



**HAL**  
open science

# Modélisation de filtres HRTF et optimisation pour le traitement de spatialisation sonore

Maxime Baelde

► **To cite this version:**

Maxime Baelde. Modélisation de filtres HRTF et optimisation pour le traitement de spatialisation sonore. Machine Learning [stat.ML]. 2015. hal-01339034

**HAL Id: hal-01339034**

**<https://inria.hal.science/hal-01339034>**

Submitted on 29 Jun 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Mémoire de recherche

# Modélisation de filtres HRTF et optimisation pour le traitement de spatialisation sonore

Maxime Baelde

Sous la direction de Christophe BIERNACKI (INRIA) et Raphaël GREFF (A-Volute)

Université de Lille 1 - Laboratoire Paul PAINLEVÉ

3 juillet 2015



**a-volute**  
3D Sound Expert



**Université  
de Lille**  
**1** SCIENCES  
ET TECHNOLOGIES

# Remerciements

Je remercie Raphaël GREFF, mon tuteur entreprise, ainsi que Christophe BIERNACKI, mon tuteur académique, pour leur aide et leur soutien pendant la réalisation de ce mémoire de recherche. Leurs expertises dans leur domaine respectif m'ont été d'une grande aide pour la réflexion et la modélisation du problème.

Merci également à Tuyen PHAM et Ambroise RECHT de la direction d'A-Volute, Damien GRANGER et Maxence FREIXA de l'équipe R&D, Maïté HOSTE et Emilie RAMAGET de l'équipe PO, Dinis ALVES, Pierre GALERNEAU et Guillaume DANIEL de l'équipe de Test, Paul ALLAIS, Laurent LEROY, Mhand BOUGHIAS, Matthieu CAMPION, Mustapha TACHOUCT, Alexis ZBIK et Gregory FOUREZ de l'équipe Développement, Cyril DELESALLE, administrateur réseau, Klara KUJAWA, du marketing, et Samantha HERMANN, designer, pour leur gentillesse, leur bonne humeur et leurs conseils avisés.

Merci à Camille CAUDRELIER pour son soutien de tous les jours, et à ma mère pour la relecture.

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Le son 3D . . . . .	4
1.2	Head-Related Transfer Function . . . . .	5
1.3	Problématique . . . . .	6
1.4	Les données . . . . .	7
1.5	Notation . . . . .	7
<b>2</b>	<b>Apprentissage statistique classique</b>	<b>9</b>
2.1	Les méthodes considérées . . . . .	9
2.1.1	Régression linéaire, <i>Ridge</i> et <i>LASSO</i> . . . . .	9
2.1.2	Régression <i>LARS</i> . . . . .	13
2.1.3	Régression <i>SVM</i> . . . . .	14
2.1.4	Régression <i>Sparse</i> . . . . .	16
2.1.5	Régression <i>PLS</i> . . . . .	17
2.2	Mesure de performance . . . . .	17
2.3	Résultats . . . . .	18
2.4	Discussion . . . . .	18
<b>3</b>	<b>Méthodes avancées d'apprentissage statistique</b>	<b>23</b>
3.1	Approche tensorielle . . . . .	23
3.1.1	Généralités sur les tenseurs . . . . .	23
3.1.2	HOSVD . . . . .	28
3.1.3	HOPLS . . . . .	32
3.2	Application aux HRTF . . . . .	37
3.2.1	Traitement des données . . . . .	37
3.2.2	Modélisation par HOPLS . . . . .	38
3.3	Résultats . . . . .	38
3.4	Discussion . . . . .	39
<b>4</b>	<b>Individualisation séparée</b>	<b>41</b>
4.1	Méthode . . . . .	41
4.1.1	Séparation des parties résonances et anti-résonances . . . . .	41
4.1.2	Extraction des lieux fréquentiels des résonances et anti-résonances . . . . .	43
4.1.3	Transformation des données . . . . .	44
4.2	Résultats . . . . .	45
4.3	Discussion . . . . .	46
<b>5</b>	<b>Conclusion</b>	<b>49</b>

# Chapitre 1

## Introduction

### 1.1 Le son 3D

La spatialisation sonore [10] a pour but de recréer un environnement sonore 3D et donner l'illusion que le son provient de n'importe quelle direction de l'espace. La technique la plus simple et la plus répandue est la stéréophonie : le son nous parvient de deux sources, situées à gauche et à droite. Il existe aussi les systèmes surround 5.1 (cinq sources sonores) et 7.1 (sept sources sonores) : ils permettent un positionnement de plus de sources sonores dans une pièce. On retrouve ce genre de système pour des applications grand public, comme le home cinema. On peut même créer virtuellement des sources de manière analogique en faisant du *panning*, c'est-à-dire en jouant sur le gain au niveau de chaque haut-parleur. L'idée est d'immerger l'utilisateur dans un environnement sonore virtuel 3D. Ces systèmes peuvent être couplés à l'outil informatique pour simuler l'environnement 3D de manière numérique. C'est l'idée fondamentale du produit proposé par A-Volute, le logiciel Nahimic. En amont de ce traitement numérique, de nombreux domaines sont utilisés comme l'acoustique, la psychoacoustique et la cognition. L'idée est de modéliser le phénomène de perception sonore afin de le simuler par la suite.

Le point central de toutes ces techniques est la perception spatiale du son. La capacité de l'Homme à localiser les sources sonores provient en outre de ses oreilles, mais aussi de tout son corps (par le biais de la diffraction et des réflexions des ondes sonores). Comme les deux oreilles ne se situent pas au même endroit sur la tête, le son n'arrive ni en même temps sur celles-ci, ni avec la même incidence ni la même intensité.

Les positions d'une source sonore par rapport à un auditeur seront indiquées en utilisant le repère suivant (voir FIGURE 1.1) :

- Le plan horizontal, ou plan d'azimuth, qui sépare le haut du bas,
- Le plan médian, qui sépare la gauche de la droite,
- Le plan frontal, qui sépare l'avant de l'arrière.

Le système de coordonnées considéré est le système sphérique  $(\theta, \phi, z)$ , respectivement angle d'azimuth, angle d'élévation et rayon.

La localisation en azimuth (dans le plan horizontal) utilise l'indice de différence entre les temps d'arrivée aux deux oreilles (ITD pour *Interaural Time Difference*) et la différence d'intensité (ILD pour *Interaural Level Difference*), qui sont des indices interauraux. Or pour un couple (ITD, ILD), plusieurs directions d'émission sont possibles. C'est pourquoi des mécanismes plus fins sont nécessaires pour bien localiser une source.

À cause de la diffraction du son sur notre corps, les différences d'intensité dépendent de la fréquence du son. À basses fréquences, les longueurs d'onde sont plus grandes que les dimensions de la tête et donc l'ILD n'est pas un indice prépondérant. À hautes fréquences, les caractéristiques morphologiques

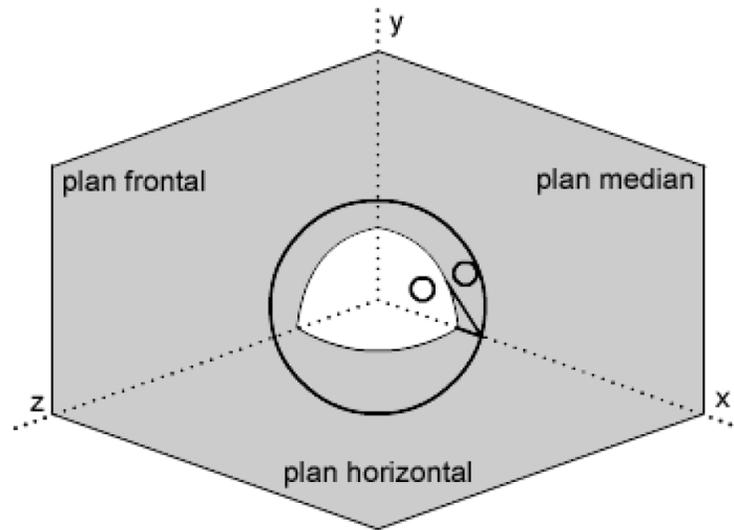


FIGURE 1.1 – Le repère pour positionner les sources sonores

ont une plus grande importance et donnent une "couleur spectrale" au son, dépendamment de sa direction d'incidence, que notre cerveau a appris à reconnaître.

Les indices spectraux (dits monauraux) servent à discriminer les directions avant/arrière et haut/bas. On suppose que les indices monauraux sont situés dans les résonances et les anti-résonances du spectre du signal [10]. Par exemple, dans le cas d'une source placée devant, on remarque souvent une forte résonance autour de 5 kHz qui est due à la conque de l'oreille, et plusieurs anti-résonances dues aux réflexions sur la paroi arrière de la conque. Néanmoins, il existe une forte variabilité interindividuelle, si bien qu'il n'y a pas de consensus sur l'effet de la position des indices spectraux sur la capacité de perception.

Pour essayer de quantifier ce phénomène de localisation d'une source sonore, l'outil le plus couramment utilisé est le HRTF.

## 1.2 Head-Related Transfer Function

La synthèse binaurale (ou spatialisation sonore) repose sur les filtres HRTF (*Head Related Transfer Function*), qui sont considérés comme des filtres RIF (Réponse Impulsionnelle Finie). Il existe des HRTF pour les oreilles gauche et droite. La spatialisation sonore peut être représentée sous forme de schéma bloc (FIGURE 1.2) :

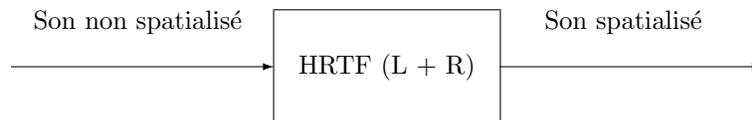


FIGURE 1.2 – Schéma bloc de fonctionnement des HRTF

Les HRTF caractérisent le filtre acoustique du système auditif périphérique d'un individu (sa tête, son cou, son torse, ses épaules et ses oreilles externes). En effet, les structures auditives périphériques

agissent comme un filtre acoustique directif pour lequel les fréquences varient avec la direction spatiale.

Les HRTF sont constitués de deux parties :

- la fonction de transfert directionnelle (DTF pour *Directional Transfert Function*), qui contient l'information sur le lien entre le filtre et la direction spatiale.
- la fonction de transfert indépendante de la direction.

Ils décrivent tous les indices acoustiques pertinents qui sont nécessaires à l'écoute spatiale dans un environnement en champ libre [10].

Soient  $x$  le signal sonore issu de la source,  $x_L$  le signal acoustique arrivant à l'oreille gauche et  $x_R$  celui arrivant à l'oreille droite. La relation entre ces signaux correspond à une convolution (notée  $\star$ ) par un filtre  $h_L$  et  $h_R$  respectivement :

$$\begin{cases} x_L(t) = x(t) \star h_L(t) \\ x_R(t) = x(t) \star h_R(t) \end{cases} \quad (1.1)$$

Dans le domaine de FOURIER, ces équations deviennent :

$$\begin{cases} X_L(f) = X(f)H_L(f) \\ X_R(f) = X(f)H_R(f) \end{cases} \quad (1.2)$$

Soit  $x_O$  le signal acoustique en champ libre à l'emplacement du milieu de la tête. Les HRTF sont définis comme :

$$\begin{cases} HRTF_L(f) = \frac{X_L(f)}{X_O(f)} \\ HRTF_R(f) = \frac{X_R(f)}{X_O(f)} \end{cases} \quad (1.3)$$

Ainsi, seuls les phénomènes de diffraction, réflexion et résonance induits par la présence de l'auditeur sont considérés.

En pratique, les HRTF sont mesurés directement sur la personne, en plaçant des microphones dans ses oreilles ([10]). Un *sweep* (un signal sinusoïdale dont la fréquence augmente au cours du temps) ou une MLD (*Maximum Length Sequence*) est émis et on enregistre la réponse au niveau de l'entrée du pavillon. Il existe aussi des outils de simulation numérique pour calculer directement les HRTF. Quelques soient l'une ou l'autre des procédures, elles sont extrêmement longues et coûteuses (en temps de calcul ou de mesure).

### 1.3 Problématique

Comme évoqué précédemment, un HRTF est un filtre qui dépend à la fois de la direction de l'espace et de la personne. De plus, la procédure pour obtenir ces HRTF personnalisés est souvent longue et fastidieuse. L'utilisation de HRTF générique (c'est-à-dire représentant une personne moyenne) pourrait permettre de palier à ce problème. Néanmoins, ces HRTF ne permettent pas un bon rendu d'une scène sonore 3D.

L'idée est donc de chercher un moyen d'individualiser facilement un HRTF. Un HRTF est principalement influencé par la morphologie d'une personne, typiquement la forme de tête, de son torse et de ses oreilles. Plusieurs approches sont envisagées au sein d'A-Volute pour permettre cet individualisation. Une première consiste à généraliser un modèle 3D de la personne puis à effectuer des calculs acoustiques pour obtenir les HRTF. Une deuxième approche consiste à réaliser un système low-cost d'enregistrement de HRTF du style des laboratoires de recherche. La dernière approche, qui sera considérée dans ce mémoire, consiste à chercher par apprentissage statistique un moyen de prédire les HRTF d'une personne connaissant des paramètres morphologiques (typiquement des distances). Ils existent beaucoup d'autres techniques possibles dans la littérature, voir notamment [7, 19, 21].

## 1.4 Les données

La base de données utilisée est la concaténation des bases *CIPIC* [17] et *LISTEN* [15]. Elle comporte les HRTF et les données morphologiques de 98 sujets. Les données présentes dans la base *CIPIC* ont subi des traitements préliminaires pour les formater au format *LISTEN*. Ensuite les deux bases ont subi des traitements correctifs tels que de l'interpolation ou de l'extrapolation. Les HRTF sont des matrices de dimension  $2016 \times 129$ , qui comportent :

- 2016 couples de positions angulaires : 72 d'azimut (de  $0^\circ$  à  $355^\circ$  par pas de  $5^\circ$ ) et 28 d'élévation (de  $-45^\circ$  à  $90^\circ$  par pas de  $5^\circ$ )
- $F = 129$  fréquences ( $F_s = 44,1$  kHz), de 0 à 22,05 kHz par pas de  $F_s/256 \approx 172$  Hz.

La partie gain en dB d'un HRTF est donnée en exemple à la FIGURE 1.3.

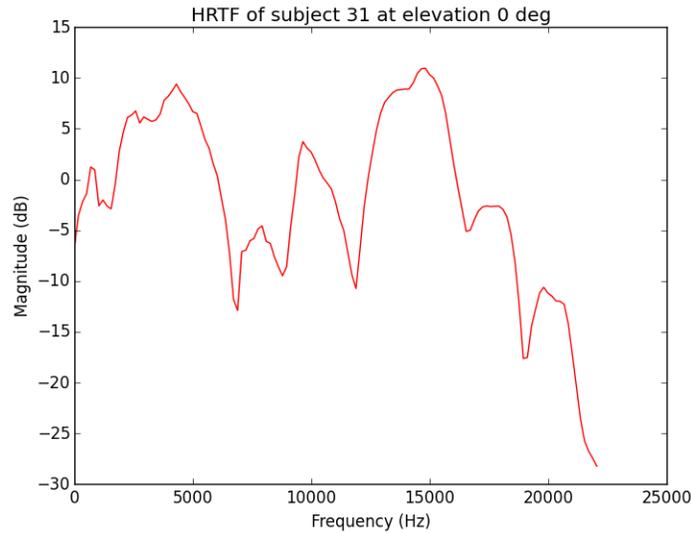


FIGURE 1.3 – Le HRTF du sujet 31, à l'élévation  $0^\circ$

Les données morphologiques comportent 27 mesures : 17 relatives à la tête et au torse, et 10 relatives aux oreilles (FIGURE 1.4).

Il s'avère que 25 sujets n'avaient pas toutes les mesures morphologiques, c'est pourquoi leurs données ne sont pas utilisées dans les expériences. Il reste donc  $P = 73$  observations disponibles.

De plus, parmi les 27 mesures, seules  $K = 21$  sont gardées car certains sujets ne les possédaient pas. Celles-ci sont  $x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}, x_{16}, d_1, d_2, d_3, d_4, d_5, d_6, d_7, \theta_1$ .

Uniquement le plan médian frontal (à azimuth nul, ce qui correspond au plan vertical) sera considéré, ce qui ramène à  $D = 28$  le nombre de positions spatiales. De plus seulement les gains en dB des HRTF sont considérés.

Seules les données pour l'oreille gauche sont utilisées, car les résultats sont comparables pour les deux oreilles. Les tâches de *machine learning* seront effectuées sur Matlab, Python ou R dépendamment des toolbox disponibles.

## 1.5 Notation

Sauf indication contraire, les scalaires sont notés  $a, b, \dots$ , les vecteurs  $\mathbf{a}, \mathbf{b}, \dots$ , les matrices  $\mathbf{A}, \mathbf{B}, \dots$  et les tenseurs  $\mathcal{A}, \mathcal{B}, \dots$ . Si  $\mathbf{a}$  est un vecteur, on note  $a_i$  une composante de ce vecteur. Si  $\mathbf{A}$  est une matrice,

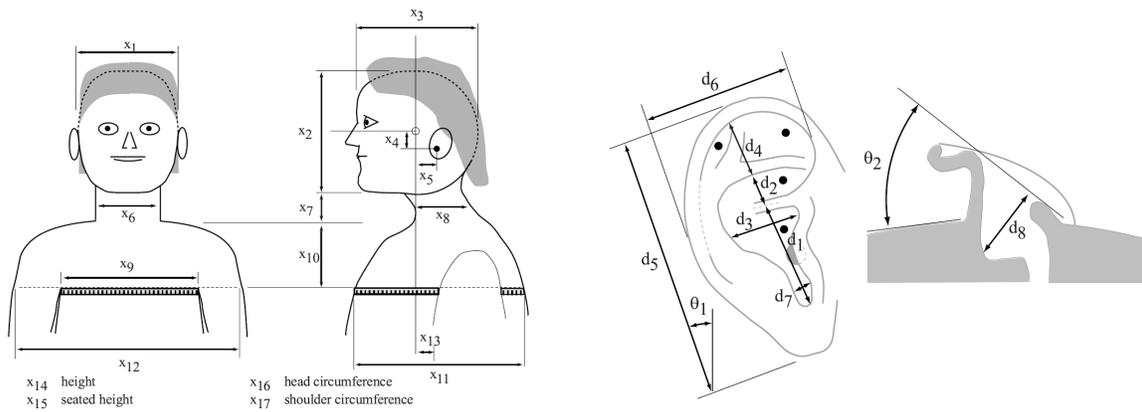


FIGURE 1.4 – Paramètres morphologiques

on note  $\mathbf{a}_i$  un vecteur colonne et  $a_{ij}$  un élément de la matrice. On note  $\star$  l'opération de conjugaison complexe.  $\text{fft}$  et  $\text{ifft}$  désignent respectivement l'opération de Transformée de FOURIER Rapide et son inverse et  $\text{Hann}_x$  désigne l'opération de fenêtrage par un demi-fenêtre de HANN de  $x$  millisecondes.

## Chapitre 2

# Apprentissage statistique classique

Une approche d'apprentissage statistique est utilisée pour trouver la relation entre les HRTF et les paramètres morphologiques d'une personne. La section 2.1 présente les techniques envisagées, la section 2.2 présente les résultats et la section 2.3 propose une discussion sur le contenu de ces résultats.

## 2.1 Les méthodes considérées

### 2.1.1 Régression linéaire, *Ridge* et *LASSO*

Cette section présente le modèle de la régression linéaire, inspiré de [11, 3].

**Définition 1** (Régression linéaire). La régression linéaire consiste en la régression d'une variable expliquée (la sortie) sur une ou plusieurs variables explicatives (les entrées), avec pour hypothèse que le modèle reliant les entrées et la sortie est **linéaire** en ses paramètres. La sortie est également appelée prédiction ou *target* en anglais. Un exemple est donné sur la FIGURE 2.1.

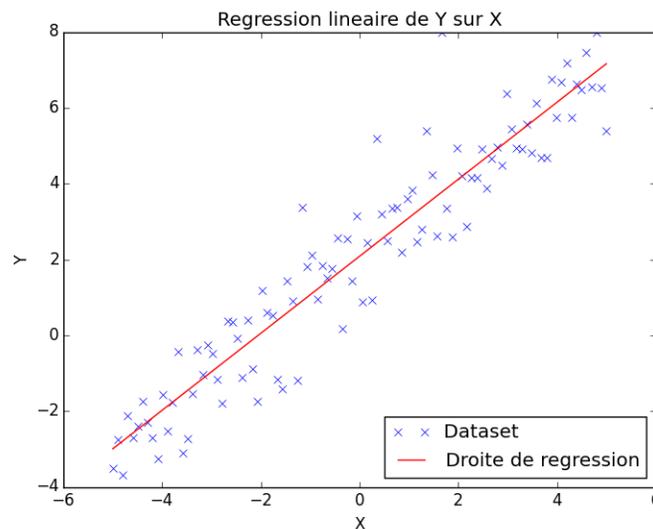


FIGURE 2.1 – Régression linéaire de Y sur X

Il arrive que l'on souhaite prédire plusieurs sorties en même temps : on parle dans ce cas de régression linéaire **multivariée**.

Une tâche de régression nécessite une base d'apprentissage :

**Définition 2** (Base d'apprentissage). La base d'apprentissage est constituée de  $n$  échantillons de  $p$  variables explicatives, qui correspondent aux entrées, et  $n$  échantillons de  $d$  variables expliquées, qui correspondent aux sorties. Par commodité d'écriture, les variables explicatives seront représentées par des vecteurs colonnes  $(\mathbf{x}_i)_{1 \leq i \leq n}$ , et regroupées dans une matrice  $\mathbf{X}$  (de dimension  $n \times (p+1)$ ) contenant en ligne tous les échantillons ainsi qu'une colonne de 1. Les variables expliquées seront représentées par des vecteurs colonnes  $(\mathbf{y}_i)_{1 \leq i \leq n}$ , et seront regroupées dans une matrice  $\mathbf{Y}$  (de dimension  $n \times d$ ) contenant en ligne tous les échantillons.

Ainsi on a :

$$\mathbf{X} = \begin{pmatrix} 1 & \mathbf{x}_1^\top \\ \vdots & \vdots \\ 1 & \mathbf{x}_n^\top \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{np} \end{pmatrix}$$

Et :

$$\mathbf{Y} = \begin{pmatrix} \mathbf{y}_1^\top \\ \vdots \\ \mathbf{y}_n^\top \end{pmatrix} = \begin{pmatrix} y_{11} & \cdots & y_{1d} \\ \vdots & \ddots & \vdots \\ y_{n1} & \cdots & y_{nd} \end{pmatrix}$$

**Définition 3** (Modèle de régression linéaire). Le modèle de la régression linéaire est défini comme :

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

avec  $\boldsymbol{\beta} = (\beta_1, \dots, \beta_d)$  la matrice contenant les vecteurs de régression (où chaque  $\beta_i$  est un vecteur colonne de taille  $p+1$ ), et  $\boldsymbol{\epsilon}$  sont les erreurs commises sur chaque régression ( $\boldsymbol{\epsilon}_i \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_n)$ ) et indépendantes) en considérant le vecteur de régression  $\beta_i$ , de dimension  $n \times d$ .

