



Rapport de Stage

DÉTERMINATION OBJECTIVE D'AUDIOGRAMME À PARTIR DES MESURES DE POTENTIELS ÉVOQUÉS AUDITIFS

Réalisé par Réjane JOYARD

Référents : Clara DUSSAUX, Jacques BOUTET DE MONVEL
Tuteur Universitaire : Renaud PETERI

Master MIX 2023/2024 - Université de La Rochelle

22 Avril 2024 - 29 Juin 2024

REMERCIEMENTS

C'est avec plaisir que je réserve cette page en signe de gratitude et de reconnaissance à tout ceux qui m'ont aidé tout au long du stage.

Je suis également reconnaissante envers l'Institut Pasteur pour son hospitalité.

À l'issu de ce travail, j'aimerais exprimer ma reconnaissance et mes sincères remerciements à : Clara Dussaux et Jacques Boutet de Monvel, mes encadrants durant ma période de stage, pour leur réceptivité, leur pédagogie et leurs qualités qui m'ont permis d'effectuer ce stage dans des meilleures conditions.

J'aimerais également remercier toutes les personnes qui ont pris le temps de me faire découvrir leur métier.

De ma part, j'espère que ma conduite et mon apprentissage ont laissé une bonne impression de La Rochelle Université, et que ma participation à la recherche effectuée a eu un apport bénéfique.

TABLE DES MATIÈRES

Introduction	1
1 Réponse auditive du tronc cérébral (ABR)	2
1.1 Fonctionnement du système auditif	2
1.2 Mesure des potentiels évoqués auditifs	3
1.2.1 Réalisation des mesures	3
1.2.2 Interprétation des enregistrements	4
2 Détermination objective du seuil auditif	4
2.1 Méthode 1 : Corrélation croisée	5
2.1.1 Principe et déroulement du processus	5
2.1.2 Résultat et interprétation	7
2.2 Méthode 2 : Transformée en ondelettes	7
2.2.1 Théorie	8
2.2.2 Application aux données	8
2.2.3 Algorithme : calcul du seuil auditif	11
2.3 Comparaison des méthodes	13
3 Méthode CNN sur les scalogrammes	15
3.1 Structure du réseau	16
3.2 Résultats	18
3.2.1 Modèle et prédictions	18
3.2.2 Validation	19
4 Discussion	21
4.1 Contribution effective au projet	21
4.2 Perspectives	21
4.3 Intérêt	21
Annexes	22
Appendix A Application : interface graphique	22
Table des figures	24
Références	25

Introduction

L'évaluation de la réponse auditive du tronc cérébral en réponse à un son, connue sous le nom de potentiels évoqués auditifs (ou ABR), constitue une méthode pour évaluer le fonctionnement du nerf auditif et des premiers relais des voies auditives dans le tronc cérébral. Ces mesures permettent d'estimer les seuils de sensibilités auditives, fournissant ainsi des informations précieuses sur la fonction auditive des individus ou des animaux. Cette méthode de phénotypage auditif consiste en l'enregistrement d'un signal électrophysiologique en réponse à une stimulation auditive. Il est constitué de cinq ondes, qui apparaissent généralement entre 1 et 10 ms suivant la stimulation sonore réalisée. Le seuil auditif, déterminé par l'intensité minimale permettant l'obtention d'une onde ABR clairement identifiable, est traditionnellement évalué de manière partiellement subjective par l'appréciation visuelle de l'expérimentateur.

La plateforme d'acquisition et de traitement du signal neuronal de l'Institut de l'Audition se penche actuellement sur le développement de nouveaux outils. Ces outils visent à acquérir et analyser de manière objective et reproductible les signaux phénotypiques, explorant ainsi diverses solutions algorithmiques pour estimer le seuil auditif.

Ce rapport de stage se propose de présenter cette démarche exploratoire et comparative, visant à améliorer la précision et la fiabilité de l'évaluation des seuils auditifs.

1 Réponse auditive du tronc cérébral (ABR)

La mesure des potentiels évoqués auditifs est une méthode utilisée en audiologie pour évaluer le fonctionnement du système auditif chez l'humain et l'animal. Une des utilisations principales de l'ABR est l'évaluation du seuil auditif : elle permet aux professionnels de santé d'établir un audiogramme afin de diagnostiquer les problèmes auditifs, tels que la surdité ou d'autres troubles de ce système. Une interprétation précise des résultats de ce test est essentielle pour évaluer la sévérité d'une atteinte auditive.

1.1 Fonctionnement du système auditif

Il est d'abord important de comprendre le mécanisme par lequel le système auditif opère, du son à la perception du cerveau. Le rôle de ce système est de convertir les fluctuations de la pression de l'air en signaux électriques (Figure 1). L'ouïe est utile non seulement pour détecter les sons environnementaux, mais elle est également cruciale pour la perception de la parole.

Le système auditif commence dans l'oreille externe, où le pavillon de l'oreille réceptionne les ondes sonores provenant de sources extérieures et les dirige dans le conduit auditif jusqu'au tympan. Ces ondes font vibrer le tympan qui lui, va communiquer les vibrations au système. Celles-ci sont ensuite transmises à travers une série d'osselets de l'oreille moyenne (marteau, enclume, étrier), qui agissent comme un système de levier amplificateur et dont les vibrations sont transmises à la cochlée, une structure en forme de coquillage (Figure 2).

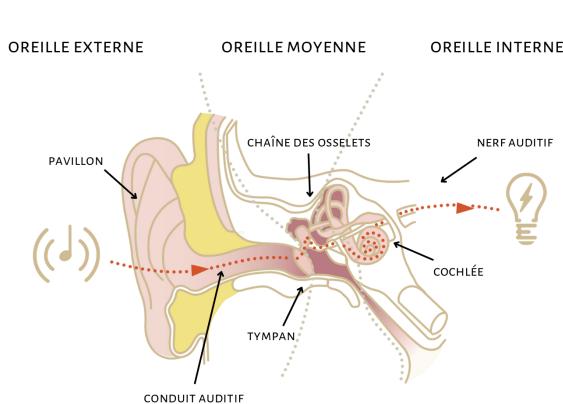


Figure 1 — Système auditif

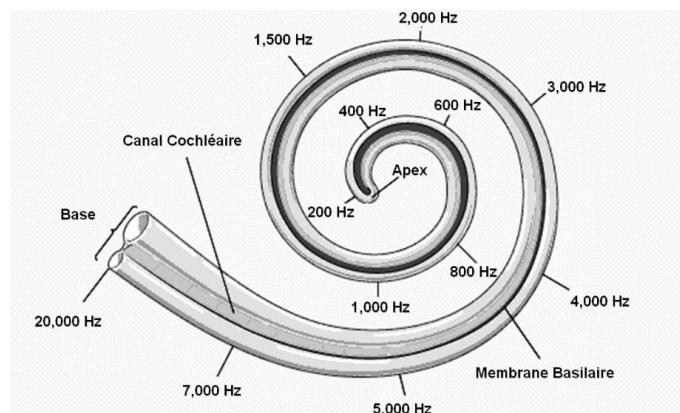


Figure 2 — Schéma de la cochlée selon les fréquences (kHz)

Dans la cochlée, organe central de l'audition, se réalise la transduction du son (vibration mécanique) en signal électrochimique, sans laquelle la sensation auditive ne serait pas possible. Les variations de fréquence sonore, donnent lieu à une diversité de perceptions auditives. Grâce à sa tonotopie, la cochlée est capable de discriminer les différentes fréquences des vibrations sonores et va ainsi permettre la distinction entre les sons aigus (fréquence haute) et les sons graves (fréquence basse).

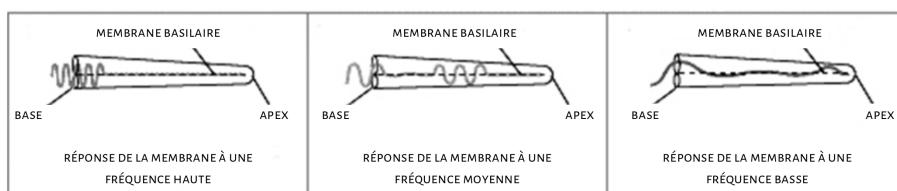


Figure 3 — Réponses de la membrane basilaire selon les fréquences

Les vibrations transmises à la cochlée sont converties en signaux électriques par des cellules ciliées. Cela entraîne la libération de neurotransmetteurs qui provoquent le déclenchement d'un potentiel d'action au niveau du nerf cochléaire. Celui-ci va transmettre ces signaux dans les différentes régions du cerveau dans lesquelles l'information sonore sera analysée, interprétée et transformée en perceptions auditives conscientes.

1.2 Mesure des potentiels évoqués auditifs

1.2.1 Réalisation des mesures

Dans notre étude, les tests auditifs sont réalisés à la fois sur des souris et des sujets humains. Toutefois, en raison de la plus grande reproductibilité des résultats obtenus chez les souris, notre analyse se concentrera principalement sur ces dernières. Cette meilleure reproductibilité résulte du fait que les souris sont anesthésierées, contrairement aux mesures d'ABR humains, qui sont réalisés sur une personne calme et immobile, mais éveillée, conduisant à une variabilité accrue de la réponse auditive.

Lors du test, les souris sont généralement placées dans une chambre insonorisée pour minimiser les interférences externes. Une première électrode est placée sur le crâne, une deuxième derrière l'oreille et une dernière sur le dos de l'animal pour enregistrer l'activité électrique générée par les voies auditives en réponse à des sons (Figure 4).

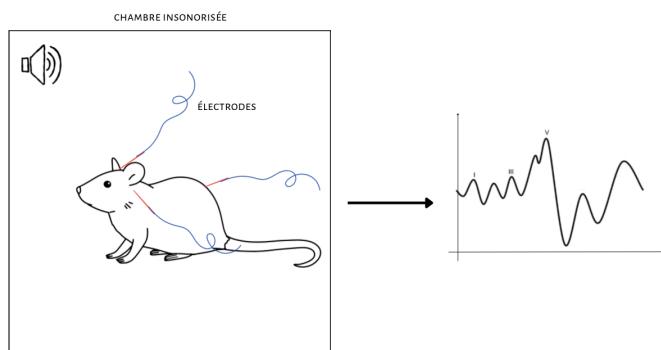


Figure 4 — Réalisation des mesures d'ABR sur une souris

En simplifiant, la première électrode située sur l'oreille va permettre de capter l'activité électrique du nerf auditif. Celle placée sur le dos, va permettre de mesurer la différence de potentiels électriques enregistrée avec les deux autres électrodes (crâne et oreille). Le signal ABR correspond donc à la différence de ces deux mesures, ce qui va permettre de supprimer les artefacts électriques comme la lumière ou l'alimentation, de chacune des électrodes. Ainsi, l'analyse de l'activité électrique du tronc cérébral en réponse à des stimuli sonores, pourra se faire avec une plus grande précision sans être affectée par les fluctuations électriques externes qui pourraient perturber les mesures.

Les stimuli sonores sont présentés à des intensités variables, généralement allant de 10 à 90 décibels. Ils peuvent être présentés sous forme d'une série de clics ou de tons purs à différentes fréquences et l'activité électrique enregistrée par les électrodes est amplifiée, filtrée ainsi qu'analysée pour identifier les potentiels évoqués auditifs spécifiques.

