réduction de dimension basée sur une analyse semblable à une analyse linéaire discriminante, qui permet de sélectionner par un apprentissage automatique les directions les plus informatives et les moins sensibles aux altérations.

1 Introduction

Le but de l'indexation audio est en quelques sortes d'identifier les signaux sonores par des séquences d'empreintes sonores, ou de codes. Cela peut être utile pour certaines applications audio, telle que la reconnaissance du titre ou de l'artiste d'un morceau de musique donné par son signal uniquement, ou bien pour le dédoublonnage d'extraits sonores d'une base de données. Ainsi, après avoir construit une base d'items de référence, des morceaux de musique par exemple, il est possible de reconnaître un extrait donné si celui-ci est inclu dans la base de référence.

Ces empreintes, dont le terme anglophone usuel est *Audio FingerPrints*, sont chacune une représentation condensée d'une partie du son, et sont le plus souvent basées sur des propriétés spécifiques des signaux audio. Cependant, dans le cas où l'extrait à identifier est perturbé par une dégradation, la modification des empreintes peut alors provoquer une détérioration significative de la reconnaissance. Ces altérations sonores sont par exemple : ajout de bruit synthétique ou environnemental, distorsion ou saturation, encodage/décodage aux formats MP3 ou GSM, changement de hauteurs, dilatation/compression du temps, égalisation ou filtrage, réverbération, compression des dynamiques. Le travail présenté dans ce papier traite de la robustesse de la reconnaissance d'extraits sonores en présence d'altérations.

Parmi les travaux pionniers du domaine, Fraunhoffer, cf. [1], a proposé l'utilisation de descripteurs audio de la norme MPEG7. Ainsi, à l'aide d'une quantification vectorielle, par clustering, une approche standard de classification permet la sélection de l'item le plus proche, au sens d'une erreur de reconstruction des codes. Pour la méthode de Philips [4], la recherche est basée sur un code binaire de grande dimension calculé sur le spectrogramme filtré par un masque dérivateur. Après une pré-sélection d'items candidats, obtenus par une recherche de sous-codes dans une table de hachage, la distance de Hamming est utilisée pour obtenir un indice de similarité. L'application la plus populaire est probablement celle de Shazam, dont la technique repose sur la recherche par table de hachage des occurrences d'empreintes sonores qui y représentent des paires de maxima du spectrogramme, cf. e.g. [23]. Après une pré-sélection des items candidats ayant le plus grand nombre de codes reconnus, une sélection plus précise est faite en étudiant la cohérence temporelle des codes détectés par un histogramme.

Plus tard, pour répondre plus spécifiquement au problème des altérations sonores, d'autres travaux ont été mis en œuvre. Par exemple dans [5], une transformation du spectre a permis d'améliorer la robustesse de la méthode de [4], Philips, au changement d'échelle fréquentielle. Par ailleurs, plus récemment dans [16], cette même méthode est encore améliorée en introduisant le principe de *Hachage Approximatif* qui permet dans ce cas une généralisation de l'approche et une amélioration de la tolérance aux erreurs, pour tout type de dégradations. Enfin, nous pouvons citer [3] qui améliore la robustesse de [23], Shazam, au changement de hauteur via l'utilisation de la Transformée à Q-Constant (CQT), et [21] qui utilise des *quadruplets* de maxima dont les positions en temps et en fréquence sont codés de façon relative, améliorant de la sorte la robustesse au changement d'échelle temporelle et fréquentielle.

Dans le contexte du projet BeeMusic, nous avons été amenés à proposer une nouvelle définition d'empreintes sonores, cf. [13]. Comparée à la précédente indexation audio développée à l'IRCAM, *IrcamAudioID* présentée dans [19, 20], cette méthode permet une robustesse accrue aux dégradations sonores, mais les vecteurs produits sont d'une si grande dimension qu'il est impossible de les traiter tels quels. C'est la raison pour laquelle une étape supplémentaire a été développée. Elle consiste à opérer une projection des données obtenues sur un sous-espace sélectionné par ses propriétés de robustesse.

Remarquons que dans le précédent système *IrcamAudioID*, une telle réduction de dimension est effectuée ; on y parle de *regroupement*. Cependant elle est arbitrairement choisie par des règles de proximité, selon les échelles (fréquences par exemple). Or, rien ne garantit que ce choix soit effectivement judicieux pour améliorer la robustesse. L'intérêt du travail présenté dans ce document, est que la réduction est ob-

tenue automatiquement par un apprentissage supervisé, similaire à une *Analyse Linéaire Discriminante*. Ainsi, grâce aux critères définis, la projection obtenue garantit une meilleure résistance aux dégradations. De plus, certaines opérations auront pour effet de pré-conditionner les données en fonction de la méthode de recherche ultérieure.

La plupart des techniques existantes sont divisées en deux étapes : d'une part, l'analyse des signaux conduit à l'obtention des empreintes sonores, et d'autre part, une phase de recherche consiste à retrouver dans une base de référence le ou les items les plus similaires par comparaison des empreintes. Ainsi, le travail présenté dans ce document est inclu à la fin du calcul des empreintes sonores, donc juste avant la phase de recherche. Nous renvoyons le lecteur intéressé à [13] pour le détail du calcul des empreintes de grande dimension, et à [14, sec. 5] et [12] pour un aperçu de la phase de recherche.

Voici le plan du document : un résumé des empreintes de grande dimension est fait en section 2. Puis la section 3 présente les différentes méthodes statistiques testées pour la sélection de directions robustes aux dégradations sonores, telles que l'*Analyse Linéaire Discriminante*. Puis, en section 4 sont décrits d'autres analyses visant au conditionnement des données pour améliorer la recherche ultérieure. Par exemple une *Analyse en Composantes Indépendantes* permet un meilleur remplissage de la table de hachage. Ensuite, pour évaluer le bénéfice de chacune de ces analyses une procédure d'évaluation est faite et est présentée en section 5. Enfin, la section 6 conclut le document.

2 Empreintes sonores de grandes dimensions : résumé

La nouvelle approche mise en œuvre dans [13] permet de définir des empreintes sonores naturellement robustes à certains types de dégradations qui sont : changement d'échelles temporelle et fréquentielle (time stretching et pitch shifting selon les termes anglophones habituels), ajout de bruit, égalisation, et variation du gain. Cette présente section donne un résumé du calcul des empreintes sonores de grande dimension, déjà détaillé dans [13]. La suite du document présente la réduction de dimension, à partir de la section suivante, sec. 3.

Spectrogramme et fenêtrage : Dans un premier temps, le spectrogramme complet du signal audio est calculé par une Transformée de Fourier à Court-Terme. Puis cette représentation temps-fréquences est fractionnée par fenêtres d'analyse d'environ 2 secondes espacée approximativement tous les quarts de secondes, avec recouvrement donc.

Conversion logarithmique temps-fréquence : De ces représentations successives du temps et de la fréquence en échelle linéaire (secondes-hertz), une conversion fournit une nouvelle représentation temps-fréquence en échelle logarithmique cette fois-ci (log-temps - log-fréquences).

Séparation en sous-bandes de fréquence : L'échelle des fréquences (logarithmique) est découpée en 5 bandes de largeurs égales, et avec recouvrement simple. A titre d'information, les fréquences centrales vont de 270Hz à 2790Hz, et leur largeur de bande représente environ 20 demi-tons en musique. Pour plus de simplicité, nous nommerons : "carreau" la représentation d'une bande, et pixels chacun de ses coefficients.

Traitement des amplitudes : C'est alors qu'une modification des amplitudes linéaires est réalisée. Premièrement, un redressement à une valeur planché est fait. Deuxièmement, une pondération 2D est réalisée pour ramener doucement à 0 les pixels aux bords du carreau. Ensuite, une normalisation de la norme L_∞ est effectuée. Enfin, un changement d'échelle quasi-logarithmique est réalisé.

Transformée de Fourier à 2 dimensions : Enfin, la transformée de Fourier discrète à 2 dimensions est réalisée sur chaque carreau modifié précédemment, et le module en est extrait.

Cette méthode fournit alors à chaque instant d'analyse (espacés d'environ 250ms) et à chacune des cinq bandes de fréquence, une représentation 2D, pour laquelle les variables sont relatives à la fréquence des log-fréquences d'une part et à la fréquence des log-temps, d'autre part. En raison de la symétrie de la transformée de Fourier, seule une moitié est conservée : l'échelle des fréquences des log-temps est alors coupée en deux. Ainsi, nous obtenons une représentation de taille (32×33), 32 pour les fréquences des log-fréquences (de la bande considérée), et 33 pour la moitié des fréquences des log-temps. Une fois vectorisée, nous obtenons une empreinte de haute dimension, 1056. Voir Fig. 1 pour un résumé du calcul.

Cette nouvelle méthode de calcul des empreintes sonores a été initialement conçue pour améliorer naturellement la robustesse à certains types de dégradations. Par exemple, nous utilisons ici le module de

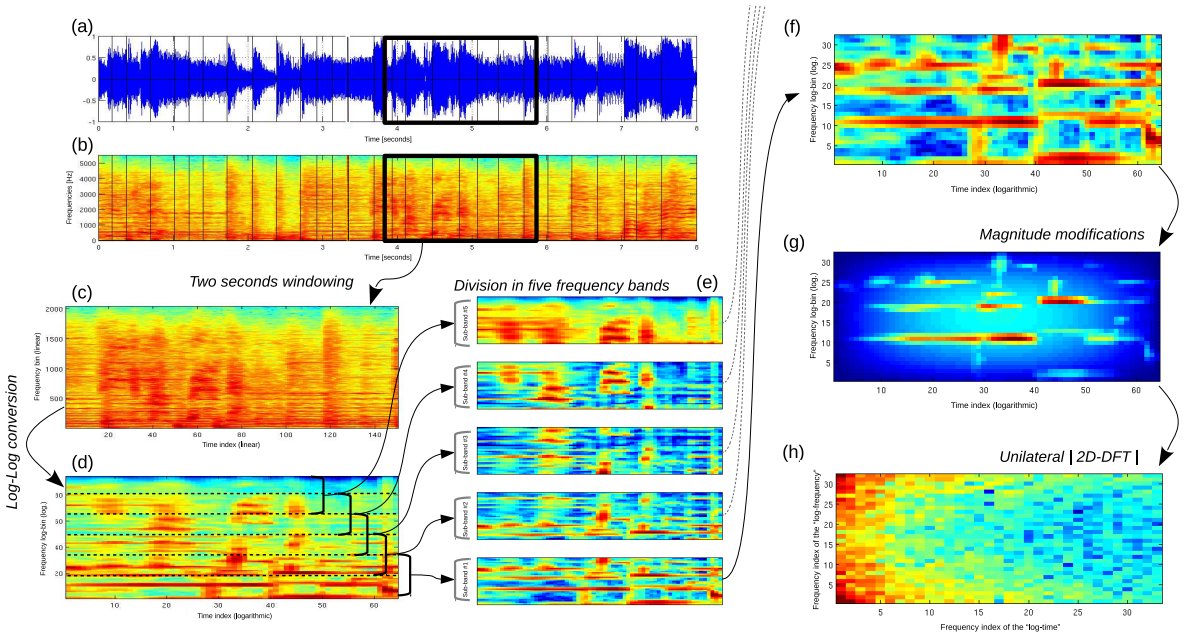

FIGURE 1 – Illustration du calcul des empreintes sonores : (a) Signal temporel. (b) Spectrogramme complet. (c) Sélection d'une fenêtre de 2 secondes environ. (d) Conversion du temps et de la fréquence en échelle logarithmique. (e) Découpage en 5 bandes de fréquences, avec recouvrement. (f) Sélection des bandes une-à-une. (g) Modification des amplitudes : redressement, conversion, normalisation, pondération. (h) Calcul de l'amplitude de la DFT2D. La représentation finale est de dimension (32×33).

la transformée de Fourier, effectuée dans les échelles logarithmiques du temps et des fréquences. Or puisqu'une dilatation/compression se manifeste par un décalage du spectrogramme en échelle logarithmique, la représentation finale est alors en principe invariante aux altérations que l'on nomme *time stretching* et *pitch shifting*. D'autre part, la robustesse au bruit additif est prise en compte deux fois dans ce travail : premièrement, la séparation en cinq bandes de fréquences différentes permet dans la plupart des cas d'isoler le bruit et de laisser un certain nombre de bandes intactes. Secondement, le redressement des pixels à un niveau minimal, permet d'atténuer relativement l'effet du bruit. Enfin, ces empreintes obtenues sont également robustes dans une certaine mesure à l'égalisation, au filtrage, au changement de gain, et à un décalage de la position des points d'analyse. Voir [13] pour plus de détails.

3 Recherche de directions robustes aux dégradations

Les étapes du processus de calcul de ces nouvelles empreintes sonores ont été élaborées de sorte à en dériver des propriétés spécifiques de robustesse à certaines dégradations. Avec le paramétrage choisi, les vecteurs obtenus sont de dimension 1056. Ce nombre peut paraître excessif, mais en réalité, cette grande dimensionnalité va nous permettre d'améliorer la performance d'indexation. En effet, une analyse statistique nous permet d'apprendre automatiquement la réduction de dimension conduisant à une robustesse accrue, via une projection dans un sous-espace discriminant. Par ailleurs, plusieurs analyses successives nous permettent de conditionner les empreintes, cf. sec. 4.

Ici l'approche développée consiste en un apprentissage supervisé. Les données de l'apprentissage sont fournies par le calcul des empreintes sonores sur le signal original et sur le signal dégradé. Remarquons que la dégradation du signal audio est réalisée par la boîte à outil *BeeAlter* présentée en [11], et développée dans le cadre de ce travail. Nous utilisons ici un grand nombre de dégradations différentes avec la possibilité de produire plusieurs altérations à la suite sur un même son, pour être très proche de la réalité. Par conséquent, en analysant les empreintes originales et les empreintes *dégradées*, les différentes méthodes statistiques présentées dans cette section permettent de définir une réduction de dimension améliorant la robustesse.

Même si certaines des techniques expliquées ne sont finalement pas retenues, nous faisons le choix de les décrire toutes, afin de présenter l'ensemble du travail fourni. Le décision des méthodes utilisées est donné en sec. 5, à l'aide d'une procédure d'évaluation.