Un estimateur par maximum de vraisemblance est utilisé pour trouver cette matrice de vecteurs  $\boldsymbol{\beta}$ .

**Proposition 1.** L'estimateur par maximum de vraisemblance des vecteurs de régression est :

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{Y} \quad (2.1)$$

**Démonstration 1.** Soit  $P(\mathbf{y}_i | \mathbf{X}, \boldsymbol{\beta}_i, \sigma)$  la vraisemblance du modèle pour une sortie  $\mathbf{y}_i = [y_{1i}, \dots, y_{ni}]$  ( $1 \leq i \leq d$ ) :

$$\begin{aligned} P(\mathbf{y}_i | \mathbf{X}, \boldsymbol{\beta}_i, \sigma) &= \prod_{j=1}^n P(y_{ji} | \mathbf{x}_j, \boldsymbol{\beta}_i, \sigma) \\ &= \prod_{j=1}^n \mathcal{N}(y_{ji} | \boldsymbol{\beta}_i^\top \mathbf{x}_j, \sigma^2) \end{aligned} \quad (2.2)$$

La log-vraisemblance est donc :

$$\ln P(\mathbf{y}_i | \mathbf{X}, \boldsymbol{\beta}_i, \sigma) = -\frac{1}{2\sigma^2} \sum_{j=1}^n (y_{ji} - \boldsymbol{\beta}_i^\top \mathbf{x}_j)^2 - \frac{n}{2} \ln(2\pi\sigma)$$

On remarque que maximiser cette vraisemblance revient à minimiser la quantité :

$$E_{LS,i} = \min_{\beta_i} \frac{1}{2} \sum_{j=1}^n (y_{ji} - \mathbf{x}_j^\top \beta_i)^2$$

Les vecteurs  $\mathbf{y}_i$  et  $\beta_i$  pour  $1 \leq i \leq d$  sont regroupés dans respectivement  $\mathbf{Y}$  et  $\beta$ . La quantité à minimiser est donc :

$$E_{LS} = \min_{\beta} \frac{1}{2} \text{tr} ((\mathbf{Y} - \mathbf{X}\beta)^\top (\mathbf{Y} - \mathbf{X}\beta))$$

Le calcul du gradient par rapport à  $\beta$  donne :

$$\frac{\partial E_{LS}}{\partial \beta} = -\mathbf{X}^\top (\mathbf{Y} - \mathbf{X}\beta)$$

L'annulation du gradient donne :

$$(\mathbf{X}^\top \mathbf{X}) \beta = \mathbf{X}^\top \mathbf{Y}$$

La matrice  $\mathbf{X}^\top \mathbf{X}$  est supposée de rang plein donc inversible. La solution au problème est donc :

$$\hat{\beta} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{Y}$$

■

Une fois connue la matrice contenant les vecteurs de régression, la prédiction est calculée avec la formule suivante :  $\hat{\mathbf{Y}} = \mathbf{X}\hat{\beta}$ .

Il peut arriver que pour certains jeux de données, on fasse du **sur-apprentissage**, ou *over-fitting* en anglais : la régression apprend parfaitement sur la base d'apprentissage mais la généralisation est très mauvaise. Or le but d'une régression est bien sûr de pouvoir généraliser. Pour remédier à ce problème, il existe des techniques de **pénalisation** du modèle. Cette pénalité peut prendre différentes formes, mais la plus commune est la pénalité en norme  $L^2$ , appelée *ridge regression*.

**Définition 4.** La pénalité en norme  $L^2$  consiste à ajouter un a priori gaussien sur le vecteur de régression  $\beta_i$  ( $1 \leq i \leq d$ ) :

$$\beta_i \sim \mathcal{N}(0, \alpha^{-1} \mathbf{I}_p)$$

**Proposition 2.** L'estimateur par maximum a posteriori des vecteurs de régression pénalisés est :

$$\hat{\beta} = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{Y} \quad (2.3)$$

Où  $\lambda = \alpha\sigma^2$ . On l'appelle **estimateur ridge**.

**Démonstration 2.** Utilisons la règle de BAYES pour calculer la posterior du modèle pour  $\beta_i$  :

$$P(\beta_i | \mathbf{X}, \mathbf{y}_i, \sigma, \alpha) \propto P(\mathbf{y}_i | \mathbf{X}, \beta_i, \sigma, \alpha) P(\beta_i | \alpha)$$

La log-posterior s'écrit :

$$\ln P(\beta_i | \mathbf{X}, \mathbf{y}_i, \sigma, \alpha) \propto -\frac{1}{2\sigma^2} \sum_{j=1}^n (y_{ji} - \mathbf{x}_j^\top \beta_i)^2 - \frac{\alpha}{2} \|\beta_i\|_2^2$$

La maximisation de cette log-posterior revient à minimiser, en notation matricielle, la quantité :

$$E_{MAP} = \min_{\beta} \frac{1}{2} \text{tr} ((\mathbf{Y} - \mathbf{X}\beta)^\top (\mathbf{Y} - \mathbf{X}\beta)) + \frac{\lambda}{2} \text{tr} (\beta^\top \beta)$$

Avec  $\lambda = \alpha\sigma^2$ . Le calcul du gradient par rapport à  $\beta$  donne :

$$\frac{\partial E_{LS}}{\partial \beta} = -\mathbf{X}^\top (\mathbf{Y} - \mathbf{X}\beta) + \lambda\beta$$

L'annulation du gradient donne :

$$\left( \mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I} \right) \beta = \mathbf{X}^\top \mathbf{Y}$$

La solution du problème est donc :

$$\hat{\beta} = \left( \mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I} \right)^{-1} \mathbf{X}^\top \mathbf{Y}$$

■

Un autre type de pénalité est la pénalité en norme  $L^1$  :

**Définition 5.** La pénalité en norme  $L^1$  consiste à ajouter une prior de LAPLACE sur le vecteur de régression  $\beta_i$  :

$$P(\beta_i | \alpha) = \left( \frac{\alpha}{2} \right)^p \exp(-\alpha \|\beta_i\|_1)$$

**Remarque 1.** Soient  $f_{\mathbf{y}_i}^{\beta_i=\mathbf{b}}(\mathbf{y})$  la densité de  $\mathbf{y}_i$  sachant que  $\beta_i = \mathbf{b}$ , et  $f_{\beta_i}(\mathbf{b})$  la densité de  $\beta_i$ , définies par :

$$\begin{aligned} f_{\mathbf{y}_i}^{\beta_i=\mathbf{b}}(\mathbf{y}) &= (2\pi\sigma^2)^{-\frac{n}{2}} \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2\right) \\ f_{\beta_i}(\mathbf{b}) &= \left( \frac{\alpha}{2} \right)^p \exp(-\alpha \|\mathbf{b}\|_1) \end{aligned} \quad (2.4)$$

La densité de probabilité jointe entre  $\mathbf{y}_i$  et  $\beta_i$  est :

$$\begin{aligned} f_{(\mathbf{y}_i, \beta_i)}(\mathbf{y}, \mathbf{b}) &= f_{\mathbf{y}_i}^{\beta_i=\mathbf{b}}(\mathbf{y}) f_{\beta_i}(\mathbf{b}) \\ &= (2\pi\sigma^2)^{-\frac{n}{2}} \left( \frac{\alpha}{2} \right)^p \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 - \alpha \|\mathbf{b}\|_1\right) \end{aligned} \quad (2.5)$$

La loi marginale de  $\mathbf{y}_i$  s'écrit donc :

$$f_{\mathbf{y}_i}(\mathbf{y}) = \int_{\mathbb{R}^p} (2\pi\sigma^2)^{-\frac{n}{2}} \left( \frac{\alpha}{2} \right)^p \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 - \alpha \|\mathbf{b}\|_1\right) d\mathbf{b}$$

Ainsi la loi a posteriori à maximiser pour chaque  $\beta_i$  est :

$$\begin{aligned} f_{\beta_i}^{\mathbf{y}_i=\mathbf{y}}(\mathbf{b}) &= \frac{f_{(\mathbf{y}_i, \beta_i)}(\mathbf{y}, \mathbf{b})}{f_{\mathbf{y}_i}(\mathbf{y})} \\ &= \frac{(2\pi\sigma^2)^{-\frac{n}{2}} \left( \frac{\alpha}{2} \right)^p \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 - \alpha \|\mathbf{b}\|_1\right)}{\int_{\mathbb{R}^p} (2\pi\sigma^2)^{-\frac{n}{2}} \left( \frac{\alpha}{2} \right)^p \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 - \alpha \|\mathbf{b}\|_1\right) d\mathbf{b}} \end{aligned} \quad (2.6)$$

Il n'existe pas de forme explicite pour cet estimateur. La méthode *LASSO* permet de le trouver. Cette méthode pénalise la norme  $L^1$  des vecteurs de régression, et ainsi force sa parcimonie : un petit nombre de coefficients de ce vecteur sera non nul.

### 2.1.2 Régression *LARS*

Cet algorithme fut pour la première fois introduit dans l'article éponyme [8]. L'algorithme LARS fait partie de la famille des algorithmes de **sélection de variables** (*features selection*), au sens où il permet d'obtenir un petit nombre d'attributs qui seront utilisés ensuite dans la régression.

L'idée de base est de changer la direction de plus forte corrélation par la direction équiangulaire à tous les prédicteurs dans une étape donnée. La difficulté théorique est de définir une direction équiangulaire dans un espace de dimension potentiellement grand.

**Définition 6** (Vecteur équiangulaire). Notons :

- $\mathcal{A}$  un sous ensemble de  $\{1, 2, \dots, p\}$  de cardinal  $p' < p$ .
- $\mathbf{X}_{\mathcal{A}} = (\dots s_j \mathbf{x}_j \dots)_{j \in \mathcal{A}} \in \mathbb{R}^{n \times p'}$  avec  $s_j = \pm 1$  une sous matrice de  $\mathbf{X}$
- $\mathbf{G}_{\mathcal{A}} = \mathbf{X}_{\mathcal{A}}^{\top} \mathbf{X}_{\mathcal{A}} \in \mathbb{R}^{p' \times p'}$
- $A_{\mathcal{A}} = (\mathbf{1}_{\mathcal{A}}^{\top} \mathbf{G}_{\mathcal{A}}^{-1} \mathbf{1}_{\mathcal{A}})^{-\frac{1}{2}} \in \mathbb{R}$  avec  $\mathbf{1}_{\mathcal{A}}$  un vecteur de dimension  $p'$  contenant des 1.
- $\mathbf{w}_{\mathcal{A}} = A_{\mathcal{A}} \mathbf{G}_{\mathcal{A}}^{-1} \mathbf{1}_{\mathcal{A}} \in \mathbb{R}^{p'}$

Le vecteur équiangulaire est donc le vecteur :

$$\mathbf{u}_{\mathcal{A}} = \mathbf{X}_{\mathcal{A}} \mathbf{w}_{\mathcal{A}} \in \mathbb{R}^n \quad (2.7)$$

C'est le vecteur unitaire faisant des angles égaux inférieurs à  $90^\circ$  avec les colonnes de  $\mathbf{X}_{\mathcal{A}}$ . Il vérifie la propriété :

$$\mathbf{X}_{\mathcal{A}}^{\top} \mathbf{u}_{\mathcal{A}} = A_{\mathcal{A}} \mathbf{1}_{\mathcal{A}} \quad (2.8)$$

L'algorithme LARS va rechercher de manière itérative ces différentes directions équiangulaires. On cherche à construire  $\hat{\mathbf{y}} = \mathbf{X}\boldsymbol{\beta}$  (cas univarié pour le moment). Le nombre de coefficients de régression équivaut au nombre de ces directions : il ne dépassera pas  $p$ . L'algorithme est construit comme suit : on commence par initialiser avec  $\hat{\mathbf{y}} = 0$ . On note  $\hat{\mathbf{y}}_{\mathcal{A}}$  le vecteur estimé courant et  $\hat{\mathbf{c}} = \mathbf{X}^{\top}(\mathbf{y} - \hat{\mathbf{y}}_{\mathcal{A}})$  le vecteur des corrélations courant. L'ensemble actif  $\mathcal{A}$  est l'ensemble des indices correspondant aux attributs avec les plus grandes corrélations :  $\hat{C} = \max |\hat{c}_j|$  et  $\mathcal{A} = \{j \mid |\hat{c}_j| = \hat{C}\}$ . Soit  $s_j = \text{sign}(\hat{c}_j) \forall j \in \mathcal{A}$ . On calcule  $\mathbf{X}_{\mathcal{A}}$ ,  $A_{\mathcal{A}}$ ,  $\mathbf{u}_{\mathcal{A}}$  comme définis précédemment ainsi que  $\mathbf{a} = \mathbf{X}^{\top} \mathbf{u}_{\mathcal{A}}$ .

On met à jour avec la formule :

$$\hat{\mathbf{y}}_{\mathcal{A}_+} \leftarrow \hat{\mathbf{y}}_{\mathcal{A}} + \hat{\gamma} \mathbf{u}_{\mathcal{A}}$$

Avec :

$$\hat{\gamma} = \min_{j \in \mathcal{A}^c} \left( \frac{\hat{C} - \hat{c}_j}{A_{\mathcal{A}} - a_j}, \frac{\hat{C} + \hat{c}_j}{A_{\mathcal{A}} + a_j} \right)$$

$\hat{\gamma}$  est la plus petite valeur positive telle qu'un certain nouveau  $\hat{j}$  entre dans  $\mathcal{A}$  :  $\mathcal{A}_+ = \mathcal{A} \cup \{\hat{j}\}$ .

**Algorithme 1 : LARS****Entrées :**  $\mathbf{X}$ ,  $\mathbf{y}$ **Initialisation :** Centrer  $\mathbf{y}$ , Normaliser  $\mathbf{X}$ ,  $\hat{\mathbf{y}}_{\mathcal{A}} = 0$ ,  $\mathcal{A} = \emptyset$ ,  $\hat{C} = 0$ ,  $\hat{\mathbf{c}} = 0$ **pour**  $i = 1$  à  $p$  **faire**

$$\hat{\mathbf{c}} = \mathbf{X}^\top (\mathbf{y} - \hat{\mathbf{y}}_{\mathcal{A}}) ;$$

$$\hat{C} = \max |\hat{c}_j| ;$$

$$\mathcal{A} = \{j \mid |\hat{c}_j| = \hat{C}\} ;$$

$$s_j = \text{sign}(\hat{c}_j) \forall j \in \mathcal{A} ;$$

$$\mathbf{X}_{\mathcal{A}} = (\dots s_j \mathbf{x}_j \dots)_{j \in \mathcal{A}}, s_j = \pm 1 ;$$

$$A_{\mathcal{A}} = (\mathbf{1}_{\mathcal{A}}^\top \mathbf{G}_{\mathcal{A}}^{-1} \mathbf{1}_{\mathcal{A}})^{-\frac{1}{2}} ;$$

$$\mathbf{w}_{\mathcal{A}} = A_{\mathcal{A}} \mathbf{G}_{\mathcal{A}}^{-1} \mathbf{1}_{\mathcal{A}} ;$$

$$\mathbf{u}_{\mathcal{A}} = \mathbf{X}_{\mathcal{A}} \mathbf{w}_{\mathcal{A}} ;$$

$$\mathbf{a} = \mathbf{X}^\top \mathbf{u}_{\mathcal{A}} ;$$

$$\hat{\gamma} = \min_{j \in \mathcal{A}^c} \left( \frac{\hat{C} - \hat{c}_j}{A_{\mathcal{A}} - a_j}, \frac{\hat{C} + \hat{c}_j}{A_{\mathcal{A}} + a_j} \right) ;$$

$$\mathcal{A}_+ = \mathcal{A} \cup \{j\} ;$$

$$\hat{\mathbf{y}}_{\mathcal{A}_+} = \hat{\mathbf{y}}_{\mathcal{A}} + \hat{\gamma} \mathbf{u}_{\mathcal{A}} ;$$

**fin****2.1.3 Régression SVM**

Les éléments de ce paragraphe sont issus de [4, 1, 5]. Les SVM (pour *Support Vector Machines*, ou Séparatrice à Vastes Marges) sont des méthodes non linéaires qui permettent de faire une transformation non linéaire des entrées (supposées appartenant à un espace  $\mathcal{X}$ ) dans un espace de dimension potentiellement infini (noté  $\mathcal{F}$ ) sans avoir à connaître la forme explicite de la transformation : c'est le *kernel trick*.

**Définition 7** (Fonction Noyau). Une fonction noyau est une fonction  $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  telle que pour  $\phi : \mathcal{X} \rightarrow \mathcal{F}$  avec  $\mathcal{F}$  un espace doté d'un produit scalaire  $\langle \cdot | \cdot \rangle$  :

$$\forall (\mathbf{x}, \mathbf{x}') \in \mathcal{X} \times \mathcal{X}, k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}) | \phi(\mathbf{x}') \rangle \quad (2.9)$$

À cette fonction noyau est associée sa matrice de GRAM :

**Définition 8** (Matrice de GRAM). La matrice de GRAM  $\mathbf{K} = (K_{ij})$  d'une fonction noyau est définie par : pour  $(\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{X} \times \mathcal{X}$ ,

$$K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$$

**Définition 9** (Fonction semi-définie positive). Une fonction  $k$  est semi-définie positive si pour  $\mu$  une mesure sur  $\mathcal{X}$  et  $\forall f \in L_2(\mathcal{X}, \mu)$  :

$$\int_{\mathcal{X} \times \mathcal{X}} k(\mathbf{x}, \mathbf{x}') f(\mathbf{x}) f(\mathbf{x}') d\mu(\mathbf{x}) d\mu(\mathbf{x}') \geq 0$$

Le théorème de MERCER est le théorème fondamental de la technique SVM car il permet de se passer des produits scalaires entre les fonctions  $\phi$  pour n'utiliser que des fonctions noyaux.

**Théorème 1** (Théorème de MERCER). Si  $k(\cdot, \cdot)$  est une fonction noyau continue symétrique d'un opérateur intégral :

$$g(y) = Af(y) = \int_a^b k(x, y) f(y) dy + h(x)$$

Vérifiant  $\forall f \in L_2(\mathcal{X})$  :

$$\int_{\mathcal{X} \times \mathcal{X}} k(\mathbf{x}, \mathbf{x}') f(\mathbf{x}) f(\mathbf{x}') d\mu(\mathbf{x}) d\mu(\mathbf{x}') \geq 0$$

Alors la fonction  $k(\cdot, \cdot)$  peut être développée en série uniformément convergente en fonction des valeurs propres positives  $\lambda_i$  et des fonctions propres  $\psi_i$  (avec  $N$  le nombre de valeurs propres positives, éventuellement infini) :

$$k(\mathbf{x}, \mathbf{x}') = \sum_{i=1}^N \lambda_i \psi_i(\mathbf{x}) \psi_i(\mathbf{x}') \quad (2.10)$$

Dans le cas de SVM et des données usuelles, les conditions du théorème seront toujours vérifiées.

Le reste du paragraphe présente la résolution du problème des Séparateurs à Vastes Marges appliqué à la régression. L'idée (en classification) est de trouver l'hyperplan séparateur ayant la marge maximale avec les points proches de cet hyperplan. En régression cela revient à chercher la fonction  $h(\mathbf{x})$  telle que pour chaque point de l'ensemble d'apprentissage  $(\mathbf{x}_i, y_i)$ ,  $h(\mathbf{x}_i)$  soit le plus proche possible de  $y_i$ .

Le modèle consiste en une représentation des sorties comme combinaison linéaire d'une fonction des entrées :

$$y = h(\mathbf{x}, \mathbf{w}) = \sum_{i=0}^{\infty} w_i \phi_i(\mathbf{x}) \quad (2.11)$$

La fonction de perte considérée est :

$$|y - h(\mathbf{x})|_{\epsilon} = \max(0, |y - h(\mathbf{x})| - \epsilon)$$

Le problème consiste à maximiser la marge entre les points de la frontière, écrit sous la forme d'un problème primal :

$$\min_{\mathbf{w}} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^m (\xi_i + \xi_i^*) \quad (2.12)$$

Sous contraintes :

$$\begin{aligned} (\mathbf{w}^{\top} \mathbf{x}_i + w_0) - y_i &\leq \epsilon + \xi_i, i = 1, \dots, m \\ y_i - (\mathbf{w}^{\top} \mathbf{x}_i + w_0) &\leq \epsilon + \xi_i^*, i = 1, \dots, m \\ \xi_i, \xi_i^* &\geq 0 \end{aligned}$$

$C$  représente la "souplesse" avec laquelle relative à la marge : elle est associée aux variables ressorts  $\xi_i$  et  $\xi_i^*$ .

En introduisant les multiplicateurs de LAGRANGE, la formulation duale du problème s'écrit :

$$\max_{\alpha, \alpha^*} W(\alpha, \alpha^*) = -\epsilon \sum_{i=1}^m (\alpha_i^* + \alpha_i) + \epsilon \sum_{i=1}^m (\alpha_i^* - \alpha_i) y_i - \frac{1}{2} \sum_{i,j=1}^m (\alpha_i^* - \alpha_i) (\alpha_j^* - \alpha_j) \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle \quad (2.13)$$

Sous contraintes :

$$\begin{aligned} 0 &\leq \alpha_i, \alpha_i^* \leq C \\ \sum_{i=1}^m (\alpha_i^* - \alpha_i) &= 0 \end{aligned}$$

Finalement, les fonctions  $\phi$  n'interviennent que par leur produit scalaire : en utilisant le Théorème de MERCER, ces produits scalaires sont remplacés par une fonction noyau  $k(\cdot, \cdot)$  :

$$\max_{\alpha, \alpha^*} W(\alpha, \alpha^*) = -\epsilon \sum_{i=1}^m (\alpha_i^* + \alpha_i) + \epsilon \sum_{i=1}^m (\alpha_i^* - \alpha_i) y_i - \frac{1}{2} \sum_{i,j=1}^m (\alpha_i^* - \alpha_i) (\alpha_j^* - \alpha_j) k(\mathbf{x}_i, \mathbf{x}_j) \quad (2.14)$$

Sous contraintes :

$$\begin{aligned} 0 &\leq \alpha_i, \alpha_i^* \leq C \\ \sum_{i=1}^m (\alpha_i^* - \alpha_i) &= 0 \end{aligned}$$

Ces coefficients  $\alpha$  et  $\alpha^*$  sont trouvés par une programmation quadratique classique. La fonction de régression s'écrit donc :

$$h(\mathbf{x}) = \sum_{i=1}^m (\alpha_i^* - \alpha_i) k(\mathbf{x}_i, \mathbf{x}) + w_0 \quad (2.15)$$

### 2.1.4 Régression *Sparse*

L'idée est tirée de l'article de BILINSKI en 2014 [2]. Le dénominateur *Sparse* est uniquement utilisé dans ce mémoire pour désigner cette technique particulière. Elle consiste à représenter les paramètres morphologiques d'un nouveau sujet à partir d'une base de données de paramètres morphologiques, et à utiliser la même représentation pour synthétiser les HRTF.