De plus, l'ABR n'est pas affecté par l'anesthésie et peut donc théoriquement être enregistré lorsque le patient ou l'animal est endormi. Toutefois, en raison de la faible amplitude ABR, elle peut être facilement perdue dans tout autre bruit qui peut être enregistré par le système. Ces bruits peuvent provenir de sources endogènes (physiologiques) comme l'activité musculaire, l'activité cardiaque ou les mouvements oculaires ; ou des sources exogènes (environnementales) comme les interférences du réseau et les équipements électriques. Pour réduire le bruit, plusieurs solutions sont mises en place en pratique : le filtrage des artefacts cardiaques dès l'acquisition, l'anesthésie de l'animal qui réduit considérablement l'activité musculaire ainsi que le filtrage du signal à l'acquisition dans la bande de fréquence qualifiée de "non utile" notamment toutes les basses fréquences. Il restera encore un peu de bruit, et le SNR (Signal-to-Noise Ratio) est faible, il est donc nécessaire de moyennner les signaux. Dans notre cas il s'agit de 300 répétitions : le signal affichera alors la moyenne entre ces 300 signaux.

Une onde ABR peut avoir typiquement une amplitude de $0,5\mu\text{V}$, alors que le bruit de fond lui, peut être supérieure à $\pm 15\mu\text{V}$.

1.2.2 Interprétation des enregistrements

Lors de la mesure d'ABR, les résultats se présentent sous la forme d'une succession ondes de latences différentes (numérotées de I à VII), reflétant l'activité électrique mesurée dans le cerveau en réponse à des stimuli sonores. Ces ondes correspondent aux différentes étapes du trajet de la réponse le long des voies auditives, de la cochlée au cortex. Les réponses à la mesure sont enregistrées sous forme d'amplitudes (en μV) fluctuant dans le temps et peuvent être utilisées pour tirer des conclusions sur l'état du système auditif d'un patient. Les enregistrements de l'ABR fournissent ainsi des informations sur la manière dont les relais successifs du système auditif répondent à des stimuli sonores de différentes intensités.

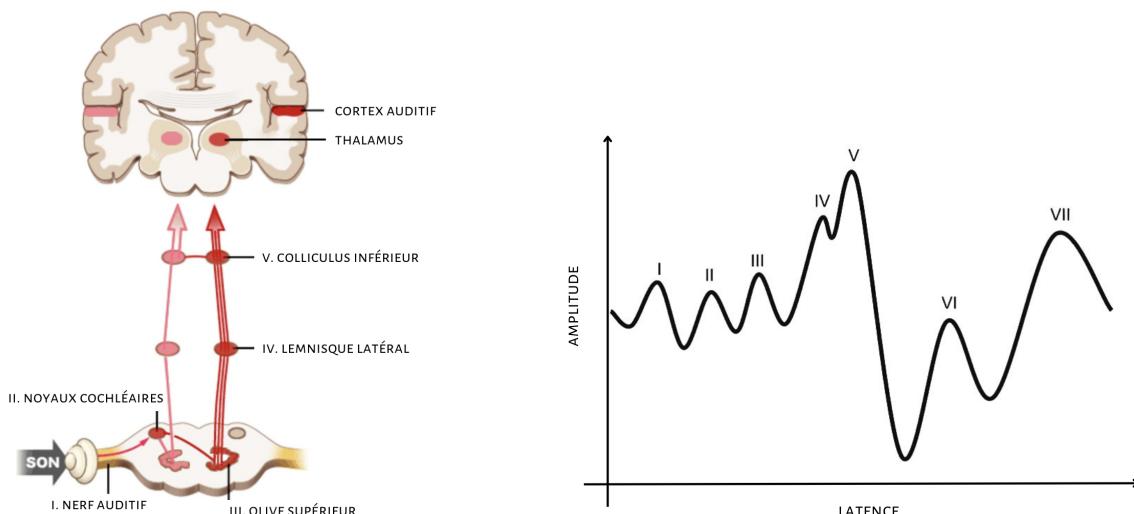


Figure 5 — Système nerveux auditif ; morphologie d'un ABR

En pratique et dans les enregistrements humains, l'onde V qui est l'onde associée à la réponse du colliculus inférieur, est particulièrement mise en avant en raison de son amplitude plus marquée, ce qui facilite sa détection et son interprétation. Chez la souris, c'est l'onde I qui est souvent la plus marquée.

La morphologie d'un enregistrement va dépendre de plusieurs paramètres : le type de stimulation, la fréquence, le niveau d'intensité ainsi que le degré et le type de perte auditive. De nombreuses morphologies peuvent être enregistrées, ce qui peut rendre l'interprétation de la forme d'onde encore plus difficile. Pour mieux repérer l'onde V pour les enregistrements humains, nous pouvons nous référer à la latence : en effet, cette onde devrait être entre 7 et 9 millisecondes environ. Aussi, l'onde V se manifeste à des latences plus courtes en réponses à des stimuli d'intensités plus fortes et à des latences plus longues pour des intensités moins fortes. Une diminution de l'amplitude est également attendue pour les intensités moins fortes.

2 Détermination objective du seuil auditif

Dans la pratique clinique, l'interprétation des ondes ABR est basée sur l'inspection visuelle de la forme d'onde moyennée de façon cohérente. Cependant, l'interprétation basée sur l'inspection visuelle est variable, même parmi les cliniciens expérimentés. Bien qu'il existe plusieurs méthodes statistiques de détection des ABR, l'inspection visuelle est toujours considérée comme étant la plus fiable et reste la base de l'analyse des ABR en pratique. L'objectif principal sera d'explorer diverses approches en vue d'automatiser l'interprétation des données de l'audiogramme, afin de déterminer la méthode la plus efficace et la plus précise possible.

Dans cette partie, deux méthodes seront présentées : une méthode de corrélation croisée (ou cross-correlation) ainsi qu'une méthode utilisant une transformée en ondelettes.

2.1 Méthode 1 : Corrélation croisée

La corrélation croisée est une opération mathématique utilisée dans divers domaines, notamment en traitement du signal, en traitement d'images et pour la reconnaissance de formes. Son objectif est de repérer les similarités entre deux signaux ou images, facilitant ainsi l'identification de motifs, la détection d'objets ou encore le suivi de mouvements.

Dans notre contexte, nous allons l'utiliser pour traiter les signaux obtenus lors de la mesure d'ABR. Cette méthode va nous permettre de quantifier la similarité entre deux signaux, en fonction d'un décalage temporel appliqué à l'un de ceux-ci.

2.1.1 Principe et déroulement du processus

Dans un premier temps, nous allons diviser les signaux enregistrés en trois segments temporels : celui qui précède le signal, correspondant à la période de pré-stimulation ; le signal lui-même, représentant la période pendant laquelle la stimulation est appliquée ; et enfin, celui qui suit le signal, correspondant à la période post-stimulation. Ceci va nous permettre de mesurer la similarité entre des parties de séquences temporelles. Nous sommes certains que le signal ABR se situe entre 0 et 12 millisecondes suivant la stimulation auditive, de ce fait, les délimitations ont été précisément définies pour garantir que le signal se trouve dans la séquence sélectionnée.

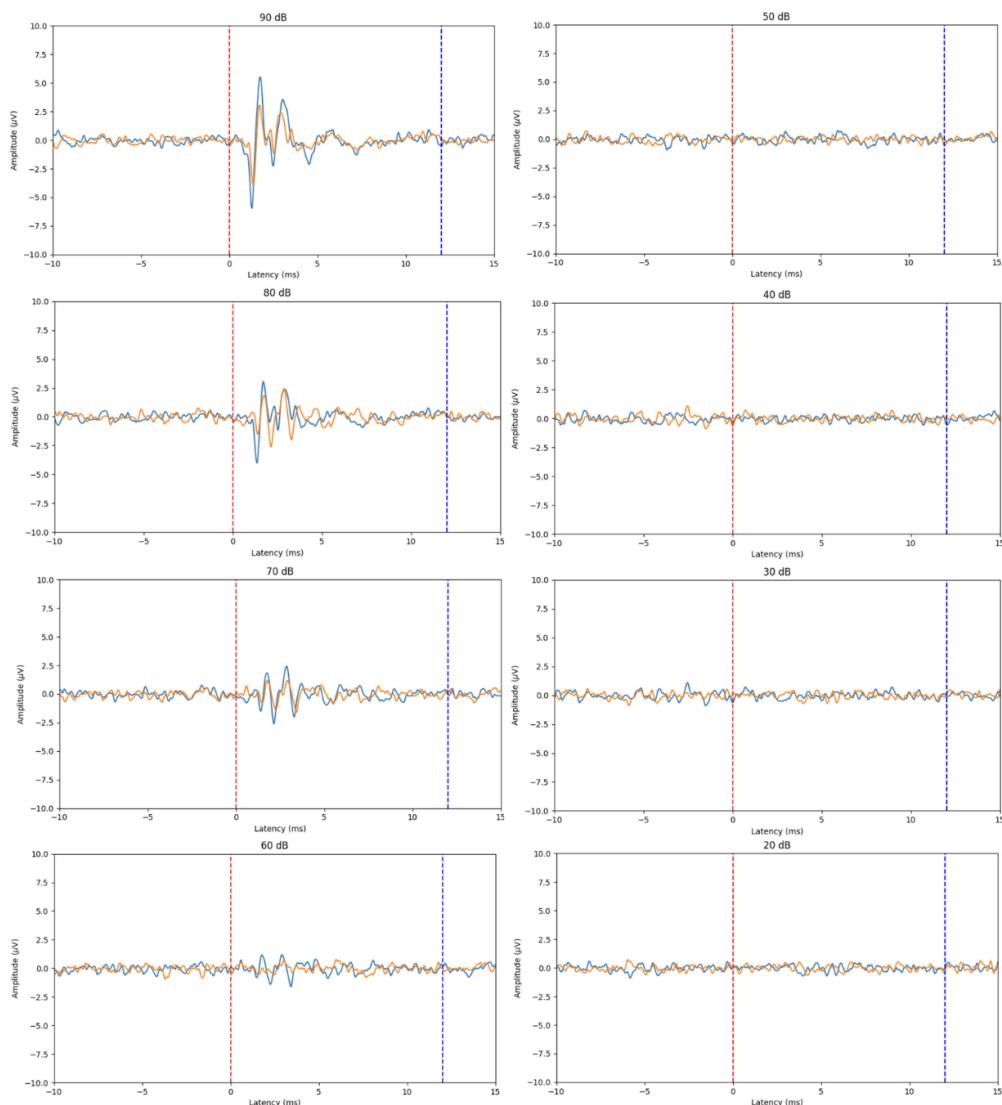


Figure 6 — Enregistrements ABR d'une souris pour une fréquence de 40kHz et d'intensité variable de 90 à 20 dB

Prenons l'exemple du premier graphe (90dB) de la Figure 6. La corrélation croisée se fait entre le signal à 90dB qu'on appellera *previous average* (courbe en bleue) et le signal à 80dB qu'on appellera *current average* (courbe en orange), sur des parties temporelles différentes.

Une première corrélation croisée est calculée entre la partie pre-signal de *previous average* (80 dB) et la partie signal (entre 0 et 12 ms) de *current average* (90 dB). Cette première corrélation est utilisée pour calculer un seuil de significativité. Puis une deuxième corrélation est calculée entre la partie signal de *previous average* et la partie signal de *current average*. Nous prenons ensuite le maximum de cette seconde corrélation qui sera comparer au seuil de significativité.