Enfin, précisons que les cinq bandes sont traitées en parallèle rigoureusement de la même manière, mais avec des données d'apprentissage distinctes, calculées séparément sur chacune d'elles. Ainsi, les réductions obtenues, même si elles ont la même forme, seront très différentes dans le sens où les coefficients des matrices et vecteurs appris ne sont pas les mêmes. En effet, le contenu fréquentiel de chaque bande étant différent, il n'y a pas de raison d'appliquer rigoureusement la même réduction.

3.1 Analyse Linéaire Discriminante

L'Analyse Linéaire Discriminante (dont l'acronyme anglophone est LDA) est une méthode parfois utilisée en classification, cf. e.g. [6, 2]. Dans un espace vectoriel représentant les éléments de plusieurs classes, elle fournit les directions optimales pour discriminer ces classes au sens de l'information de Fisher. Nous avons alors formalisé le problème d'indexation sous forme d'un problème de classification et avons appliqué la LDA pour les empreintes sonores de grande dimension : une classe est ici donnée par un ensemble d'empreintes correspondant toutes à un même instant d'analyse et un même signal original, mais chacune calculée après une série d'altérations différentes. Ainsi, discriminer les classes dans notre cas permet implicitement de réduire significativement la sensibilité aux altérations sonores.

3.1.1 Variances totale, intra-classes et inter-classes

Commençons par donner la décomposition de la variance totale d'un nuage de points séparé en un nombre fini de groupes différents : soient X le vecteur aléatoire, de grande dimension J, et Y une variable aléatoire d'appartenance à un groupe, ou classe. Alors le théorème de la variance totale s'écrit :

$$\mathrm{Var}(X) = \mathrm{E}_Y[\,\mathrm{Var}(X/Y)\,] + \mathrm{Var}_Y[\,\mathrm{E}(X/Y)\,] \tag{1}$$

Précédemment, $\mathrm{E}(Z)$ est le vecteur $(J\times 1)$ de l'espérance de Z, et $\mathrm{Var}(Z) \triangleq \mathrm{E}\big([(Z-\mathrm{E}(Z)][Z-\mathrm{E}(Z)]^T\big)$ sa matrice de covariance de dimension $(J\times J)$, quelque soit Z. Le symbole $.^T$ représente la transposition de matrices. Pour simplifier les expressions, notons : $\mu \triangleq \mathrm{E}(X)$ et $\mu^c \triangleq \mathrm{E}(X/Y=y_c)$. Le vecteur μ n'est autre que le "centre de gravité" de l'ensemble complet des points, et μ^c celui des éléments appartenant à la classe y_c. Alors, en considérant chaque point de poids égal, l'estimateur sans biais suivant donne chaque coefficient de la matrice $T \triangleq \mathrm{Var}(X)$:

$$T_{j,k} = \mathrm{Var}(X)_{j,k} = \frac{1}{N-1}\sum_{i=1}^{N}\big(X_j(i)-\mu_j\big)\,\big(X_k(i)-\mu_k\big). \tag{2}$$

Notons que si les composantes de X n'ont pas de relation affine, alors la matrice T est symétrique définie positive, ce qui sera vérifié dans notre cas après la *Réjection des Composantes Mal-Conditionnées* présentée en sec. 4.1. Maintenant, avec N_c le nombre de points appartenant à la classe y_c, les coefficients de la matrice de covariance empirique de la classe y_c sont :

$$\mathrm{Var}(X/Y=y_c)_{j,k} = \frac{1}{N_c-1}\sum_{\substack{i\in[1,N]\\ Y(i)=y_c}}\big(X_j(i)-\mu_j^c\big)\,\big(X_k(i)-\mu_k^c\big), \tag{3}$$

et avec $P(Y=y_c)=N_c/N$ la probabilité qu'un point quelconque appartienne à la classe y_c, on a alors :

$$\begin{aligned}\mathrm{E}_Y[\,\mathrm{Var}(X/Y)\,]_{j,k} &= \sum_{c=1}^{C} P(Y=y_c)\,\mathrm{Var}(X/Y=y_c)_{j,k}, &(4)\\ &= \frac{N_c}{N(N_c-1)}\sum_{c=1}^{C}\sum_{\substack{i\in[1,N]\\ Y(i)=y_c}}\big(X_j(i)-\mu_j^c\big)\,\big(X_k(i)-\mu_k^c\big). &(5)\end{aligned}$$

La matrice $W \triangleq \mathrm{E}_Y[\mathrm{Var}(X/Y)]$ s'interprète alors comme la moyenne des matrices de covariance de chaque classe, et on parle de *Matrice de Covariance Intra-Classe*. Enfin, pour ce qui concerne l'autre terme de l'eq. (2), en rappelant que $\mu = E(X)$ et que $\mu^c = E(X/Y=y_c)$, l'estimateur sans biais de la variance donne :

$$\mathrm{Var}_Y[\,E(X/Y)\,]_{j,k} = \frac{1}{C-1}\sum_{c=1}^{C}\big(\mu_j^c-\mu_j\big)\,\big(\mu_k^c-\mu_k\big). \tag{6}$$

Il s'agit cette fois de la *Matrice de Covariance Inter-Classe*, puisqu'elle s'interprète comme la matrice de covariance des centre de gravité μ^c de chaque classe. On la note alors $B \triangleq \mathrm{Var}_Y[\,E(X/Y)\,]$, et on remarque qu'elle est symétrique positive de rang inférieur ou égal à $C-1$.

3.1.2 Pouvoir de discrimination et analyse linéaire discriminante

L'indice de Sobol est un critère permettant de quantifier le pouvoir de discrimination des classes. Dans le cas d'une seule variable, il est donné par la formule : $S(X_1) = \mathrm{Var}_Y[\,E(X_1/Y)\,] \,/\, \mathrm{Var}(X_1)$, et est relatif à l'information de Fisher. On voit alors qu'à variance totale invariante, S est d'autant plus grand que les centres de gravité des groupes sont éloignés les uns des autres, et donc les classes sont d'autant discriminées. Remarquons que $0 \geq S(X_1) \geq 1$. Ainsi, l'analyse linéaire discriminante consiste à déterminer le vecteur g de la combinaison linéaire $\sigma_X = g^T X = \sum_i g_i X_i$ qui maximise l'indice de Sobol :

$$\begin{aligned} S_g &= \mathrm{Var}_Y[\,E(\sigma_X/Y)\,] \,/\, \mathrm{Var}(\sigma_X) && (7)\\ &= \mathrm{Var}_Y\big[\,E(g^T X/Y)\,\big] \,/\, \mathrm{Var}(g^T X), && (8) \end{aligned}$$

qui devient après simplification :

$$S_g = \big(g^T B g\big) \,/\, \big(g^T T g\big)\,. \tag{9}$$

Rappelons que T et B sont deux matrices $(J \times J)$ symétriques, avec T définie positive. Alors, selon le théorème d'*Orthogonalisation Simultanée de Formes Quadratiques*, il existe deux matrices de dimension $(J \times J)$: G inversible et D diagonale, telles que

$$G^T B G = D \quad \text{et} \quad G^T T G = I_J. \tag{10}$$

Ainsi, si les valeurs diagonales de D sont rangées par ordre décroissant, alors la combinaison linéaire qui maximise l'indice de Sobol est donnée par la première colonne de G. Pour résoudre le problème, il suffit d'écrire, $G^T = (TG)^{-1} = G^{-1}T^{-1}$, parce que P et T sont carrés de même dimension, ce qui conduit à :

$$G^{-1}(T^{-1}B)G = D. \tag{11}$$

Par conséquent, l'analyse linéaire discriminante se résout par une diagonalisation de $T^{-1}B$. En rangeant par ordre décroissant les valeurs propres, la matrice $P = G^T$ fournit un changement de base $Z = PX$, sur laquelle les nouvelles variables sont directement rangées par ordre décroissant de pouvoir discriminant, puisque les indices de Sobol sont respectivement donnés par les valeurs propres associées.

Notons qu'avec un nombre de C classes, B est de rang inférieur strictement à C, si bien qu'au plus, les $C-1$ premières valeurs de D sont non-nulles. Ainsi dans le cas d'une projection, seule les $C-1$ premières direction sont utiles pour la discrimination.

3.1.3 Réduction de dimension par LDA

Dans notre cas de réduction des empreintes sonores pour la robustesse aux dégradations, chaque classe est donnée par un ensemble d'empreintes toutes calculées sur un même signal, mais après des dégradations différentes. Ainsi, la base d'apprentissage est construite en dégradant plusieurs fois, plusieurs signaux tests. Rappelons que ces signaux ont une durée de 2 secondes minimum car c'est la taille d'une fenêtre d'analyse pour une empreinte. Puis en groupant les empreintes par classe, les matrices T et B sont calculées et la matrice de passage P est obtenue par diagonalisation de $T^{-1}B$.

Pour réduire la dimension de J aux K directions les plus discriminantes, avec $K \leq C-1$, il suffit alors de ne conserver que les K premières lignes de P. Nous obtenons donc une projection non-orthogonale sur un sous-espace, plus robuste aux dégradations et qui contient en principe une information plus pertinente du point de vue musical. Avec P_{lda} la matrice $(K \times J)$ de la projection, le changement de variable Z de taille $(K \times 1)$ s'opère par le produit matriciel :

$$Z = P_{\mathrm{lda}}\, X, \quad \text{avec} \quad (P_{\mathrm{lda}})_{k,j} = G_{j,k}, \quad \forall k \leq K \text{ et } \forall j \leq J. \tag{12}$$

Cependant, le calcul de l'apprentissage nécessite l'utilisation d'un très grand volume de données. En effet, rappelons que la dimension de départ dépasse 1 000, et nous avons choisi une base d'apprentissage de 150 000 signaux tests. Dans le but d'avoir une très bonne représentativité des dégradations possibles, chaque signal test est dégradé par 300 dégradations différentes, ce qui constitue alors des classes de 300 empreintes dégradées. Notons que l'empreinte originale peut aussi être inclue dans la classe.

Pour ne pas avoir à placer en mémoire les 45 milliards de coefficients ($1\,000 \times 150\,000 \times 300 = 45.10^9$), les matrices B et T sont construites de façon incrémentale, classe après classe. Cependant, telles quelles, les équations (2) et (6) nécessitent de connaître à l'avance μ, la moyenne totale des empreintes. Par

conséquent, ces deux équations sont chacune modifiées en :

$$T = \frac{1}{N-1}\sum_{i=1}^{N} X(i)X(i)^T - \frac{N}{N-1}\,\mu\,\mu^T, \tag{13}$$

$$B = \frac{1}{C-1}\sum_{c=1}^{C} \mu_c\,\mu_c^T - \frac{C}{C-1}\,\mu\,\mu^T. \tag{14}$$

De la sorte, pour se donner une idée du processus, calculons pas-à-pas :

$$\begin{aligned}
\Gamma_n &= \textstyle\sum_{i=1}^{n} X(i) &&= \Gamma_{n-1} + X(n),\\
\Phi_n &= \textstyle\sum_{i=1}^{n} X(i)X(i)^T &&= \Phi_{n-1} + X(n)X(n)^T,\\
\Psi_m &= \textstyle\sum_{c=1}^{m} \mu_c\,\mu_c^T &&= \Psi_{m-1} + \mu_m\,\mu_m^T,\\
&\text{avec} \quad \mu_c = \frac{1}{N_c}\sum_{Y(i)=y_c} X(i).
\end{aligned}$$

Finalement à la fin de la récurrence, on trouve T et B par :

$$T = \frac{1}{N-1}\,\Phi_N - \frac{N}{N-1}\,\mu\,\mu^T, \tag{15}$$

$$B = \frac{1}{C-1}\,\Psi_C - \frac{C}{C-1}\,\mu\,\mu^T, \tag{16}$$

$$\text{avec} \quad \mu = \frac{1}{N}\,\Gamma_N.$$

Remarquons qu'il est possible de rencontrer des problèmes numériques en fin de récurrence. En effet, par exemple les composantes de $X(n)$ peuvent être en module inférieures à la résolution numérique de Γ_{n-1}, pour n très élevé. Pour éviter ce problème nous avons fait des récurrences multiples "imbriquées", de sorte à diminuer les chances d'additions mal conditionnées. Enfin, notons que nous n'avons pas rencontré de problème spécifique sous Matlab à calculer et diagonaliser $T^{-1}B$, malgré sa dimension dépassant $(1\,000 \times 1\,000)$. De plus, avec 150 000 classes, B et $T^{-1}B$ sont en principe de rang plein, puisque $1\,000 \ll 150\,000 - 1$.

Les résultats sont illustrés en figure 2. On y voit que les 4 premières directions de la LDA ont pour effet de regrouper les empreintes appartenant à la même classe, issue de la dégradation du même signal. A l'inverse, sur les 4 directions les moins discriminantes au sens se l'information de Fisher, les empreintes d'une même classe sont très espacées les unes des autres. En conséquence, conserver les premières directions au moyen d'une simple projection permet alors non seulement de réduire la sensibilité au bruit, mais aussi implicitement d'augmenter la pertinence des données retenues du point de vue musicale. Nous verrons plus tard précisément la taille du nouveau sous-espace.

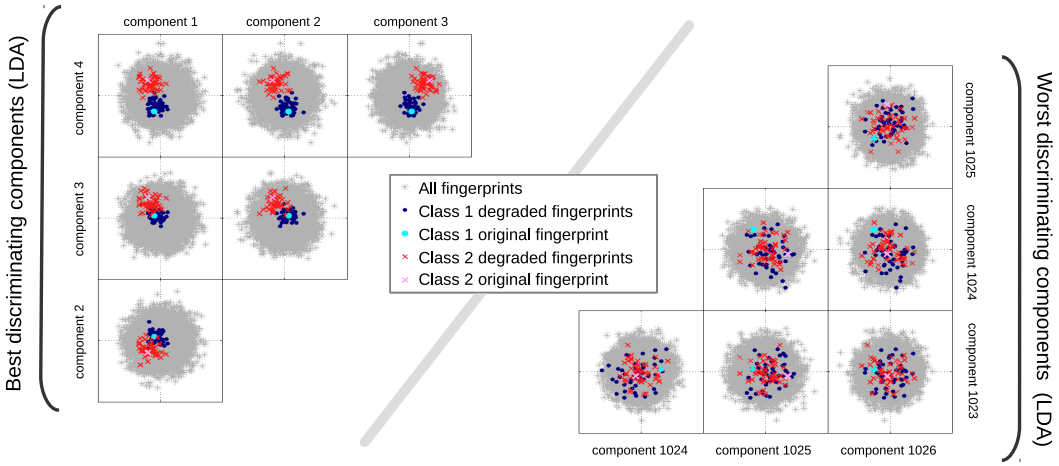

FIGURE 2 – Représentation de 2 classes avec projection des empreintes après LDA sur des plans de 2 variables. Directions les plus discriminantes à gauche, et les moins discriminantes à droite.