Soient  $\mathcal{H} \in \mathbb{R}^{P \times D \times F}$  le tenseur contenant les HRTF (comportant  $D$  directions et  $F$  fréquences) pour les  $P$  sujets de la base, et  $\mathbf{A} \in \mathbb{R}^{P \times K}$  les  $K$  paramètres morphologiques pour les  $P$  sujets.

Soit  $\mathbf{a}^{new} \in \mathbb{R}^K$  les paramètres morphologiques d'un nouveau sujet. La méthode consiste à les représenter de manière parcimonieuse en utilisant les paramètres de la base, ce qui définit donc le modèle :

$$\mathbf{a}^{new} = \boldsymbol{\beta}^\top \mathbf{A} \quad (2.16)$$

Avec  $\boldsymbol{\beta} = [\beta_1, \dots, \beta_P]^\top$  le vecteur parcimonieux de représentation. Ce vecteur est appris par la méthode LASSO (voir 2.1.1) :

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\| \mathbf{a}^{new} - \boldsymbol{\beta}^\top \mathbf{A} \right\|_2^2 + \lambda \|\boldsymbol{\beta}\|_1 \quad (2.17)$$

Ensuite les HRTF sont prédits fréquence par fréquence ( $f$ ) et position par position ( $d$ ) :

$$\hat{H}_{d,f} = \sum_{p=1}^P \hat{\beta}_p H_{p,d,f} \quad (2.18)$$

Le pré-traitement des données s'inspire de l'article de HE et *al.* en 2015 [12]. Leur conclusion montre que le meilleur pré-traitement est le suivant :

- Utiliser des paramètres parcimonieux non négatifs : il s'agit d'une contrainte supplémentaire dans l'algorithme de minimisation qui porte sur le signe des coefficients du vecteur  $\beta$ .
- Normaliser les paramètres morphologiques (centrer puis réduire)
- Utiliser les log-amplitudes des HRTF
- Normaliser les coefficients par le maximum.

### 2.1.5 Régression PLS

**Définition 10.** La PLS (*Partial Least Square*) consiste à chercher un ensemble de vecteurs latents en décomposant simultanément deux matrices  $\mathbf{X} \in \mathbb{R}^{I \times J}$  et  $\mathbf{Y} \in \mathbb{R}^{I \times M}$  sous contrainte que les composantes de ces vecteurs latents expliquent autant que possible la covariance entre  $\mathbf{X}$  et  $\mathbf{Y}$ . Soient  $\mathbf{T}, \mathbf{U} \in \mathbb{R}^{I \times R}$ ,  $\mathbf{P} \in \mathbb{R}^{R \times J}$  et  $\mathbf{Q} \in \mathbb{R}^{R \times M}$  et  $\mathbf{E} \in \mathbb{R}^{I \times J}$  et  $\mathbf{F} \in \mathbb{R}^{I \times M}$ . Le modèle de la PLS s'écrit :

$$\begin{aligned}\mathbf{X} &= \mathbf{TP}^\top + \mathbf{E} = \sum_{r=1}^R \mathbf{t}_r \mathbf{p}_r^\top + \mathbf{E} \\ \mathbf{Y} &= \mathbf{UQ}^\top + \mathbf{F} = \sum_{r=1}^R \mathbf{u}_r \mathbf{q}_r^\top + \mathbf{F}\end{aligned}\tag{2.19}$$

$\mathbf{T}$  contient les  $R$  vecteurs latents orthogonaux extraits de  $\mathbf{X}$ ,  $\mathbf{U}$  contient les  $R$  vecteurs latents  $\mathbf{Y}$  qui ont le maximum de covariance avec  $\mathbf{T}$  colonne par colonne.  $\mathbf{P}$  et  $\mathbf{Q}$  sont les matrices loadings, et  $\mathbf{E}$  et  $\mathbf{F}$  sont les résidus.

Pour trouver le premier ensemble de composantes, il suffit de résoudre :

$$\max_{(\mathbf{w}, \mathbf{q})} \mathbf{w}^\top \mathbf{X}^\top \mathbf{Y} \mathbf{q}$$

Sous contraintes :

$$\mathbf{w}^\top \mathbf{w} = 1, \quad \mathbf{q}^\top \mathbf{q} = 1$$

On en déduit le vecteur latent  $\mathbf{t} = \mathbf{X}\mathbf{w}$ . On cherche à présent une relation linéaire du type  $\mathbf{u} = d\mathbf{t}$ . Si on pose :

$$\begin{aligned}d_{rr} &= \frac{\mathbf{u}_r^\top \mathbf{t}_r}{\mathbf{t}_r^\top \mathbf{t}_r} \\ \mathbf{D} &= \text{diag}(d_{rr})\end{aligned}\tag{2.20}$$

On en déduit donc  $\mathbf{Y} = \mathbf{TDQ}^\top$ . Par ce biais il n'y a plus qu'à chercher des variables latentes communes à  $\mathbf{X}$  et  $\mathbf{Y}$ .

## 2.2 Mesure de performance

Pour mesurer les performances des différents algorithmes, la technique du *leave-one-out* est utilisée. Elle consiste à utiliser un sujet pour le test et garder le reste pour l'apprentissage, et répéter ceci pour tous les sujets de la base d'apprentissage.

Les méthodes nécessitant le réglage d'un hyperparamètre utilisent la validation croisée. Une plage de valeur est définie à l'avance et pour chaque valeur, l'ensemble d'apprentissage est divisé en un sous-ensemble d'apprentissage et un sous-ensemble de validation (appelé *fold*). La valeur qui optimise un certain critère sur chaque *fold* sera utilisée par la méthode.

Une procédure d’essai-erreur a permis de choisir des plages de valeurs utiles pour les validations croisées. La pénalisation de la méthode *Ridge* a été choisie parmi 100 valeurs entre 1 et 1000 par espacement logarithmique. La pénalisation de la méthode *SVM* a été choisie parmi 5 valeurs entre 0,01 et 10 par espacement logarithmique. La pénalisation pour la méthode *LARS* a été fixée à 4 (le nombre de coefficients non nuls) et pour la méthode la méthode *Sparse* a été fixée à  $10^{-6}$  pour des raisons de temps de calculs.

La mesure objective de ressemblance entre des HRTF originaux et prédits considérée ici est la corrélation de PEARSON.

**Définition 11** (Corrélation de PEARSON). Soient  $X$  et  $Y$  deux variables aléatoires réelles. Le coefficient de corrélation de PEARSON, noté  $r_{X,Y}$ , entre  $X$  et  $Y$  est défini par :

$$r_{X,Y} = \frac{\text{cov}(X, Y)}{\sqrt{\text{var}(X) \text{var}(Y)}}$$

Une manière visuelle (et subjective) d’étudier la ressemblance entre deux jeux de HRTF est de construire une image dans laquelle l’axe des abscisses représente une direction angulaire, l’axe des ordonnées les fréquences, et la nuance de gris la magnitude du HRTF en dB.

## 2.3 Résultats

La FIGURE 2.2 montre le HRTF du sujet 31 à l’élévation  $0^\circ$ , original (rouge) et prédit par les méthodes décrites précédemment. Les prédictions semblent être toutes dans les mêmes gammes d’amplitude. Par contre, elles décrochent toutes après 5000 Hz : dans les hautes fréquences, le spectre est ”lissé”.

La FIGURE 2.3 montre la distribution des corrélations pour tous les sujets et toutes les directions. Celles-ci sont fortement concentrées autour de 0,9. Le TABLEAU 2.1 donne le nombre de corrélations plus grandes que 0,9 pour chaque méthode. Les corrélations semblent être du même ordre de grandeur pour chaque méthode, avec la *Ridge* et la *PLS* qui ont le plus grand nombre de corrélations supérieures à 0,9 (respectivement 1118 sur 2044 et 1115 sur 2044), et la *SVM* qui a le plus petit nombre (1014 sur 2044).

Méthode	<i>Ridge</i>	<i>LARS</i>	<i>SVM</i>	<i>Sparse</i>	<i>PLS</i>
Nombre	1118	1098	1014	1068	1115

TABLE 2.1 – Nombre de corrélations supérieures à 0,9 pour chaque méthode

Enfin, la FIGURE 2.4 montre les différentes ”cartes” des HRTF, originaux et prédits, pour le sujet 31. La carte des HRTF prédits par la méthode *Ridge* est plus lisse que les autres, alors que la carte des HRTF prédits par la méthode *SVM* comporte plus de détails, elle semble plus rugueuse. Ces cartes montrent que la première résonance est bien prédite par toutes les méthodes (trainée blanche aux alentours de 4 kHz).

## 2.4 Discussion

La FIGURE 2.2 montre que les prédictions sont ”lissées” comparées aux HRTF originaux. La résonance principale est plutôt bien représentée mais les informations en haute fréquence (anti-résonance principale, résonances et anti-résonances secondaires) ont disparu. Or, dans le plan médian, ce sont précisément ces indices qui permettent une bonne localisation sonore.

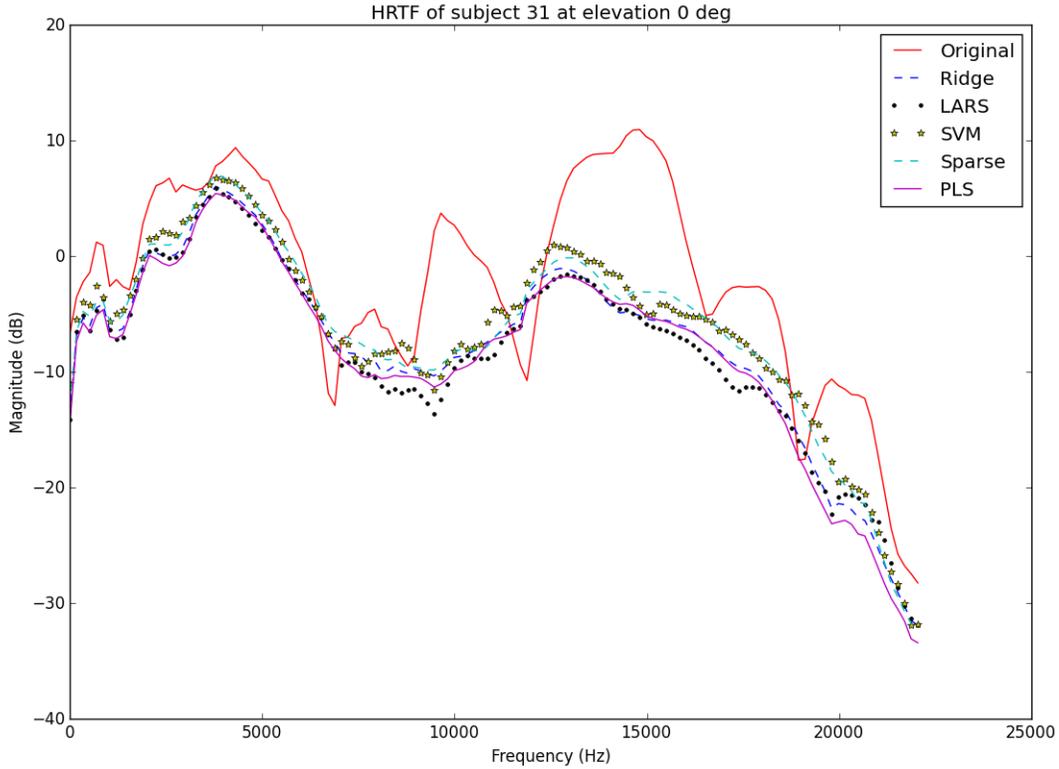


FIGURE 2.2 – Exemple de HRTF original (rouge) et prédit par différentes méthodes pour le sujet 31 à l'élévation  $0^\circ$

Les valeurs des corrélations semblent indiquer que les méthodes linéaires simples donnent de bons résultats.

A l'inverse, si on regarde à FIGURE 2.4, les méthodes comme la SVM ou la PLS semblent donner des prédictions plus proches de la réalité.

La première technique envisagée, la régression *Ridge*, est la plus communément utilisée dans la communauté du *machine learning*. Le principal problème de cette technique est qu'elle ne donne pas de bons résultats quand le nombre d'observations est petit. Dans notre cas,  $P = 73$  est à la limite de ce qui est acceptable pour l'utilisation d'une régression *Ridge*. La pénalisation permet une meilleure prédiction, mais elle enlève une partie de l'interprétabilité du modèle.

La technique *LARS* impose une parcimonie sur les paramètres morphologiques, et réduit donc la complexité du modèle. Néanmoins, dans le cas d'entrées comportant des corrélations entre elles, elle peut donner des résultats de mauvaise qualité. Si on étudie la matrice des corrélations des paramètres morphologiques, on se rend compte que beaucoup de paramètres possèdent des corrélations significatives avec d'autres paramètres. Par exemple, la corrélation de PEARSON entre  $d_6$  pour l'oreille gauche et  $d_6$  pour l'oreille droite est de 0,7, celle entre  $x_8$  et  $x_{12}$  est de 0,9 et celle entre  $x_6$  et  $x_{12}$  est de 0,8. Une étude plus détaillée de ces corrélations a été réalisée par ZHANG et al. [22].

La technique *SVM*, par construction, permet une prédiction de bonne qualité en général, car elle va chercher un espace de dimension potentiellement infinie dans lequel les données seront le mieux

expliquées. Elle est conseillée dans le cas où le nombre d'observations est petit, comme dans notre cas. Néanmoins, c'est un outil "boîte noire", car on ne sait pas quelles sont les transformations qui ont été réalisées, et donc l'interprétation du modèle est plus compliquée.

La technique *Sparse* est en fait le *LASSO* appliqué uniquement pour trouver une représentation des paramètres morphologiques. Son intérêt principal ici est qu'elle permet un apprentissage et une prédiction très rapide car les données mises en jeu sont en quantité réduite. Malheureusement elle souffre des mêmes défauts que le LARS, à savoir que comme les paramètres morphologiques présentent des corrélations entre eux, le modèle peut être de moins bonne qualité.

Enfin, la technique *PLS* possède l'avantage de pouvoir bien fonctionner avec peu de données et en plus d'être interprétable facilement.

Pour tenter de palier au problème du "lissage" dû aux techniques linéaires présentées dans ce chapitre, les prochaines techniques vont tenter de prédire les HRTF, non plus fréquence par fréquence et direction par direction, mais en considérant la structure globale des HRTF, sous la forme tensorielle. C'est l'objet du chapitre 3.

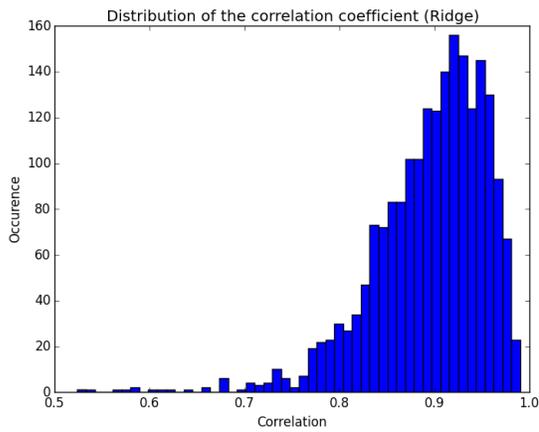
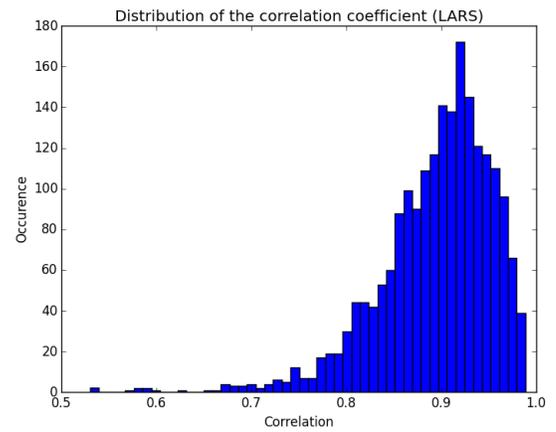
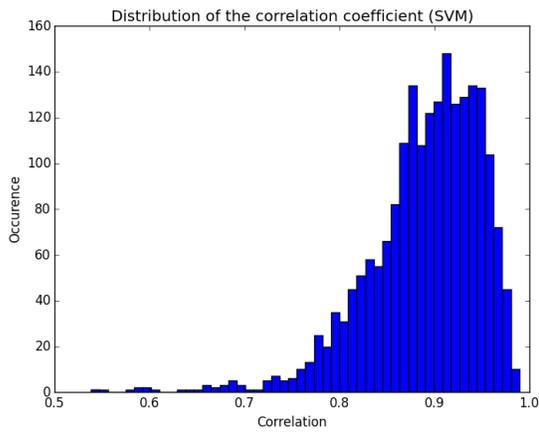
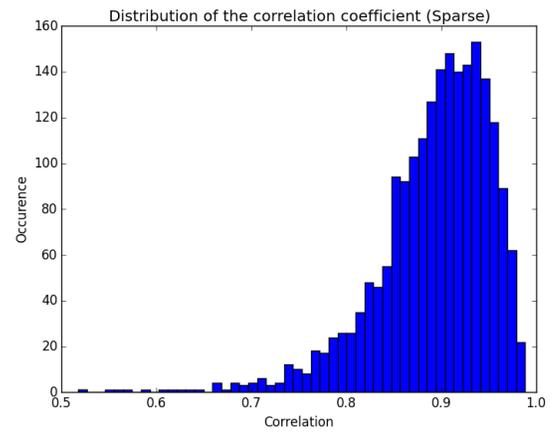
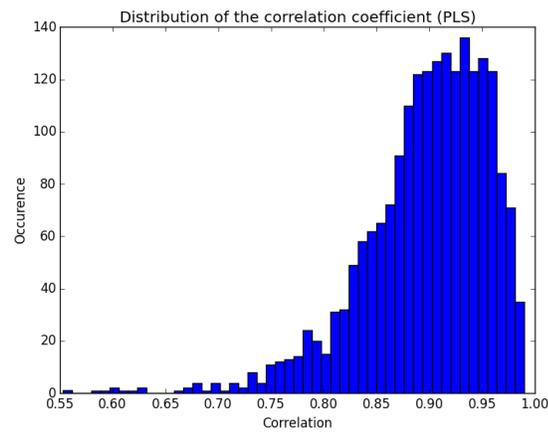
(a) Corrélations pour la méthode *Ridge*(b) Corrélations pour la méthode *LARS*(c) Corrélations pour la méthode *SVM*(d) Corrélations pour la méthode *Sparse*(e) Corrélations pour la méthode *PLS*

FIGURE 2.3 – Distributions des corrélations pour chaque méthode

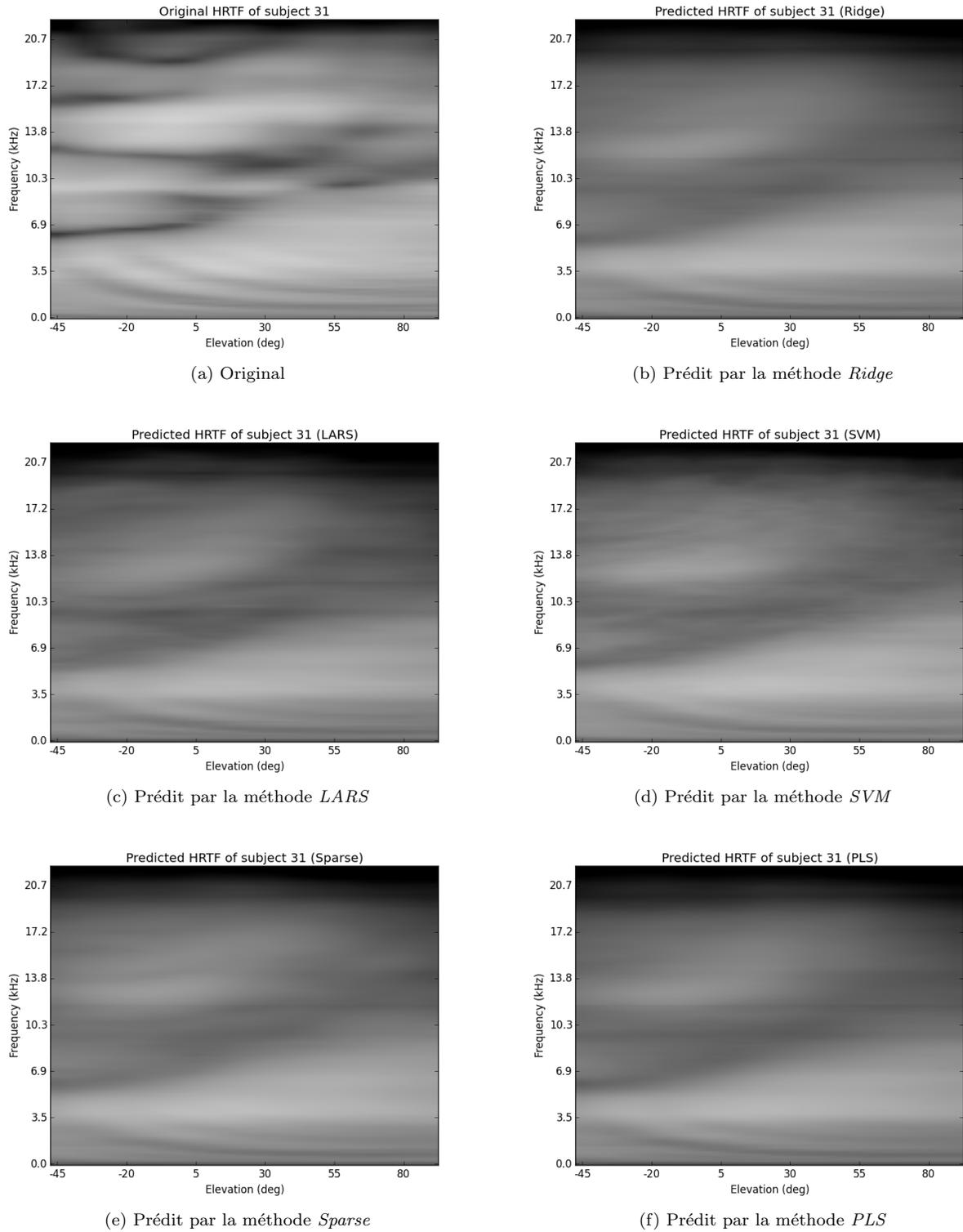


FIGURE 2.4 – Cartes des HRTF du sujet 31, original et prédits

## Chapitre 3

# Méthodes avancées d'apprentissage statistique

Pour tenter de résoudre le problème des prédictions "lissées", une approche tensorielle basée sur une décomposition de type SVD tensorielle puis sur une régression PLS pour tenseurs est mise en oeuvre dans ce chapitre. La section 3.1 présente les outils de travail génériques sur les tenseurs (notamment la HOSVD et la HOPLS), la section 3.2 présente l'application de ces méthodes aux HRTF et paramètres morphologiques, et les sections 3.3 présente les résultats et la section 3.4 présente la discussion.