La corrélation croisée discrète entre deux séquences de données f et g est calculée comme suit :

$$(f \star g)[n] = \sum_{m=-\infty}^{\infty} f^*[m]g[n+m] \quad (1)$$

où,

- f^* est le complexe conjugué de f ,
- \star est l'opération de corrélation croisée,
- n est l'indice de décalage entre les deux séquences f et g . Nous remarquons que si $n = 0$, cela signifie qu'elles sont parfaitement superposées. Si n est positif, cela signifie que g est traduite vers la gauche par rapport à f , et si n est négatif, cela signifie que g est traduite vers la droite par rapport à f ,
- m est l'indice d'échantillonnage dans les fréquences f et g . Théoriquement m varie de $-\infty$ à ∞ mais en pratique pour des séquences de longueur finie, nous sommes limités à un nombre fini d'échantillons.

Pour chaque n , est calculée la somme des produits de conjugaison entre les deux séquences et pour tous les décalages possibles entre f et g .

Ainsi, à l'issue de cette opération nous pourrons récupérer la séquence de corrélation croisée calculée entre les deux séquences choisies.

Par la suite, le seuil de significativité d'une réponse ABR est calculé en fonction de la moyenne μ et l'écart-type σ de la séquence récupérée par la première corrélation croisée effectuée (en code python):

$$\text{threshold} = \mu + \text{norm.ppf}(1 - \alpha) \sigma \quad (2)$$

où $\text{norm.ppf}(1 - \alpha)$ est une fonction de la bibliothèque `scipy`, qui renvoie la valeur du quantile de la distribution normale centrée réduite avec une probabilité de $1 - \alpha$, où α est un nombre compris entre 0 et 1 qui correspond à la probabilité d'erreur tolérée. Ici, nous choisissons $\alpha = 10^{-5}$ qui correspond à un intervalle de confiance de 99,999%, i.e. environ 4,3 σ .

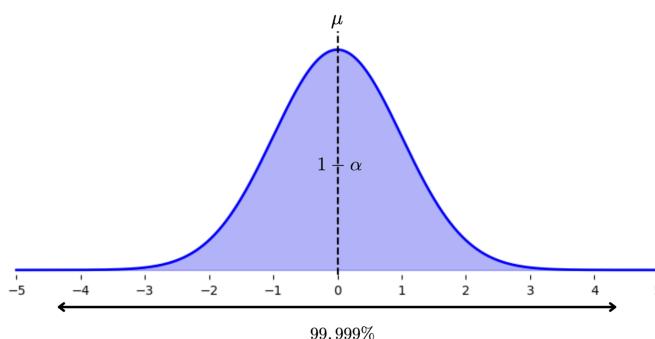


Figure 7 — Distribution normale centrée réduite avec un intervalle de confiance de 99,999%

2.1.2 Résultat et interprétation

Ci-dessous, le résultat de l'application de l'algorithme qui renvoie la corrélation croisée sur les enregistrements de la Figure 6 :

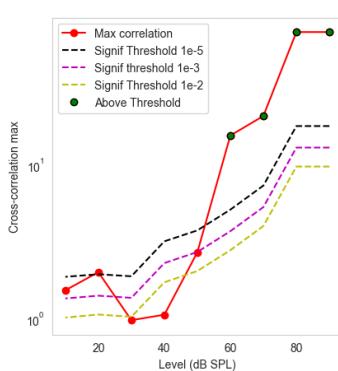


Figure 8 — Graphe corrélation croisée avec différents seuils de significativité

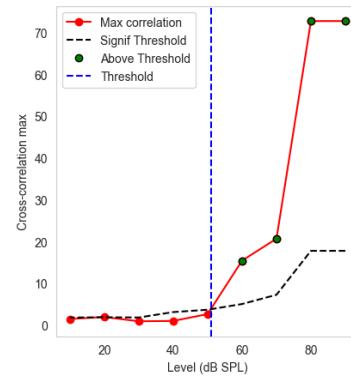


Figure 9 — Seuil exacte récupéré avec l'interpolation

Si la corrélation croisée est supérieure au seuil de bruit, le signal ABR est dit significatif. Lorsque l'on passe sous le seuil de significativité, on en conclut que nous sommes arrivés au seuil. Sur cette représentation graphique, un point représente la corrélation entre 2 signaux : 90 dB avec 90 dB, 90 dB avec 80 dB, 80 dB avec 70 dB, etc... Ceci nous permet de donner un seuil auditif et pour cette donnée, le seuil renvoyé par l'algorithme est de **60 dB** (Figure 8). Afin d'affiner le résultat, nous pouvons réaliser une interpolation pour trouver le point de croisement de la courbe du seuil significatif avec celui la corrélation maximum. Nous obtenons alors un seuil à **51 dB** (Figure 9).

2.2 Méthode 2 : Transformée en ondelettes

La transformée de Fourier et la transformée en ondelettes sont deux outils mathématiques très utilisés pour analyser les signaux dans divers domaines. Les séries de Fourier, ont été développées par Jean-Baptiste Joseph Fourier au début du 19e siècle, dans le but d'obtenir un algorithme permettant de décomposer un signal en une somme de sinusoïdes de différentes fréquences. Cette décomposition est utile pour les signaux stationnaires, où les caractéristiques fréquentielles ne diffèrent pas au cours du temps. Elle présente donc des limitations lorsqu'il s'agit de signaux non stationnaires. De cette façon, la transformée en ondelettes a été développée et introduite dans les années 1980 par des mathématiciens tels que Ingrid Daubechies. Elle va permettre une analyse à la fois temporelle et fréquentielle des signaux, décomposant un signal en composantes appelées ondelettes qui sont localisées à la fois en temps et en fréquence. Cette capacité à effectuer une analyse multi-résolution en fait un outil puissant pour l'étude de signaux transitoires et de structures locales complexes.

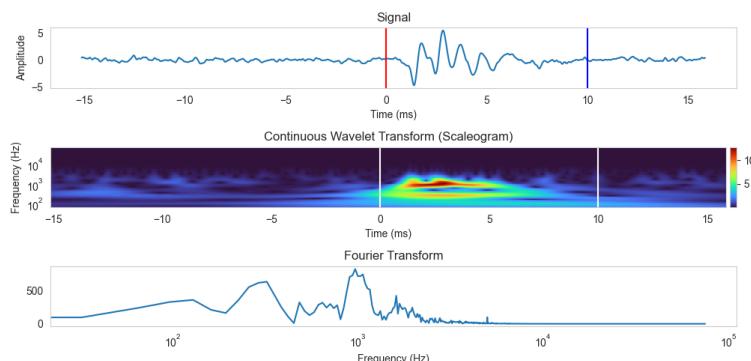


Figure 10 — Comparaison Transformée de Fourier/Transformée en ondelettes pour un signal (90 dB)

Comparée à la transformée de Fourier, la transformée en ondelettes offre une flexibilité accrue pour capturer des détails fins, et des phénomènes transitoires tout en conservant une vue d'ensemble du signal. Elle est très utilisée dans le traitement de signal, et particulièrement pour l'analyse des signaux enregistrés lors des tests ABR, nous permettant ainsi d'isoler des composantes significatives liées aux différents potentiels du système auditif.

2.2.1 Théorie

Mathématiquement la transformée en ondelettes utilise une fonction ψ dans $L^2(\mathbb{R})$, appelée ondelette mère et localisée en espace ou en temps et de moyenne nulle :

$$\int_{-\infty}^{+\infty} \psi(t) dt = 0$$

Contrairement à une décomposition de Fourier qui utilise toujours des fonctions de base exponentielles complexes (sinus et cosinus), une décomposition en ondelettes utilise une fonction oscillatoire localisée dans le temps comme ondelette mère.

L'ondelette mère est une fonction continue en temps et en fréquence qui sert de fonction source à partir de laquelle sont construites les fonctions de base mises à l'échelle et translatées. Elle peut être complexe ou réelle, et comprend généralement un paramètre réglable qui contrôle les propriétés de l'oscillation localisée. L'analyse en ondelettes est plus compliquée que l'analyse de Fourier car il faut spécifier complètement l'ondelette mère à partir de laquelle les fonctions de base seront construites. Cette ondelette doit en particulier vérifier la continuité, elle doit être intégrable et d'énergie finie.

Le principe de cette transformation est de représenter un signal comme une combinaison linéaire de ces ondelettes que l'on peut translater et dilater ou contracter. La famille des ondelettes rassemble des fonctions analysantes, traduisant les transformations de l'ondelette mère ψ et sont définies comme suit :

$$\psi_{x,s}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-x}{s}\right)$$

avec $(s, x) \in \mathbb{R}^{+*} \times \mathbb{R}$. Ainsi, toutes les ondelettes ont la même énergie.

La transformée en ondelettes est définie comme le résultat d'un opérateur intégral qui transforme une fonction d'énergie finie $f(t) \in L^2(\mathbb{R})$ en utilisant un ensemble de fonction $\psi_{x,s}$. Elle est définie par le produit scalaire entre la fonction ondelette mère $\psi_{x,s}$ et une fonction réelle ou complexe $f(t)$:

$$\text{cwt}(s, x) = \langle f(t), \psi_{x,s}(t) \rangle = \frac{1}{\sqrt{s}} \int_{-\infty}^{+\infty} f(t) \cdot \psi^*\left(\frac{t-x}{s}\right) dt$$

où ψ^* est le conjugué complexe de ψ .

Cette transformation modifie l'espace dans lequel on se place à un espace à deux dimensions : l'espace des coefficients d'ondelettes. D'après cette interprétation, il est possible de représenter un signal temporel monodimensionnel $f(t)$ sous forme d'un champ à deux dimensions $\text{cwt}(s, x)$ en faisant varier x , qui est homogène à un temps et s , qui est homogène à une échelle.

2.2.2 Application aux données

En principe cette transformation continue fonctionne en utilisant directement la définition de la transformée en ondelettes, c'est-à-dire en calculant le produit scalaire du signal par l'ondelette mise à l'échelle. Dans notre contexte, nous allons utiliser la fonction `cwt(data, scales, wavelet)` de la bibliothèque PyWavelets qui prendra en paramètres les enregistrements ABR, l'échelle à laquelle nous voulons observer l'ondelette ainsi que le type ψ d'ondelettes. Cette fonction va nous permettre de réaliser la transformation en ondelettes continues d'un signal unidimensionnel et de renvoyer un résultat à deux dimensions.

Le choix du type d'ondelettes ψ est important pour l'interprétation des résultats. Chacun des types peut mettre en évidence différents détails selon ce que l'on cherche à étudier ou observer précisément. Il va permettre de régulariser la décomposition temps-fréquence qui a une influence sur la résolution : ces deux

paramètres sont importants car ils peuvent faire ressortir divers résultats. Si ces choix ne correspondent pas au signal étudié au départ, nous risquons de passer à côté d'informations cruciales pour notre analyse. Les différents types d'ondelettes compatibles avec la fonction `pywt.cwt` sont présentés ci-dessous :

- *Morlet* :

C'est une ondelette continue qui est à la fois définie dans le domaine temporel et dans le domaine fréquentiel, elle est ainsi souvent utilisée pour analyser des signaux ayant des caractéristiques périodiques. Cette ondelette est définie par l'expression suivante :

$$\psi(t) = e^{\frac{-t^2}{2}} \cos(5t)$$

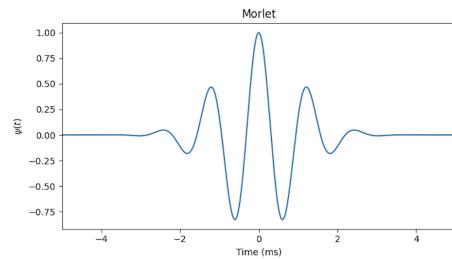


Figure 11 — Morlet Wavelet

- *Complex Morlet* :

Sa particularité est de fournir à la fois des informations sur l'amplitude et sur la phase du signal, elle est donc utile pour l'analyse de signaux non-stationnaires. Cette ondelette est définie par l'expression suivante :

$$\psi(t) = \frac{1}{\sqrt{\pi B}} e^{-\frac{t^2}{B}} e^{2j\pi Ct}$$

avec B la bande passante et C la fréquence centrale.