3.2 Analyse Linéaire Insensible

La LDA de la section 3.1 permet en principe de choisir les K directions qui discriminent au mieux ce que nous avons défini comme classes d'empreintes. Cependant, même si les points de chaque classe sont relativement bien regroupés dans le sous-espace sélectionné, cf. fig. 2, on remarque très souvent que l'empreinte originale se trouve vers l'extérieur du groupe. Or pour réduire plus efficacement la sensibilité de l'indexation à tout type de dégradation, il sera préférable que dans le nouveau sous-espace les empreintes originales soient proche de la moyenne de leur classe. Nous proposons une légère variante de la LDA pour une meilleure adaptation à notre problème.

Commençons par réécrire les formules avec W la matrice de covariance intra-classe, au lieu de T la matrice de covariance totale. Sachant que $T = W + B$, cf. eq. (1) du théorème de la variance totale, et que (~~$BG = GTD$~~) $BG = TGD$, cf. eq. (11), il vient alors que

$$G^{-1}(W^{-1}B)G = D(I_J - D)^{-1} = \Lambda \tag{17}$$

Puisque D est diagonale, de coefficients d_j, la matrice Λ l'est aussi et ses valeurs diagonales valent : $\lambda_j = d_j/(1-d_j)$. En conséquence $T^{-1}B$ et $W^{-1}B$ ont les mêmes vecteurs propres, donnés par la matrice G, et la LDA peut se résoudre en diagonalisant l'une ou l'autre des deux formes. Notons de plus que les indices de Sobol correspondant aux valeurs de la diagonale de D sont positifs et inférieurs à 1, si bien que si les d_j sont rangés dans l'ordre décroissant, les λ_j le sont aussi.

Puisque la LDA peut finalement être calculée via la matrice W et non T, nous proposons ici de la modifier pour adapter l'algorithme à notre problème spécifique. Remplaçons dans eq. (5) de W, les centres de gravité des classes μ^c par l'empreinte $X^*(c)$ originale de la classe, obtenue sur le signal non dégradé. Il vient alors :

$$W^0_{j,k} \triangleq \frac{N_c}{N(N_c-1)} \sum_{c=1}^{C} \sum_{\substack{i\in[1,N] \\ Y(i)=y_c}} \big(X_j(i) - X^*_j(c)\big)\,\big(X_k(i) - X^*_k(c)\big), \tag{18}$$

La matrice W^0 est donc une matrice similaire à la covariance des écarts avec l'empreinte originale du groupe. Ainsi, réduire cet écart permet de réduire la sensibilité à dégradation, ce qui est plus pertinent pour le problème d'indexation robuste.

Par conséquent, pour chercher les directions qui minimisent l'éloignement des empreintes dégradées par rapport à l'empreinte originale, nous proposons de remplacer la matrice de covariance intra-classe W par la matrice W^0 donnée en eq. (18). Cette variante de la LDA, et qui semble en principe mieux adaptée à notre problème spécifique, est nommée : *Analyse Linéaire Insensible*, ou LISA pour la version anglaise *Linear InSensitive Analysis*. Au final, la diagonalisation de $(W^0)^{-1}B$ donne la nouvelle matrice G^0 $(J\times J)$ des vecteurs propres, et la matrice de passage P_{lisa} $(K\times J)$ du changement de variables :

$$Z = P_{\text{lisa}}\, X, \quad \text{avec} \quad (P_{\text{lisa}})_{k,j} = G^0_{j,k}, \quad \forall k \le K \text{ et } \forall j \le J. \tag{19}$$

Précisons que la matrice W^0 est obtenue par le même type de récurrence que celle utilisée pour B et T en sec. 3.1.3, pour des raisons d'utilisation de mémoire.

Nous verrons en section 5 que cette variante produit de petites améliorations comparée à LDA, sans être pour autant significatives. En effet, on constate après la méthode LISA que même si les empreintes originales sont légèrement recentrées au milieu de leur groupe respectif, ce centrage ne semble par suffisant. Alors, nous avons envisagé une autre méthode dans le but d'améliorer cela, cf. sec. 3.3.2.

3.3 Méthodes dans l'Espace Quadratique

Dans cette section, nous présentons deux méthodes visant à améliorer les résultats obtenus en secs. 3.1 et 3.2, pour les méthodes LDA et LISA respectivement. Une première méthode connue de la littérature, est la QDA (*Quadratic Discriminant Analysis*), et est semblable à une LDA mais en considérant des variables quadratiques, cf. sec. 3.3.1. La seconde méthode que nous avons développée consiste à apprendre automatiquement dans l'espace quadratique une correction visant à résoudre plus efficacement le problème de centrage des empreintes originales présenté précédemment, cf. sec. 3.3.2.

Ici l'utilisation d'un espace quadratique nous permet d'avoir un modèle de données plus complexe. Cependant pour que les méthodes restent réalisables en pratique, nous choisissons de le définir à partir d'un sous-espace de dimension K donné par la LDA ou la LISA. Pour tout point X de l'espace linéaire de dimension K, les composantes de sa représentation $Q(X)$ dans l'espace quadratique sont données par :

$$Q(X) = [\ X_1,\ X_2,\ \ldots\ X_K,\ X_1^2,\ X_1X_2,\ \ldots\ X_1X_K,\ X_2^2,\ X_2X_3,\ \ldots\ X_2X_K,\ \ldots\ldots\ X_K^2\ ]^T,$$

où $Q(X)$ est un vecteur de taille $K_q = K(K/2+1)$. Par exemple, avec $K = 40$, nous avons $K_q = 840$, alors qu'en partant de l'espace initiale de dimension 1000, nous aurions un espace quadratique de dimension dépassant les 500 000.

3.3.1 Analyse Quadratique Discriminante

L'analyse quadratique discriminante suit la même méthode que la LDA mais en remplaçant le vecteur X aléatoire par le vecteur $Q(X)$ de l'espace quadratique, cf. e.g. [6]. Alors de la même manière, sont calculées les matrices de covariance totale T_q, intra-classes W_q et inter-classes B_q, et la diagonalisation de $T_q^{-1}B_q$ ou $W_q^{-1}B_q$ fournit la matrice G_q des vecteurs propres et donc la matrice P_q de la projection.

Par conséquent, d'un espace *linéaire* de dimension K, issu de la LDA par exemple pour des raisons de faisabilité, nous passons dans l'espace quadratique de dimension $K_q \gg K$, et la QDA permet alors de réduire une fois de plus la dimension à $K' \ll K_q$. En principe, l'avantage est ici de pouvoir définir implicitement des frontières de décisions sous formes d'hyper-surfaces quadratiques et non seulement des hyper-plans comme avec la LDA. En effet, la combinaison linéaire des coefficients de $Q(X)$ donnée par un vecteur $g \in \mathbb{R}^{K_q}$, s'écrit maintenant :

$$\begin{array}{rcrrrrrrrrrr} g^T Q(X) & = & & g_1 & X_1 & + & g_2 & X_2 & + & g_3 & X_3 & + \ \ldots + & g_K & X_K \\ & & + & g_{K+1} & X_1^2 & + & g_{K+2} & X_1X_2 & + & g_{K+3} & X_1X_3 & + \ \ldots + & g_{2K} & X_1X_K \\ & & & & & + & g_{2K+1} & X_2^2 & + & g_{2K+2} & X_2X_3 & + \ \ldots + & g_{3K-1} & X_2X_K \\ & & & & & & & & & & & \ \ldots + & g_{K_q} & X_K^2 \end{array} , \tag{20}$$

alors pour L une constante, $g^T Q(X) = L$ est l'équation d'une hyper-surface quadratique d'un espace de dimension K (cône, cylindre, paraboloïde, ellipsoïde, par exemple).

Donc en pratique, après le calcul des empreintes sonores de haute dimension J et la projection (LDA ou LISA) sur le sous-espace de dimension K, les vecteurs de l'espace quadratique $Q(X)$ de dimension K_q sont calculés. C'est alors que la QDA donne la matrice P_{qda} de dimension $(K' \times K_q)$ pour une seconde projection sur un sous-espace (quadratique) de dimension K' :

$$Z = P_{\text{qda}}\, Q(X), \quad \text{avec} \quad (P_{\text{dda}})_{k,j} = (G_q)_{j,k}, \quad \forall k \leq K' \text{ et } \forall j \leq K_q. \tag{21}$$

Notons qu'il ne s'agit pas ici réellement d'une réduction de dimension puisque que celle-ci a provisoirement beaucoup augmenté à $K_q = K(K/2+1)$. L'intérêt espéré est en fait de généraliser l'analyse dans le but d'améliorer significativement la discrimination des classes et donc la robustesse. Par exemple, lors des évaluations faites en sec. 5, les dimensions choisies sont $K = 48$, ce qui implique $K_q = 1200$, et la dimension finale est $K' = K = 48$. Nous y verrons que le bénéfice de cette méthode est tout à fait négligeable dans notre cas.

3.3.2 Correction Quadratique

En section 3.2, nous avons détaillé le fait qu'il arrive souvent que les empreintes originales se retrouvent vers l'extérieur de leur groupe respectif d'empreintes dégradées, et avons adapté la LDA en LISA pour tenter de limiter ce problème de centrage. Dans cette nouvelle section, nous proposons une nouvelle méthode visant à corriger ce problème plus efficacement.

L'idée est d'appliquer sur les empreintes dégradées X une translation $\tau(X)$, dépendant dans la position de départ, permettant de minimiser l'écart avec l'empreinte originale. Dans le cas de la reconnaissance de flux audio, cette translation a lieu uniquement sur les empreintes du signal à reconnaître, potentiellement dégradées, et pas sur les signaux originaux de référence lors de la construction de la base, et de la table de hachage.

L'apprentissage consiste en une procédure d'optimisation qui a pour but la minimisation des écarts moyens entre l'empreinte dégradée puis translatée d'une part, et son empreinte originale respective d'autre part. Le critère est alors donné par :

$$\mathcal{C} \triangleq \sum_{k=1}^{K} \mathcal{C}_k = \sum_{k=1}^{K} \frac{1}{N} \sum_{i=1}^{N} \left| X_k^*(c(i)) - \Big(X_k(i) + \tau(X(i))_k \Big) \right|^2, \tag{22}$$

où $k \leq K$ est l'indice de la composante, $i \leq N$ l'indice du point et $c(i) \leq C$ l'indice de la classe associée au point i. Ainsi $X^*(c(i))$ est l'empreinte originale associée à l'empreinte dégradée $X(i)$.

Maintenant, pour avoir une dépendance plus complexe entre la translation à appliquer et la position de l'empreinte dégradée X d'origine, nous faisons le choix de la placer dans l'espace quadratique, donné

par $Q(X)$. Enfin, pour avoir un modèle simple à résoudre, le vecteur de translation est donné par un produit matriciel de $Q(X)$ avec une matrice B de dimension $(K \times K_q)$ de la manière suivante :

$$\tau(X) = B\ Q(X). \tag{23}$$

Ainsi, même si la relation entre $\tau(X)$ et X est non-linéaire, celle avec $Q(X)$ est linéaire, ce qui nous permet d'appliquer une procédure simple d'optimisation par les moindres carrés.

Pour ce faire, remarquons dans un premier temps que les *sous-critères* $\mathcal{C}_k$, écarts moyens de la composante k, sont indépendants les uns des autres. Par conséquent, les lignes B_k de B sont tour à tour optimisées par minimisation des $\mathcal{C}_k$. Définissons : ϕ_k le vecteur $(N\times1)$ des données, Θ la matrice $(N\times K_q)$ du modèle (indépendante de k), et β_k le vecteur $(K_q\times1)$ du régresseur, tels que

$$(\phi_k)_i \triangleq X_k^*(c(i)) - X_k(i), \qquad \Theta_{i,q} \triangleq Q(X(i))_q, \qquad \text{et} \qquad \beta_k \triangleq B_k^T. \tag{24}$$

Les critères $\mathcal{C}_k$ de l'équation (22) se réécrivent alors :

$$\mathcal{C}_k = \frac{1}{N}\Big\|\phi_k - \Theta\ \beta_k\Big\|_2^2 = \frac{1}{N}\Big(\phi_k - \Theta\ \beta_k\Big)\Big(\phi_k - \Theta\ \beta_k\Big)^T, \tag{25}$$

et l'optimisation des moindres carrés, cf. e.g. [22], donne les vecteurs β_k minimisant $\mathcal{C}_k$:

$$\beta_k^* = \big(\Theta^T\Theta\big)^{-1}\big(\Theta^T\phi_k\big). \tag{26}$$

Par conséquent, la translation des empreintes dégradées X en Z permettant de recentrer optimalement les empreintes originales X^* au sens de $\mathcal{C}$ est donnée par :

$$Z = X + B_{\text{qc}}\ Q(X), \quad \text{avec} \quad B_{\text{qc}} = [\ \beta_1^*,\ \beta_2^*,\ \beta_3^*, \ldots \beta_K^*\ ]^T. \tag{27}$$

Cette méthode que nous nommons *Correction Quadratique*, QC, est donc appliquée sur les signaux à reconnaître, potentiellement dégradés, et non sur les signaux de référence, a priori sans dégradation. Rappelons que pour des raisons de faisabilité, la dimension d'origine des X doit être réduite pour avoir un espace quadratique de dimension K_q raisonnable. C'est pourquoi cette opération a lieu après la réduction de la méthode LDA ou LISA. Cet apprentissage fournit alors la matrice B_{qc} de dimension $(K\times K_q)$ utilisée pour la reconnaissance de flux audio dégradé.

3.4 Analyse en Composante Principale de Mahalanobis

La méthode proposée dans cette section a comme les méthodes LDA ou LISA le double but de réduire la dimension des empreintes tout en améliorant la robustesse aux dégradations sonores. Elle est soit une alternative aux méthodes LDA et LISA, soit une technique complémentaire. Nous verrons en section 5 qu'elle est dans ce travail utilisée de façon complémentaire, et elle sert a corriger une perte de performance après l'indépendance des variables opérée par l'ICA de la section 4.3. Nous faisons ici le choix de la présenter pour une utilisation générale.