### 3.1 Approche tensorielle

Le but de cette approche est de considérer la structure tensorielle des HRTF pour garder le maximum d'informations sur l'imbrication entre les sujets, les directions et les fréquences.

#### 3.1.1 Généralités sur les tenseurs

**Définition 12.** Soit  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  un tenseur d'ordre  $N$ . La matricisation suivant le mode  $n$  de  $\mathcal{A}$  est une matrice  $\mathbf{A}_{(n)} = (a_{i_n, l_n}) \in \mathbb{C}^{I_n \times (I_{n+1} \dots I_N I_1 \dots I_{n-1})}$  contenant les éléments  $a_{i_1, i_2, \dots, i_N}$  à la ligne  $i_n$  et à la colonne  $l_n = (i_{n+1} - 1) I_{n+2} \dots I_N I_1 \dots I_{n-1} + (i_{n+2} - 1) I_{n+3} \dots I_N I_1 \dots I_{n-1} + \dots + (i_N - 1) I_1 \dots I_{n-1} + (i_1 - 1) I_2 \dots I_{n-1} + \dots + i_{n-1}$ .

**Exemple 1.** Soit  $\mathcal{A} \in \mathbb{R}^{3 \times 2 \times 3}$  défini par :

$$\begin{aligned} a_{111} &= a_{112} = a_{211} = -a_{212} = 1 \\ a_{213} &= a_{311} = a_{313} = a_{121} = a_{122} = a_{221} = -a_{222} = 2 \\ a_{223} &= a_{321} = a_{323} = 4 \\ a_{112} &= a_{312} = a_{123} = a_{322} = 0 \end{aligned} \tag{3.1}$$

La matricisation suivant le mode 1  $\mathbf{A}_{(1)}$  vaut :

$$\mathbf{A}_{(1)} = \begin{pmatrix} 1 & 1 & 0 & 2 & 2 & 0 \\ 1 & -1 & 2 & 2 & -2 & 4 \\ 2 & 0 & 2 & 4 & 0 & 4 \end{pmatrix} \tag{3.2}$$

En effet c'est bien une matrice  $3 \times (2 * 3) = 3 \times 6$ , avec  $N = 3$ ,  $I_1 = 3$ ,  $I_2 = 2$ ,  $I_3 = 3$ .

**Définition 13** (Vecteurs du mode  $n$ ). Soit  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  un tenseur d'ordre  $N$ . Les vecteurs du mode  $n$  de  $\mathcal{A}$  sont les vecteurs de dimension  $I_n$  obtenus à partir de  $\mathcal{A}$  en faisant varier l'indice  $i_n$  tout en gardant les autres fixes.

**Définition 14** (Rang- $n$ ). Soit  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  un tenseur d'ordre  $N$ . Le rang- $n$  de  $\mathcal{A}$ , noté  $R_n = \text{rang}_n(\mathcal{A})$ , est la dimension de l'espace vectoriel engendré par ses vecteurs du mode  $n$ .

**Propriété 1.** Soit  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  un tenseur d'ordre  $N$ .

- (i) Les vecteurs du mode  $n$  de  $\mathcal{A}$  sont les vecteurs colonnes de la matricisation du mode  $n$   $\mathbf{A}_{(n)}$ .
- (ii)  $\text{rang}_n(\mathcal{A}) = \text{rang}(\mathbf{A}_{(n)})$

**Démonstration 3.** La propriété 1 (i) découle de la définition de la matricisation. La propriété 1 (ii) découle de la (i) car les vecteurs de mode  $n$  sont les vecteurs colonne de  $\mathbf{A}_{(n)}$  et le rang d'une matrice est égal au rang de ses vecteurs colonne. ■

**Définition 15** (Tenseur de rang 1). Soit  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  un tenseur d'ordre  $N$ .  $\mathcal{A}$  est dit tenseur de rang 1 s'il est égal au produit dyadique de  $N$  vecteurs  $\mathbf{u}^{(1)} \in \mathbb{C}^{I_1}, \dots, \mathbf{u}^{(N)} \in \mathbb{C}^{I_N}$  :

$$\mathcal{A} = \mathbf{u}^{(1)} \circ \dots \circ \mathbf{u}^{(N)} \quad (3.3)$$

**Définition 16** (Rang d'un tenseur). Soit  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  un tenseur d'ordre  $N$ . Le rang de  $\mathcal{A}$ , noté  $R = \text{rang}(\mathcal{A})$ , est le nombre minimal de tenseurs de rang 1 dont une combinaison linéaire forme  $\mathcal{A}$ .

**Remarque 2.** En général,  $R_n \leq R$ .

**Exemple 2.** Soit  $\mathcal{A} \in \mathbb{R}^{2 \times 2 \times 2}$  défini par :

$$\begin{aligned} a_{111} &= a_{112} = a_{211} = a_{212} = 3 \\ a_{121} &= a_{221} = -a_{122} = -a_{222} = 2 \end{aligned} \quad (3.4)$$

On a  $R = 2$ , alors que  $R_1 = 1$  et  $R_2 = R_3 = 2$ . En effet,  $\mathcal{A}$  peut s'écrire comme la somme de deux tenseurs de rang 1 :

$$\mathcal{A} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \circ \begin{bmatrix} 1 \\ 1 \end{bmatrix} \circ \begin{bmatrix} 3 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \circ \begin{bmatrix} 1 \\ -1 \end{bmatrix} \circ \begin{bmatrix} 0 \\ 2 \end{bmatrix} \quad (3.5)$$

De plus, si on écrit les trois matricisations de  $\mathcal{A}$  :

$$\begin{aligned} \mathbf{A}_{(1)} &= \begin{pmatrix} 3 & 3 & 2 & -2 \\ 3 & 3 & 2 & -2 \end{pmatrix} \\ \mathbf{A}_{(2)} &= \begin{pmatrix} 3 & 3 & 2 & 2 \\ 3 & 3 & -2 & -2 \end{pmatrix} \\ \mathbf{A}_{(3)} &= \begin{pmatrix} 3 & 3 & 3 & 3 \\ 2 & 2 & -2 & -2 \end{pmatrix} \end{aligned} \quad (3.6)$$

On obtient bien les rangs ci-dessus.

**Définition 17** (Produit scalaire). Soient  $\mathcal{A}, \mathcal{B} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  deux tenseurs d'ordre  $N$ . Le produit scalaire entre  $\mathcal{A}$  et  $\mathcal{B}$ , noté  $\langle \mathcal{A}, \mathcal{B} \rangle$  est défini par :

$$\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \dots \sum_{i_N=1}^{I_N} b_{i_1, \dots, i_N}^* a_{i_1, \dots, i_N} \quad (3.7)$$

**Définition 18** (Norme de tenseur). Soit  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  un tenseur d'ordre  $N$ . La norme de FROBENIUS de  $\mathcal{A}$  est la quantité définie par :

$$\|\mathcal{A}\|_{\mathcal{F}} = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle} \quad (3.8)$$

**Définition 19** (Produit mode  $n$ ). Soit  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  un tenseur d'ordre  $N$  et  $\mathbf{U} \in \mathbb{C}^{J_n \times I_n}$ . Le produit mode  $n$  de  $\mathcal{A}$  par  $\mathbf{U}$ , noté  $\mathcal{A} \times_n \mathbf{U}$ , est un tenseur  $(I_1 \times \dots \times I_{n-1} \times J_n \times I_{n+1} \times \dots \times I_N)$  dont les composantes sont :

$$(\mathcal{A} \times_n \mathbf{U})_{i_1, \dots, i_{n-1}, j_n, i_{n+1}, \dots, i_N} = \sum_{i_n} a_{i_1, \dots, i_{n-1}, i_n, i_{n+1}, \dots, i_N} u_{j_n, i_n} \quad (3.9)$$

**Lemme 1.** Soient  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  un tenseur d'ordre  $N$  et  $\mathbf{F} \in \mathbb{C}^{J_n \times I_n}$  une matrice. Soit  $\mathcal{B} = \mathcal{A} \times_n \mathbf{F}$ . Alors  $\mathbf{B}_{(n)} = \mathbf{F} \mathbf{A}_{(n)}$ .

**Démonstration 4.** La matricisation du mode  $n$  de  $\mathcal{A}$  donne une matrice  $\mathbf{A}_{(n)} = (a_{i_n, l_n})$  avec  $l_n = (i_{n+1} - 1) I_{n+2} \dots I_N I_1 \dots I_{n-1} + (i_{n+2} - 1) I_{n+3} \dots I_N I_1 \dots I_{n-1} + \dots + (i_N - 1) I_1 \dots I_{n-1} + (i_1 - 1) I_2 \dots I_{n-1} + \dots + i_{n-1}$ . Ecrivons les composantes de  $\mathcal{B}$  :

$$b_{i_1, \dots, i_{n-1}, j_n, i_{n+1}, \dots, i_N} = \sum_{i_n=1}^{I_n} f_{j_n, i_n} a_{i_1, \dots, i_{n-1}, i_n, i_{n+1}, \dots, i_N}$$

La matricisation du mode  $n$  de  $\mathcal{B}$  donne une matrice  $\mathbf{B}_{(n)} = (b_{i_n, k_n})$  avec  $l_n = k_n$  car les indices sont identiques sauf pour  $i_n$ . De plus,  $\mathbf{F} \mathbf{A}_{(n)} = (c_{j_n, l_n})$  avec  $c_{j_n, l_n}$  défini par :

$$c_{j_n, l_n} = \sum_{i_n=1}^{I_n} f_{j_n, i_n} a_{i_n, l_n}$$

Comme :

$$b_{j_n, k_n} = \sum_{i_n=1}^{I_n} f_{j_n, i_n} a_{i_n, k_n}$$

Il vient que  $\mathbf{B}_{(n)} = \mathbf{F} \mathbf{A}_{(n)}$ . ■

**Propriété 2.** Soient  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  un tenseur d'ordre  $N$  et  $\mathbf{F} \in \mathbb{C}^{J_n \times I_n}$ ,  $\mathbf{G} \in \mathbb{C}^{J_m \times I_m}$ ,  $\mathbf{H} \in \mathbb{C}^{K_n \times J_n}$  trois matrices. Alors :

- (i)  $(\mathcal{A} \times_n \mathbf{F}) \times_m \mathbf{G} = (\mathcal{A} \times_m \mathbf{G}) \times_n \mathbf{F} = \mathcal{A} \times_n \mathbf{F} \times_m \mathbf{G}$
- (ii)  $(\mathcal{A} \times_n \mathbf{F}) \times_n \mathbf{H} = \mathcal{A} \times_n (\mathbf{H} \mathbf{F})$

**Démonstration 5.** Commençons par démontrer la propriété 2 (i). Soient  $\mathcal{B} = \mathcal{A} \times_n \mathbf{F}$  et  $\mathcal{C} = \mathcal{A} \times_m \mathbf{G}$ . Supposons que  $n < m$ . Par définition, on a :

$$\begin{aligned} (\mathcal{B})_{i_1, \dots, i_{n-1}, j_n, i_{n+1}, \dots, i_N} &= \sum_{i_n} a_{i_1, \dots, i_{n-1}, i_n, i_{n+1}, \dots, i_N} f_{j_n, i_n} \\ (\mathcal{C})_{i_1, \dots, i_{m-1}, j_m, i_{m+1}, \dots, i_N} &= \sum_{i_m} a_{i_1, \dots, i_{m-1}, i_m, i_{m+1}, \dots, i_N} g_{j_m, i_m} \end{aligned}$$

Le produit de  $\mathcal{B}$  par  $\mathbf{G}$  s'écrit :

$$\begin{aligned} (\mathcal{B} \times_m \mathbf{G})_{i_1, \dots, i_{n-1}, j_n, i_{n+1}, \dots, i_{m-1}, j_m, i_{m+1}, \dots, i_N} &= \sum_{i_m} \left( \sum_{i_n} a_{i_1, \dots, i_{n-1}, i_n, i_{n+1}, \dots, i_N} f_{j_n, i_n} \right) g_{j_m, i_m} \\ &= \sum_{i_n} \left( \sum_{i_m} a_{i_1, \dots, i_{m-1}, i_m, i_{m+1}, \dots, i_N} g_{j_m, i_m} \right) f_{j_n, i_n} \\ &= (\mathcal{C} \times_n \mathbf{F})_{i_1, \dots, i_{n-1}, j_n, i_{n+1}, \dots, i_{m-1}, j_m, i_{m+1}, \dots, i_N} \end{aligned} \quad (3.10)$$

Donc  $(\mathcal{A} \times_n \mathbf{F}) \times_m \mathbf{G} = (\mathcal{A} \times_n \mathbf{G}) \times_m \mathbf{F}$ .

Montrons 2 (ii). En considérant les matricisations du mode  $n$ , soient  $\mathcal{B} = \mathcal{A} \times_n \mathbf{F}$ ,  $\mathcal{C} = \mathcal{B} \times_n \mathbf{H}$  et  $\mathcal{D} = \mathcal{A} \times_n (\mathbf{HF})$ . Alors :

$$\begin{aligned} \mathbf{B}_{(n)} &= \mathbf{FA}_{(n)} \\ \mathbf{C}_{(n)} &= \mathbf{HB}_{(n)} = \mathbf{HFA}_{(n)} \\ \mathbf{D}_{(n)} &= \mathbf{HFA}_{(n)} \end{aligned} \quad (3.11)$$

Donc  $(\mathcal{A} \times_n \mathbf{F}) \times_n \mathbf{H} = \mathcal{A} \times_n (\mathbf{HF})$ . ■

**Proposition 3.** Soient  $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ . Alors :

$$\|\mathcal{X} - \mathcal{Y}\|_{\mathcal{F}}^2 = \|\mathcal{X}\|_{\mathcal{F}}^2 - 2\langle \mathcal{X}, \mathcal{Y} \rangle + \|\mathcal{Y}\|_{\mathcal{F}}^2$$

**Démonstration 6.**

$$\begin{aligned} \|\mathcal{X} - \mathcal{Y}\|_{\mathcal{F}}^2 &= \langle \mathcal{X} - \mathcal{Y}, \mathcal{X} - \mathcal{Y} \rangle \\ &= \langle \mathcal{X}, \mathcal{X} - \mathcal{Y} \rangle - \langle \mathcal{Y}, \mathcal{X} - \mathcal{Y} \rangle \\ &= \langle \mathcal{X}, \mathcal{X} \rangle - \langle \mathcal{X}, \mathcal{Y} \rangle - \langle \mathcal{Y}, \mathcal{X} \rangle + \langle \mathcal{Y}, \mathcal{Y} \rangle \\ &= \|\mathcal{X}\|_{\mathcal{F}}^2 - 2\langle \mathcal{X}, \mathcal{Y} \rangle + \|\mathcal{Y}\|_{\mathcal{F}}^2 \end{aligned} \quad (3.12)$$

**Proposition 4.** Soient  $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_{n-1} \times J \times I_{n+1} \times \dots \times I_N}$ ,  $\mathcal{Y} \in \mathbb{R}^{I_1 \times \dots \times I_{n-1} \times K \times I_{n+1} \times \dots \times I_N}$  et  $\mathbf{A} \in \mathbb{R}^{J \times K}$ . Alors :

$$\langle \mathcal{X}, \mathcal{Y} \times_n \mathbf{A} \rangle = \langle \mathcal{X} \times_n \mathbf{A}^\top, \mathcal{Y} \rangle$$

**Démonstration 7.** On note :

$$\begin{aligned} (\mathcal{Y} \times_n \mathbf{A})_{i_1, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N} &= \sum_k y_{i_1, \dots, i_{n-1}, k, i_{n+1}, \dots, i_N} a_{jk} = y'_{i_1, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N} \\ (\mathcal{X} \times_n \mathbf{A}^\top)_{i_1, \dots, i_{n-1}, k, i_{n+1}, \dots, i_N} &= \sum_j x_{i_1, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N} a_{kj} = x'_{i_1, \dots, i_{n-1}, k, i_{n+1}, \dots, i_N} \end{aligned} \quad (3.13)$$

Alors :

$$\begin{aligned} \langle \mathcal{X}, \mathcal{Y} \times_n \mathbf{A} \rangle &= \sum_{i_1, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N} x_{i_1, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N} y'_{i_1, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N} \\ &= \sum_{i_1, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N} x_{i_1, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N} \left( \sum_k y_{i_1, \dots, i_{n-1}, k, i_{n+1}, \dots, i_N} a_{jk} \right) \\ &= \sum_{i_1, \dots, i_{n-1}, k, i_{n+1}, \dots, i_N} \left( \sum_j x_{i_1, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N} a_{kj} \right) y_{i_1, \dots, i_{n-1}, k, i_{n+1}, \dots, i_N} \\ &= \sum_{i_1, \dots, i_{n-1}, k, i_{n+1}, \dots, i_N} x'_{i_1, \dots, i_{n-1}, k, i_{n+1}, \dots, i_N} y_{i_1, \dots, i_{n-1}, k, i_{n+1}, \dots, i_N} \\ &= \langle \mathcal{X} \times_n \mathbf{A}^\top, \mathcal{Y} \rangle \end{aligned} \quad (3.14)$$

■

**Proposition 5.** Soient  $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_{n-1} \times J \times I_{n+1} \times \dots \times I_N}$  et  $\mathbf{A} \in \mathbb{R}^{I_n \times I_n}$  une matrice orthogonale. Alors :

$$\|\mathcal{X} \times_n \mathbf{A}\|_{\mathcal{F}} = \|\mathcal{X}\|_{\mathcal{F}}$$

**Démonstration 8.**

$$\begin{aligned} \|\mathcal{X} \times_n \mathbf{A}\|_{\mathcal{F}} &= \langle \mathcal{X} \times_n \mathbf{A}, \mathcal{X} \times_n \mathbf{A} \rangle \\ &= \left\langle \mathcal{X}, (\mathcal{X} \times_n \mathbf{A}) \times_n \mathbf{A}^{\top} \right\rangle \\ &= \left\langle \mathcal{X}, \mathcal{X} \times_n (\mathbf{A}^{\top} \mathbf{A}) \right\rangle \\ &= \langle \mathcal{X}, \mathcal{X} \times_n I_n \rangle \\ &= \langle \mathcal{X}, \mathcal{X} \rangle \\ &= \|\mathcal{X}\|_{\mathcal{F}} \end{aligned} \tag{3.15}$$

■

**Lemme 2.** Soient  $\mathcal{A}, \mathcal{S} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  deux tenseurs d'ordre  $N$ , et  $\mathbf{U}^{(1)} \in \mathbb{C}^{I_1 \times I_1}, \dots, \mathbf{U}^{(N)} \in \mathbb{C}^{I_N \times I_N}$   $N$  matrices orthogonales, tels que :

$$\mathcal{S} = \mathcal{A} \times_1 \mathbf{U}^{(1)\top} \dots \times_N \mathbf{U}^{(N)\top}$$

Alors :

$$\mathbf{A}_{(n)} = \mathbf{U}^{(n)} \mathbf{S}_{(n)} \left( \mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \dots \otimes \mathbf{U}^{(n-1)} \right)^{\top}$$

**Démonstration 9.** Par associativité du produit mode  $n$ , on peut écrire :

$$\mathcal{S} = \left( \mathcal{A} \times_1 \mathbf{U}^{(1)\top} \dots \times_{n-1} \mathbf{U}^{(n-1)\top} \times_{n+1} \mathbf{U}^{(n+1)\top} \times_{n+2} \dots \times_N \mathbf{U}^{(N)\top} \right) \times_n \mathbf{U}^{(n)\top}$$

La matricisation mode  $n$  de  $\mathcal{S}$  donne donc :

$$\mathbf{S}_{(n)} = \mathbf{U}^{(n)\top} \left( \mathcal{A} \times_1 \mathbf{U}^{(1)\top} \dots \times_{n-1} \mathbf{U}^{(n-1)\top} \times_{n+1} \mathbf{U}^{(n+1)\top} \times_{n+2} \dots \times_N \mathbf{U}^{(N)\top} \right)_{(n)}$$

Montrons maintenant les deux propriétés suivantes :

- (i) Soient  $\mathbf{U} \in \mathbb{R}^{n \times m}$  et  $\mathbf{V} \in \mathbb{R}^{p \times q}$ . Alors  $(\mathbf{U} \otimes \mathbf{V})^{\top} = \mathbf{U}^{\top} \otimes \mathbf{V}^{\top}$ .  
(ii)  $\left( \mathcal{A} \times_1 \mathbf{U}^{(1)\top} \dots \times_{n-1} \mathbf{U}^{(n-1)\top} \times_{n+1} \mathbf{U}^{(n+1)\top} \times_{n+2} \dots \times_N \mathbf{U}^{(N)\top} \right)_{(n)} =$   
 $\mathbf{A}_{(n)} \left( \mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \dots \otimes \mathbf{U}^{(n-1)} \right)^{\top}$

Pour montrer (i), il suffit d'écrire les expressions des matrices issues du produit de KRONECKER :

$$\begin{aligned} \mathbf{U} \otimes \mathbf{V} &= \begin{pmatrix} u_{11} \mathbf{V} & \cdots & u_{1m} \mathbf{V} \\ \vdots & \ddots & \vdots \\ u_{n1} \mathbf{V} & \cdots & u_{nm} \mathbf{V} \end{pmatrix} \\ (\mathbf{U} \otimes \mathbf{V})^{\top} &= \begin{pmatrix} u_{11} \mathbf{V}^{\top} & \cdots & u_{n1} \mathbf{V}^{\top} \\ \vdots & \ddots & \vdots \\ u_{1m} \mathbf{V}^{\top} & \cdots & u_{nm} \mathbf{V}^{\top} \end{pmatrix} \\ \mathbf{U}^{\top} \otimes \mathbf{V}^{\top} &= \begin{pmatrix} u_{11} \mathbf{V}^{\top} & \cdots & u_{n1} \mathbf{V}^{\top} \\ \vdots & \ddots & \vdots \\ u_{1m} \mathbf{V}^{\top} & \cdots & u_{nm} \mathbf{V}^{\top} \end{pmatrix} \end{aligned} \tag{3.16}$$

Ainsi  $(\mathbf{U} \otimes \mathbf{V})^{\top} = \mathbf{U}^{\top} \otimes \mathbf{V}^{\top}$ .