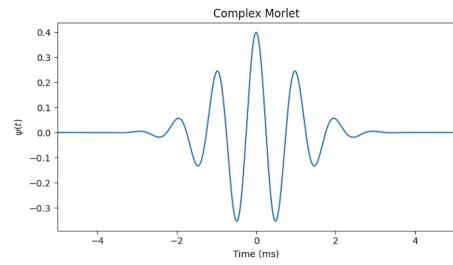


Figure 12 — Complex Morlet Wavelet

- *Frequency B-Spline* :

Cette ondelette est efficace pour détecter et analyser les transitoires ou les changements brusques dans un signal qui peuvent être des pics soudains par exemple. Elle peut révéler des structures temporelles et fréquentielles complexes dans des signaux, ce qui en fait un outil précieux pour la décomposition et la compréhension de ceux-ci. Cette ondelette est définie par l'expression suivante :

$$\psi(t) = \sqrt{B} \left[\frac{\sin(\pi B \frac{t}{M})}{\pi B \frac{t}{M}} \right]^M e^{2j\pi Ct}$$

avec M l'ordre de la spline, B la bande passante et C la fréquence centrale.

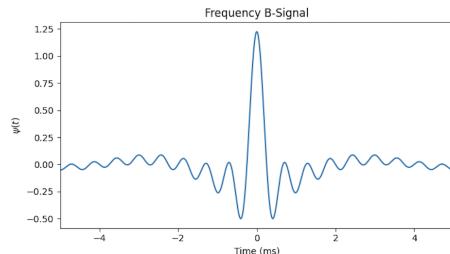


Figure 13 — Frequency B-Signal Wavelet

- *Mexican Hat* :

Cette fonction est aussi connue sous le nom d'ondelette de Ricker, et est souvent utilisée pour le traitement des données sismiques. Elle est définie comme la dérivée seconde de la fonction gaussienne et sa formule est la suivante :

$$\psi(t) = \frac{2}{\sqrt{4\sqrt[4]{\pi}}} e^{\frac{t^2}{2}} (1 - t^2)$$

Sa forme va permettre de mettre en évidence les points où il y a un changement brusque dans le signal, comme des pics ou des bords.

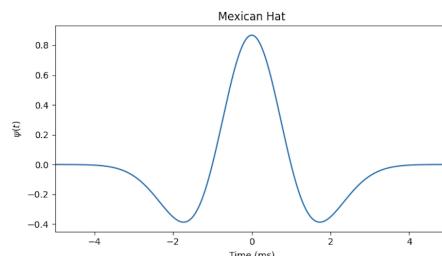


Figure 14 — Mexican Hat Wavelet

- *Shannon Wavelet* :

L'ondelette de Shannon est définie par la transformée de Fourier d'une fonction porte, ce qui signifie qu'elle a une bande passante finie dans le domaine fréquentiel. Elle est moins localisée en temps, elle est donc mieux adaptée aux signaux où l'information fréquentielle est plus importante que l'information temporelle. Cette ondelette est définie par l'expression suivante :

$$\psi(t) = \sqrt{B} \frac{\sin(\pi B t)}{\pi B t} e^{j2\pi C t}$$

avec B la bande passante et C la fréquence centrale.

Une ondelette très localisée en fréquence offre une haute résolution fréquentielle, mais cela ne signifie pas pour autant qu'elle est mal résolue. Cela signifie plutôt qu'elle est spécifiquement bien résolue en fréquence au détriment de la résolution temporelle. Donc si une ondelette est très localisée en fréquence comme l'ondelette Morlet par exemple, elle a une bonne résolution fréquentielle et une moins bonne résolution temporelle. Ci-dessous nous retrouvons les résultats des différents types d'ondelettes précédentes appliquées à un signal à 90 dB.

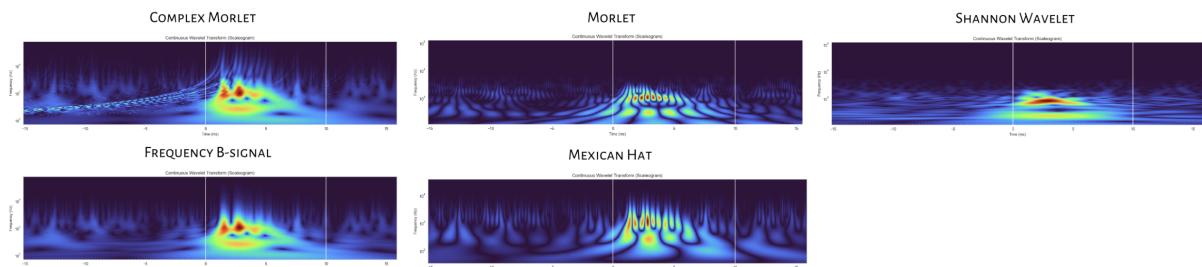


Figure 16 — Différents types d'ondelettes

Les types d'ondelettes dépendant d'une bande passante ainsi que d'une fréquence centrale peuvent prendre différentes formes selon le réglage de ces paramètres. C'est pourquoi nous choisissons de prendre l'ondelette *Complex Morlet*, qui va nous permettre d'adapter avec exactitude la forme que nous souhaitons. La fréquence centrale détermine la fréquence principale autour de laquelle l'ondelette oscille et la largeur de la bande passante affecte l'étalement de l'ondelette. La figure 17 illustre les différentes formes que peut prendre l'ondelette *Complex Morlet* selon la variation des paramètres.

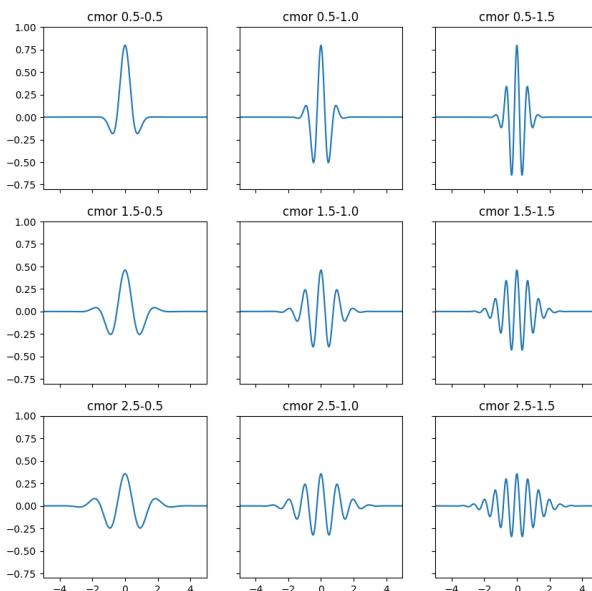


Figure 17 — Variations des paramètres **bandwidths** et **center frequencies**

Nous remarquons qu'en augmentant la fréquence centrale, le nombre d'oscillations dans l'ondelette augmente également. Concernant le paramètre de bande passante : une bande plus étroite signifie que l'ondelette est plus localisée en temps, ce qui donne une forme plus concentrée. Inversement, une bande plus large signifie que l'ondelette est plus dispersée, couvrant une plus grande étendue de temps.

Pour en savoir un peu plus sur le choix, il est essentiel de faire varier ces différents paramètres sur un enregistrement ABR et de les comparer. Nous allons appliquer les différents changements de paramètres sur un signal d'une intensité de 90 dB, où l'ABR est le plus visible.

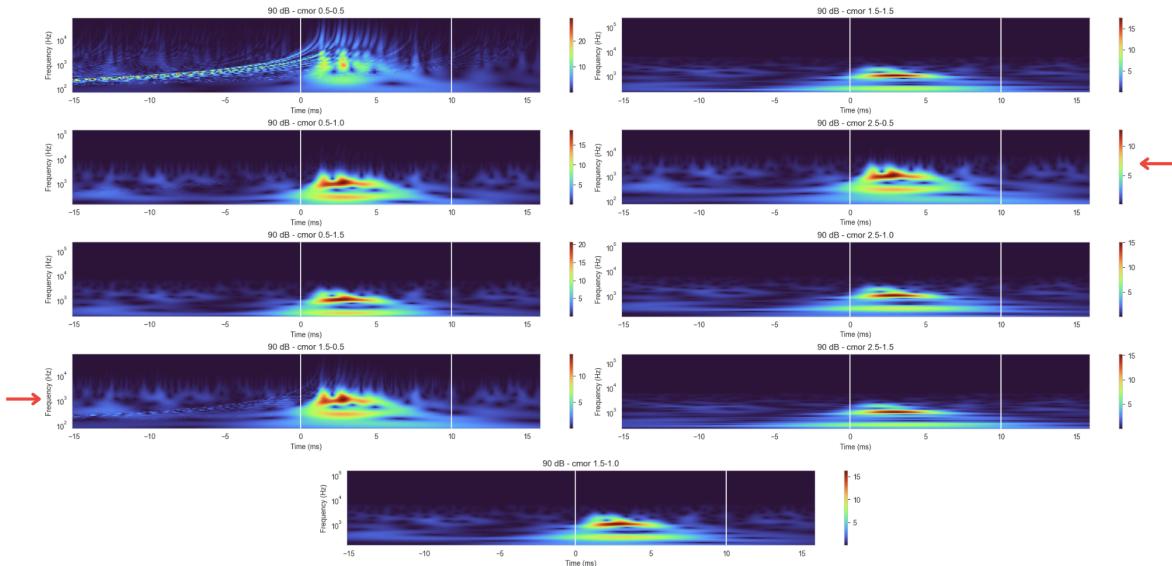


Figure 18 — Visualisation des ondelettes selon les variation **bandwidths** et **center frequencies**

Certains choix de paramètres correspondent au même résultat, par exemple avec 1,5 comme largeur de bande passante et 0,5 de fréquence centrale, nous obtenons un résultat similaire à celui obtenu lorsque la largeur de la bande passante est à 2,5 et la fréquence centrale à 0,5. Nous pourrions penser qu'une forme d'onde avec plus d'oscillations correspondrait mieux à la forme de l'ABR, mais nous observons bien que l'ABR ressort beaucoup plus avec des ondes ayant peu d'oscillations comme l'ondelette "cmor 1.5-0.5".

2.2.3 Algorithme : calcul du seuil auditif

Nous souhaitons à présent déterminer un seuil auditif à partir de ces scalogrammes. Une donnée est constituée de 9 scalogrammes, allant de 90 dB à 10 dB.

Nous allons appliquer un algorithme classique qui consiste à mettre en place trois boxes sur chaque scalogramme (Figure 18). Elles sont placées de façon à être certain de prendre en compte les zones que l'on cible, en particulier nous voulons que la première soit bien avant le signal, la deuxième doit être située entre 1 et 5 ms afin d'être sûr de la présence du signal à l'intérieur et enfin la troisième après le signal (*i.e.* après 10 ms).