Dans un premier temps, la méthode *standard* est présentée en sec. 3.4.1, puis une version *orthogonale* est détaillée en sec. 3.4.2. Contrairement à la première, elle garantit l'obtention d'une matrice de passage orthogonale, utile par exemple pour conserver la décorrélation des variables, cf. sec. 5.

3.4.1 MPCA standard

Commençons par parler de l'*Analyse en Composantes Principales* ou PCA, cf. e.g. [9]. Pour un ensemble de points donné, la PCA retourne du point de vue géométrique les directions qui maximisent la variance des nouvelles variables. Ici, nous proposons d'adapter le problème pour accroître encore la robustesse des empreintes aux dégradations sonores. Nous définissons alors les deux distributions suivantes :

1. La première est donnée pas les différences des empreintes des *positifs*, pour laquelle chaque vecteur $\mathcal{P}_i$ est la différence d'une empreinte dégradée $X(i)$ et de son empreinte originale associée $X^*(c(i))$, correspondant au même signal et au même instant d'analyse. Il s'agit donc de l'erreur engendrée par une dégradation sur les empreintes.

$$\mathcal{P}_i = X(i) - X^*(c(i)). \tag{28}$$

2. La seconde distribution correspond aux différences d'empreintes des *négatifs*, c'est-à-dire pour laquelle l'empreinte originale ne correspond pas au même signal. Il s'agit donc cette fois-ci d'un ensemble de vecteurs $\mathcal{N}_i$ représentatifs de l'écart entre empreintes ne venant pas du même signal.

$$\mathcal{N}_i = X(i) - X^*(j(i)), \quad \text{avec} \quad j(i) \neq c(i). \tag{29}$$

Le but ici est de trouver les directions qui à la fois : minimisent l'erreur engendrée par une dégradation, donnée par la variance de la première distribution $\mathcal{P}$ des positifs sur la nouvelle base ; et maximisent l'écart des empreintes pour des signaux et/ou instants différents, donné par variance de la seconde distribution $\mathcal{N}$ des négatifs. Cela aura pour effet de diminuer la sensibilité aux dégradations, et augmenter le caractère informatif des variables obtenues. Définissons alors les matrices de covariance des deux distributions

$$\begin{aligned} C_{\mathcal{P}} &\triangleq \mathrm{Var}(\mathcal{P}) &&= \mathrm{E}\left[\; (\mathcal{P} - \mathrm{E}(\mathcal{P})) \; (\mathcal{P} - \mathrm{E}(\mathcal{P}))^T \; \right], \\ C_{\mathcal{N}} &\triangleq \mathrm{Var}(\mathcal{N}) &&= \mathrm{E}\left[\; (\mathcal{N} - \mathrm{E}(\mathcal{N})) \; (\mathcal{N} - \mathrm{E}(\mathcal{N}))^T \; \right]. \end{aligned} \tag{30}$$

L'idée de la méthode que nous proposons ici vient de l'interprétation suivante de la LDA présentée en section 3.1 : *l'algorithme de la LDA est le même que celui de la PCA du nuage des centres de gravité des classes, de covariance B, où l'espace des individus est muni de la métrique dite de Mahalanobis associée à T^{-1}*, cf. [10]. Ainsi, la LDA permet de maximiser la variance d'une distribution, donnée par B, tout en minimisant celle d'une seconde distribution, donnée par T. Enfin, rappelons que sa solution s'obtient en diagonalisant la matrice $T^{-1}B$.

Par conséquent, la méthode que nous nommons *Mahalanobis-PCA* (MPCA) est obtenue en remplaçant T par $C_{\mathcal{P}}$, et B par $C_{\mathcal{N}}$. Alors les directions qui maximisent la variances des négatifs et minimisent la variance des positifs sont les vecteurs propres de $C_{\mathcal{P}}^{-1}C_{\mathcal{N}}$ associés aux valeurs propres les plus fortes. Dans un espace de départ de dimension J, avec $\mathcal{G}$ la matrice $(J \times J)$ des vecteurs propres, et $\mathcal{D}$ la matrice diagonale de J valeurs propres rangées par ordre décroissant, nous avons :

$$\mathcal{G}^{-1} \, (\, C_{\mathcal{P}}^{-1} \, C_{\mathcal{N}} \,) \, \mathcal{G} = \mathcal{D}, \tag{31}$$

et les K premières colonnes de $\mathcal{G}$ donnent la matrice P de projection sur le sous-espace dont les directions maximisent le critère donné. On a alors le changement de variable :

$$Z = P_{\mathrm{mpca}} \, X, \quad \text{avec} \quad (P_{\mathrm{mpca}})_{k,j} = \mathcal{G}_{j,k}, \quad \forall k \leq K \text{ et } \forall j \leq J. \tag{32}$$

3.4.2 MPCA Orthogonale

Cependant, alors que la PCA sur une distribution renvoie une matrice de passage orthogonale, cette MPCA ne retourne pas de matrice orthogonale, à moins que $C_{\mathcal{P}}$ soit unitaire. Or avoir une projection orthogonale peut être très utile dans certains cas. Nous verrons notamment que pour conserver la décorrélation des variables, après l'ICA de la section 4.3, l'orthogonalité du projecteur est nécessaire. C'est la raison pour laquelle nous définissons une version orthogonale, nommée *Orthogonal MPCA* ou OMPCA.

Pour garantir l'orthogonalité du changement de base, nous disposons de deux méthodes. La première consiste simplement à faire une orthogonalisation de Gram-Schmidt de la matrice $\mathcal{G}$ obtenue après diagonalisation de $C_{\mathcal{P}}^{-1}C_{\mathcal{N}}$. Avec g_j la j-ième colonne de $\mathcal{G}$, et $\overline{g}_j = g_j / \|g_j\|_2$ sa version normalisée, cette orthogonalisation est réalisée récursivement sur l'indice j de 1 à $J-1$, par l'opération suivante :

$$g'_i = g_i - \langle \, g_i, \, \overline{g}_j \, \rangle \, \overline{g}_j, \quad \forall i > j, \tag{33}$$

$$\text{puis} \quad \left\{ \begin{array}{l} g_i \leftarrow g_i, \quad \forall i \leq j, \\ g_i \leftarrow g'_i, \quad \forall i > j. \end{array} \right. \tag{34}$$

Tandis que les colonnes g_i pour $i \leq j$ de $\mathcal{G}$ ne changent pas, les colonnes suivantes sont modifiées par g'_i, leur projeté sur le sous-espace complémentaire à g_j, eq. (33). L'indice de récurrence j va de 1 à $J-1$. La nouvelle matrice $\mathcal{G}'$ est orthogonale, et remarquons que sa première colonne, la composante principale de la MPCA, ne change pas.

Malheureusement, même si la matrice obtenue est orthogonale, on constate pour J élevé que l'indice de Sobol associée à cette méthode, $S_{\mathrm{MPCA}}(g_j) = (g_j^T C_{\mathcal{N}} \, g_j)/(g_j^T C_{\mathcal{P}} \, g_j)$, n'est pas toujours décroissant quand j augmente. La seconde méthode est plus longue à calculer mais garantit la décroissance de l'indice de Sobol associé à la MPCA.

Cette fois-ci à chaque itération j, plutôt que de projeter les colonnes suivantes de $\mathcal{G}$ sur le sous-espace complémentaire à g_j, la MPCA est recalculée après la projection orthogonale des points des deux distributions $\mathcal{P}$ et $\mathcal{N}$ sur le sous-espace complémentaire à g_j, de dimension $J-j$. La matrice du projecteur P est alors mise à jour récursivement de sorte à tenir compte des projections successives. Pour plus de clarté, le code Matlab de la méthode est donné en fig. 3. Remarquons que la base orthogonale du sous-espace complémentaire à un vecteur y est obtenue par un factorisation QR. Enfin, une illustration donnée en figure 4 compare les directions obtenues par la PCA et l'OMPCA sur des nuages de points.

```
function  P = OMPCA( dist_P, dist_N )
% Dimension de l'espace d'origine, et tailles des distributions
[ J, p ]  = size( dist_P );
[ J, n ]  = size( dist_N );
% Centrage des distributions P et N
dist_P = dist_P - repmat( mean( dist_P, 2 ), 1, p );
dist_N = dist_N - repmat( mean( dist_N, 2 ), 1, n );
% Boucle de récurrence
for  j = 1:J-1
    % Calcul des matrices empiriques de covariance
    C_P   = 1/(p-1)* dist_P * dist_P';
    C_N   = 1/(n-1)* dist_N * dist_N';
    % Diagonalisation de C_P^-1 C_N
    [ G, Lambda ] = eig( C_P \ C_N  );
    % rq : ici G et Λ sont de dimension ((J-j+1)×(J-j+1)).
    % Récupération de la composante principale
    [ ~, Iprincipal ] = max( diag(Lambda) );
    g = G( :, Iprincipal );
    % Factorisation QR de g :
    [ Q, a ]  = qr( g );
    Q         = Q / a(1);
    % rq : la première colonne de Q est égale à g, les suivantes
    %      constituent une base orthogonale du sous-espace complémentaire.
    % Initialisation ou mise à jour de la matrice de passage P :
    if  j == 1
        P = Q';
    else
        P = [ eye( j-1 )          , zeros( j-1, J-j+1 ) ;
              zeros( J-j+1, j-1 ) , Q'                  ] * P;
    end
    % Projection des distributions P et N sur le sous-espace complémentaire,
    dist_P = Q( :, 2:end )' * dist_P;
    dist_N = Q( :, 2:end )' * dist_N;
end
% Fin du code :
return
```

FIGURE 3 – Code Matlab de l'OMPCA. Ici les deux arguments en entrée sont `dist_P` et `dist_N` les matrices contenant les nuages de points des deux distributions, respectivement $\mathcal{P}$ et $\mathcal{N}$. Elle retourne en sortie la matrice de passage P, de dimension $(J \times J)$, permettant la projection sur un sous-espace de dimension K en sélectionnant les K premières lignes.

Pour réduire la dimension de l'espace des empreintes sonores de J à K à partir de la matrice P de passage obtenue, il suffit de conserver les K premières lignes, on a alors le changement de variables :

$$Z = P_{\text{ompca}}\, X, \quad \text{avec} \quad (P_{\text{ompca}})_{k,j} = P_{k,j}, \quad \forall k \leq K \text{ et } \forall j \leq J. \tag{35}$$

Cette OMPCA peut être aussi utile pour des applications autres que la réduction de dimension des empreintes sonores. Notons que l'un des problèmes de la PCA est de savoir sur la variance des variables obtenues est dues à de l'information utile, ou à du bruit. Ce qui pose entre autres la question de la normalisation ou non des variables d'origine. Avec la MPCA ou l'OMPCA, s'il est possible de caractériser le bruit dans le même espace euclidien, alors on peut obtenir les directions qui réduise la variance du bruit et augmentent celle de l'information. C'est ce que nous faisons ici, la distribution $\mathcal{P}$ caractérise le bruit, tandis que la distribution $\mathcal{N}$ caractérise d'une certaines manière l'information que l'on souhaite mettre en valeur.

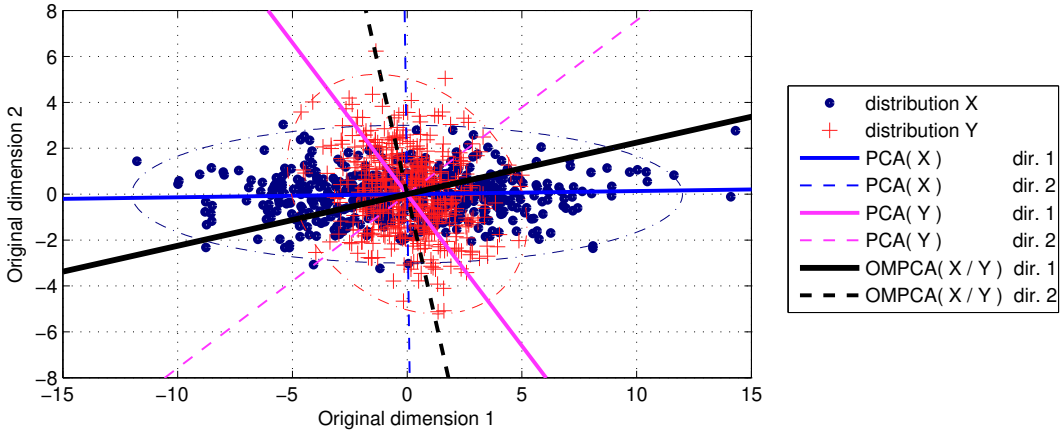

FIGURE 4 – Comparaison entre la PCA et l'OMPCA, en dimension 2.

4 Conditionnement et préparation des données

En section précédente nous avons présenté plusieurs méthodes statistiques visant à améliorer la robustesse aux altérations sonores en projetant les empreintes sur un sous-espace moins sensible, et sélectionnant en même temps l'information pertinente du point de vue musical. Cette nouvelle section présente d'autres transformations qui visent à reconditionner les données, soit pour garantir qu'il n'existe pas de relation linéaire entre les variables, soit pour améliorer le traitement postérieur, tel que la binarisation des données pour le calcul des codes de hachage.

4.1 Réjection des composantes mal conditionnées

Avant d'appliquer une analyse linéaire discriminante et autres méthodes statistiques, il est important de s'assurer que les composantes des empreintes sonores de dimensions 1056 ne soient pas linéairement dépendantes, même faiblement. Cela est notamment nécessaire pour garantir du bon conditionnement des matrices à inverser.

Par exemple, avec X le vecteur aléatoire représentant les $J = 1056$ composantes des empreintes sonores de grande dimension obtenue en sec. 2, si les variables ont une dépendance linéaire, alors il existe un vecteur $g \in \mathbb{R}^J$, tel que $g^T X = \sum_j g_j X_j = 0$. Alors la matrice $R = E(XX^T)$ semblable à la matrice de covariance de X, mais sans centrage, est telle que : $g^T R g = 0$. On comprend alors que le principe consiste à trouver les vecteurs propres de R de valeurs propres non-nulles, fournissant donc une base du sous-espace où les variables n'ont pas de relation linéaire.