Pour montrer (ii), notons  $\mathcal{D} = \left( \mathcal{A} \times_1 \mathbf{U}^{(1)\top} \dots \times_{n-1} \mathbf{U}^{(n-1)\top} \times_{n+1} \mathbf{U}^{(n+1)\top} \times_{n+2} \dots \times_N \mathbf{U}^{(N)\top} \right)_{(n)}$ .

Je n'ai pas réussi à montrer la fin de lemme.

**Définition 20** (Modèle de TUCKER). Soient  $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ ,  $\mathcal{S} \in \mathbb{R}^{R_1 \times R_2 \times \dots \times R_N}$  deux tenseurs d'ordre  $N$  (avec  $R_n \leq I_n$ ), et  $\mathbf{U}^{(1)} \in \mathbb{R}^{I_1 \times R_1}$ , ...,  $\mathbf{U}^{(N)} \in \mathbb{R}^{I_N \times R_N}$   $N$  matrices. Le modèle de TUCKER de rang- $(R_1, \dots, R_N)$  de  $\mathcal{A}$  est :

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{U}^{(1)} \times_2 \dots \times_N \mathbf{U}^{(N)} = \llbracket \mathcal{S}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)} \rrbracket$$

$\mathcal{S}$  est appelé tenseur coeur, et les matrices  $\mathbf{U}^{(n)}$  sont appelées les matrices facteurs.

**Définition 21** (Modèle CP). Soient  $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$  un tenseur d'ordre  $N$ , et  $\mathbf{U}^{(1)} \in \mathbb{R}^{I_1 \times R_1}$ , ...,  $\mathbf{U}^{(N)} \in \mathbb{R}^{I_N \times R_N}$   $N$  matrices. Le modèle CP de  $\mathcal{A}$  est :

$$\mathcal{A} = \sum_{r=1}^R \lambda_r \mathbf{u}_r^{(1)} \circ \dots \circ \mathbf{u}_r^{(N)}$$

Les vecteurs  $\mathbf{u}_r^{(n)}$  sont les colonnes des  $\mathbf{U}^{(n)}$ . Lorsque le tenseur coeur du modèle de TUCKER est super diagonal (c'est-à-dire si  $R_1 = \dots = R_N$  et si  $\forall n \neq m, i_n \neq i_m$ , on a  $s_{i_1, \dots, i_N} = 0$ ), on retrouve le modèle CP.

**Définition 22** (Covariance croisée). Soient  $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_n \times \dots \times I_N}$  un tenseur d'ordre  $N$  et  $\mathcal{Y} \in \mathbb{R}^{J_1 \times \dots \times I_n \times \dots \times J_M}$  un tenseur d'ordre  $M$  tels que leur mode  $n$  ait la même dimension  $I_n$ . On note  $\langle \cdot, \cdot \rangle_{(n;n)}$  la multiplication des modes  $n$  de deux tenseurs. La covariance croisée du mode  $n$  entre  $\mathcal{X}$  et  $\mathcal{Y}$  est :

$$\mathcal{C} = \text{cov}_{(n;n)}(\mathcal{X}, \mathcal{Y}) = \langle \mathcal{X}, \mathcal{Y} \rangle_{(n;n)} \in \mathbb{R}^{I_1 \times \dots \times I_{n-1} \times I_{n+1} \times \dots \times I_N \times J_1 \times \dots \times J_{n-1} \times J_{n+1} \times \dots \times J_M}$$

### 3.1.2 HOSVD

Cette technique a été développée par DE LATHAUWER en 2000 [6]. Le but est de généraliser la SVD aux tenseurs.

**Théorème 2** (SVD tensorielle). Soient  $\mathcal{A}, \mathcal{S} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  deux tenseurs d'ordre  $N$ , et  $\mathbf{U}^{(1)} \in \mathbb{C}^{I_1 \times I_1}$ , ...,  $\mathbf{U}^{(N)} \in \mathbb{C}^{I_N \times I_N}$   $N$  matrices unitaires. Alors  $\mathcal{A}$  peut être décomposé comme le produit suivant :

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{U}^{(1)} \times_2 \dots \times_N \mathbf{U}^{(N)}$$

Chaque sous-tenseur  $\mathcal{S}_{i_n=\alpha}$  est obtenu en fixant le  $n^{\text{ie}}$  indice à  $\alpha$ . Le tenseur  $\mathcal{S}$  possède les propriétés suivantes :

- (i) Orthogonalité complète :  $\forall n, \forall \alpha \neq \beta$  les sous-tenseurs  $\mathcal{S}_{i_n=\alpha}$  et  $\mathcal{S}_{i_n=\beta}$  sont orthogonaux :  $\langle \mathcal{S}_{i_n=\alpha}, \mathcal{S}_{i_n=\beta} \rangle = 0$ .
- (ii)  $\forall n, \|\mathcal{S}_{i_n=1}\|_{\mathcal{F}} \geq \|\mathcal{S}_{i_n=2}\|_{\mathcal{F}} \geq \dots \geq \|\mathcal{S}_{i_n=I_n}\|_{\mathcal{F}} \geq 0$ . Les normes  $\|\mathcal{S}_{i_n=i}\|_{\mathcal{F}}$  sont les valeurs singulières du mode  $n$  de  $\mathcal{A}$  et sont notées  $\sigma_i^{(n)}$ , et le vecteur  $\mathbf{U}_i^{(n)}$  est le  $i^{\text{ie}}$  vecteur singulier du mode  $n$ .

**Démonstration 10.** Soient  $\mathcal{A}, \mathcal{S} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  deux tenseurs d'ordre  $N$ , et  $\mathbf{U}^{(1)} \in \mathbb{C}^{I_1 \times I_1}$ , ...,  $\mathbf{U}^{(N)} \in \mathbb{C}^{I_N \times I_N}$   $N$  matrices orthogonales, tels que :

$$\mathcal{S} = \mathcal{A} \times_1 \mathbf{U}^{(1)\top} \dots \times_N \mathbf{U}^{(N)\top}$$

Cette relation peut s'écrire en forme matricielle grâce au Lemme 2 :

$$\mathbf{A}_{(n)} = \mathbf{U}^{(n)} \mathbf{S}_{(n)} \left( \mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \dots \otimes \mathbf{U}^{(n-1)} \right)^\top$$

Considérons le cas où  $\mathbf{U}^{(n)}$  est obtenue à partir de la SVD de  $\mathbf{A}_{(n)}$ . Soit  $\mathbf{V}^{(n)}$  une matrice orthogonale et  $\mathbf{\Sigma}^{(n)} = \text{diag}(\sigma_1^{(n)}, \dots, \sigma_{I_n}^{(n)})$  avec  $\sigma_1^{(n)} \geq \dots \geq \sigma_{I_n}^{(n)} \geq 0$ . Alors :

$$\mathbf{A}_{(n)} = \mathbf{U}^{(n)} \mathbf{\Sigma}^{(n)} \mathbf{V}^{(n)\top}$$

Soit  $r_n$  le plus grand indice tel que  $\sigma_{r_n}^{(n)} > 0$ . Le produit de KRONECKER dans le Lemme 2 est orthogonal donc on peut identifier les deux expressions, ce qui donne :

$$\mathbf{S}_{(n)} = \mathbf{\Sigma}^{(n)} \mathbf{V}^{(n)\top} \left( \mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \dots \otimes \mathbf{U}^{(n-1)} \right)$$

Donc  $\forall \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(n-1)}, \mathbf{U}^{(n+1)}, \dots, \mathbf{U}^{(N)}$  :

$$\forall \alpha \neq \beta, \langle \mathcal{S}_{i_n=\alpha}, \mathcal{S}_{i_n=\beta} \rangle = 0$$

et

$$\|\mathcal{S}_{i_n=1}\|_{\mathcal{F}} = \sigma_1^{(n)} \geq \|\mathcal{S}_{i_n=2}\|_{\mathcal{F}} = \sigma_2^{(n)} \geq \dots \geq \|\mathcal{S}_{i_n=I_n}\|_{\mathcal{F}} = \sigma_{I_n}^{(n)} \geq 0$$

De plus, si  $r_n < I_n$ , alors :

$$\|\mathcal{S}_{i_n=r_n+1}\|_{\mathcal{F}} = \sigma_{r_n+1}^{(n)} = \dots = \|\mathcal{S}_{i_n=I_n}\|_{\mathcal{F}} = \sigma_{I_n}^{(n)} = 0$$

$\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(n-1)}, \mathbf{U}^{(n+1)}, \dots, \mathbf{U}^{(N)}$  sont construites de la même manière. ■

**Propriété 3** (Unicité). Soit la HOSVD d'un tenseur  $\mathcal{A}$  donnée par le théorème 2. Alors les valeurs singulières du mode  $n$  sont uniques.

**Démonstration 11.** Supposons qu'il existe  $(\gamma_j^{(n)})_{1 \leq n \leq N; 1 \leq j \leq I_n}$  un autre jeu de valeurs singulières de la HOSVD de  $\mathcal{A}$ . Soient  $n \in \{1, \dots, N\}$ ,  $i, j \in \{1, \dots, I_n\}$ . Par définition de la HOSVD de  $\mathcal{A}$ , on a :

$$\begin{aligned} \|\mathcal{S}_{i_n=i}\|_{\mathcal{F}} &= \sigma_i^{(n)} \\ \|\mathcal{S}_{i_n=j}\|_{\mathcal{F}} &= \gamma_j^{(n)} \end{aligned} \tag{3.17}$$

$$\begin{aligned} \sigma_1^{(n)} &\geq \dots \geq \sigma_{I_n}^{(n)} \geq 0 \\ \gamma_1^{(n)} &\geq \dots \geq \gamma_{I_n}^{(n)} \geq 0 \end{aligned} \tag{3.18}$$

Supposons maintenant que  $\exists k \in \{1, \dots, I_n\}$  tel que  $\sigma_k^{(n)} \neq \gamma_k^{(n)}$ . Alors au moins deux valeurs singulières sont permutées, de telle sorte que  $\sigma_k^{(n)} = \gamma_{k+1}^{(n)}$  et  $\sigma_{k+1}^{(n)} = \gamma_k^{(n)}$  par exemple. Mais dans ce cas l'ordre des valeurs singulières est modifié, ce qui est impossible par construction de la HOSVD. Donc on a :

$$\forall n \in \{1, \dots, N\}, \forall i \in \{1, \dots, I_n\}, \sigma_i^{(n)} = \gamma_i^{(n)}$$

Donc les valeurs singulières sont uniques. ■

**Propriété 4** (Généralisation). La HOSVD d'un tenseur d'ordre 2 revient à la SVD matricielle de ce tenseur.

**Démonstration 12.** Soient  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2}$  un tenseur d'ordre deux et  $\mathbf{A} \in \mathbb{C}^{I_1 \times I_2}$  une matrice. La SVD de  $\mathbf{A}$  s'écrit :

$$\mathbf{A} = \mathbf{S}^{\text{SVD}} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)}$$

Avec  $\mathbf{U}^{(n)} \in \mathbb{C}^{I_n \times I_n}$  des matrices unitaires et  $\mathbf{S}^{\text{SVD}} \in \mathbb{C}^{I_1 \times I_2}$  une matrice pseudo-diagonale et ordonnée :



**Propriété 6** (Norme). Soit la HOSVD d'un tenseur  $\mathcal{A}$  donnée par le théorème 2. Alors :

$$\|\mathcal{A}\|_{\mathcal{F}}^2 = \sum_{i=1}^{R_1} (\sigma_i^1)^2 = \dots = \sum_{i=1}^{R_N} (\sigma_i^N)^2 = \|\mathcal{S}\|_{\mathcal{F}}^2$$

**Démonstration 14.** Soit  $n \in \{1, \dots, N\}$ . On a :

$$\begin{aligned} \|\mathcal{A}\|_{\mathcal{F}}^2 &= \|\mathbf{A}_{(n)}\|_{\mathcal{F}}^2 \\ &= \sum_{i=1}^{r_n} \|\mathcal{S}_{i_n=i}\|_{\mathcal{F}}^2 \\ &= \sum_{i=1}^{r_n} \sigma_i^{(n)2} \\ &= \|\mathcal{S}\|_{\mathcal{F}}^2 \end{aligned} \tag{3.21}$$

■

**Propriété 7** (Approximation). Soit la HOSVD d'un tenseur  $\mathcal{A}$  donnée par le théorème 2. Soit  $R_n$  le rang du mode  $n$  de  $\mathcal{A}$ . Soit  $\hat{\mathcal{A}}$  le tenseur obtenu en supprimant les plus petites valeurs singulières du mode  $n$ ,  $\sigma_{I'_n+1}^{(n)}, \dots, \sigma_{R_n}^{(n)}$  pour une valeur donnée  $I'_n$  ( $1 \leq n \leq N$ ). Alors :

$$\|\mathcal{A} - \hat{\mathcal{A}}\|_{\mathcal{F}}^2 \leq \sum_{i_1=I'_1+1}^{R_1} (\sigma_{i_1}^{(1)})^2 + \dots + \sum_{i_N=I'_N+1}^{R_N} (\sigma_{i_N}^{(N)})^2$$

**Démonstration 15.** On a :

$$\begin{aligned} \|\mathcal{A} - \hat{\mathcal{A}}\|_{\mathcal{F}}^2 &= \sum_{i_1=1}^{R_1} \sum_{i_2=1}^{R_2} \dots \sum_{i_N=1}^{R_N} s_{i_1, i_2, \dots, i_N}^2 - \sum_{i_1=1}^{I'_1} \sum_{i_2=1}^{I'_2} \dots \sum_{i_N=1}^{I'_N} s_{i_1, i_2, \dots, i_N}^2 \\ &= \sum_{i_1=I'_1+1}^{R_1} \sum_{i_2=I'_2+1}^{R_2} \dots \sum_{i_N=I'_N+1}^{R_N} s_{i_1, i_2, \dots, i_N}^2 \\ &\leq \sum_{i_1=I'_1+1}^{R_1} \sum_{i_2=1}^{R_2} \dots \sum_{i_N=1}^{R_N} s_{i_1, i_2, \dots, i_N}^2 + \sum_{i_1=1}^{R_1} \sum_{i_2=I'_2+1}^{R_2} \dots \sum_{i_N=1}^{R_N} s_{i_1, i_2, \dots, i_N}^2 + \dots + \sum_{i_1=1}^{R_1} \sum_{i_2=1}^{R_2} \dots \sum_{i_N=I'_N+1}^{R_N} s_{i_1, i_2, \dots, i_N}^2 \\ &= \sum_{i_1=I'_1+1}^{R_1} (\sigma_{i_1}^{(1)})^2 + \dots + \sum_{i_N=I'_N+1}^{R_N} (\sigma_{i_N}^{(N)})^2 \end{aligned} \tag{3.22}$$

■

**Propriété 8** (Lien entre la HOSVD et la SVD matricielle). Soit la HOSVD d'un tenseur  $\mathcal{A}$  donnée par le théorème 2. Alors :

$$\mathbf{A}_{(n)} = \mathbf{U}^{(n)} \mathbf{\Sigma}^{(n)} \mathbf{V}^{(n)H}$$

est une SVD de  $\mathbf{A}_{(n)}$ , où la matrice diagonale  $\mathbf{\Sigma}^{(n)} \in \mathbb{R}^{I_n \times I_n}$  et la matrice orthogonale par colonne  $\mathbf{V}^{(n)} \in \mathbb{C}^{I_{n+1} I_{n+2} \dots I_N I_1 \dots I_{n-1} I_n}$  sont définies par :

$$\begin{aligned} \mathbf{\Sigma}^{(n)} &= \text{diag}(\sigma_1^{(n)}, \dots, \sigma_{I_n}^{(n)}) \\ \mathbf{V}^{(n)H} &= \tilde{\mathbf{S}}_{(n)} \left( \mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \dots \otimes \mathbf{U}^{(n-1)} \right)^\top \end{aligned} \tag{3.23}$$

où  $\tilde{\mathbf{S}}_{(n)}$  est une version normalisée de  $\mathbf{S}_{(n)}$ , avec les lignes de longueur unitaire :

$$\mathbf{S}_{(n)} = \mathbf{\Sigma}^{(n)} \tilde{\mathbf{S}}_{(n)}$$

**Démonstration 16.** La HOSVD de  $\mathcal{A}$  s'écrit :

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{U}^{(1)} \times_2 \dots \times_N \mathbf{U}^{(N)}$$

En matricisant  $\mathcal{A}$  suivant le mode  $n$ , il vient :

$$\mathbf{A}_{(n)} = \mathbf{U}^{(n)} \mathbf{S}_{(n)} \left( \mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \dots \otimes \mathbf{U}^{(n-1)} \right)$$

En posant :

$$\begin{aligned} \mathbf{\Sigma}^{(n)} &= \text{diag} \left( \sigma_1^{(n)}, \dots, \sigma_{I_n}^{(n)} \right) \\ \mathbf{V}^{(n)H} &= \tilde{\mathbf{S}}_{(n)} \left( \mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \dots \otimes \mathbf{U}^{(n-1)} \right)^\top \\ \mathbf{S}_{(n)} &= \mathbf{\Sigma}^{(n)} \tilde{\mathbf{S}}_{(n)} \end{aligned} \quad (3.24)$$

Ces quantités vérifient les propriétés requises pour être une SVD de  $\mathbf{A}_{(n)}$  car les matrices  $\mathbf{U}^{(n)}$  et  $\mathbf{V}^{(n)H}$  sont unitaires et la matrice  $\mathbf{\Sigma}^{(n)}$  est diagonale (donc pseudo diagonale) et ordonnée (par définition de la HOSVD). ■

### 3.1.3 HOPLS

Cette technique a été développée par ZHAO en 2013 [23]. Elle généralise la PLS aux tenseurs d'ordre  $N$ . L'objectif de la HOPLS est de chercher (comme pour la PLS) le sous-espace d'approximation optimal de  $\mathcal{X}$  et  $\mathcal{Y}$  dans lequel les vecteurs latents de  $\mathcal{X}$  et  $\mathcal{Y}$  ont le maximum de covariance colonne par colonne.

**Définition 23.** Soient  $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$  un tenseur d'ordre  $N$  et  $\mathcal{Y} \in \mathbb{R}^{J_1 \times \dots \times J_M}$  un tenseur d'ordre  $M$ , avec  $I_1 = J_1$ . Le modèle de la HOPLS est basé sur la décomposition  $\mathcal{X}$  comme une somme de blocs de TUCKER de rang-(1,  $L_2, \dots, L_N$ ) et  $\mathcal{Y}$  en blocs de rang-(1,  $K_2, \dots, K_M$ ), c'est-à-dire :

$$\begin{aligned} \mathcal{X} &= \sum_{r=1}^R \mathcal{G}_r \times_1 \mathbf{t}_r \times_2 \mathbf{P}_r^{(1)} \times_3 \dots \times_N \mathbf{P}_r^{(N-1)} + \mathcal{E}_R \\ \mathcal{Y} &= \sum_{r=1}^R \mathcal{D}_r \times_1 \mathbf{t}_r \times_2 \mathbf{Q}_r^{(1)} \times_3 \dots \times_M \mathbf{Q}_r^{(M-1)} + \mathcal{F}_R \end{aligned} \quad (3.25)$$

Avec :

- $R$  le nombre de vecteurs latents
- $\mathbf{t}_r \in \mathbb{R}^{I_1}$  le  $r^{\text{ie}}$  vecteur latent
- $\mathbf{P}_r^{(n)} \in \mathbb{R}^{I_{n+1} \times L_{n+1}}$  et  $\mathbf{Q}_r^{(m)} \in \mathbb{R}^{J_{m+1} \times K_{m+1}}$  les matrices loadings
- $\mathcal{G}_r \in \mathbb{R}^{1 \times L_2 \times \dots \times L_N}$  et  $\mathcal{D}_r \in \mathbb{R}^{1 \times K_2 \times \dots \times K_M}$  les tenseurs coeur.

Pour assurer l'unicité des quantités précédentes, on impose les conditions suivantes :

- $\mathcal{G}_r$  et  $\mathcal{D}_r$  sont orthogonales complètes
- $\forall r, \mathbf{P}_r^{(n)\top} \mathbf{P}_r^{(n)} = \mathbf{I}$
- $\forall r, \mathbf{Q}_r^{(m)\top} \mathbf{Q}_r^{(m)} = \mathbf{I}$
- $\forall r, \|\mathbf{t}_r\|_F = 1$ .

**Remarque 3.** En posant :

$$\begin{aligned}
\mathbf{T} &= [\mathbf{t}_1, \dots, \mathbf{t}_R] \in \mathbb{R}^{I_1 \times R} \\
\mathbf{P}^{(n)} &= \left[ \mathbf{P}_1^{(n)}, \dots, \mathbf{P}_R^{(n)} \right] \in \mathbb{R}^{I_{n+1} \times RL_{n+1}} \\
\mathbf{Q}^{(m)} &= \left[ \mathbf{Q}_1^{(m)}, \dots, \mathbf{Q}_R^{(m)} \right] \in \mathbb{R}^{J_{m+1} \times RK_{m+1}} \\
\mathcal{G} &= \text{blockdiag}(\mathcal{G}_1, \dots, \mathcal{G}_R) \in \mathbb{R}^{R \times RL_2 \times \dots \times RL_N} \\
\mathcal{D} &= \text{blockdiag}(\mathcal{D}_1, \dots, \mathcal{D}_R) \in \mathbb{R}^{R \times RK_2 \times \dots \times RK_M}
\end{aligned} \tag{3.26}$$

Le modèle prend la forme :

$$\begin{aligned}
\mathcal{X} &= \mathcal{G} \times_1 \mathbf{T} \times_2 \mathbf{P}^{(1)} \times_3 \dots \times_N \mathbf{P}^{(N-1)} + \mathcal{E}_R \\
\mathcal{Y} &= \mathcal{D} \times_1 \mathbf{T} \times_2 \mathbf{Q}^{(1)} \times_3 \dots \times_M \mathbf{Q}^{(M-1)} + \mathcal{F}_R
\end{aligned} \tag{3.27}$$

On doit résoudre deux problèmes d'optimisation simultanément :

- (i) Approximer  $\mathcal{X}$  et  $\mathcal{Y}$  par un modèle de TUCKER orthogonal
- (ii) Avoir le même vecteur latent pour un mode spécifique

On peut se ramener pour cela à la détermination de l'ensemble des loadings orthogonaux  $\mathbf{P}_r^{(n)}$  et  $\mathbf{Q}_r^{(m)}$  et des vecteurs latents  $\mathbf{t}_r$ ,  $r = 1, \dots, R$ . Pour simplifier l'écriture, considérons la recherche du premier  $\mathbf{t}$  et des deux premières matrices  $\mathbf{P}^{(n)}$  et  $\mathbf{Q}^{(m)}$ .