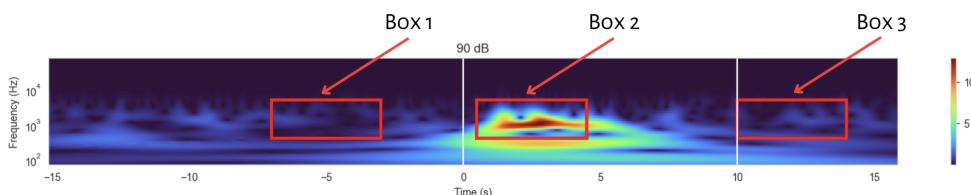


Figure 19 — Scalogramme (90 dB) avec placement des boxes

Dans un premier temps, nous calculons pour chaque scalogramme la moyenne des amplitudes de la box 2. Puis, nous concaténons les amplitudes des boxes 1 et 3 afin de former une nouvelle box, que nous nommerons

box 4 et récupérons la moyenne des amplitudes de cette box.

Reprendons le calcul du seuil (2), vu dans la partie corrélation croisée :

$$\text{threshold} = \mu + \text{norm.ppf}(1 - \alpha) \sigma$$

avec toujours de même intervalle de confiance, c'est-à-dire avec un risque de $\alpha = 10^{-5}$. Ici, μ sera la moyenne des amplitudes de la box 4 et σ en sera l'écart-type. Nous allons pouvoir comparer ce seuil récupéré avec la moyenne des amplitudes de la box 2, *i.e* la moyenne des amplitudes contenant le signal. Ainsi, lorsque le seuil est strictement inférieur à cette moyenne l'algorithme retournera 1 sinon 0. La donnée aura pour seuil l'intensité associée au dernier 1 avant le premier 0 affiché.

Algorithm 1 Algorithme Seuil Méthode Boxes

Entrée: Donnée signal

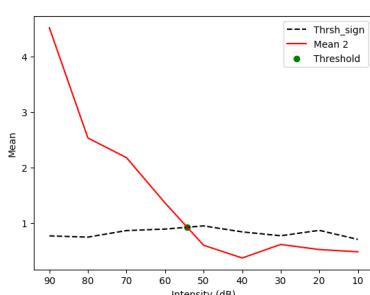
Sortie: Seuil thrsh

```

for i in range(signal.shape[1]) do
    Transformer signal en ondelette
    Placer les boxes
    for j in enumerate(boxes) do
        Récupérer les amplitudes
        Calculer la moyenne de la box 2 ← mean 2
        Calculer la somme des boxes 1 et 3 ← box 4
        Calculer la moyenne de la box 4 ← mean 4
    end for
end for
Intervalle de confiance ← alpha
mean 4 + norm.ppf(1-alpha,0,1)*std(mean 4) ← thrsh_sign
if thrsh_sign < mean 2 then
    Afficher 1
else
    Afficher 0
end if
Intervalle de 90 à 10 dB ← interval
for t in range(90, 9, -10) do
    interval[t] ← Index
    if condition[Index] == 0 then
        Break
    end if
end for
Index ← thrsh
return thrsh

```

De la même manière que pour la première méthode, nous pouvons affiner le résultat du seuil en affichant les courbes de **mean2** et de **thrsh_sign** et en les interpolant. Le croisement de ces courbes nous donnera un seuil plus précis.



Ci-contre, le graphe associé à la même donnée étudiée lors de la corrélation croisée (Figure 9) où nous avions récupéré un seuil à **51 dB**. Cette méthode nous donne un seuil à **54 dB**. La différence de quelques décibels entre les deux seuils renvoyés est en réalité négligeable. Ainsi nous fixerons par la suite un seuil arrondi à la dizaine près : pour cette donnée le seuil est donc défini à **50 dB** pour les deux méthodes.

Figure 20 — Croisement des moyennes

2.3 Comparaison des méthodes

Jusqu'à présent, nous avons réalisé notre analyse sur des données individuelles. Il est maintenant intéressant d'appliquer ces deux méthodes sur un jeu de données, afin de déterminer la méthode la plus appropriée à notre problème, de comprendre les différences entre elles et de les comparer aux résultats visuels évalués par un expérimentateur. Pour cela, nous avons développé un algorithme permettant de récupérer les seuils auditifs en utilisant ces deux méthodes sur notre jeu de données labélisé par un expérimentateur expert.

On se proposera d'observer les résultats sur deux intervalles de confiance différents en variant la probabilité d'erreur α . Nous choisissons d'abord α égal à 10^{-3} puis à 10^{-5} qui correspondent respectivement à un intervalle de confiance de 99,9% et de 99,999%.

Dans un premier temps, nous allons nous concentrer sur une analyse regroupant la totalité des données. Une première méthode classique de visualiser les répartitions des résultats est la représentation de diagrammes en boîtes (ou boîtes à moustaches). Elle va permettre d'identifier selon les méthodes, certains éléments comme les médianes, les moyennes, la dispersion des valeurs de seuils ainsi que d'éventuelles valeurs qui peuvent ressortir (Figure 21 et 23, (a) Boxplot).

La deuxième représentation sera sous forme de courbe, appelée courbe de densité (Figure 21 et 23, (b) Densité) : elle va nous permettre d'observer la distribution des résultats et ainsi nous permettre de faire ressortir ou non une méthode en particulier. Puis, nous regarderons les données une à une afin de préciser certains éléments qui ressortent (Figure 22 et 24).

- Figure 21 (a) : nous pouvons remarquer que la méthode scalogramme à une répartition similaire à la méthode visuelle. Nous constatons également que des valeurs aberrantes ressortent sur les deux premières méthodes. Il sera intéressant par la suite d'observer ces données seules. La cross corrélation donne des valeurs largement inférieures à l'appréciation visuelle.
- Figure 21 (b) : les trois méthodes ont des distributions semblables, à savoir trois pics au niveau des seuils 10, 20 et 30 dB. Nous trouvons des résultats cohérents avec le fait que le jeu de données correspond à des animaux sains normo-entendants.

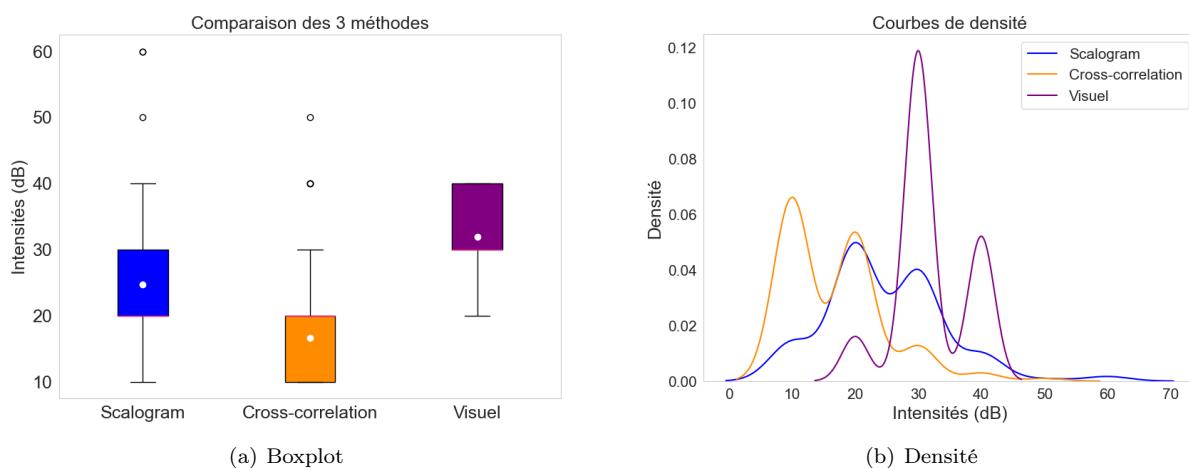


Figure 21 — Comparaison des résultats avec $\alpha = 10^{-3}$

Le graphique ci-dessous (Figure 22) nous montre pour chaque donnée, le seuil renvoyé par chaque méthode. De cette façon nous pouvons facilement visualiser les données qui devient significativement des tendances observées. Sur ce graphique, nous retrouvons les seuils de la méthode scalogramme qui sont les points bleus, ceux de la méthode de corrélation croisée qui sont les étoiles jaunes et enfin les seuils visuels qui sont en violet. La tendance de la méthode de la corrélation croisée est confirmée : nous observons bien les valeurs de seuils plus basses que les valeurs des deux autres méthodes.

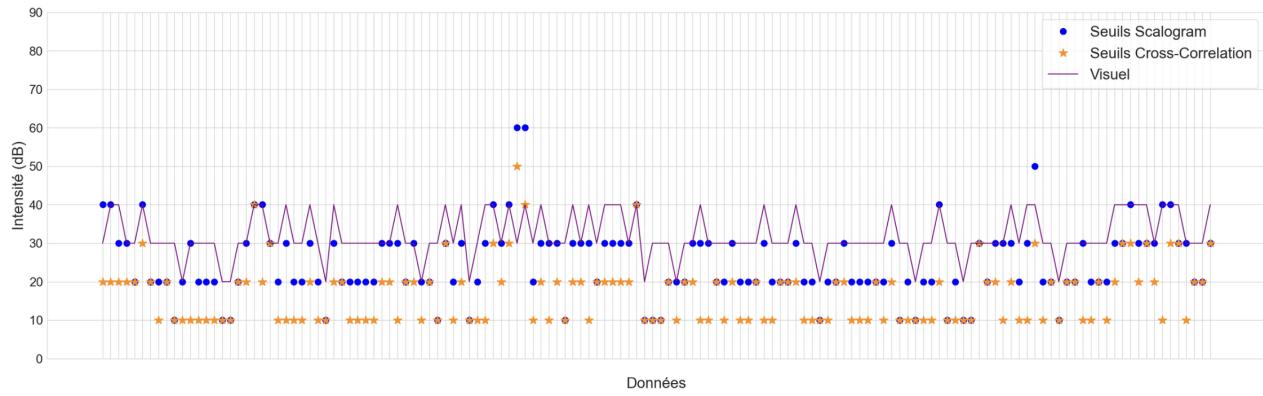


Figure 22 — Comparaison des méthodes donnée par donnée $\alpha = 10^{-3}$

L'écart des valeurs de seuils entre chaque méthode est de 10 ou de 20 dB. Cependant, certaines données ressortent : par exemple la 52ème donnée nous dit que l'expérimentateur a donné un seuil visuel à 30 dB tandis que les méthodes automatiques renvoient 60 ou 70 dB. Après avoir visualisé la donnée, nous avons pu en déduire que les algorithmes permettent d'identifier des erreurs d'appréciation visuelle et de faire un contrôle qualité a posteriori ; dans ce cas les deux algorithmes des méthodes automatiques ont été plus pertinentes, et ont donné un seuil qui correspondait et se rapprochait au mieux des signaux récupérés.

Nous observons une tendance pour la méthode des scalogrammes. Elle renvoie régulièrement des seuils à 50 dB alors que la corrélation croisée ou le seuil visuel indique 40 ou 30 dB : cet écart vient probablement de la méthode de calcul du seuil par les boxes. En effet, il peut y avoir parfois la présence d'artefacts sur le pré-signal ou le post-signal qui engendrerait ce genre d'erreur. Il pourrait être intéressant d'affiner cette méthode afin d'éviter la prise en compte de ces artefacts.

Observons à présent les résultats correspondant à un intervalle de confiance encore plus précis en utilisant α égal à 10^{-5} (Figure 22 et 23).