Commençons par construire une matrice $\mathbf{X}$ de dimension $(J \times N)$ constituée de N réalisations de X, avec $N \gg J$. Cet ensemble d'analyse est ici constitué d'un grand nombre d'empreintes sonores calculées sur des signaux différents. Selon la *Décomposition en Valeurs Singulières* (SVD en anglais) on peut exprimer la matrice $\mathbf{X}$ sous la factorisation suivante :

$$\mathbf{X} = USV^T, \tag{36}$$

avec U matrice unitaire de taille $(J \times J)$, V matrice de taille $(N \times J)$ dont les colonnes sont orthogonales deux à deux, et S matrice diagonale de taille $(J \times J)$ constituée des valeurs singulière (non-négatives) rangées par ordre décroissant. En découpant ces matrice par blocs selon la nullité des valeurs singulière, on obtient

$$\mathbf{X} = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} S_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}. \tag{37}$$

Avec j_0 le nombre de valeurs singulières non-nulles, la matrice U_1 de taille $(J \times j_0)$ fournit alors une base orthogonale de l'espace image de $\mathbf{X}$, tandis que V_2 de taille $(N \times (J - j_0))$ donne une base orthogonale du noyau de $\mathbf{X}$. Par conséquent, avec $P_{iccr} \triangleq U_1^T$ comme matrice de projection, pour supprimer toutes dépendances linéaires de X, il suffit d'appliquer le changement de variables :

$$Z = P_{iccr}\, X, \quad \text{avec} \quad P_{iccr} = U_1^T. \tag{38}$$

Dans notre cas, pour assurer un bon conditionnement des matrices à inverser, nous faisons le choix de supprimer non seulement les composantes associées aux valeurs propres nulles, mais aussi celles dont les valeurs propres sont inférieures à un seuil dépendant de la valeur maximale $s_{max} = s_1$. Nous appelons alors cette opération : *Réjection des Composantes Mal Conditionnées*, ou ICCR en anglais, pour *Ill-Conditionned Component Rejection*.

En pratique, nous avons utilisé une valeur très grande $N = 150\,000$ afin d'avoir une bonne représentativité. Pour se faire, 5 empreintes ont été calculés à des instants différents, pour chacun des 30 000 morceaux choisi dans la base BeeMusic, disponible à l'IRCAM dans le cadre du projet. Notons que nous avons choisi un petit nombre d'empreintes par morceaux, afin d'éviter des similarités dues à d'éventuelles répétitions. De plus, ajoutons que puisqu'il s'agit ici seulement d'étudier la dépendance linéaire, seules les empreintes originales ont été utilisées, donc sans dégradation.

En résultat de cette ICCR appliquée aux empreintes sonores de sec. 2, on remarque que 30 composantes sont rejetées, conduisant à une nouvelle représentation des empreintes de taille 1026. Il est intéressant de noter que ces 30 composantes sont toutes associées à des valeurs propres nulles, provenant d'une symétrie de la transformée de Fourier 2D. En effet, bien qu'ayant supprimé la moitié de la représentation, il demeure une symétrie sur les axes verticaux à la fréquence 0 et à la fréquence de Nyquist du log-temps (cf. fig. 1-h, première et dernière colonnes). En outre, les 1026 valeurs propres restantes ont toutes une valeur très supérieure au seuil fixé qui est de $s_{max}10^{-5}$. Par conséquent, cela assure un très bon conditionnement des données pour la suite.

Remarquons que cette ICCR est très semblable à une *Analyse en Composantes Principales*, avec cependant deux différences minimes : le but n'est pas de conserver un minimum de variables expliquant les données, mais d'en conserver un maximum garantissant un bon conditionnement. Et surtout en pratique, la matrice $\mathbf{X}$ n'est pas centrée. Rappelons notamment que les composantes de X sont toutes non-négatives puisqu'elles proviennent du module de la transformée de Fourier 2D, cf. sec. 2.

Enfin, précisons que cette opération a systématiquement lieu après le calcul des empreintes sonores de sec. 2. Si bien que pour les différentes méthodes statistiques présentées en sec. 3, la dimension J de l'espace de départ est inférieure à 1026.

4.2 Blanchiment des données

Le blanchiment des données, parfois nommé ZCA, permet par une simple transformation affine d'obtenir une nouvelle représentation où les nouvelles variables sont : de moyenne nulle, de variance unité et décorrélées entre elles. Soit Z le vecteur des nouvelles variables obtenues par le changement : $Z = P_{\text{zca}}\, X + t_{\text{zca}}$, avec P_{zca} une matrice $(J \times J)$, et t_{zca} un vecteur $(J \times 1)$ pour le centrage. Alors, la propriété souhaitée du blanchiment est :

$$R_Z \triangleq \mathrm{E}(ZZ^T) = I_J. \tag{39}$$

Cette équation résume le fait que les composantes de Z sont centrées ($\mathrm{E}(Z_j) = 0$), de variance unité ($\mathrm{E}(Z_j Z_j) = 1$), et décorrélées deux à deux ($\mathrm{E}(Z_j Z_k) = 0$, pour $j \neq k$), Pour commencer la recherche du changement de variable, centrons chaque composante de X en 0 :

$$Y \triangleq X - t_0, \quad \text{avec} \quad t_0 \triangleq \mathrm{E}(X). \tag{40}$$

Ensuite, on définit un changement de base $Z = PY$ quelconque, et on exprime R_Z en fonction de Y :

$$R_Z = \mathrm{E}(PYY^TP^T) = P\,\mathrm{E}(YY^T)P^T = PR_YP^T. \tag{41}$$

Puisque R_Y est symétrique et définie positive, elle possède une racine carrée $R_Y^{1/2}$ symétrique et définie positive, et donc inversible, telle que $R_Y = R_Y^{1/2}R_Y^{1/2}$. Ainsi, choisir $P = R_Y^{-1/2}$ permet de retrouver la propriété souhaitée :

$$R_Z = PR_YP^T = I_J. \tag{42}$$

Remarquons que la racine carré de $R_Y = \mathrm{E}(YY^T)$ s'obtient comme d'habitude, par diagonalisation ou décomposition en valeurs singulières de R_Y :

$$UR_YV^T = S \quad \Rightarrow \quad R_Y^{-1/2} = US^{-1/2}U^T, \tag{43}$$

où $S^{-1/2}$ est la matrice diagonale d'éléments $1/\sqrt{s_j}$ avec s_j les valeurs singulières. En conséquence le changement de variable qui donne le blanchiment de X, est donné par :

$$Z = P_{\text{zca}}\, X + t_{\text{zca}}, \quad \text{avec} \quad P_{\text{zca}} = R_Y^{-1/2}, \quad \text{et} \quad t_{\text{zca}} = -P_{\text{zca}}t_0. \tag{44}$$

4.3 Analyse en Composantes Indépendantes

L'indépendance de variables aléatoires est une condition plus forte que la décorrélation. En effet, alors que la décorrélation de variables x_1 et x_2 se traduit de manière générale par

$$\mathrm{E}(x_1 x_2) = \mathrm{E}(x_1)\,\mathrm{E}(x_2), \tag{45}$$

l'indépendance est que pour toute fonction h,

$$\mathrm{E}\,[\,h(x_1)\,h(x_2)\,] = \mathrm{E}\,[\,h(x_1)\,]\;\mathrm{E}\,[\,h(x_2)\,]\,. \tag{46}$$

Evidemment, l'indépendance implique la décorrélation, il suffit de prendre $h(x) = x$.

4.3.1 Indépendance pour le hachage

Cette propriété d'indépendance est très intéressante dans le cas de l'indexation par une technique à base d'empreintes sonores et de table de hachage. Dans la plupart des techniques, une *fonction de hachage* produit des codes dits *codes de hachage* en fonction des empreintes sonores que l'on nomme *clef de hachage*. Ces codes sont des nombres entiers bornés. Alors, avec cette méthode générale, l'identifiant du morceau est ajouté dans l'élément de la *table de hachage* dont le numéro est le code. Il s'agit donc d'un index inversé.

Cependant, si la fonction de hachage ou les clefs sont mal conçus, la table peut être remplie de façon peu uniforme. Si c'est le cas, la performance de la reconnaissance peut significativement décroître, parce que les codes les plus souvent produits contiennent peu d'information, et les codes les moins produits portent d'avantage d'information mais sont très peu utilisés. Avec un remplissage uniforme, on garantit la bonne répartition de la pertinence des codes. Dans le système précédent, IrcamAudioID, ce problème était très marqué et seule 10% de la base était réellement utile.

Dans ce nouveau travail, les composantes des empreintes sonores réduites de dimension K sont chacune binarisées par rapport à 0, cf. [14, sec. 5] pour plus de détails. Ainsi, avec z_k la k-ième composante de l'empreinte réduite Z, la fonction de binarisation h donne la composante binaire b_k du code :

$$b_k = h(z_k) = \begin{cases} 0, & si \quad z_k \leq 0, \\ 1, & si \quad z_k > 0. \end{cases} \tag{47}$$

Le code de hachage est alors donné bit-à-bit par $\Gamma = (\,b_1,\, b_2,\, b_3,\, \ldots\, b_K\,)$. Par conséquent, l'indépendance des variables continues z_k garantit l'indépendance statistique des variables binaires b_k. Alors toutes les valeurs de Γ entre 0 et $2^K - 1$ ont la même probabilité de se produire, si bien que les éléments de la table de hachage seront uniformément remplis.

Remarquons que finalement dans notre approche, le hachage utilisé est plus complexe que celui résumé ici. Néanmoins, le bénéfice de l'indépendance des variables reste justifié.

4.3.2 Principe de l'ICA

Nous résumons maintenant la technique permettant d'obtenir par une transformation affine du type $Z = P_{\mathrm{ica}}\,X + t_{\mathrm{ica}}$, des composantes Z_k indépendantes, centrées et de variance unité. Nous ne détaillons pas l'algorithme qui vient de la littérature, mais nous présentons les principes, pour en comprendre les limites.

Tout d'abord considérons des variables aléatoires Y_j centrées et statistiquement indépendantes. Si leur densité de probalité n'est pas gaussienne, alors on sait par le *théorème central limite* que toute combinaison linéaire des Y_j produit une nouvelle variable aléatoire dont la densité se rapproche d'une gaussienne. Par conséquent, si les variables observées X_j sont obtenues par mélange des Y_j, sous la forme d'un changement de base $X = AY$, avec A une matrice de rang plein de dimension $(J \times J)$, alors ces variables X_j sont d'avantage gaussiennes que les Y_j, et perdent l'indépendance (sauf si A est une matrice de permutation).

L'idée de base de l'*Analyse en Composantes Indépendantes*, ou ICA, est en quelques sortes de faire un mélange inverse de sorte à revenir, à une permutation prêt, aux variables non-gaussiennes de départ, et donc indépendantes. Ce fait est important, parce que l'ICA ne fonctionne en principe que sous l'hypothèse où les variables indépendantes de départ sont non-gaussiennes. Néanmoins, dans le cas où les variables de départ Y_k sont gaussiennes et que A est orthogonale, alors les variables X_j ont la même distribution, et sont aussi indépendantes. Cela ne pose pas de réel problème, et l'ICA reste applicable sans précaution préalable.

L'ICA consiste donc à déterminer une matrice de passage B qui permet de maximiser le caractère *non-gaussien* des composantes Z_j données par le vecteur $Z = BX$. Pour mesurer le caractère *non-gaussien* d'une variable, il existe plusieurs possibilités.

La première est le Kurtosis défini par

$$\mathrm{Kurt}(z_j) \triangleq \mathrm{E}(z_j^4) - 3\,\mathrm{E}(z_k^2)^2. \tag{48}$$

Pour une variable gaussienne il vaut précisément 0, et est non-nul pour "presque" toute autre distribution. Ainsi, un premier critère de *non-gaussiennité* est $\mathcal{C}_{\mathrm{ng}} \triangleq |\mathrm{Kurt}(z_j)|$.

Une autre possibilité est la *neg-entropie*, qui est la notion inverse de l'entropie en théorie de l'information. L'entropie permet de quantifier l'information portée par une variable aléatoire. Pour un vecteur aléatoire Z de taille $(J \times 1)$ et de densité $f(Z)$, elle vaut :

$$H_Z = \int_{\mathbb{R}^J} f(Z)\ \log(f(Z))\ dZ. \tag{49}$$

Une propriétés importante exploitée par l'ICA est que parmi toutes les distributions possibles de même variance, la distribution gaussienne a l'entropie la plus grande. Par conséquent, avec H_G l'entropie d'une gaussienne de même variance que Z, un nouveau critère de *non-gaussiennité* est donné par la *neg-entropie* : $\mathcal{C}'_{\mathrm{ng}} \triangleq \overline{H}_Z \triangleq H_G - H_Z$. Elle est non-négative, nulle pour une variable ou gaussienne, et est d'autant plus élevée que Z est éloignée d'une distribution gaussienne.

4.3.3 Algorithme *Fast*-ICA pour les empreintes sonores

A partir du vecteur X des variables d'observation, l'algorithme *fast*-ICA est un algorithme "rapide" pour l'obtention de la transformation affine qui fournit le vecteur Z des variables indépendantes, centrée et de variance unité. Voir par exemple [8] pour plus de détail. Après centrage et blanchiment des données, il est basé sur une maximisation itérative d'une version approchée de la *neg-entropie*.

Dans ce travail, la distribution X est donnée par un nuage de 150 000 empreintes sonores obtenues après ICCR, et éventuellement d'autres méthodes statistiques décrites en section 3. Comme pour les autres méthodes, ce grand nombre permet une bonne représentativité des empreintes sonores. Précisons que cette base d'apprentissage ne contient que les empreintes non dégradées. Il ne s'agit pas ici d'améliorer la robustesse aux altérations sonores, mais d'une préparation des variables pour optimiser le remplissage de la table de hachage.

On utilise l'implémentation de [15] qui nous retourne alors la transformation affine suivante :

$$Z = P_{ica} X + t_{ica}, \tag{50}$$

avec P_{ica} une matrice de passage de taille $(J \times J)$ et t_{ica} un vecteur de taille $(J \times 1)$. Remarquons qu'il n'y a pas de réduction de dimension.

4.4 Transformée de Hadamard

Certaines des méthodes statistiques présentées en section 3 retournent des vecteurs d'empreintes réduites, où les composantes sont rangées par ordre de pouvoir discriminant décroissant. C'est le cas des méthodes : ICA, LISA, QDA, MPCA et OMPCA. Sans modification, les bits des codes de hachage Γ sont alors rangés par ordre de robustesse décroissante. Cependant, en raison du hachage approximatif que nous faisons, voir [14, sec. 5], il semble intéressant d'avoir des variables de robustesse égale.