**Proposition 6.** Soient  $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$  un tenseur d'ordre  $N$ ,  $\mathbf{P}^{(n)} \in \mathbb{R}^{I_{n+1} \times L_{n+1}}$  une matrice orthogonale, et  $\mathbf{t} \in \mathbb{R}^{I_1}$  avec  $\|\mathbf{t}\|_F = 1$ . La solution au problème

$$\min_{\mathcal{G}} \left\| \mathcal{X} - \mathcal{G} \times_1 \mathbf{t} \times_2 \mathbf{P}^{(1)} \times_3 \dots \times_N \mathbf{P}^{(N-1)} \right\|_{\mathcal{F}}^2$$

est le tenseur suivant :

$$\mathcal{G} = \mathcal{X} \times_1 \mathbf{t}^\top \times_2 \mathbf{P}^{(1)\top} \times_3 \dots \times_N \mathbf{P}^{(N-1)\top}$$

**Démonstration 17.** Considérons :

$$\mathbf{X}_{(1)} = \mathbf{t} \mathbf{G}_{(1)} \left( \mathbf{P}^{(N-1)} \otimes \dots \otimes \mathbf{P}^{(1)} \right)^\top$$

Comme  $\mathbf{t}$  et le produit de KRONECKER  $\mathbf{P}^{(N-1)} \otimes \dots \otimes \mathbf{P}^{(1)}$  est orthogonal, on peut inverser la relation et on trouve :

$$\mathbf{G}_{(1)} = \mathbf{t}^\top \mathbf{X}_{(1)} \left( \mathbf{P}^{(N-1)} \otimes \dots \otimes \mathbf{P}^{(1)} \right)$$

En prenant la forme tensorielle on obtient le résultat voulu. ■

**Proposition 7.** Soit  $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$  un tenseur d'ordre  $N$ . Les propositions suivantes sont équivalentes :

- (i)  $\min_{(\mathbf{P}^{(n)}, \mathbf{t})} \left\| \mathcal{X} - \mathcal{G} \times_1 \mathbf{t} \times_2 \mathbf{P}^{(1)} \times_3 \dots \times_N \mathbf{P}^{(N-1)} \right\|_{\mathcal{F}}^2$  sous contraintes que  $\mathbf{P}^{(n)}$  sont orthogonales par colonne et que  $\mathbf{t}$  est normé.
- (ii)  $\max_{(\mathbf{P}^{(n)}, \mathbf{t})} \left\| \mathcal{X} \times_1 \mathbf{t}^\top \times_2 \mathbf{P}^{(1)\top} \times_3 \dots \times_N \mathbf{P}^{(N-1)\top} \right\|_{\mathcal{F}}^2$  sous contraintes que  $\mathbf{P}^{(n)}$  sont orthogonales par colonne et que  $\mathbf{t}$  est normé.

**Démonstration 18.** On souhaite résoudre :

$$\min_{(\mathbf{P}^{(n)}, \mathbf{t})} \left\| \mathcal{X} - \mathcal{G} \times_1 \mathbf{t} \times_2 \mathbf{P}^{(1)} \times_3 \dots \times_N \mathbf{P}^{(N-1)} \right\|_{\mathcal{F}}^2$$

Sous contraintes que  $\mathbf{P}^{(n)}$  sont orthogonales par colonne et que  $\mathbf{t}$  est normé. Supposons que l'on soit à l'optimalité, avec  $\mathbf{t}$ ,  $\mathbf{P}^{(1)}$ , ...,  $\mathbf{P}^{(N-1)}$  fixés, alors le tenseur coeur s'écrit :

$$\mathcal{G} = \mathcal{X} \times_1 \mathbf{t}^\top \times_2 \mathbf{P}^{(1)\top} \times_3 \dots \times_N \mathbf{P}^{(N-1)\top}$$

En effet, on vectorise l'expression dans la minimisation :

$$\left\| \mathcal{X} - \mathcal{G} \times_1 \mathbf{t} \times_2 \mathbf{P}^{(1)} \times_3 \dots \times_N \mathbf{P}^{(N-1)} \right\|_{\mathcal{F}}^2 = \left\| \text{vec}(\mathcal{X}) - \left( \mathbf{P}^{(N-1)} \otimes \dots \otimes \mathbf{P}^{(1)} \otimes \mathbf{t} \right) \text{vec}(\mathcal{G}) \right\|_{\mathcal{F}}^2$$

La solution à ce problème est donnée par les moindres carrés. De plus, le produit de KRONECKER étant orthogonal, on a :

$$\begin{aligned} \text{vec}(\mathcal{G}) &= \left( \mathbf{P}^{(N-1)} \otimes \dots \otimes \mathbf{P}^{(1)} \otimes \mathbf{t} \right)^\top \text{vec}(\mathcal{X}) \\ &= \left( \mathbf{P}^{(N-1)\top} \otimes \dots \otimes \mathbf{P}^{(1)\top} \otimes \mathbf{t}^\top \right) \text{vec}(\mathcal{X}) \end{aligned} \quad (3.28)$$

Par la proposition 3, il vient :

$$\left\| \mathcal{X} - \mathcal{G} \times_1 \mathbf{t} \times_2 \mathbf{P}^{(1)} \times_3 \dots \times_N \mathbf{P}^{(N-1)} \right\|_{\mathcal{F}}^2 = \|\mathcal{X}\|_{\mathcal{F}}^2 - 2 \left\langle \mathcal{X}, \mathcal{G} \times_1 \mathbf{t} \times_2 \mathbf{P}^{(1)} \times_3 \dots \times_N \mathbf{P}^{(N-1)} \right\rangle + \|\mathcal{G}\|_{\mathcal{F}}^2$$

Or par la proposition 4 :

$$\left\langle \mathcal{X}, \mathcal{G} \times_1 \mathbf{t} \times_2 \mathbf{P}^{(1)} \times_3 \dots \times_N \mathbf{P}^{(N-1)} \right\rangle = \left\langle \mathcal{X} \times_1 \mathbf{t}^\top \times_2 \mathbf{P}^{(1)\top} \times_3 \dots \times_N \mathbf{P}^{(N-1)\top}, \mathcal{G} \right\rangle = \|\mathcal{G}\|_{\mathcal{F}}^2$$

Et par la proposition 5 :

$$\left\| \mathcal{G} \times_1 \mathbf{t} \times_2 \mathbf{P}^{(1)} \times_3 \dots \times_N \mathbf{P}^{(N-1)} \right\|_{\mathcal{F}}^2 = \|\mathcal{G}\|_{\mathcal{F}}^2$$

Donc :

$$\left\| \mathcal{X} - \mathcal{G} \times_1 \mathbf{t} \times_2 \mathbf{P}^{(1)} \times_3 \dots \times_N \mathbf{P}^{(N-1)} \right\|_{\mathcal{F}}^2 = \|\mathcal{X}\|_{\mathcal{F}}^2 - \|\mathcal{G}\|_{\mathcal{F}}^2$$

Donc la minimisation initiale revient à maximiser  $\|\mathcal{G}\|_{\mathcal{F}}^2$ . Ainsi le problème se réécrit :

$$\max_{(\mathbf{P}^{(n)}, \mathbf{t})} \left\| \mathcal{X} \times_1 \mathbf{t}^\top \times_2 \mathbf{P}^{(1)\top} \times_3 \dots \times_N \mathbf{P}^{(N-1)\top} \right\|_{\mathcal{F}}^2$$

Sous contraintes que  $\mathbf{P}^{(n)}$  sont orthogonales par colonne et que  $\mathbf{t}$  est normé. ■

Si on dispose de  $\mathbf{P}^{(n)}$ ,  $\mathbf{Q}^{(m)}$  et  $\mathbf{t}$ , alors on peut calculer les tenseurs coeur :

$$\begin{aligned} \mathcal{G} &= \mathcal{X} \times_1 \mathbf{t}^\top \times_2 \mathbf{P}^{(1)\top} \times_3 \dots \times_N \mathbf{P}^{(N-1)\top} \\ \mathcal{D} &= \mathcal{Y} \times_1 \mathbf{t}^\top \times_2 \mathbf{Q}^{(1)\top} \times_3 \dots \times_M \mathbf{Q}^{(M-1)\top} \end{aligned} \quad (3.29)$$

D'après la Proposition 7, minimiser les normes  $\|\mathcal{E}_R\|_{\mathcal{F}}$  et  $\|\mathcal{F}_R\|_{\mathcal{F}}$  revient à maximiser les normes  $\|\mathcal{G}\|_{\mathcal{F}}$  et  $\|\mathcal{D}\|_{\mathcal{F}}$ . On ne va pas maximiser les deux normes en même temps mais plutôt leur produit au carré  $\|\mathcal{G}\|_{\mathcal{F}}^2 \cdot \|\mathcal{D}\|_{\mathcal{F}}^2$ .

**Proposition 8.** Soient  $\mathcal{G} \in \mathbb{R}^{1 \times L_2 \times \dots \times L_N}$  un tenseur d'ordre N et  $\mathcal{D} \in \mathbb{R}^{1 \times K_2 \times \dots \times K_M}$  un tenseur d'ordre M. Alors :

$$\left\| \langle \mathcal{G}, \mathcal{D} \rangle_{(1;1)} \right\|_{\mathcal{F}}^2 = \|\mathcal{G}\|_{\mathcal{F}}^2 \cdot \|\mathcal{D}\|_{\mathcal{F}}^2$$

**Démonstration 19.**

$$\begin{aligned}
\left\| \langle \mathcal{G}, \mathcal{D} \rangle_{(1;1)} \right\|_{\mathcal{F}}^2 &= \left\| \text{vec}(\mathcal{G}) \text{vec}^\top(\mathcal{D}) \right\|_{\mathcal{F}}^2 \\
&= \text{tr} \left( \text{vec}(\mathcal{D}) \text{vec}^\top(\mathcal{G}) \text{vec}(\mathcal{G}) \text{vec}^\top(\mathcal{D})^\top \right) \\
&= \left\| \text{vec}(\mathcal{G}) \right\|_{\mathcal{F}}^2 \cdot \left\| \text{vec}(\mathcal{D}) \right\|_{\mathcal{F}}^2 \\
&= \|\mathcal{G}\|_{\mathcal{F}}^2 \cdot \|\mathcal{D}\|_{\mathcal{F}}^2
\end{aligned} \tag{3.30}$$

Avec  $\text{vec}()$  l'opération de vectorisation d'un tenseur. ■

Or on a :

$$\begin{aligned}
\left\| \langle \mathcal{G}, \mathcal{D} \rangle_{(1;1)} \right\|_{\mathcal{F}}^2 &= \left\| \left[ \langle \mathcal{X}, \mathcal{Y} \rangle_{(1;1)} ; \mathbf{P}^{(1)\top}, \dots, \mathbf{P}^{(N-1)\top}, \mathbf{Q}^{(1)\top}, \dots, \mathbf{Q}^{(M-1)\top} \right] \right\|_{\mathcal{F}}^2 \\
&= \left\| \left[ \text{cov}_{(1;1)}(\mathcal{X}, \mathcal{Y}); \mathbf{P}^{(1)\top}, \dots, \mathbf{P}^{(N-1)\top}, \mathbf{Q}^{(1)\top}, \dots, \mathbf{Q}^{(M-1)\top} \right] \right\|_{\mathcal{F}}^2 \\
&= \left\| \left[ \mathcal{C}; \mathbf{P}^{(1)\top}, \dots, \mathbf{P}^{(N-1)\top}, \mathbf{Q}^{(1)\top}, \dots, \mathbf{Q}^{(M-1)\top} \right] \right\|_{\mathcal{F}}^2
\end{aligned} \tag{3.31}$$

Ainsi le problème de maximisation des normes devient :

$$\max_{(\mathbf{P}^{(n)}, \mathbf{Q}^{(m)})} \left\| \left[ \mathcal{C}; \mathbf{P}^{(1)\top}, \dots, \mathbf{P}^{(N-1)\top}, \mathbf{Q}^{(1)\top}, \dots, \mathbf{Q}^{(M-1)\top} \right] \right\|_{\mathcal{F}}^2$$

Sous contraintes :

$$\mathbf{P}^{(n)\top} \mathbf{P}^{(n)} = \mathbf{I}_{L_{n+1}}, \quad \mathbf{Q}^{(m)\top} \mathbf{Q}^{(m)} = \mathbf{I}_{K_{m+1}}$$

La proposition 7 ainsi que l'orthogonalité des  $\mathbf{P}^{(n)}$  et  $\mathbf{Q}^{(m)}$  permettent de modifier le problème précédent en la recherche du meilleur sous-espace d'approximation de  $\mathcal{C}$ . On cherche donc la HOSVD de rang  $(L_2, \dots, L_N, K_2, \dots, K_M)$  de  $\mathcal{C}$  :

$$\mathcal{C} \approx \left[ \mathcal{G}^{(C)}; \mathbf{P}^{(1)}, \dots, \mathbf{P}^{(N-1)}, \mathbf{Q}^{(1)}, \dots, \mathbf{Q}^{(M-1)} \right]$$

Grâce à la proposition 6, il suffit d'optimiser  $\left\| \mathcal{G}^{(C)} \right\|_{\mathcal{F}}$ , en utilisant l'algorithme HOOI (Higher-order orthogonal iteration). Cet algorithme cherche les  $\mathbf{P}^{(n)}$  et  $\mathbf{Q}^{(m)}$  par une décomposition orthogonale de TUCKER de  $\mathcal{C}$ . Finalement, il ne reste plus qu'à chercher  $\mathbf{t}$  par :

$$\mathbf{t} = \underset{\mathbf{t}}{\text{argmin}} \left\| \mathcal{X} - \left[ \mathcal{G}; \mathbf{t}, \mathbf{P}^{(1)}, \dots, \mathbf{P}^{(N-1)} \right] \right\|_{\mathcal{F}}^2$$

La solution à ce problème est le premier vecteur singulier à gauche de la matrice  $(\mathcal{X} \times_2 \mathbf{P}^{(1)} \times_3 \dots \times_N \mathbf{P}^{(N-1)})_{(1)}$ .

On peut ainsi écrire l'algorithme de HOPLS :

---

**Algorithme 2** : HOPLS entre deux tenseurs  $\mathcal{X}$  et  $\mathcal{Y}$ 


---

**Entrées** :  $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ ,  $\mathcal{Y} \in \mathbb{R}^{J_1 \times \dots \times J_M}$ ,  $N \geq 3$ ,  $M \geq 3$  et  $I_1 = J_1$ .

$R$  le nombre de vecteurs latents et  $\{L_n\}_{n=2}^N$ ,  $\{K_m\}_{m=2}^M$  le nombre de loadings.

**Output** :  $\{\mathbf{P}_r^{(n)}\}$ ;  $\{\mathbf{Q}_r^{(m)}\}$ ;  $\{\mathcal{G}_r\}$ ;  $\{\mathcal{D}_r\}$ ;  $\mathbf{T}$

$r = 1, \dots, R$ ;  $n = 1, \dots, N - 1$ ;  $m = 1, \dots, M - 1$

**Initialisation** :  $\mathcal{E}_1 \leftarrow \mathcal{X}$ ;  $\mathcal{F}_1 \leftarrow \mathcal{Y}$

**pour**  $r = 1$  à  $R$  **faire**

**si**  $\|\mathcal{E}_r\|_{\mathcal{F}} > \epsilon$  **et**  $\|\mathcal{F}_r\|_{\mathcal{E}} > \epsilon$  **alors**

$\mathcal{C}_r \leftarrow \langle \mathcal{E}_r, \mathcal{F}_r \rangle_{(1;1)}$  ;

    Décomposition orthogonale de TUCKER de rang- $(L_2, \dots, L_N, K_2, \dots, K_M)$  de  $\mathcal{C}_r$  :

$\mathcal{C} \approx \llbracket \mathcal{G}^{(C)}; \mathbf{P}^{(1)}, \dots, \mathbf{P}^{(N-1)}, \mathbf{Q}^{(1)}, \dots, \mathbf{Q}^{(M-1)} \rrbracket$  ;

$\mathbf{t}_r \leftarrow$  le premier vecteur singulier à gauche de

    SVD  $\left( \left( \mathcal{E}_r \times_2 \mathbf{P}^{(1)\top} \times_3 \dots \times_N \mathbf{P}^{(N-1)\top} \right)_{(1)} \right)$  ;

$\mathcal{G}_r \leftarrow \llbracket \mathcal{E}_r; \mathbf{t}_r^\top, \mathbf{P}^{(1)\top}, \dots, \mathbf{P}^{(N-1)\top} \rrbracket$  ;

$\mathcal{D}_r \leftarrow \llbracket \mathcal{F}_r; \mathbf{t}_r^\top, \mathbf{Q}^{(1)\top}, \dots, \mathbf{Q}^{(M-1)\top} \rrbracket$  ;

$\mathcal{E}_{r+1} \leftarrow \mathcal{E}_r - \llbracket \mathcal{G}_r; \mathbf{t}_r, \mathbf{P}^{(1)}, \dots, \mathbf{P}^{(N-1)} \rrbracket$  ;

$\mathcal{F}_{r+1} \leftarrow \mathcal{F}_r - \llbracket \mathcal{D}_r; \mathbf{t}_r, \mathbf{Q}^{(1)}, \dots, \mathbf{Q}^{(M-1)} \rrbracket$  ;

**sinon**

    | Break;

**fin**

**fin**

---

Une fois que le modèle a été entraîné, on peut faire une prédiction à partir d'un nouveau tenseur  $\mathcal{X}^{new}$ . Si on note :

$$\begin{aligned} \mathbf{W} &= [\mathbf{w}_1, \dots, \mathbf{w}_R], \quad \mathbf{w}_r = \left( \mathbf{P}_r^{(N-1)} \otimes \dots \otimes \mathbf{P}_r^{(1)} \right) \mathcal{G}_{r(1)}^+ \\ \mathbf{Q}^* &= [\mathbf{q}_1^*, \dots, \mathbf{q}_R^*], \quad \mathbf{q}_r^* = \mathcal{D}_{r(1)} \left( \mathbf{Q}_r^{(M-1)} \otimes \dots \otimes \mathbf{Q}_r^{(1)} \right)^\top \end{aligned} \quad (3.32)$$

La prédiction sous forme matricielle s'écrit :

$$\widehat{\mathbf{Y}}_{(1)}^{new} = \mathbf{X}_{(1)}^{new} \mathbf{W} \mathbf{Q}^{*\top}$$

Cet algorithme possède plusieurs propriétés :

- Robuste au bruit : la contrainte sur le fait de garder les plus grands  $\{L_n\}_{n=2}^N$  vecteurs loadings résulte en un modèle plus flexible qui permet un compromis entre un bon entraînement et le sens des variables latentes.
- Problème "grand  $p$ , petit  $n$ " : cet algorithme permet de résoudre ce problème car les paramètres du modèle à optimiser sont d'une valeur faible.
- Facile à interpréter : les vecteurs loadings dans  $\mathbf{P}^{(n)}$  révèlent de nouveaux patterns de sous-espace correspondant aux attributs du mode  $n$ .
- Calcul : deux alternatives : soit  $\forall n, m, L_n = K_m = \lambda$ , soit pour  $0 < \eta \leq 1$ ,  $L_n = \eta R_n$  et  $K_m = \eta R_m$

## 3.2 Application aux HRTF

Cette partie s'inspire des travaux de HUANG en 2014 [14] pour l'application des méthodes HOSVD et HOPLS à l'individualisation des HRTF.

### 3.2.1 Traitement des données

#### Réduction de dimension

L'ensemble des magnitudes en dB des HRTF forme un tenseur d'ordre 3, noté  $\mathcal{H} \in \mathbb{R}^{P \times D \times F}$ , avec  $P$  le nombre de sujets,  $D$  le nombre de directions, et  $F$  le nombre de fréquences. Une HOSVD sur de tenseur est réalisée, uniquement pour les modes 2 et 3 :

$$\mathcal{H} = \mathcal{W} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}$$

Dans les matrices  $\mathbf{U}^{(2)} \in \mathbb{R}^{D \times D}$  et  $\mathbf{U}^{(3)} \in \mathbb{R}^{F \times F}$ , uniquement les  $D'$  et  $F'$  premières composantes sont conservées afin de réduire la dimension des tenseurs. Pour choisir ces nombres, il faut étudier les valeurs propres  $\lambda^{(2)}$  et  $\lambda^{(3)}$  des matrices  $\mathbf{U}^{(2)}$  et  $\mathbf{U}^{(3)}$ . Soient  $E^{(2)}$  le ratio entre la somme des  $D'$  premières valeurs propres et la somme des  $D$  valeurs propres de  $\mathbf{U}^{(2)}$  et  $E^{(3)}$  le ratio entre la somme des  $F'$  premières valeurs propres et la somme des  $F$  valeurs propres de  $\mathbf{U}^{(3)}$ , calculés avec :

$$\begin{aligned} E^{(2)} &= \frac{\sum_{d=1}^{D'} \lambda_d^{(2)}}{\sum_{d=1}^D \lambda_d^{(2)}} \\ E^{(3)} &= \frac{\sum_{f=1}^{F'} \lambda_f^{(3)}}{\sum_{f=1}^F \lambda_f^{(3)}} \end{aligned} \quad (3.33)$$

$D'$  et  $F'$  sont calculés tels que  $E^{(2)} \geq 0,98$  et  $E^{(3)} \geq 0,98$ . Soient  $\tilde{\mathbf{U}}^{(2)}$  la matrice  $\mathbf{U}^{(2)}$  à laquelle on a extrait les  $D'$  premières colonnes, et  $\tilde{\mathbf{U}}^{(3)}$  la matrice  $\mathbf{U}^{(3)}$  à laquelle on a extrait les  $F'$  premières colonnes. Le tenseur coeur individualisé est calculé par :

$$\tilde{\mathcal{W}} = \mathcal{H} \times_2 \tilde{\mathbf{U}}^{(2)} \times_3 \tilde{\mathbf{U}}^{(3)} \in \mathbb{R}^{P \times D' \times F'}$$

Ce tenseur va constituer les variables expliquées du modèle de régression.

#### Sélection des attributs

La phase de sélection des paramètres morphologiques les plus importants se fait en 3 étapes.

Tout d'abord, une analyse de corrélation entre les paramètres morphologiques et le tenseur coeur individualisé matricisé suivant le mode 1 est réalisée. Pour cela la matrice de corrélation de PEARSON est calculée et les paramètres qui ont les plus grandes corrélations avec le tenseur coeur sont gardés.