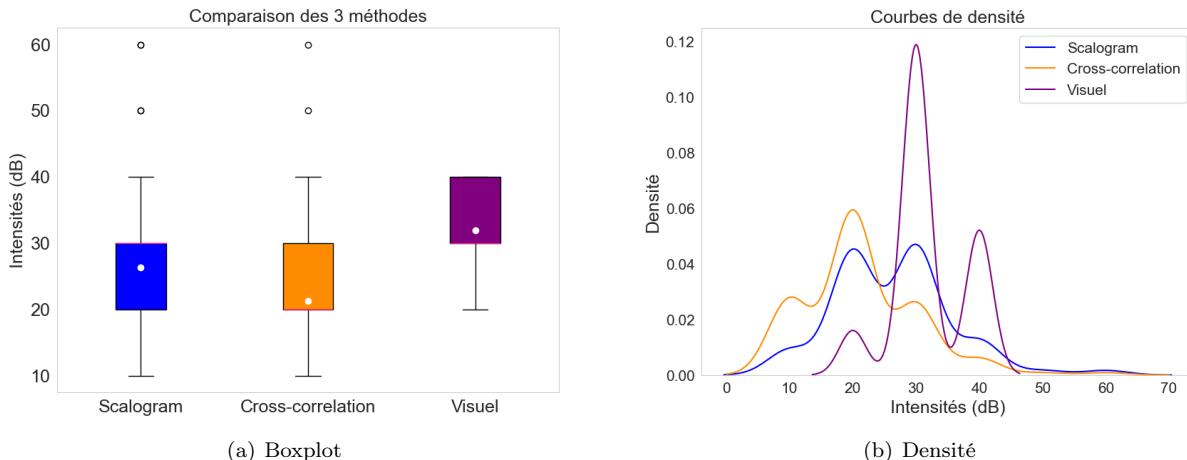


Figure 23 — Comparaison des résultats avec $\alpha = 10^{-5}$

Nous observons que la moyenne de la corrélation croisée reste légèrement plus basse avec une médiane à 20 dB. En revanche, la méthode du scalogramme affiche une médiane à 30 dB, ce qui est comparable à celle obtenue par la méthode visuelle.

Les courbes de densité nous indiquent que la méthode de corrélation croisée se rapproche plus de la tendance des autres méthodes avec cet intervalle de confiance. Son changement de distribution nous démontre qu'elle n'est pas très stable lorsque l'on varie l'intervalle de confiance, alors que pour la méthode scalogramme, la distribution reste quasiment inchangée.

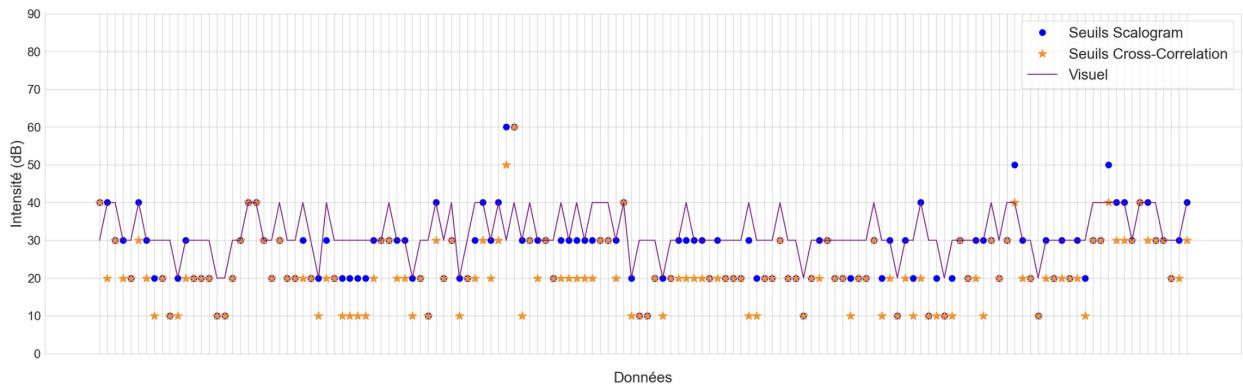


Figure 24 — Comparaison des méthodes donnée par donnée $\alpha = 10^{-5}$

De toute ces observations, nous pouvons conclure que l'intervalle de confiance est crucial selon les méthodes et en particulier avec celle de la corrélation croisée, et qu'elle doit être par conséquent choisie avec précaution si l'on souhaite confirmer les seuils avec celle-ci. Aussi, les deux méthodes algorithmiques se rapprochent beaucoup de l'évaluation visuelle par un expérimentateur bien que la méthode scalogramme nécessite encore des améliorations afin d'éviter que les artefacts faussent les résultats. Nous concluons qu'un algorithme est plus pertinent et peut permettre d'éviter les erreurs d'appréciation visuelle.

Cette analyse nous amène à explorer une méthode algorithmique plus avancée pour améliorer la précision de nos résultats, en particulier dans la détection et la classification des scalogrammes.

3 Méthode CNN sur les scalogrammes

Le CNN (Convolutional Neural Network) désigne une sous-catégorie de réseaux de neurones qui sont très utilisés pour la classification d'images. Pour ce processus, l'utilisateur fournit en entrée une image sous forme d'une matrice de pixels : une matrice à deux dimensions pour les images en niveaux de gris et une à trois dimensions pour les images en couleurs à 3 canaux (rouge, vert, bleu).

Contrairement à un réseau de neurones général de type MLP (Multi Layers Perceptron) qui contient une succession de couches pleinement connectées, l'architecture du CNN se distingue en deux parties, une partie convulsive suivie d'une partie classification.

La partie convulsive est adaptée à la structure en grille d'images : elle consiste à une succession de couches convolutionnelles, chacune étant elle-même la combinaison d'un filtre de convolution, suivi d'une fonction d'activation non-linéaire. Cette structure va permettre d'extraire des caractéristiques propres à chaque image : l'image d'entrée se voit passer à travers divers filtres, créant de nouvelles images que nous appelons cartes de convolutions. Ces cartes seront par la suite concaténées dans un vecteur de caractéristiques.

La partie classification reprend quant à elle ce vecteur de caractéristiques en tant qu'entrée d'un perceptron multicouches qui va permettre de combiner les caractéristiques pour pouvoir classifier l'image.

Dans notre cas, nous allons appliquer ce processus sur les scalogrammes de chaque donnée récupérés. Et ainsi permettre une classification et une identification de la présence ou l'absence de l'ABR.

Premièrement il est important de savoir comment nous allons construire un jeu de données cohérents avec nos objectifs. Nous disposons des 140 données précédentes, chacune renvoyant 9 scalogrammes, soit 1 260 scalogrammes au total. C'est un jeu de données conséquent qui va nous permettre d'avoir de nombreuses formes de scalogrammes afin d'entraîner un modèle avec le plus de cas possible.

Pour la construction de notre réseau de neurones, nous allons utiliser la bibliothèque TensorFlow Keras de Python.

3.1 Structure du réseau

Notre modèle de réseau de neurones a été conçu pour traiter les tâches de classification d'images en tirant parti des convolutions afin d'extraire les caractéristiques essentielles des images.

La structure réalisée suit un schéma classique utilisé pour concevoir un CNN : elle alterne généralement des couches de convolution et de pooling.

Avant de commencer à construire un modèle de type CNN, les données sont généralement transformées.

Donnée d'entrée. En entrée du réseau de neurones, nous devons avoir un vecteur. Chaque image est un tableau de taille 375×82 ayant des entrées entre 0 et 255 (image RGB). Nous la transformons en un vecteur de taille $375 \times 82 = 30\,750$ et nous normalisons les données dans l'intervalle [0, 1], en divisant par 255 (Figure 25).

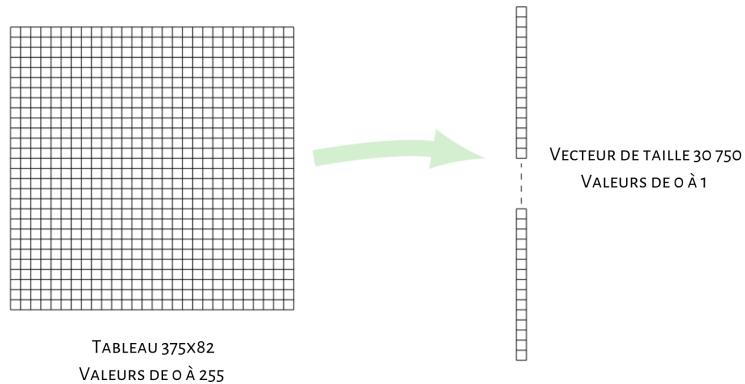


Figure 25 — Donnée d'entrée du réseau de neurones : transformation de l'image vers un vecteur

Ainsi, une entrée X est un "vecteur-image", c'est-à-dire un vecteur de taille 30 750 représentant une image, donc dans notre cas un scalogramme.

Donnée de sortie. Notre réseau de neurones va renvoyer un chiffre, soit 1 si le modèle a détecté la présence d'un ABR, soit 0 si aucun ABR n'a été détecté.

Après avoir fixé les objectifs du modèle, l'entrée et la sortie attendues, nous allons construire le réseau de neurones. L'architecture est composée de 5 couches (Figure 26).

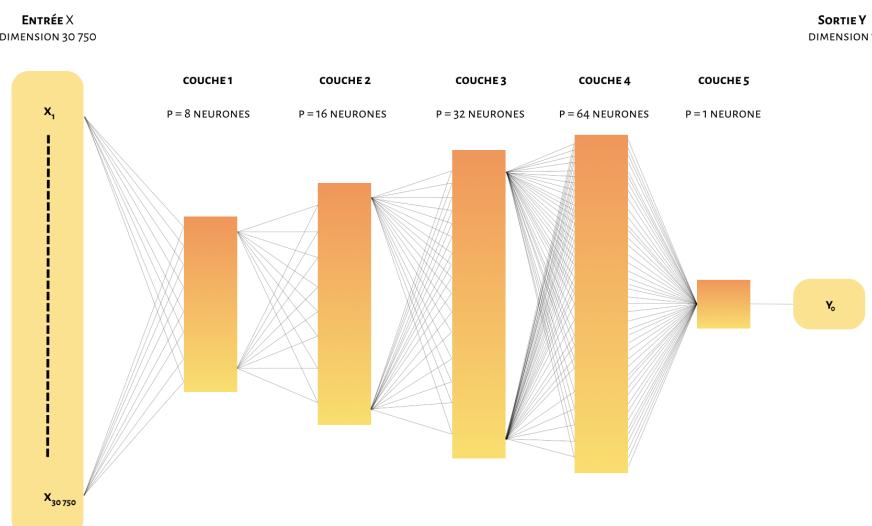


Figure 26 — Architecture du réseau de neurones

Dans notre cas, nous ne transformons pas les données d'entrée en un vecteur mais nous conservons une grille 2D des images jusqu'aux couches de classification. La première couche applique 8 filtres de convolution de taille 3×3 sur l'entrée afin d'extraire des caractéristiques de bas niveau comme les textures ou les motifs simples. C'est une couche qui va produire une première carte de caractéristiques de 8 canaux.

La deuxième couche applique 16 filtres de convolution de taille 3×3 permettant d'extraire des caractéristiques de niveau intermédiaire qui correspondent à des motifs plus complexes.

Enfin, une dernière couche de convolution permettant d'appliquer 32 filtres va extraire des caractéristiques de haut niveau comme des motifs abstraits de l'image. Toutes les couches de convolution sont suivies d'une fonction d'activation **ReLU** (Rectified Linear Unit) définie comme suit :

$$\text{ReLU}(x) = \max(0, x)$$

et renvoie 0 si x est inférieur ou égal à 0 et x si x est strictement positif. C'est une fonction généralement utilisée pour sa capacité à réaliser un apprentissage plus efficace de modèles profonds.