Ainsi, pour égaliser la robustesse des composantes des empreintes réduites avant la binarisation en b_k, nous proposons un dernier changement de base : la transformation de Hadamard, cf. e.g. [7]. Cette transformation est donnée par une matrice orthogonale dont les éléments ont tous la même valeur en module. En dimension K, ils valent $\pm 1/\sqrt{K}$, où les signes sont déterminés pour assurer l'orthogonalité de la matrice. Par exemple, pour $K = 4$, on a :

$$H_{hadamard} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}. \tag{51}$$

Le changement de variable est alors donné par

$$Z = P_{\mathrm{hadamard}} X. \tag{52}$$

Le choix de cette transformation vient du simple fait que tous les coefficients ont la même valeur en module. Par conséquent, chaque composante X_k a la même influence sur les variables Z_k, si bien que ces nouvelles variables ont en principe une sensibilité égale aux altérations sonores.

Remarquons d'une part que si les variables de départ X sont indépendantes, alors les variables Z perdent cette indépendance. Néanmoins elles restent décorrélées grâce à l'orthogonalité de P_{hadamard}. Nous verrons en section suivante que la décorrélation est suffisante. D'autre part, précisons que la transformée de Hadamard existe pour un certain nombre de dimension. Par exemple, l'ensemble des K acceptables jusque 1024 sont :

$$\{\ 1, 2, 4, 8, 12, 16, 20, 24, 32, 40, 48, 64, 80, 96, 128, 160, 192, 256, 320, 384, 512, 640, 768, 1024\ \}\,.$$

5 Stratégies pour l'indexation et évaluations

Nous proposons dans cette section plusieurs évaluations aidant au choix des transformations utilisées pour la réduction *discriminante* des empreintes sonores.

Il faut préciser que de nombreux tests informels ont été réalisés durant le projet ; et nous nous sommes efforcés d'en formaliser quelques uns de sorte à proposer une évaluation claire des techniques présentées précédemment. Néanmoins, tous les tests ne peuvent pas être présentés parce que soit les données sont perdues, soit ils utilisent des méthodes ou variantes abandonnées et non-pertinentes. Par conséquent, certains des choix faits ne peuvent pas être clairement explicités.

Par exemple, la méthode LISA, assez semblable à la LDA, est testée lors du dernier test uniquement. La raison est que l'idée de cette méthode est venue vers la fin du travail ; et il manque les résultats du test informel nous ayant fait préférer cette méthode LISA à la LDA.

Puisque chaque réduction d'empreintes testée inclut plusieurs des techniques présentées, nous définissons des scénarii correspondant chacun à une série de transformations. Par exemple le scénario **ICCR + LDA + ICA** correspond à la réduction composée de l'ICCR pour le bon conditionnement des composantes, puis la LDA pour la discrimination des classes, et l'ICA pour l'indépendance des variables. Si toutes les transformations sont affines, alors par factorisation la réduction des empreintes est opérée par un simple produit matriciel. Pour l'exemple présenté, on a alors :

$$Z = \mathbf{P}_{\text{totale}} \times X + \mathbf{t}_{\text{totale}}, \tag{53}$$
$$\text{avec} \quad \begin{cases} \mathbf{P}_{\text{totale}} &= P_{\text{ica}} \times P_{\text{lda}} \times P_{\text{iccr}}, \\ \mathbf{t}_{\text{totale}} &= t_{\text{ica}}, \end{cases}$$

Notons que si une méthode de l'espace quadratique est utilisée, alors le temps de calcul est un peu plus long parce que la factorisation est moins simple.

5.1 Premiers tests basés sur des critères intermédiaires

Nous présentons dans cette sous-partie un certain nombre d'évaluations. Elles sont toutes basées sur deux critères intermédiaires prédisant la performance de l'indexation audio pour un choix donné de transformations. Un certain nombre de tests effectués durant le travail sont manquant ici, nous avons choisi les évaluations les plus informatives pour comparer les méthodes.

5.1.1 Critères

Pour évaluer la performance de chaque série de transformations, nous devons définir un critère d'évaluation. Pour éviter de calculer le processus d'indexation complet, ce critère intermédiaire a pour but de prédire le taux de performance final. Pour cela, nous avons repris les deux critères déjà utilisés pour la recherche des valeurs de paramètres de [13, sec. 5]. Nous donnons maintenant un résumé.

Information de Fisher Le premier des deux critères est basé sur l'information de Fisher ou l'indice de Sobol qui résume la discrimination de groupes de points. Ces variables scalaires sont ici égales aux normes euclidiennes des points des distributions $\mathcal{P}$ et $\mathcal{N}$ de sec. 3.4.2, *positifs* et *négatifs*. On définis donc les variables : $D_{\mathcal{P}} = \|\mathcal{P}\|_2$ et $D_{\mathcal{N}} = \|\mathcal{N}\|_2$. Alors $D_{\mathcal{P}}$ représente la distance d'une empreinte dégradée par rapport à l'originale associée, et $D_{\mathcal{N}}$ la distance d'une empreinte dégradée par rapport à l'empreinte d'un autre signal ou autre instant d'analyse.

La robustesse aux dégradations se traduit alors par des distances $D_{\mathcal{P}}$ faibles et des distances $D_{\mathcal{N}}$ élevées. Pour résumer cela dans un coefficient unique, nous utilisons ici l'information de Fisher permettant

de décrire combien ces deux distributions sont séparées, cf. par exemple fig. 5. Cette information de Fisher constitue le premier critère utilisé. Elle est calculée séparément pour chaque bande, et une version commune est faite.

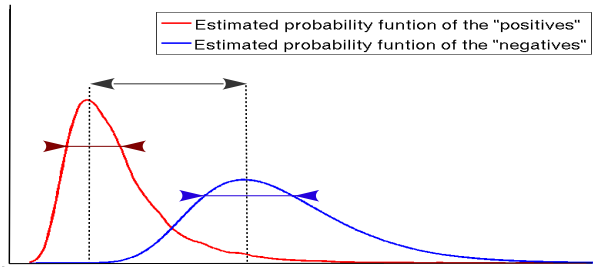

FIGURE 5 – Discrimination des deux distributions : $D_{\mathcal{P}}$ et $D_{\mathcal{N}}$. Une bonne discrimination est donnée par une variance faible (variance intra-classe) et un fort écart des moyennes (variance inter-classe). Aussi, la moyenne de la distribution $D_{\mathcal{P}}$, des positifs, doit être inférieure à celle de la distribution $D_{\mathcal{N}}$.

Probabilité de succès des k-nn Un second critère est utilisé pour prédire les performances de reconnaissance. Dans le travail initial IrcamAudioID de [19, 20], la reconnaissance était basée sur un calcul des *k-plus proches voisins* (k-nn), alors nous prédisons ici le taux de réussite du problème. Pour se faire, normalisons $D_{\mathcal{P}}$ et $D_{\mathcal{N}}$ pour obtenir les densités de probabilité de la distance euclidienne pour deux empreintes correspondante au même signal, $D_{\mathcal{P}}$, et ne correspondant pas, $D_{\mathcal{N}}$. Ainsi, ce nouveau critère donne la probabilité de réussite, c'est-à-dire la probabilité que l'empreinte du signal original $Z^*(c(i))$ est parmi les k-plus proche voisins de $Z(i)$, sachant une base contenant V empreintes originales.

Soient $P_{\mathcal{P}}(\delta)$, et $P_{\mathcal{N}}(\delta)$ respectivement, la densité de probabilité que la distance d'un positif, resp. négatif, vale δ. Ces densités sont obtenues par estimation et normalisation sur les distributions $\mathcal{P}$ et $\mathcal{N}$, cf. fig. 5. Alors on a P_V^k la probabilité que parmi V tirages de négatifs, exactement k ont une distance inférieure à la distance du positif :

$$P_V^k = C_V^k \int_{-\infty}^{\infty} P_{\mathcal{P}}(\delta) Q_{\mathcal{N}}(\delta)^k \left(1 - Q_{\mathcal{N}}(\delta)\right)^{V-k} d\delta, \quad \text{avec} \quad Q_{\mathcal{N}}(\delta) = \int_{-\infty}^{\delta} P_{\mathcal{N}}(x) dx. \tag{54}$$

Alors le critère associé vaut : $\bar{Q}_V^K = \sum_{k=0}^{K} P_V^k$, il s'agit de la probabilité que parmi V tirages de négatifs, moins de K ont une distance inférieure à la distance du positif. Ce critère traduit donc la probabilité de succès pour un problème K-nn avec une base de V empreintes.

Remarquons que pour V et K relativement grand, le calcul du coefficient du binôme C_V^k pose des problèmes de résolution numérique avec l'implémentation de Matlab, fonction `nchoosek`. Même si nous avons pu résoudre ce problème d'implémentation, la lenteur du calcul impose de les choisir petits. Dans les tests nous choisissons alors $V \approx 1000$ et K de l'ordre de 0 à 2.

Remarquons que les deux distributions $\mathcal{P}$ et $\mathcal{N}$ sont ici obtenues après toutes les transformations testées, et réduction sur le sous-espace final de dimension K. De plus, pour évaluer en même temps l'influence de K, sa valeur n'est pas fixe, mais varie entre 1 et 40, voir l'abscisse des figures.

5.1.2 Evaluations des méthodes de l'espace quadratique

En premier lieu, testons les deux méthodes de l'espace quadratique, des sections 3.3.1 et 3.3.2. En utilisant la procédure d'évaluation décrite précédemment, nous avons comparé la réduction de dimension en utilisant les scénarii suivant :

- ✓ **ICCR+LDA** : Dans ce premier scénario, seule la LDA est testée, de sorte à avoir un témoin sans raffinement. Evidemment, la projection ICCR est préalablement utilisée pour résoudre le problème de dépendance linéaire, cf. sec. 4.1.
- ✓ **ICCR+LDA+QDA** : Ici, après une ICCR préalable et une LDA qui réduit les empreintes à une dimension intermédiaire de $K = 48$, la QDA est appliquée. Avec une dimension de l'espace quadratique de 1200, nous réduisons à nouveau les dimensions à K', qui varie, voir l'abscisse.
- ✓ **ICCR+LDA+QC** : Encore après la suite ICCR+LDA, la dimension intermédiaire est de $K = 48$. Cette fois-ci la *Correction Quadratique* est appliquée avec l'espace quadratique de dimension 1200 pour le calcul de la translation correctrice. Enfin, encore une fois, la dimension de la réduction finale varie pour voir son influence.

Rappelons que pour avoir un espace quadratique de dimension raisonnable, les méthodes de l'espace quadratique sont systématiquement faites après LDA.

Premièrement en regardant en détail la figure 6, comparé à la LDA seule on voit une nette dégradation des performances de la QDA que ce soit pour l'information de Fisher que pour le critère des k-nn. La QC semble donner des résultats acceptables pour l'information de Fisher, mais il apparaît clairement qu'elle échoue autant que la QDA pour le critère k-nn.

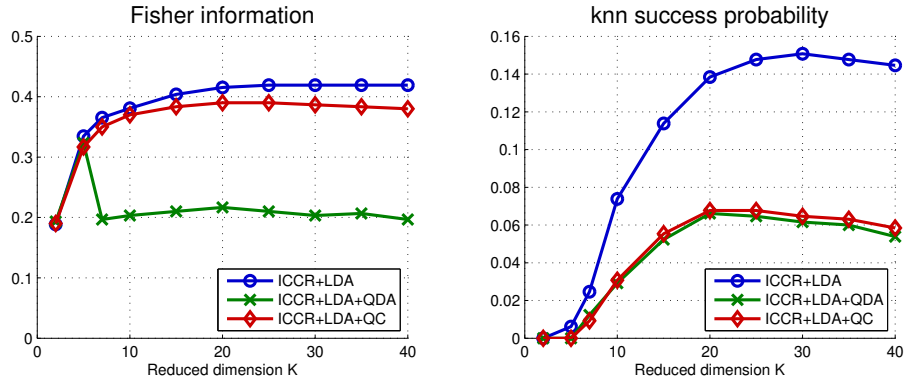

FIGURE 6 – Evaluation et comparaison de la LDA seule, et des méthodes de l'espace quadratique. Les deux critères relatifs à l'information de Fisher et à l'algorithme k-nn sont donnés en fonction du nombre K de variables utilisées.

Nous ne comprenons pas précisément les raisons de cet échec, et une étude plus poussée devrait être faite. Néanmoins ces résultats justifient le fait que nous n'utiliserons aucune de ces deux méthodes. De plus, le calcul des empreintes dans l'espace demande plus du temps comparé à une suite de transformations affines qui peuvent être factorisée. Cela renforce donc notre choix.

En regardant les résultats de la LDA seule, nous remarquons pour l'information de Fisher un palier qui commence à environ $K = 20$, et un maximum atteint à $K = 30$ pour le critère k-nn. Cette observation est très intéressante car elle justifie le fait d'utiliser une dimension finale de cet ordre là.

5.1.3 Evaluations de la PCA et de l'ICA

Nous cherchons à étudier ici le bénéfice de l'ICA et de la PCA, présentées en section 4.3 et 3.4. Nous définissons donc les trois scénarii suivant :

- ✓ **ICCR+LDA** : Il s'agit ici du même scénario qu'en section 5.1.2.
- ✓ **ICCR+LDA+PCA** : Après la suite ICCR+LDA, une analyse en composante principale est testée. Notons que la PCA est exécutée sur un sous-espace de dimension intermédiaire 80, après sélection des directions les plus discriminantes de la LDA. Cependant, au final, l'évaluation est fait sur les K composantes principales de la PCA, avec K variant.
- ✓ **ICCR+LDA+ICA** : Cette fois-ci, l'ICA est testée après LDA+ICCR. Encore une fois la dimension intermédiaire est 80. L'abscisse représente K, le nombre de composantes utilisée après l'ICA pour l'évaluation des critères.