Ces paramètres vont servir de base au calcul des score de Laplace [13]. Ces scores sont basés sur l'hypothèse que deux paramètres sont probablement liés à un même topic s'ils sont proches l'un de l'autre. Supposons qu'il y ait  $P$  sujets et  $K$  paramètres. Soit  $a_{pk}$  le  $k^{\text{ie}}$  paramètre morphologique de du sujet  $p$ . Les paramètres morphologiques sont regroupés dans une matrice  $\mathbf{A}_{all} = [\mathbf{a}_1, \dots, \mathbf{a}_P]^\top \in \mathbb{R}^{P \times K}$ . Chaque vecteur  $\mathbf{a}_p$  est considéré comme un point et correspond au noeud  $p$  d'un graphe des plus proches voisins. Si  $\mathbf{a}_p$  est proche de  $\mathbf{a}_{p'}$ , un lien est inséré et un poids est attribué :

$$S_{pp'} = \frac{\|\mathbf{a}_p - \mathbf{a}_{p'}\|^2}{t}$$

avec  $t$  un paramètre constant. Sinon un poids nul est attribué. Ce graphe pondéré représente la structure locale de l'espace des paramètres. Pour un paramètre  $k$ , son score de LAPLACE est :

$$L_k = \frac{\sum_{p,p'} (a_{pk} - a_{p'k})^2 S_{pp'}}{\text{var}(\mathbf{b}_k)}$$

avec  $\mathbf{b}_k = [a_{1k}, \dots, a_{pk}]^\top$ . Au final, les paramètres retenus sont ceux pour lesquels le score de LAPLACE est plus petit que 0,4.

La dernière étape consiste en une nouvelle analyse de corrélation sur les variables précédentes pour supprimer celles qui auraient encore une trop grande corrélation avec les autres.

Cette phase de sélection d'attributs permet de construire la matrice  $\mathbf{A} \in \mathbb{R}^{P \times K'}$ , qui contient les  $K'$  paramètres sélectionnés.

### 3.2.2 Modélisation par HOPLS

Le modèle de HOPLS considéré est :

$$\begin{aligned} \widetilde{\mathcal{W}} &= \mathcal{Y} \times_1 \mathbf{T} \times_2 \overline{\mathbf{Q}}^{(2)} \times_3 \overline{\mathbf{Q}}^{(3)} + \mathcal{F} \\ \mathbf{A} &= \mathbf{D} \times_1 \mathbf{T} \times_2 \mathbf{V} + \mathbf{E} \end{aligned} \quad (3.34)$$

Avec :

- $\mathbf{T} = [\mathbf{t}_1, \dots, \mathbf{t}_R] \in \mathbb{R}^{P \times R}$  la matrice des  $R$  vecteurs latents
- $\mathcal{Y} \in \mathbb{R}^{R \times R J_2 \times R J_3}$  le tenseur bloc diagonal contenant les tenseurs  $\mathcal{Y}_r$  de rang-(1,  $J_2$ ,  $J_3$ ) sur la diagonale
- $\mathbf{D} \in \mathbb{R}^{R \times R I}$  la matrice bloc diagonale contenant les  $\mathbf{d}_r$  sur la diagonale
- $\overline{\mathbf{Q}}^{(2)} \in \mathbb{R}^{D' \times R J_2}$ ,  $\overline{\mathbf{Q}}^{(3)} \in \mathbb{R}^{F' \times R J_3}$ ,  $\mathbf{V} \in \mathbb{R}^{K' \times R I}$  les matrices loadings
- $\mathcal{F}, \mathbf{E}$  les résidus

On pose  $J_2 = J_3 = I = \lambda$ . Le nombre de vecteurs latents  $R$  et de loadings  $\lambda$  sont trouvés par validation croisée.

Une fois que l'entraînement du modèle est réalisé, un nouveau jeu de HRTF connaissant les paramètres morphologiques  $\mathbf{a}^{new}$  est prédit. Le nouveau tenseur coeur individualisé est calculé avec :

$$\widetilde{\mathcal{W}}^{new} = \mathbf{a}^{new} \mathbf{V} \mathbf{D} + \mathbf{Y}_{(1)} \left( \overline{\mathbf{Q}}^{(3)} \otimes \overline{\mathbf{Q}}^{(2)} \right)^\top$$

Le nouveau jeu de HRTF est reconstruit à partir de :

$$\widehat{\mathcal{H}}^{new} = \widetilde{\mathcal{W}}^{new} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}$$

## 3.3 Résultats

La FIGURE 3.1 montre le HRTF du sujet 31 à l'élévation  $0^\circ$ , original (rouge), prédit par la PLS (bleu) et la HOPLS (vert). La prédiction du HRTF par la méthode HOPLS est encore plus lisse que la PLS, mais elle enveloppe mieux le HRTF original. Néanmoins, les indices acoustiques pertinents pour l'écoute dans le plan médian ne sont toujours pas présents.

La FIGURE 3.2 montre la distribution des corrélations pour tous les sujets et toutes les directions. La majorité des corrélations se situe autour de 0,8, ce qui est moins bon que les méthodes classiques.

Enfin, la FIGURE 3.3 montre les différentes "cartes" des HRTF, originaux et prédits, pour le sujet 31. La carte des HRTF prédits par la HOPLS est bien plus lissée que la carte des HRTF prédits par la PLS.

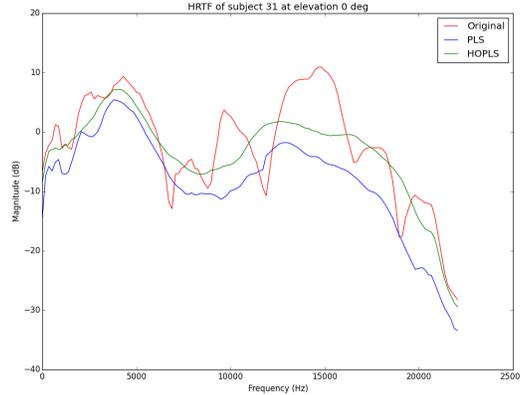


FIGURE 3.1 – Exemple de HRTF original (rouge), prédit par la PLS (bleu) et la HOPLS (vert) pour le sujet 31 à l’élévation  $0^\circ$

### 3.4 Discussion

L’analyse de la FIGURE 3.2 semble indiquer que la prédiction par HOPLS est de moins bonne qualité que par la PLS ou la Ridge. Ce résultat est en contradiction avec l’article de HUANG [14]. En effet leur conclusion révèle que la méthode HOPLS donne de meilleurs résultats que la PLS. Il faut néanmoins faire attention au fait que les données ne sont pas les mêmes dans leur étude et la nôtre. Ils n’utilisent que la base de données CIPIC, et considèrent toutes les directions de l’espace. Il se peut donc que les prédictions dans les autres plans soient de meilleure qualité que dans le plan médian, et qu’en moyenne les prédictions soient meilleures pour la HOPLS que la PLS. Or notre étude s’est uniquement focalisée sur le plan médian, révélant ainsi une faiblesse de la méthode HOPLS. Néanmoins, cela ne veut pas dire que celle-ci soit mauvaise. En effet, son application sur d’autres plans que le plan médian peut s’avérer judicieuse car la localisation en dehors du plan médian ne nécessite pas forcément les indices que nous recherchons, à savoir les résonances et anti-résonances du spectre.

Ainsi, la méthode tensorielle ne donne pas les résultats attendus, à savoir une meilleure prédiction des résonances et anti-résonances des HRTF. Cela peut être dû au réglage des paramètres de la méthode, c’est-à-dire le nombre de directions  $D'$  et de fréquences  $F'$  à garder après la HOSVD, le nombre de vecteurs latents  $R$  et le nombre de loadings  $\lambda$  pour la HOPLS, et les paramètres morphologiques clés  $K'$ . Néanmoins, ces paramètres ont été choisis par validation croisée. Il semble donc difficile de trouver un meilleur jeu de paramètres, à moins de changer le critère à optimiser.

Après examen des résultats des deux approches précédentes (méthodes classiques ou tensorielles), il ressort que le point crucial est la prédiction des résonances et des anti-résonances des filtres HRTF. En effet, toutes les prédictions sont en quelque sorte des enveloppes de HRTF, et l’information sur les pics et les creux a été perdue. Si on analyse les HRTF, on se rend compte que la première résonance se situe plus ou moins aux alentours de 4 kHz pour tous les sujets, mais les autres résonances et anti-résonances sont situées à des positions qui dépendent fortement de la personne. Une nouvelle méthode, expliquée dans le chapitre 4, consiste donc à ramener toutes les résonances et anti-résonances de tous les sujets au niveau des résonances et anti-résonances moyennes, puis à prédire séparément les enveloppes et les positions de ces pics et creux.

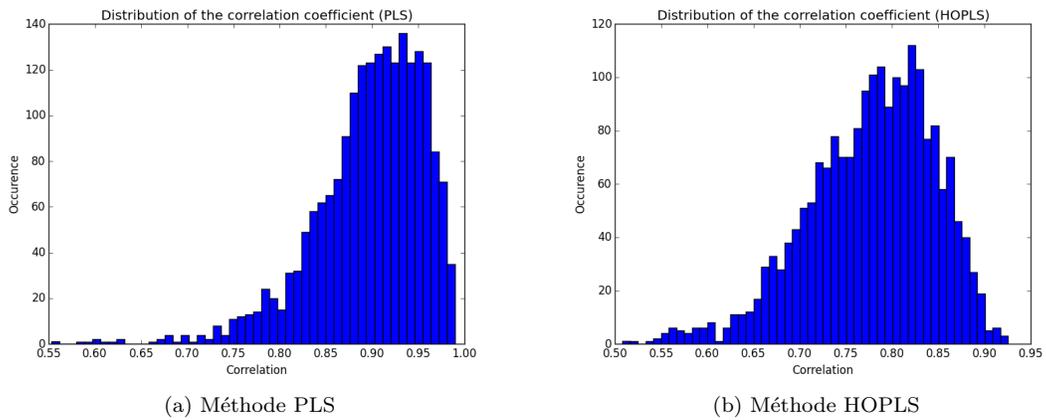


FIGURE 3.2 – Distributions des corrélations : comparaison de la méthode HOPLS avec la méthode PLS

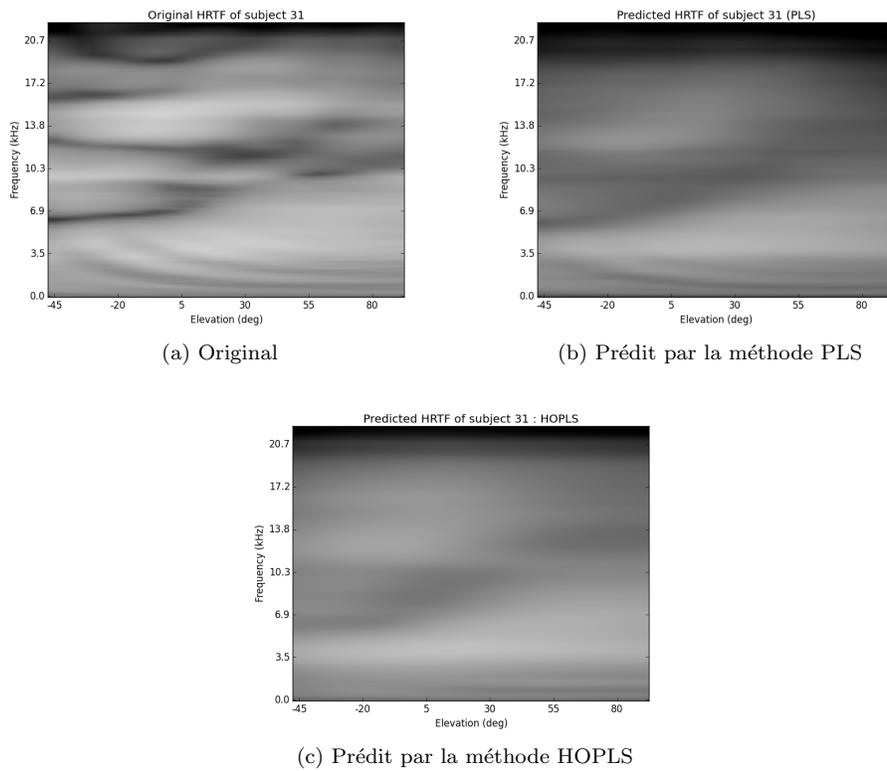


FIGURE 3.3 – Cartes des HRTF du sujet 31, original et prédits

# Chapitre 4

## Individualisation séparée

Basée sur le constat que la plus grande difficulté est de prédire la position et l'amplitude exacte des résonances et anti-résonances des filtres dans le plan médian, ce chapitre présente une méthode nouvelle dont la démarche générale est la suivante :

1. Séparation des parties résonante et anti-résonante : un algorithme permet de séparer la contribution résonante et anti-résonante de chaque filtre.
2. Extraction des lieux fréquentiels des résonances et anti-résonances : chaque partie comprend des trajectoires de résonances ou d'anti-résonances qui ont été utilisées pour extraire les points pertinents pour l'étude.
3. Morphing des parties résonante et anti-résonante : en utilisant les lieux fréquentiels des résonances et anti-résonances précédentes, les parties résonantes et anti-résonantes sont déformées de sorte que chaque résonance et anti-résonance soit déplacée vers les résonances et anti-résonances moyennes.
4. Une étape de régression sur ces parties résonante et anti-résonante est réalisée séparément pour prédire la nouvelle enveloppe. Les nouveaux lieux fréquentiels des résonances et anti-résonances sont également prédits. Un morphing inverse est appliqué pour obtenir les parties résonante et anti-résonante prédites, puis sont sommées pour obtenir le HRTF prédit.

### 4.1 Méthode

#### 4.1.1 Séparation des parties résonances et anti-résonantes

Cette partie est réalisée par un algorithme développé par SPAGNOL en 2010 [9].

**Définition 24** (PRTF). Le PRTF (*Pinna-Related Transfer Function*) est la transformée de FOURIER du fenêtrage d'un HRIR (*Head-Related Impulse Response*) avec une demi-fenêtre de Hann d'une milliseconde. C'est un modèle qui tente de supprimer l'influence de la tête et du torse.

$$PRTF = \text{fft}(\text{Hann}_{1ms}(HRIR))$$

Ce fenêtrage calcule en fait une approximation du PRFT : en effet la durée du fenêtrage n'est peut-être pas optimale. L'idée est de séparer le PRTF en une partie résonante  $H_{res}$  et une partie anti-résonante (ou réflexive)  $H_{refl}$ , de manière itérative.

**Définition 25** (Cepstre réel). Soit  $x(t)$  un signal réel dont la transformée de FOURIER est notée  $X(f)$ . Le cepstre réel de  $x(t)$ , noté  $C$ , est défini par :

$$C = \text{Re}(\text{ifft}(20 \log_{10}(|X(f)|)))$$

L'initialisation commence par mettre  $H_{res}$  au PRTF et  $H_{refl}$  à 1. Ensuite, le cepstre réel de  $H_{res}$  est calculé, qu'on liffre avec les  $N_{ceps}$  premiers coefficients du cepstre et enfin une FFT est appliquée pour obtenir une enveloppe de  $H_{res}$ , notée  $C_{res}$ . Le résidu  $E_{res}$  est la différence des amplitudes en dB de  $H_{res}$  et  $C_{res}$ .

Une recherche des minima locaux de  $E_{res}$  plus petit qu'un paramètre  $D_{min}$  est effectuée, de manière à obtenir le nombre d'anti-résonances à trouver  $N_{nch}$ . Le but est de compenser chaque anti-résonance par un filtre dont les paramètres sont la fréquence centrale  $f_C$ , la largeur de bande  $f_B$  et l'amplitude de l'anti-résonance  $D$ . Pour chaque minimum, la fréquence centrale correspond à la position de ce minimum, et la profondeur vaut  $D = |E_{res}(f_C)|$ . La largeur de bande est plus compliquée à trouver. Dans le cas général, il s'agit de la largeur de bande à -3dB. Néanmoins le calcul est différent dans les trois cas suivants :

1. Si  $D < 3\text{dB}$ , les fréquences à gauche  $f_l$  et à droite  $f_r$  de  $f_C$  à un niveau intermédiaire entre 0 et  $-D$  sont utilisées (typiquement le milieu)
2. Si le maximum local de  $E_{res}$  précédant (ou suivant) immédiatement  $f_C$  n'est pas plus grand que 0dB alors que le maximum local suivant (ou précédant) immédiatement  $f_C$  l'est, alors  $f_B$  est calculé comme le double de la bande entre  $f_C$  et  $f_r$  (ou  $f_l$ ).
3. Si les deux maxima ne sont pas plus grand que 0dB,  $E_{res}$  est déplacé verticalement jusqu'à ce que l'un des maxima soit égal à 0dB.  $f_B$  est calculé comme précédemment, sauf si la nouvelle anti-résonance est plus petite que  $D_{min}$  : dans ce cas la procédure pour cette anti-résonance est arrêtée et  $N_{nch}$  est diminué de 1.

Il arrive que le cas 1 puisse arriver en même temps que le 2 et le 3, dans ce cas tous les effets précédents sont pris en compte dans le calcul de  $f_B$ .

A partir de ces trois paramètres, le filtre est construit de la manière suivante :

$$H_{nch}(z) = \frac{1 + (1+k)\frac{H_0}{2} + l(1-k)z^{-1} + (-k - (1+k)\frac{H_0}{2})z^{-2}}{1 + l(1-k)z^{-1} - kz^{-2}}$$

Avec :

$$\begin{aligned} f_B &= \frac{f_B}{\rho} \\ k &= \frac{\tan(\pi \frac{f_B}{f_s}) - V_0}{\tan(\pi \frac{f_B}{f_s}) + V_0} \\ l &= -\cos(2\pi \frac{f_C}{f_s}) \\ V_0 &= 10^{-\frac{D}{20}} \\ H_0 &= V_0 - 1 \end{aligned}$$

Le paramètre  $\rho$  sert à contrôler la largeur de bande pour une meilleure réduction de l'anti-résonance dans la partie résonante. Le filtre *multi-notch* est construit en convoluant les paramètres des filtres simples.

La mise à jour des filtres se fait avec :

$$\begin{aligned} H_{refl} &= H_{refl} \cdot H_{nch} \\ H_{res} &= H_{res} / H_{nch} \end{aligned}$$

Ce procédé est itéré jusqu'à ce que l'algorithme ne détecte plus d'anti-résonances à compenser. Une étape de calibration des différents paramètres a été nécessaire car les valeurs présentées dans l'article ne permettaient pas d'obtenir une séparation de bonne qualité. Au final, les valeurs considérées ici sont :

$$\begin{aligned}
 N_{ceps} &= 6 \\
 D_{min} &= 1 \\
 \rho &= 2 \\
 N_{fft} &= 8192
 \end{aligned}$$

Exceptionnellement, un grand nombre de coefficients pour la FFT a été utilisé de manière à avoir une courbe lissée pour la recherche des minima et maxima. La FIGURE 4.1 montre un exemple de séparation en partie résonante et anti-résonante.

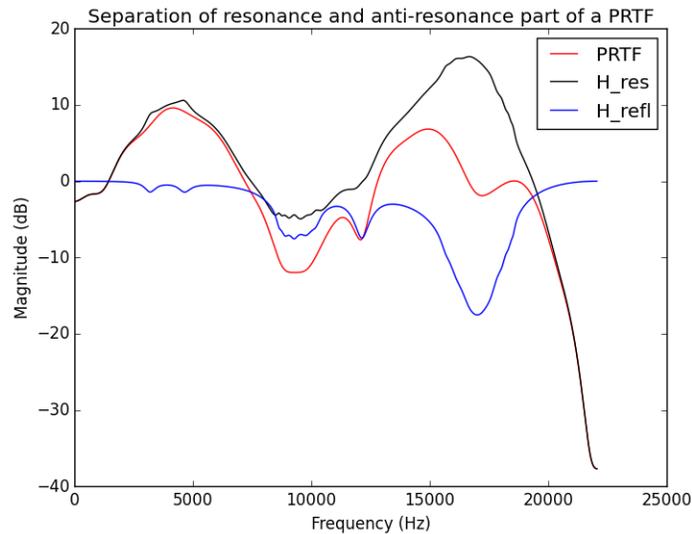


FIGURE 4.1 – Séparation d’un PRTF (blue) en sa partie résonante (noire) et anti-résonance (rouge)

#### 4.1.2 Extraction des lieux fréquentiels des résonances et anti-résonances

L’idée est de chercher des trajectoires de résonances et d’anti-résonances sur les cartes de HRTF, en l’occurrence sur les cartes des parties résonantes et anti-résonantes. RODRIGUEZ en 2005 [18] a identifié trois pattern d’anti-résonance, dont chacun présente des trajectoires d’anti-résonances, voir FIGURE 4.2a. Le pattern 1 est considéré car il représente 70% des cas. De même, on cherche à identifier trois trajectoires dans les cartes de parties résonantes, voir FIGURE 4.2b.

L’idée est de retrouver les trajectoires A et B dans chaque carte formée par les filtres d’anti-résonance. La trajectoire C est compliquée à distinguer pour certaines images, et comme elle se situe dans les hautes fréquences elle n’a pas beaucoup d’intérêt. Pour trouver ces trajectoires, chaque filtre est modélisé avec un modèle de STEIGLITZ-MCBRIDE (STMCB) à six pôles et six zéros. Les pôles sont ensuite récupérés pour la partie résonante et les minima pour la partie anti-résonante. Le résultat est visible sur la FIGURE 4.3.

On remarque que certaines trajectoires sont bien visibles, mais que des points parasites sont également présents. Une seconde passe manuelle a donc été effectuée. Pour cela, un programme en fenêtre (*Graphical User Interface*) a été codé et permet de sélectionner les points d’intérêt et de les sauvegarder dans des matrices (voir FIGURE 4.4). Elle permet de charger la banque des filtres de résonance ou d’anti-résonance, puis de choisir un sujet. Les points rouges correspondent aux points

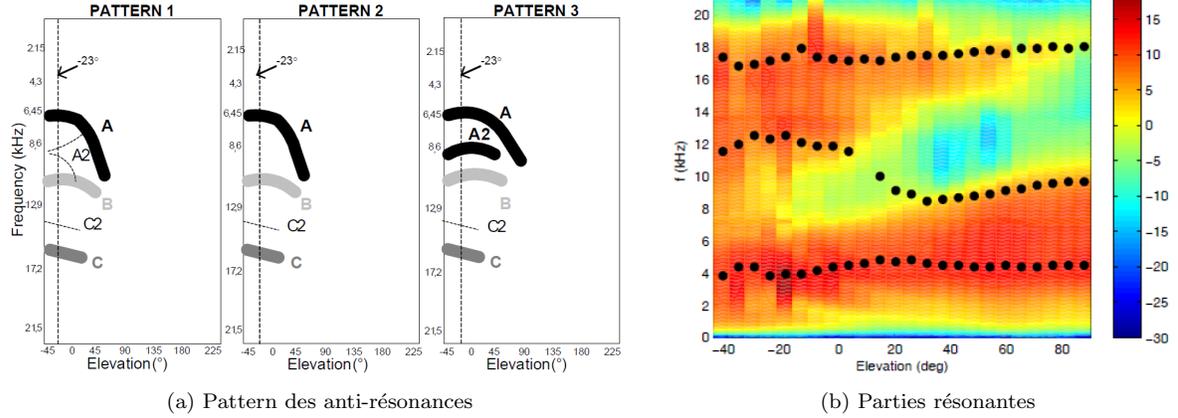


FIGURE 4.2 – Les trajectoires remarquables dans les filtres

trouvés de manière automatique avec le modèle STMCB, et les points bleus sont ceux édités à la main. Les deux lignes violettes correspondent à une estimation des fréquences haute et basse de réflexion sur les parois de l'oreille et de la conque [20] : elles permettent de mieux identifier les trajectoires A et B.