Le modèle termine avec 2 couches denses contenant respectivement 64 et 1 neurones : la première apprend les combinaisons non linéaires des caractéristiques extraites par les couches précédentes, et la deuxième va produire la probabilité pour la classification binaire. Sur cette dernière couche dense, la fonction **sigmoid** est utilisée pour convertir la sortie en une probabilité interprétable. Cette fonction est adaptée au problème, c'est-à-dire à une classification binaire.

Entre chaque couche de convolution (Figure 27), un pooling est réalisé afin de réduire la taille spatiale de la carte caractéristiques. Ceci va aider à diminuer la dimension des données et à conserver les informations les plus importante sur les images.

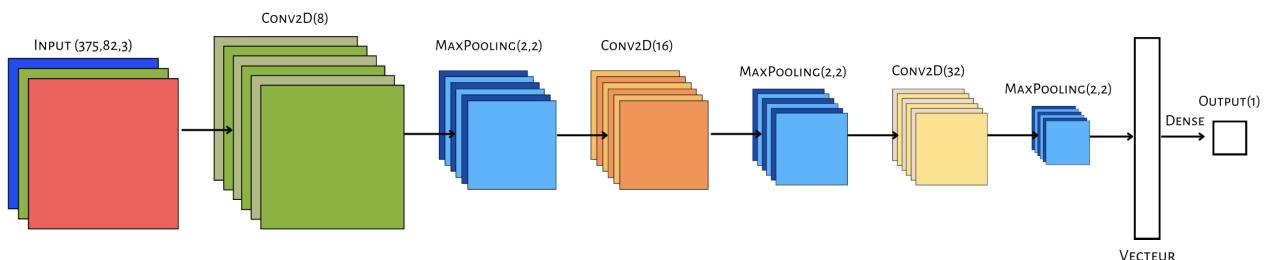


Figure 27 — Structure réseau de classification avec ajout des pooling

Il existe différents types de pooling, dans notre cas nous utilisons un **MaxPooling** qui, pour chaque région de la carte de caractéristiques va sélectionner la valeur maximale. L'avantage d'utiliser ce pooling est qu'il capture les informations les plus importantes dans chaque région ce qui va permettre de conserver les éléments significatifs de la carte. Cette partie du modèle encourage également la prévention du surapprentissage en réduisant le nombre de paramètres.

Avant d'entraîner notre jeu de données sur notre modèle, il faut d'abord configurer notre réseau de neurones en utilisant la commande suivante :

```
1 model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
```

Elle spécifie une fonction de perte **loss**, un optimiseur **optimizer** ainsi que les métriques **metrics** à utiliser. Nous avons choisi un **loss** de type '**binary_crossentropy**' qui est une fonction utilisée pour les classifications binaires. Elle va permettre de mesurer à quel point les prédictions du modèle s'écartent des étiquettes de départ de chaque image.

Cette fonction est définie comme suit :

$$\text{loss} = -\frac{1}{N} \sum_{i=1}^N [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)]$$

où y_i est l'étiquette de départ attribuée à l'image i et p_i est la probabilité prédictive pour cette image en sortie de notre CNN. Elle permet de réduire considérablement les prédictions incorrectes faites sur l'ensemble d'entraînement en attribuant un coût élevé aux prédictions trop éloignées de l'étiquette de départ.

Ensuite nous avons **optimizer** qui va permettre l'ajustement des poids du modèle afin de minimiser la fonction **loss** lors de l'entraînement. Nous choisissons l'optimiseur '**adam**' qui utilise des estimations adaptatives de premier et deuxième ordre des moments pour ajuster le taux d'apprentissage.

Pour finir, le paramètre **metrics** est utilisé pour évaluer les performances du modèle. À contrario de la fonction **loss**, elle ne va pas ajuster les poids du modèle mais plutôt fournir des informations sur la qualité de celui-ci. L'**accuracy** est une métrique courante et en particulier pour les classifications. Elle mesure le pourcentage de prédictions correctes parmi toutes les prédictions. Ce pourcentage est calculé concurremment sur l'ensemble d'entraînement et sur l'ensemble de test.

3.2 Résultats

3.2.1 Modèle et prédictions

Nous estimons la performance de notre réseau sur les données de test. Nous choisissons d'entraîner 70% du jeu de données et de garder les 30% restant pour le test. Puis, à partir d'un nouveau jeu de données, une validation ou non du modèle sera faite.

Afin de suivre l'apprentissage lors de l'entraînement ainsi que l'évaluation du modèle sur l'ensemble de test, nous pouvons observer le comportement du **loss** et celui de l'**accuracy** :

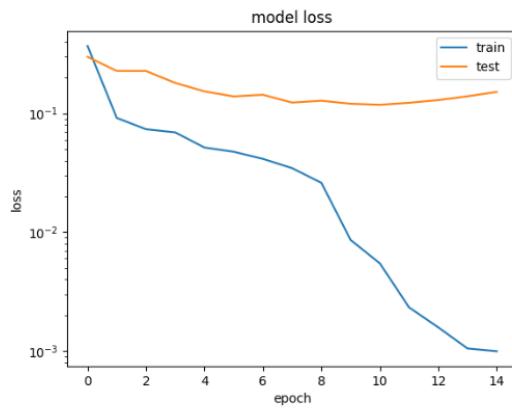


Figure 28 — Graphe du loss d'entraînement et du test

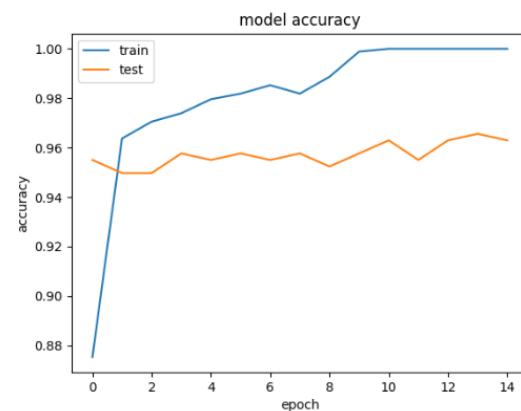


Figure 29 — Graphe de l'accuracy d'entraînement et du test

Nous pouvons constater que le **loss** de l'entraînement décroît rapidement et atteint les 10^{-3} au bout de 15 époques : ceci nous traduit une bonne convergence du modèle (Figure 28, courbe bleue). La pente de cette fonction au cours des époques nous démontre que le modèle est assez stable. L'**accuracy** de l'entraînement (Figure 29, courbe bleue), nous indique une tendance tout à fait classique : elle augmente puis sature à 1.

Le **loss** de l'ensemble du test diminue peu et remonte légèrement à partir de 9 époques tandis que la perte d'entraînement continue de diminuer, ce qui nous indique que l'entraînement conduit à un léger surapprentissage à partir de l'époque 9. Pour remédier à cela, il pourrait être bénéfique d'appliquer une régularisation plus forte ou de mettre en place un arrêt précoce pour éviter le surapprentissage.

L'accuracy du test reste autour de 95% et traduit donc de très bon résultats. Cependant il faut rester prudent sur ce résultat car une accuracy aussi élevée sur des classes équilibrées est un bon indicateur, mais dans notre cas nous avons deux classes assez déséquilibrées : nous avons environ 3 fois plus d'exemples de scalogrammes comprenant un ABR (284) plutôt que sans ABR (94). Ce taux peut-être donc trompeur et peut nous cacher la détection de faux positifs : ABR détecté alors qu'il est absent, ou faux négatifs : ABR non détecté alors qu'il est présent.

Afin de visualiser où le modèle se trompe le plus, nous pouvons afficher la matrice de confusion. Cela peut révéler des biais ou des difficultés spécifiques à certaines classes.

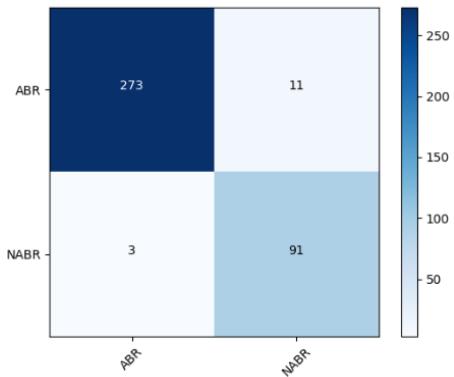


Figure 30 — Matrice de prédictions sur la partie test

La précision pour la classe ABR est très élevée avec un taux à 96,2%.
Pour la suite, nous noterons :

- TP(réel ABR, prédit ABR) = 273,
- FN(réel ABR, prédit NABR) = 11,
- FP(réel NABR, prédit ABR) = 3,
- TN(réel NABR, prédit NABR) = 91.

Ainsi, nous pouvons calculer la précision de chaque classe à partir des données ci-dessus : pour la classe ABR nous avons une précision de 98,9% et pour la classe NABR elle est de 89,2%.

Nous pouvons également calculer le rappel, qui est le rapport entre le nombre de prédictions correctes pour une classe et le nombre total des instances réelles de cette classe : 96,1% pour ABR et 96,8% pour NABR. Ce qui nous permettra par la suite de connaître le F1-score.

Le F1-score est une mesure statistique qui combine à la fois la précision et le rappel du modèle et est calculée de la manière suivante :

$$\text{F1-score} = \frac{TP}{TP + \frac{1}{2}(FN + FP)}$$

Il est utile lorsque les classes sont déséquilibrées, comme dans notre cas. Le F1-score de la classe ABR est de 97,5% et traduit une excellente performance pour cette classe.

Bien que la performance soit très bonne pour les deux classes, il peut être bénéfique de se concentrer sur l'amélioration de la précision pour la classe NABR, afin de réduire au mieux les erreurs de prédiction.

3.2.2 Validation

Nous arrivons maintenant à une étape importante qui est la validation du modèle à partir d'un nouveau jeu de données non connu du modèle. En effet, ce sont ces résultats qui pourront non seulement nous montrer la performance du modèle mais aussi être comparés aux autres méthodes.

La validation sera réalisée sur un petit jeu de 12 données et donc au total 108 scalogrammes. Ci-dessous nous retrouvons les boîtes à moustaches contenant toutes les méthodes appliquées sur ce jeu de données.

Premièrement, nous remarquons que les résultats des seuils ne descendent pas en dessous de 20 dB pour les méthodes visuel et CNN. Les méthodes scalogramme et corrélation croisée indiquent toujours des seuils en dessous des autres méthodes.

Les moyennes des méthodes visuel et CNN sont assez proches : une moyenne de 36 dB pour la méthode visuel et 40 dB pour la méthode CNN tandis que les deux autres méthodes ont des moyennes inférieures à 30 dB.