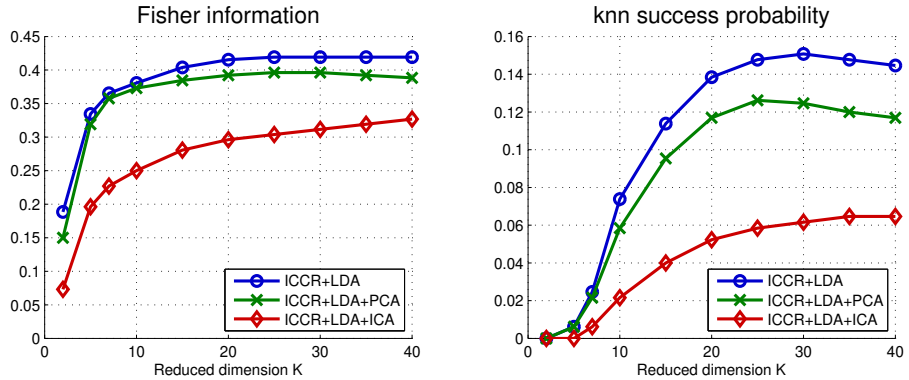

FIGURE 7 – Evaluation et comparaison de la LDA, de la PCA et de l'ICA après LDA.

Premièrement, selon la figure 7, on observe que la PCA n'apporte aucun bénéfice comparé à la LDA seule. Cela n'est pas étonnant parce qu'il n'est pas nécessairement judicieux de maximiser la variance des composantes sans prise en compte des dégradations. Deuxièmement, on constate que l'ICA provoque une détérioration très significative des performances. Rappelons que le but de l'ICA est le bon remplissage de la table de hachage, or cela n'est pas pris en compte dans les deux critères proposés, information de Fisher et critère des k-nn, nous ne nous attendions donc pas à une amélioration des performances. Néanmoins la détérioration est relativement surprenante.

Pour comprendre cela, notons que l'ICA n'ordonne pas les directions en fonction d'un critère. Il s'agit seulement de la séparation de composantes indépendantes avec un ordre qui n'agit pas sur les résultats présentés ici. Sur les 80 variables du sous-espace intermédiaire, il n'y a pas de raison de prendre les K première ou les K dernières. Cette remarque est importante parce qu'elle a justifié le développement de la méthode OMPCA, qui fournit une transformation orthogonale avec un ordonnancement des composantes de la plus discriminante à la moins discriminante au sens du critère de la MPCA. Nous verrons le bénéfice apporté en prochaine section.

5.1.4 Evaluations de l'OMPCA

Comme il a été discuté en section 4.3, l'intérêt de l'indépendance des variables est le bon remplissage de la table de hachage. Nous verrons que c'est effectivement le cas plus bas. Cependant, la perte de performance au sens des deux critères étudiés ici nous ont poussés à ajouter une étape en sortie de l'ICA. Il s'agit de l'OMPCA qui a un objectif similaire à la LDA et LISA, et peut les remplacer. Mais nous l'utilisons ici en complément à l'ICA pour compenser la perte de performance de cette dernière méthode. Puisque l'ICA n'ordonne pas les composantes par pouvoir discriminant, l'OMPCA permet en quelque sorte de remettre de l'ordre dans les variables.

Néanmoins, ajouter une transformation linéaire après l'ICA, nous fait perdre l'indépendance des variables. Cela n'est pas forcément grave parce que grâce à l'orthogonalité de la matrice, la décorrélation des variables est préservée.

Nous réaffichons en fig. 8 les résultats des scénarii **ICCR+LDA** et **ICCR+LDA+ICA** pour mieux comparer l'apport de l'OMPCA. Le nouveau scénario est donc **ICCR+LDA+ICA+OMPCA**. Après l'ICA qui retourne 80 composantes indépendantes, l'OMPCA permet de réordonner les variables en fonction du pouvoir discriminant. Alors l'évaluation est faite dans le nouveau sous-espace, en sélectionnant les K premières composantes, K variant.

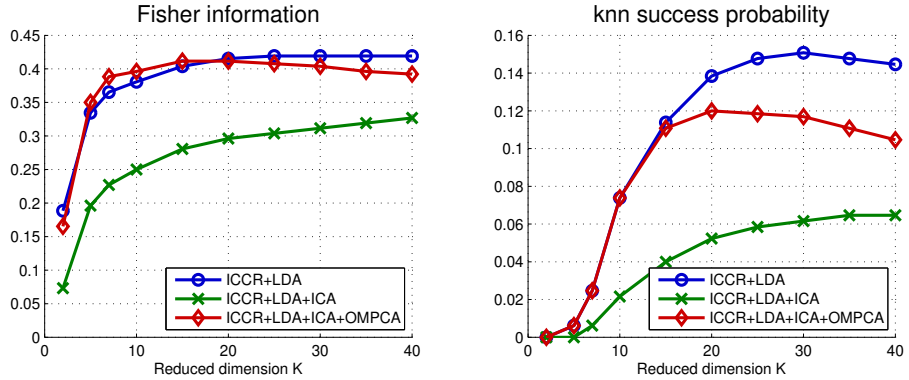

FIGURE 8 – Evaluation du bénéfice de l'OMPCA après l'ICA.

Nous remarquons ici un regain très significatif des performances grâce à l'OMPCA. On constate même que pour K entre 5 et 15, les résultats semblent meilleurs que la LDA d'origine. Ces résultats justifient clairement l'utilisation de l'OMPCA après l'ICA. Cependant, on constate avec le critère k-nn que la LDA seule semble rester malgré tout la meilleure méthode, mais nous verrons plus tard que des variables indépendantes ou décorrélées produisent de meilleurs résultats quand les critères d'évaluation sont plus proche de l'indexation.

5.1.5 Influence de la transformée de Hadamard

Nous étudions ici l'influence de la transformée de Hadamard après l'OMPCA. Nous reprenons donc les scénarii **ICCR+LDA** et **ICCR+LDA+ICA+OMPCA** à titre de comparaison, et ajoutons le scénario **ICCR+LDA+ICA+OMPCA+HT** avec la transformée de Hadamard en fin de chaîne.

Donnons des précisions sur les dimensions intermédiaires parce que cela se complique : comme d'habitude la dimension d'origine est de 1056, qui devient 1026 après l'ICCR ; puis la LDA permet une première réduction à une dimension intermédiaire de 80 ; l'ICA rend les variables indépendantes sans changer la dimension ; puis l'OMPCA sélectionne les 40 meilleures composantes ; qui sont modifiées par Hadamard. C'est à ce moment que l'on teste plusieurs dimension K de 1 à 40. Les résultats sont donnés en fig. 9.

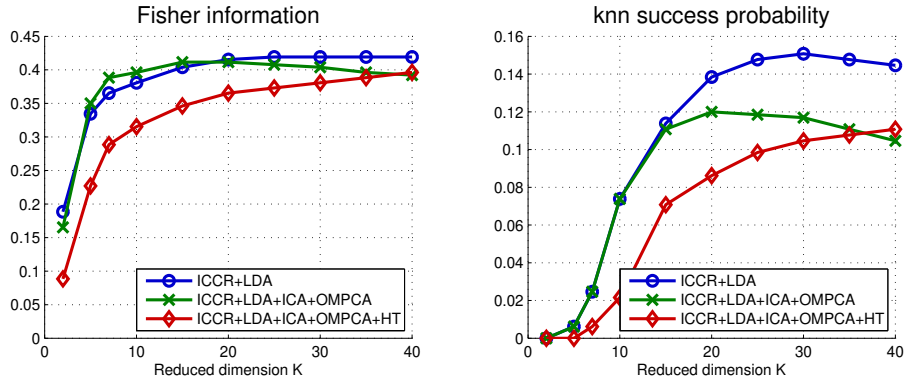

FIGURE 9 – Influence de la transformée de Hadamard.

Alors que les performances de l'OMPCA après l'ICA montent brusquement pour des valeurs faibles de K et redescendent légèrement après $K \approx 15$, on constate une profession plus lente avec l'ajout de la transformée de Hadamard. Les deux scénarii finissent par avoir des résultat équivalents pour $K \approx 40$.

Cette observation confirme le fait que la transformée de Hadamard a égalisé l'efficacité des 40 composantes obtenues après la réduction de l'OMPCA. Les résultats de l'OMPCA croissent brusquement pour K faible parce qu'à ce moment seules les composantes les plus performantes sont utilisées, puis avec l'ajout des composantes moins discriminantes, les résultats chutent un peu. Comme la transformation de Hadamard égalise le pouvoir discriminant, la progression est plus lente, mais plus stable.

Ces résultats pourraient inciter à n'utiliser que les 15 premières composantes issues de l'OMPCA, mais en fonction du hachage approximatif utilisé après, cf. [14, sec. 5], il est préférable d'utiliser un nombre plus grand de composantes. C'est pourquoi ces résultats justifient l'intérêt de la transformée de Hadamard, contrairement aux premiers abords.

5.1.6 Discussion a propos de tests non présentés

De très nombreux autres tests ont été réalisés et ne sont pas présentés ici. Par exemples :

- ✓ D'autres tests ont été faits avec des dimensions différentes des sous-espaces intermédiaires.
- ✓ Aussi, nous avons testé de nombreux autres scénarii, tels que **ICCR+LDA+MPCA+ICA**, **ICCR+LDA+HT** ou **ICCR+LDA+ZCA**.
- ✓ Quelques ensembles de valeurs de paramètres des empreintes de hautes dimensions ont été comparés, voir par exemple [13].
- ✓ Enfin, dans certains cas, nous avons utilisé d'autres bases d'apprentissage pour certaines méthodes : par exemple l'ICA a été testés tantôt sur les empreintes originales (ce que nous avons présentés), et tantôt sur des empreintes dégradées.

Nous faisons le choix de ne pas les présenter parce que d'une part ils n'apportent le plus souvent pas de conclusion différente, et d'autres part, certains des scénarii ont été imaginés au moment de l'évaluation qui suit, et sont donc présentés dans la prochaine section.

5.2 Remplissage de la table et robustesse moyenne des bits

Lors de ces derniers tests, nous avons évalué plusieurs scénarii différents à partir de critères nouveaux. Le premier critère concerne le remplissage de la table de hachage, et le second la robustesse moyenne des bits des codes de hachage obtenus après binarisation, brièvement présentée en sec. 4.3. Même si le second semble plus important, le premier l'est tout autant, parce qu'en plus de pouvoir assurer la robustesse, un remplissage uniforme de la table permet d'améliorer la quantité d'information contenu par chaque case, et donc améliore l'indexation.

5.2.1 scénarii testés

Nous avons dans ce test évalué les scénarii suivant :

1. **ICCR + LISA + HT**
2. **ICCR + LDA + HT**
3. **ICCR + LISA + ICA**
4. **ICCR + LISA + ICA + HT**
5. **ICCR + LISA + ICA + OMPCA + HT**
6. **ICCR + LISA + MPCA + HT**
7. **ICCR + LISA + ZCA + HT**
8. **ICCR + LISA + ZCA + OMPCA + HT**

Remarquons avant toute chose que les scénarii 1 et 2 permettront de comparer les deux méthodes LDA et LISA. Les autres scénarii ont tous remplacer la méthode LDA par LISA, parce que des tests informels non présentés ici avaient déjà montré une très légère amélioration de la robustesse et du remplissage de la table.

Aussi, nous testons à la place de l'ICA, d'une part le blanchiment ZCA et la MPCA seule. Remarquons que la ZCA a pour fonction de décorréler les données avec un minimum de modification. Cela signifie que les composantes restent dans un ordre décroissant de pouvoir discriminant si elles l'étaient auparavant.

Ajoutons que contrairement aux évaluations précédentes, la taille du sous-espace final est fixé à $K = 40$, et les dimensions intermédiaires sont les mêmes.

5.2.2 Remplissage des éléments de la table de hachage

La première des deux valeurs affichées concerne le remplissage de la table de hachage. Il s'agit de la variance de la taille de ses éléments. Une variance forte signifie que les éléments de la table ont des tailles très différentes, alors qu'une variance faible signifie que la table est remplie plus uniformément. Les résultats sont donnés en figure 10.

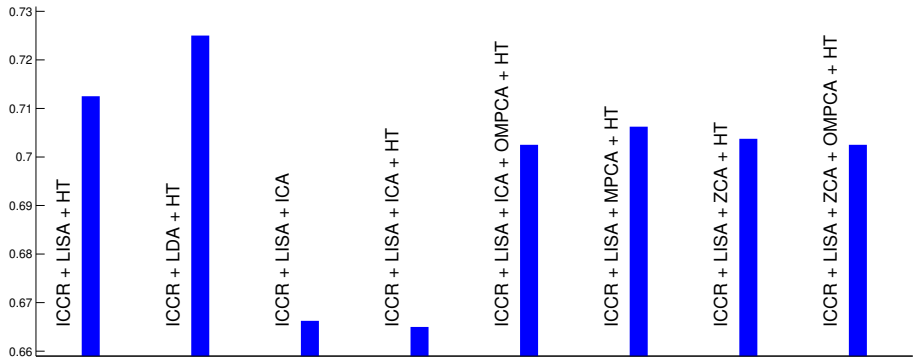

FIGURE 10 – Variance de la taille des cases de la table de hachage pour les 8 scénarii testés.

En analysant la figure 10, on remarque comme attendu que l'ICA (scenario 3) permet un remplissage plus uniforme de la table. Cependant on constate que l'ajout de la transformée de Hadamard ne rend pas le remplissage moins uniforme malgré la perte d'indépendance des variables. En comparant les méthodes LDA et LISA, il semble que la seconde permet un remplissage légèrement plus uniforme. Enfin, pour les quatre autres scénarii, la variance est à peu près la même, nettement plus élevée que pour l'ICA seule, mais légèrement plus faible que pour les deux premiers scénarii.

5.2.3 Robustesse moyenne des bits

Dans ce dernier test, nous avons construit les codes de hachage après binarisation des empreintes dans les différents espaces de dimension réduite. Cela a été fait à la fois pour le signal original mais aussi pour les signaux dégradés. Après cela, la robustesse moyenne est estimée, comme étant $r = 1 - \epsilon$, où ϵ est le taux moyen de bits erronés. Les résultats sont présentés en fig. 11.

Ajoutons que $r \geq 0.5$ car 0.5 est la valeur pour un hachage aléatoire. En effet comme les bits ne prennent que 2 valeurs, par un processus purement aléatoire, il y a 1 chance sur 2 que la valeur du bit ne change pas.