### 4.1.3 Transformation des données

Ayant obtenu ces lieux fréquentiels de résonances et anti-résonances, l'étape de transformation s'inspire des techniques de morphing d'images pour transformer les parties résonantes et anti-résonantes de manière à ramener chaque (anti-)résonance en face de la (anti-)résonance moyenne pour une direction donnée. La FIGURE 4.5a illustre le point de départ et le point d'arrivée de la méthode.

Pour cela, l'axe des fréquences est déformé de manière à faire passer les résonances par les points moyens. Comme illustré sur la FIGURE 4.5b, les points moyens sont placés en abscisse et les résonances en ordonnée. Ensuite une interpolation par spline cubique permet d'obtenir une nouvelle échelle de fréquence. Deux points artificiels ont été ajoutés aux coordonnées (1500, 1500) et (21000, 21000) pour éviter que la nouvelle échelle possède des valeurs négatives ou au dessus de 22050Hz ( $F_s/2$ ).

La même méthode est appliquée pour les parties anti-résonantes. Néanmoins, pour certains sujets, les trajectoires A et B se recoupent, et pour d'autres non. Au final, il y a deux trajectoires moyennes d'anti-résonances. Pour déformer ces filtres, nous avons artificiellement ajouté un second point à un  $\epsilon$  près (ici deux pas fréquentiels) lorsque la trajectoire est unique. Il y a un problème pour l'interpolation car deux points sont très proches en ordonnées. En effet, avec une interpolation linéaire il y a l'apparition d'un plateau dans la reconstruction du filtre, et avec une interpolation par spline cubique il peut y avoir des anti-résonances fantômes qui sont des artefacts de l'interpolation. Il a donc fallu créer une spline particulière pour éviter ce problème. La solution finale a été la suivante.

Soit  $S$  la spline,  $x = (x_0, x_1, x_2, x_3, x_4, x_5)$  les points moyens et  $y = (y_0, y_1, y_2, y_3, y_4, y_5)$  les anti-résonances, avec comme point particulier  $x_0 = y_0 = 0$ ,  $x_1 = y_1 = 1500$ ,  $x_4 = y_4 = 21000$  et  $x_5 = y_5 = 22050$ . On cherche cinq polynômes, un pour chaque intervalle  $[x_i, x_{i+1}]$  :

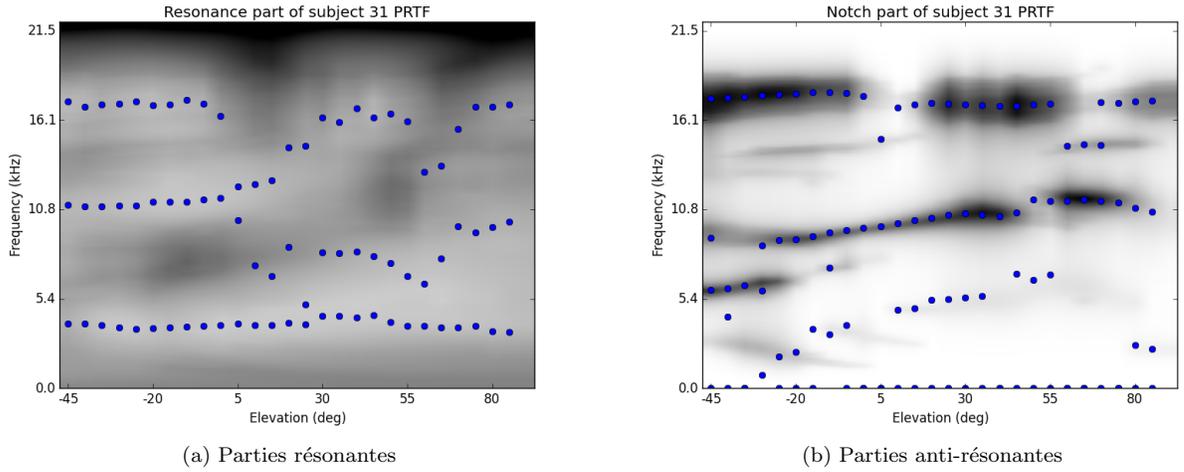


FIGURE 4.3 – Les filtres et les trajectoires identifiées

$$\begin{aligned}
 S(t) &= P_0(t) = t && \text{si } t \in [x_0, x_1] \\
 S(t) &= P_1(t) = a_1 t^2 + b_1 t + c_1 && \text{si } t \in [x_1, x_2] \\
 S(t) &= P_2(t) = a_2 t^3 + b_2 t^2 + c_2 t + d_2 && \text{si } t \in [x_2, x_3] \\
 S(t) &= P_3(t) = a_3 t^3 + b_3 t^2 + c_3 t + d_3 && \text{si } t \in [x_3, x_4] \\
 S(t) &= P_4(t) = t && \text{si } t \in [x_4, x_5]
 \end{aligned}$$

Les contraintes relatives à la courbe interpolée sont le passage par les points  $(x_i, y_i)$ , la continuité entre les polynômes aux points  $(x_i, y_i)$ , la continuité des dérivées des polynômes aux points  $(x_i, y_i)$  et trois contraintes supplémentaires :

$$\begin{aligned}
 P_1'(x_2) &= e_1 \\
 P_2'(x_3) &= 0 \\
 P_3'(x_4) &= e_2
 \end{aligned}$$

En effet, il faut qu'il y ait un maximum entre les deux anti-résonances (pour voir apparaître deux anti-résonances), que la courbe ne passe pas en dessous de la deuxième anti-résonance, et que la courbe soit strictement monotone après le deuxième minimum. Les degrés des polynômes ont été choisis de manière à avoir autant d'équations que d'inconnues. Les dérivées sont choisies par essai-erreur et finalement  $e_1 = 1,5$  et  $e_2 = 0,5$ . Le résultat final est illustré sur les FIGURES 4.6a et 4.6b.

## 4.2 Résultats

La prédiction pour le sujet 1 des parties résonante et anti-résonance est disponible en FIGURE 4.7. Les points bleus sont les points originaux, les points rouges les points prédits par la méthode SVM et les carrés verts les points moyens.

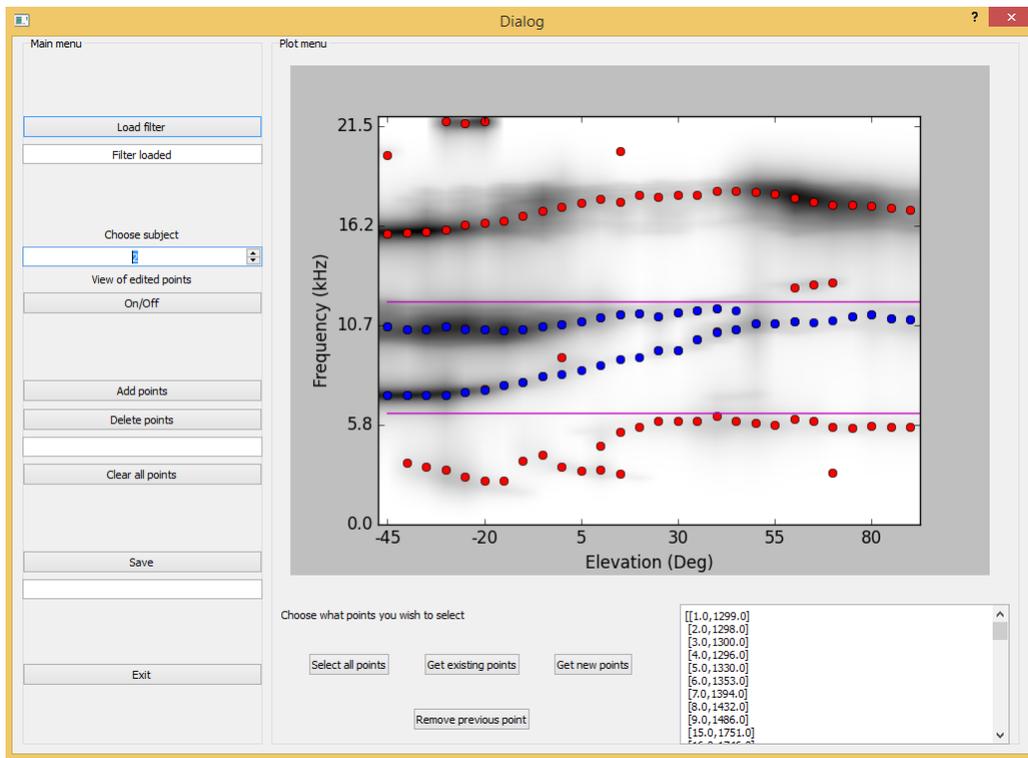


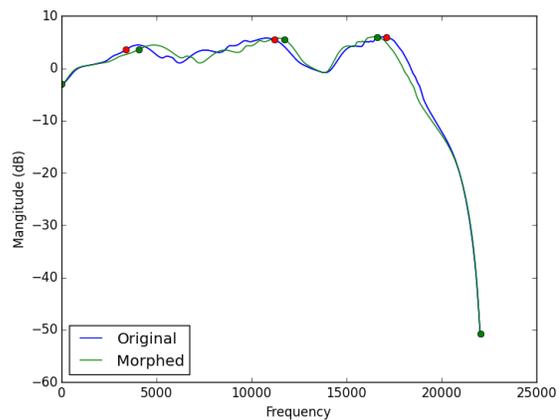
FIGURE 4.4 – L’interface permettant la sélection de points

### 4.3 Discussion

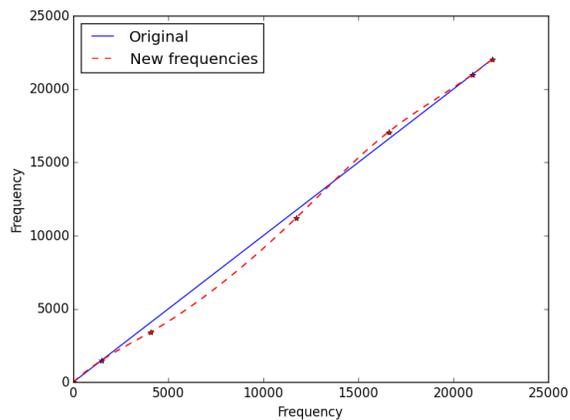
La régression *Sparse* avait déjà montré son efficacité au niveau calculatoire, car elle permet de prédire des HRTF de manière rapide. En effet, la phase d’apprentissage ne nécessite que les paramètres morphologiques et donc elle va très vite. Ensuite il suffit d’appliquer le vecteur de régression pour chaque position et chaque fréquence pour obtenir le HRTF. Ici, une représentation parcimonieuse des paramètres morphologiques est recherchée puis appliquée pour calculer les enveloppes de résonance et d’anti-résonance. La phase de morphing précédente permet d’éviter l’effet de lissage au niveau des résonances et anti-résonances.

Par contre, la régression *Sparse* ne fonctionne pas pour la prédiction des positions fréquentielles. C’est pourquoi une régression linéaire pénalisée  $L^2$  et une régression SVM ont été testés. Malheureusement, les résultats ne sont pas du tout concluants : soit la méthode est trop prudente et ne modifie pas la position des résonances ou anti-résonances, soit elle prédit des valeurs aberrantes. Dans le cas de certains sujets (dont le sujet 1), la trajectoire des anti-résonances se recoupe au-delà d’une certaine direction spatiale : la méthode ne parvient pas à prédire ce comportement. En analysant la FIGURE 4.7a, on peut penser que les prédictions concordent avec les points originaux, mais celles-ci sont en fait bien plus proches des points moyens que des points originaux. Ceci est confirmé par l’étude de la FIGURE 4.7b, qui montre un chevauchement quasi parfait des points moyens et des points prédits.

Cette approche est pertinente car elle permet de séparer la contribution ”basse résolution” du HRTF (son enveloppe) de la ”haute résolution” (les pics et les creux). Malheureusement, les techniques actuelles de régression ne permettent pas de prédire avec une bonne précision les positions de ces pics et creux. En effet, ils sont sujets à une forte variabilité, et cela doit impacter la qualité de la régression. Il serait sans doute judicieux de créer deux groupes de sujets : ceux pour lesquels il existe deux trajectoires



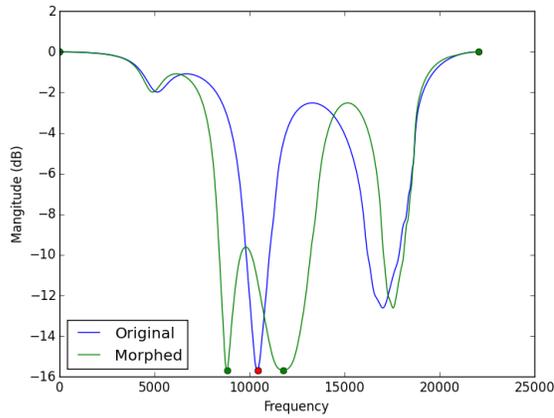
(a) Le filtre original (bleu) avec ses résonances (points rouges), et le filtre morphé (vert) avec les résonances moyennes (points verts)



(b) La déformation de l'axe des fréquences

FIGURE 4.5 – Cas des résonances

d'anti-résonances, et ceux pour lesquels il n'en existe qu'une seule. Néanmoins le manque de données ne permet pas de faire ce genre de clustering sans que cela n'impacte les prédictions.



(a) La déformation d'un filtre ayant une seule anti-résonance

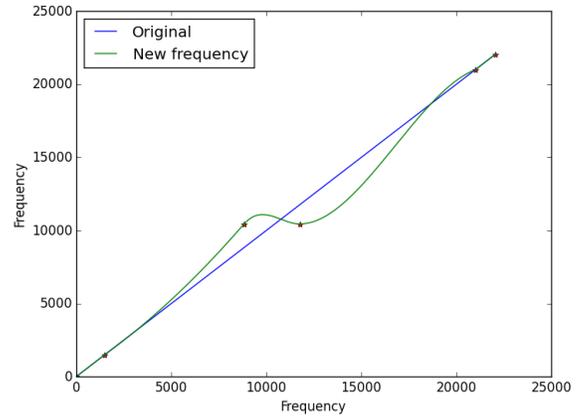
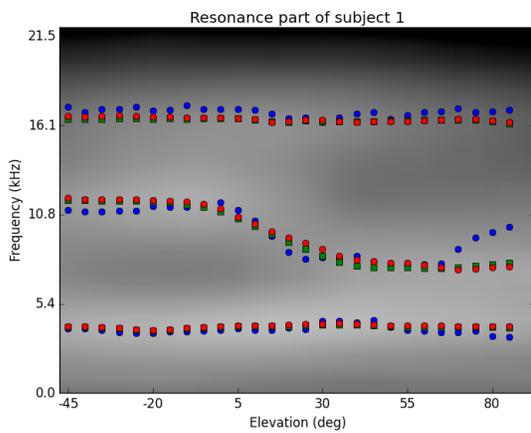
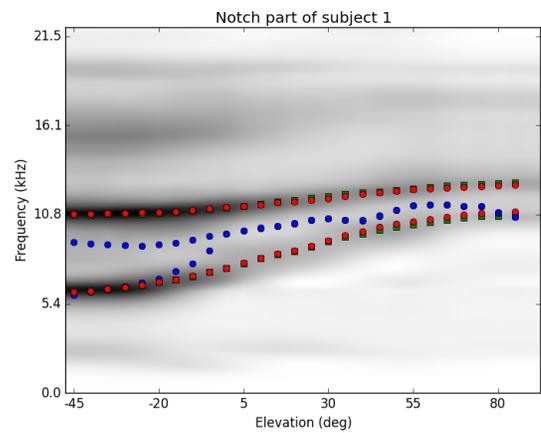
(b) La déformation de l'axe des fréquences avec la spline  $S$ 

FIGURE 4.6 – Cas des anti-résonances



(a) Résonance



(b) Anti-résonance

FIGURE 4.7 – Prédiction des parties résonante et anti-résonante pour le sujet 1 : en rouge les points prédits, en bleu les points originaux, en vert les points moyens

## Chapitre 5

# Conclusion

Ce mémoire a étudié la possibilité d'individualiser des filtres HRTF à partir de données morphologiques par une approche d'apprentissage statistique. Pour cela, les bases de données *CIPIC* et *LISTEN* ont été utilisées pour les tâches de régressions. Des méthodes conventionnelles comme la régression linéaire *Ridge* ou *LARS*, la régression non paramétrique *SVM* ou encore la régression *PLS* ont été envisagées, sans donner de résultats probants. Afin d'augmenter la précision des prédictions, une approche tensorielle basée sur la *HOSVD* et la *HOPLS* a été mise en oeuvre. Malheureusement elle souffrait des mêmes problèmes que les méthodes linéaires classiques, à savoir que la prédiction des résonances et anti-résonances était de piètre qualité. C'est pourquoi une dernière approche basée sur la séparation des parties résonante et anti-résonante du HRTF a été considérée. Le but était de ramener tous les HRTF dans un même "cadre" permettant une prédiction de meilleure qualité.

Cette étude a mis en évidence que l'individualisation de filtres HRTF n'est pas une tâche aisée, et que la principale difficulté réside dans la prédiction des résonances et anti-résonances. En effet, ces indices sont sujet à une forte variabilité, ce qui implique que les tâches de régressions classiques "moyennent" la prédiction. Une future étude devrait être menée pour élaborer une nouvelle technique permettant de prédire ces informations fréquentielles avec plus de précision. Une autre approche pourrait consister en une transformation des données dans un espace qui se rapproche de la perception humaine. Par exemple, au lieu de considérer les fréquences de manière linéaire, plusieurs banques de filtres de MEL ou de BARK pourraient être utilisées.

Enfin, la dernière technique relative à la séparation des HRTF a mis en évidence la présence d'au moins deux groupes parmi les individus : ceux pour lesquels la partie anti-résonante comporte deux trajectoires d'anti-résonances bien distinctes, et ceux pour lesquels elle possède deux trajectoires qui se recoupent. Il serait alors judicieux d'appliquer un modèle de mélange de régression [11, 16]. Ce type de modèle suppose la présence de groupes sous-jacents dans les données, mais que l'on ne connaît pas à l'avance. Une fois que les groupes sont identifiés, une régression est apprise dans chaque groupe. Enfin, lorsqu'on souhaite faire une prédiction sur un nouveau sujet, on détecte d'abord à quel groupe il appartient puis on applique la régression adéquate.

# Bibliographie

- [1] Francis Bach. *Méthodes à noyaux en apprentissage statistique*, 2008.
- [2] P. Bilinski, J. Ahrens, M. R. P. Thomas, I. J. Tashev, and J. C. Platt. Hrtf magnitude synthesis via sparse representation of anthropometric features. *IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, pages 4468–4472, May 2014.
- [3] Christopher Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [4] Stéphane Canu. Machine à noyaux pour l'apprentissage statistique. *Techniques de l'Ingénieur*, 2007.
- [5] Antoine Cornuéjol and Laurent Miclet. *Apprentissage artificiel*. Eyrolles, 2010.
- [6] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. & Appl.*, 21(4) :1253–1278, April 2000.
- [7] Matteo Dellepiane, Nico Pietroni, Nicolas Tsingos, Manuel Asselot, and Roberto Scopigno. Reconstructing head models from photographs for individualized 3d-audio processing. *Computer Graphics Forum (Special Issue - Pacific Graphics 2008 Proc.)*, 27(7) :1719–1727, 2008.
- [8] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshinari. Least angle regression. *The Annals of Statistics*, 32(2) :407–499, 2002.
- [9] Michele Geronazzo and Simone Spagnol. Estimation and modeling of pinna-related transfer functions. *Conference on Digital Audio Effects*, 2010.
- [10] Raphael Greff. *Holophonie binaurale. Spatialisation sonore sur réseaux de haut-parleurs circumauraux*. PhD thesis, Université Paris VI, 2008.
- [11] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer, 2009.
- [12] Jianjun He, Woon-Seng Gan, and Ee-Leng Tan. On the preprocessing and postprocessing of HRTF individualization based on sparse representation of anthropometric features. *IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, 2015.
- [13] Xiaofei He, Deng Cai, and Niyogin Partha. Laplacian score for feature selection. *Advances in Neural Information Processing System*, 2005.
- [14] Qinghua Huang and Lin Li. Modeling individual hrtf tensor using high-order partial least squares. *EURASIP Journal on Advances in Signal Processing*, 2014.
- [15] IRCAM. <http://recherche.ircam.fr/equipes/salles/listen/>.
- [16] L. Montuelle and E. Le Pennec. Mixture of gaussian rregression model with logistic weights, a penalized maximum likelihood approach. *Electronic Journal Of Statistics*, 8 :1661–1695, 2014.
- [17] University of California Davis. <http://interface.cipic.ucdavis.edu/sound/hrtf.html/>.
- [18] Sergio G. Rodriguez and Miguel A. Ramirez. Extracting and modeling approximated pinna-related transfer functions from hrtf data. *Proceedings of ICAD*, 2005.
- [19] David Schönstein and Brian FG Katz. Sélection de HRTF dans une base de données en utilisant des paramètres morphologiques pour la synthèse binaurale. *10ème Congrès Français d'Acoustique*, 2010.

- [20] Simone Spagnol, Michele Geronazzo, and Federico Avanzini. Fitting pinna-related transfer functions to anthropometry for binaural sound rendering. *IEEE International Workshop on Multimedia Signal Processing*, 2010.
- [21] Song Xu, Zhizhong Li, and Gavriel Salvendy. Improved method to individualize head-related transfer function using anthropometric measurements. *Acoust. Sci. & Tech.*, 29(6) :388–390, 2008.
- [22] M. Zhang, R. A. Kennedy, T. D. Abhayapala, and W. Zhang. Statistical method to identify key anthropometric parameters in hrtf individualization. *Joint Workshop on Hands-free Speech Communication and Microphone Arrays*, 2011.
- [23] Qibin Zhao, Cesar F. Caiafa, Danilo P. Mandic, Zenas C. Chao, Yasuo Nagasaka, NAotaka Fujii, Liqing Zhang, and Andrzej Cichocki. Higher-order partial least squares (hopls) : A generalized multi-linear regression method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013.