En regardant la figure 11 de près, on observe déjà que l'ICA avec ou sans transformée de Hadamard produit les plus mauvais résultats. Et comme attendu des traitements postérieurs permettent de rétablir

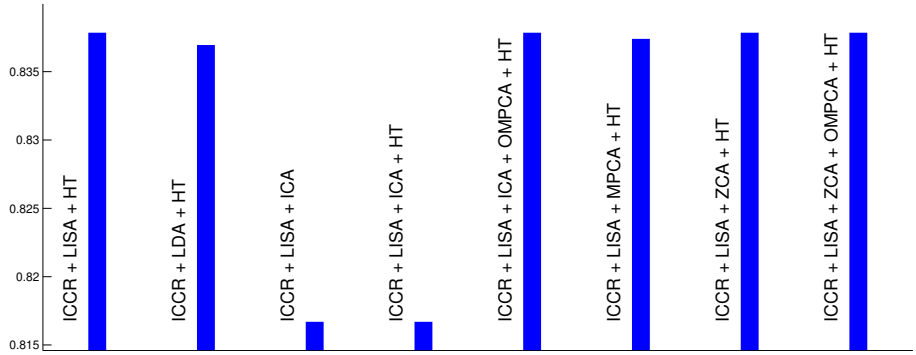

FIGURE 11 – Robustesse moyenne des bits des codes de hachage.

la performance. Ensuite on constate que les performances des autres tests sont très très similaires, avec une différence quasi insignifiante.

Notons que le scénario 1, de la méthode LISA, semble donner une performance très légèrement meilleure que pour le scénario 2. Cela ne semble pas signifiant, mais confirme les tests informels préalables.

5.2.4 Conclusion de l'évaluation

A la vue des évaluations menées dans cette section, et en considérant la technique de hachage mise en œuvre par la suite, les 4 derniers scénarii semblent être les plus adaptés. Pour rappel, il s'agit de :

5. **ICCR + LISA + ICA + OMPCA + HT**
6. **ICCR + LISA + MPCA + HT**
7. **ICCR + LISA + ZCA + HT**
8. **ICCR + LISA + ZCA + OMPCA + HT**

D'autres évaluations plus poussées nous auraient permis de trancher, mais le manque de temps ne nous l'a pas permis. Finalement, notre choix s'est porté sur le scénario 6 : **ICCR + LISA + ICA + OMPCA + HT**, parce qu'il nous semble le plus justifié du point de vue théorique. D'autres sont plus simples à mettre en œuvre, mais étant donné qu'il ne s'agit que de transformations affines, le temps de calcul des empreintes réduites est le même quelque soit les méthodes utilisées.

Par conséquent, à partir du vecteur d'empreintes X de dimension 1056, nous obtenons le vecteur Z des empreintes réduites par la transformation affine :

$$Z = \mathbf{P}_{\text{totale}} \times X + \mathbf{t}_{\text{totale}}, \tag{55}$$

$$\text{avec} \quad \begin{cases} \mathbf{P}_{\text{totale}} = P_{\text{hadamard}} \times P_{\text{ompca}} \times P_{\text{ica}} \times P_{\text{lda}} \times P_{\text{iccr}}, \\ \mathbf{t}_{\text{totale}} = P_{\text{hadamard}} \times P_{\text{ompca}} \times t_{\text{ica}}, \end{cases}$$

Pour rappel des dimensions intermédiaires utilisées, voici un schéma qui résume les dimensions avant et après chaque transformation. Cela donne entre autre la taille des matrices :

$$1056 \overset{ICCR}{\longrightarrow} 1026 \overset{LISA}{\longrightarrow} 80 \overset{ICA}{\longrightarrow} 80 \overset{OMPCA}{\longrightarrow} 40 \overset{HT}{\longrightarrow} 40$$

Rappelons que l'ICCR réduit la dimension pour ne conserver que les composantes avec un bon conditionnement, LISA sélectionne les 80 directions parmi 1026 qui discriminent aux mieux les classes détaillées en sec. 3.1, et de même l'OMPCA réduit une dernière fois la dimension de 80 à 40, pour sélectionner les composantes les plus robustes et les plus informatives. Les autres, ICA et HT, ne modifient pas la dimension. Par conséquent la matrice $\mathbf{P}_{\text{totale}}$ est de taille (40×1056) et $\mathbf{t}_{\text{totale}}$ de taille (40×1).

Comme il a déjà été discuté, le même type de traitement est réalisé sur les cinq bandes de fréquence, mais puisque le contenu de chacune d'elle peut avoir une nature différente, les données d'apprentissage ne sont pas mélangées, si bien que cinq apprentissages différents sont réalisés en parallèle, conduisant à des transformations affines de coefficients différents. En effet, en échelle logarithmique, la densité d'harmoniques ou de partiels est bien plus forte pour les fréquences élevées que pour les fréquences basses. La nature du contenu des cinq bandes est par conséquent potentiellement différent.

La figure 12 représente la contribution de chaque fréquence (log-fréquence et log-temps) de la transformée de Fourier 2D à l'origine des empreintes de haute dimension. On y remarque que les principaux bins utilisés sont proches des axes (verticaux et horizontaux de fréquences nulles). Cependant, l'axe vertical semble assez peu utilisé. Notons que la conversion quasi-logarithmique des amplitudes avant la

transformée de Fourier 2D a pour effet d'y renvoyer la contribution du filtrage et de l'égalisation. On constate alors que l'apprentissage a automatiquement choisi de moins les utiliser.

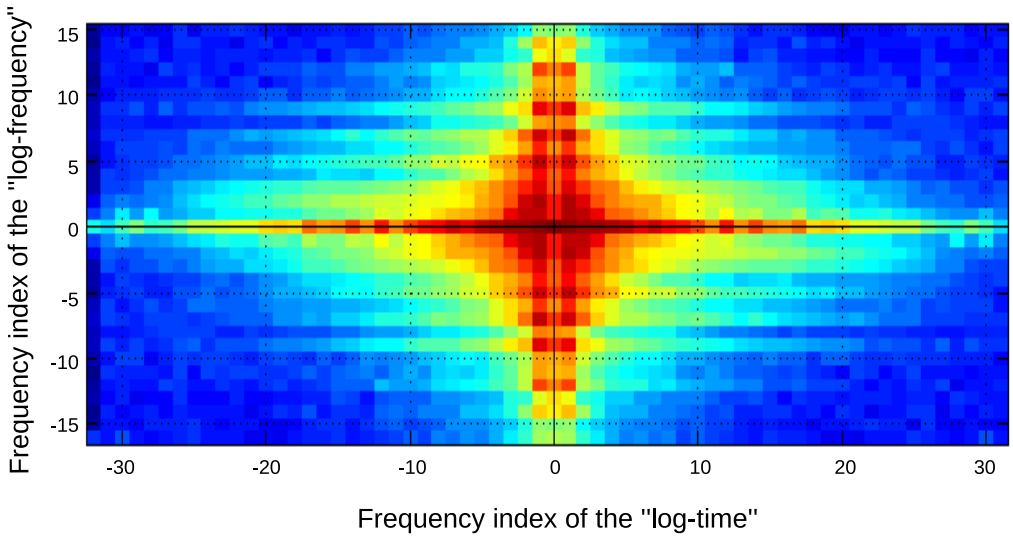

FIGURE 12 – Contribution des bins DFT-2D pour les composantes des empreintes réduites. La figure illustre la contribution des bins pour l'ensemble des 40 composantes des empreintes réduites.

6 Conclusion

Dans le contexte de l'indexation audio et de la reconnaissance d'extraits musicaux robuste aux dégradations sonores, nous avons proposé dans le projet BeeMusic de nouvelles empreintes sonores de hautes dimensions et naturellement robustes à certaines dégradations. Ce document traite de l'apprentissage automatique et supervisé de la réduction de dimension de ces empreintes sonores. Le but étant de rendre réalisable l'indexation audio avec un nombre de coefficients acceptable, et surtout de déterminer le sous-espace le moins sensible aux altérations et éventuellement contenant l'information la plus pertinente du point de vue musicale.

Après avoir présenté plusieurs méthodes statistiques en sec. 3 et d'autres transformations en sec. 4 pour le conditionnement, l'indépendance/décorrélation des variables, nous avons réalisé un certain nombre de tests pour faire le choix des méthodes finalement utilisées pour le système d'indexation. La chaîne de traitements finalement retenue est : **ICCR+LISA+ICA+OMPCA+HT**, cf. sec. 5.2.4. Cependant, le bénéfice de ce choix comparé à d'autres semble minime, et peut-être des tests plus poussés pourraient améliorer les résultats de la reconnaissance. Entre autres, le choix de la dimension intermédiaire, qui est 80 après LISA, pourrait être affiné.

L'un des points forts de la technique réalisée ici, est que le réglage de la base d'apprentissage peut être fait en fonction du type de signaux considérés, et du type de dégradations. Par exemple, si les extraits musicaux attendus sont des titres de rock sans changement d'échelles, mais avec du bruit additif principalement, alors il est possible de constituer la base d'apprentissage avec uniquement ce genre musical et les dégradations considérées. Cela spécialisera les matrices retournées pour cette tâche, et les performances de reconnaissance seront très probablement accrues.

Rappelons que la prochaine étape pour l'indexation audio est la construction des codes de hachage pour la table de hachage. La technique utilisée est présentée en [14, sec. 5], et les résultats finaux de la reconnaissance sont détaillés dans [12].

Remarquons également que d'autres techniques de la littérature ont été étudiées, et n'ont pas été retenues. Par exemple, l'IRMFSP de [18] permet une sélection de composantes évitant une dépendance linéaire. Cependant, elle est longue à calculer parce qu'à chaque itération sélectionnant une composante, les matrices de covariance T et B doivent être recalculées dans un nouveau sous-espace. Cela n'est pas compatible avec la procédure mise en œuvre. De plus, l'ICCR remplit quelque part cette fonction avec efficacité.

Précisons, qu'une autre méthode statistique (ODA) a été mise en œuvre mais n'est pas présentée dans ce document. Dans le cas d'une indexation utilisant l'approche des *k-plus proches voisins*, elle a

pour but d'apprendre automatiquement la distance optimale associée à l'algorithme k-nn. Elle a donné des résultats satisfaisant, mais puisque finalement la procédure de recherche est basée sur une technique de table de hachage, la méthode a été abandonnée.

Pour finir, ces empreintes réduites qui décrivent une portion de signal peuvent être considérées comme descripteurs audio au même titre que d'autres de la littérature (e.g. MFFC, barycentre et variance spectrale, *spectral flatness*, cf. [17]). Par conséquent, l'approche développée ici, pourrait être appliquée à d'autres tâches d'apprentissage automatique, telles que classification ou segmentation, à l'aide d'une adaptation des méthodes statistiques utilisées et des données utilisées comme base de l'apprentissage.

Références

[1] E. Allamanche, J. Herre, O. Hellmuth, B. Fröba, T. Kastner, and M. Cremer. Content-based identification of audio material using MPEG-7 low level description. In *Int. Symposium on Music Information Retrieval (ISMIR'01)*, October 2001.

[2] R. Duda, P. Hart, and D. Stork. *Pattern classification*. John Wiley & Sons, 2nd edition, 2000. 680 pages.

[3] S. Fenet, G. Richard, and Y. Grenier. A scalable audio fingerprint method with robustness to pitch-shifting. In *Int. Symposium on Music Information Retrieval (ISMIR'11)*, pages 121–126, October 2011.

[4] J. Haitsma and T. Kalker. A highly robust audio fingerprinting system. In *Int. Symposium on Music Information Retrieval (ISMIR'02)*, volume 2002, pages 107–115, Octobre 2002.

[5] J. Haitsma and T. Kalker. Speed-change resistant audio fingerprinting using auto-correlation. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP'03)*, volume 4, pages IV–728, 2003.

[6] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin. The elements of statistical learning : data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2) :83–85, 2005.

[7] A. Hedayat and W.D. Wallis. Hadamard matrices and their applications. *The Annals of Statistics*, 6(6) :1184–1238, 1978.

[8] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural computation*, 9(7) :1483–1492, 1997.

[9] I. Jolliffe. *Principal component analysis*. John Wiley & Sons, 2nd edition, 2002.

[10] P.C. Mahalanobis. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2 :49–55, 1936.

[11] R. Mignot. BeeAlter Toolbox : Boîte à outils de dégradations sonores, pour tests. Technical report, 2015. Rapport interne IRCAM – CNRS, projet BeeMusic.

[12] R. Mignot. Ircam AudioID : Evaluation et résultats de la reconnaissance d'extraits audio dégradés. Technical report, 2015. Rapport interne IRCAM – CNRS, projet BeeMusic.

[13] R. Mignot. Ircam AudioPrint : calcul des empreintes sonores et choix des paramètres. Technical report, 2015. Rapport interne IRCAM – CNRS, projet BeeMusic.

[14] R. Mignot. RAPPORT GLOBAL – Ircam AudioID : Indexation audio avec robustesse aux dégradations sonores. Technical report, 2015. Rapport interne IRCAM – CNRS, projet BeeMusic.

[15] B. Moore. PCA and ICA Package. MathWorks File Exchange, April 2015. Matlab toolbox available at : http ://www.mathworks.com/matlabcentral/fileexchange/38300-pca-and-ica-package.

[16] K. Moravec and I.J. Cox. A comparison of extended fingerprint hashing and locality sensitive hashing for binary audio fingerprints. In *Proc. of the 1st ACM Int. Conf. on Multimedia Retrieval (ICMR'11)*, page 31, April 2011.

[17] G. Peeters. A large set of audio features for sound description (similarity and classification) in the cuidado project. 2004.

[18] G. Peeters and X. Rodet. Hierarchical gaussian tree with inertia ratio maximization for the classification of large musical instruments databases. In *Proc. Int. Conf. on Digital Audio Effects (DAFx'03)*, volume 3, pages 8–11, 2003.

[19] M. Ramona and G. Peeters. Audio identification based on spectral modeling of bark-bands energy and synchronization through onset detection. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP'11)*, pages 477–480, 2011.

[20] M. Ramona and G. Peeters. Audioprint : An efficient audio fingerprint system based on a novel cost-less synchronization scheme. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP'13)*, pages 818–822, May 2013.

[21] R. Sonnleitner and G. Widmer. Quad-based audio fingerprinting robust to time and frequency scaling. In *Proc. Int. Conf. on Digital Audio Effects (DAFx'14)*, September 2014.

[22] E. Walter and L. Pronzato. Identification of parametric models. *Communications and Control Engineering*, 1997. 413 pages.

[23] A. Wang. An industrial strength audio search algorithm. In *Int. Symposium on Music Information Retrieval (ISMIR'03)*, pages 7–13, October 2003.