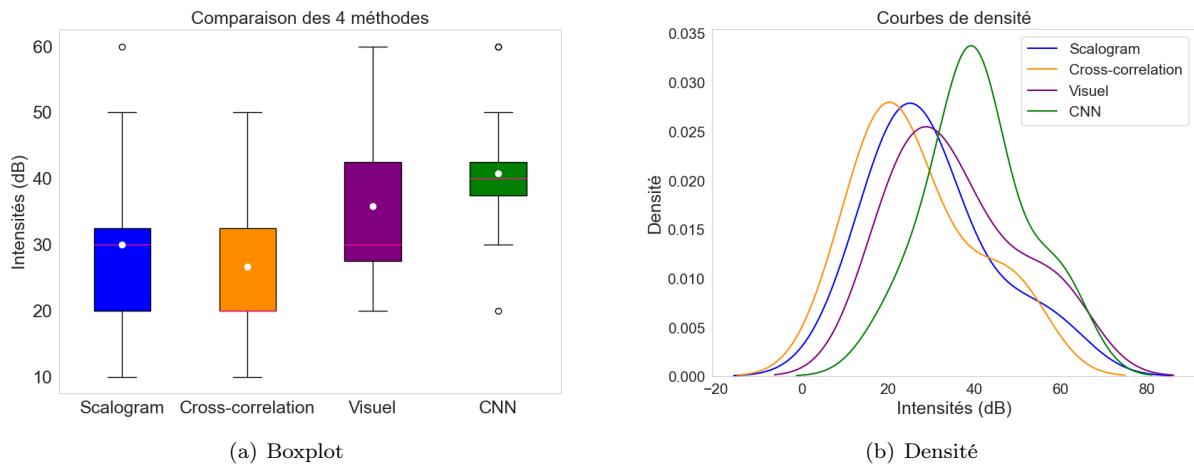


Figure 31 — Graphes de comparaison des 4 méthodes étudiées

La tendance des courbes de densité sont similaires avec chacune deux pics. Cependant, nous remarquons bien que la méthode CNN indique des seuils plus concentrés autour de 40 dB et donc légèrement plus haut que les autres méthodes qui ont plutôt des pics vers 20 ou 30 dB.

Ci-dessous, nous visualisons les 12 données afin de représenter plus en détails les données qui ressortent.

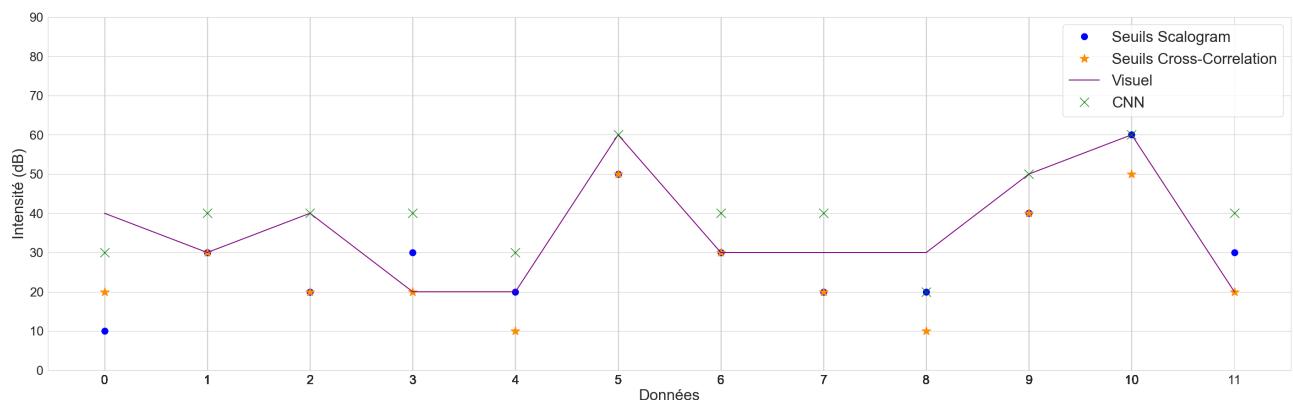


Figure 32 — Visualisation des seuils de chaque méthode pour chaque donnée

Nous voyons que la valeur aberrante à 60 dB qu'il y avait pour la méthode CNN sur la figure 31 (a) est en réalité cohérente avec celle donnée par le seuil visuel.

Après avoir visualisé les données en détail, nous constatons que l'automatisation, par les biais de ces méthodes, apporte un contrôle qualité post-évaluation visuelle et une analyse plus fine des données. Aussi, nous avons pu constater qu'il était important pour chaque donnée, d'appliquer les différentes méthodes afin de les analyser et d'adapter au mieux le seuil : certaines méthodes correspondent mieux à certaines données que pour d'autres.

4 Discussion

4.1 Contribution effective au projet

Dans le cadre de ce projet, j'ai pu réaliser différentes tâches techniques et analytiques. J'ai d'abord converti le code existant de corrélation croisée de MatLab vers Python, et approfondi l'étude de cette méthode en examinant l'impact des intervalles de confiance et en affinant l'estimation du seuil par interpolation.

Concernant la méthode des ondelettes, j'ai également effectué la conversion du code de MatLab vers Python, tout en élargissant la portée de l'analyse : en MatLab l'ondelette mère par défaut était utilisée, j'ai exploré différentes ondelettes ainsi que différents paramètres de fréquence centrale et de bande passante. J'ai identifié la nécessité de définir une nouvelle méthode, au-delà de l'évaluation visuelle, pour estimer et déterminer au mieux les paramètres. De plus, j'ai mis en place une nouvelle méthode d'estimation du seuil, autre que celle utilisée en MatLab (méthode des boxes), et j'ai examiné l'impact des intervalles de confiance en intégrant l'interpolation pour une estimation plus précise du seuil audiométrique.

J'ai comparé les performances des différentes méthodes sur un ensemble de données, incluant des données labellisées, ce qui n'avait pas été réalisé auparavant. Ceci a permis une comparaison complète, plus détaillée et plus pertinente. Cette analyse approfondie des trois méthodes sur un dataset plus large a permis de généraliser les résultats de manière plus fiable.

Enfin, j'ai implémenté une nouvelle approche basée sur un réseau de neurones convolutionnel (CNN). Pour ce faire, j'ai dû comprendre en profondeur les données, les organiser, les trier et explorer un premier modèle. J'ai pu interpréter, discuter et comparer ce modèle avec les trois autres méthodes. En complément, j'ai développé une interface graphique (Annexe A), améliorant ainsi la visualisation rapide des données.

4.2 Perspectives

Pour aller plus loin, plusieurs perspectives d'amélioration et d'approfondissement peuvent être envisagées. Dans un premier temps, la définition d'une métrique objective et la mise en place d'une procédure de benchmarking pour les méthodes utilisant des ondelettes permettraient de quantifier et de comparer de manière rigoureuse les performances des différents paramètres (fréquence centrale, bande passante, etc...), afin d'identifier les plus appropriés et les plus efficaces pour le problème.

En ce qui concerne l'approche par CNN, l'augmentation du dataset serait essentielle, en incluant différents types d'ABR : des ABR bruités, des ABR de souris malentendantes avec donc des seuils plus élevés, des ABR de souris présentant des pathologies auditives qui pourraient montrer des ondes légèrement différentes des ondes classiques et enfin à plus long terme, des ABR humains. Ainsi, nous pourrions améliorer significativement la robustesse et la généralisation du modèle. Ces ajouts permettraient également d'évaluer et d'améliorer également les autres méthodes offrant une grande diversité de cas d'étude.

Pour affiner le CNN, nous pourrions explorer et tester d'autres structures ce qui permettrait de comprendre encore mieux leur impact sur les performances et choisir des configurations adaptées aux caractéristiques spécifiques des ABR. Ces perspectives ouvrent des améliorations autant en termes de précision des méthodes actuelles que d'adaptation à une plus grande étendue de données.

4.3 Intérêt

L'intérêt de ce projet réside dans les contributions à l'automatisation et à la standardisation de l'évaluation des seuils auditifs au sein de l'Institut. En automatisant le processus, nous augmentons l'efficacité globale des analyses. Elle permet également de maintenir un contrôle de qualité post-évaluation visuelle en offrant une méthode reproductible pour vérifier les résultats par les expérimentateurs.

En utilisant des algorithmes pour analyser les ABR, nous pouvons minimiser les variations et assurer la cohérence des résultats à travers les différentes études et équipes. Cela est essentiel pour garantir la fiabilité et la comparabilité des données collectées. Et enfin, les algorithmes développés dans ce projet ne se limitent pas à l'évaluation des seuils : ils permettent également l'extraction d'autres informations physiologiques comme l'amplitude, la latence ou encore la visibilité des différentes ondes ABR ouvrant la voie à de nouvelles découvertes et applications cliniques.

ANNEXES

A Application : interface graphique

À l'occasion de cette étude, nous avons développé une application pour les utilisateurs. Elle va permettre de retourner le seuil auditif d'une donnée selon la méthode que l'on souhaite : corrélation croisée ou scalogramme.

Au lancement de l'application, nous choisissons une donnée en appuyant sur le bouton **Load File**, qui s'affichera dans l'onglet **GRAPH**.

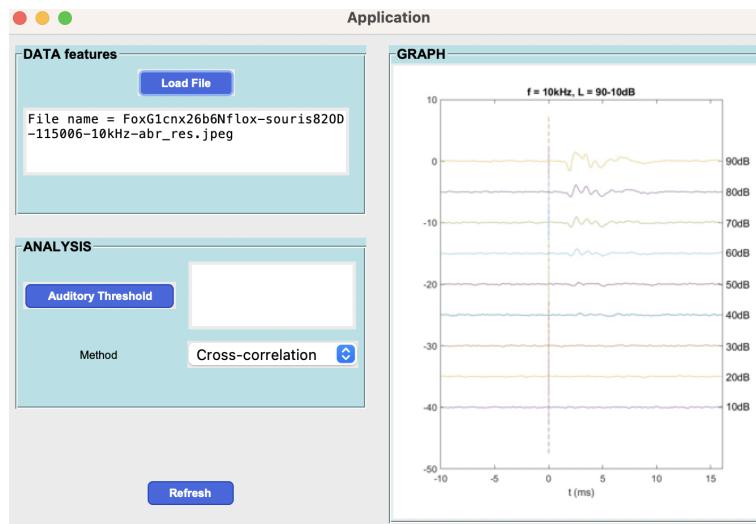


Figure 33 — Application

Par la suite, l'utilisateur va pouvoir choisir à l'aide d'une liste déroulante la méthode avec laquelle il souhaite utiliser pour obtenir le seuil.

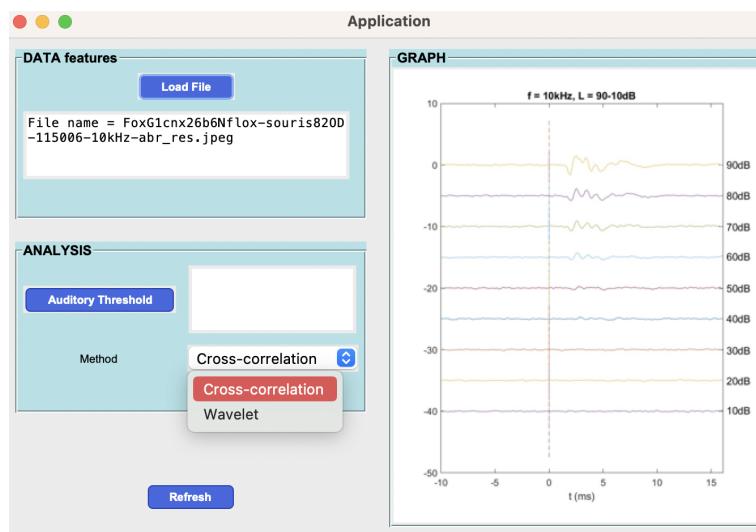


Figure 34 — Choix de la méthode

Lorsque la méthode *cross-correlation* est sélectionnée, nous pouvons afficher le seuil dans la zone de texte dédiée, ainsi que le graphe de la corrélation croisée dans une nouvelle fenêtre à l'aide du bouton **Auditory Threshold**.

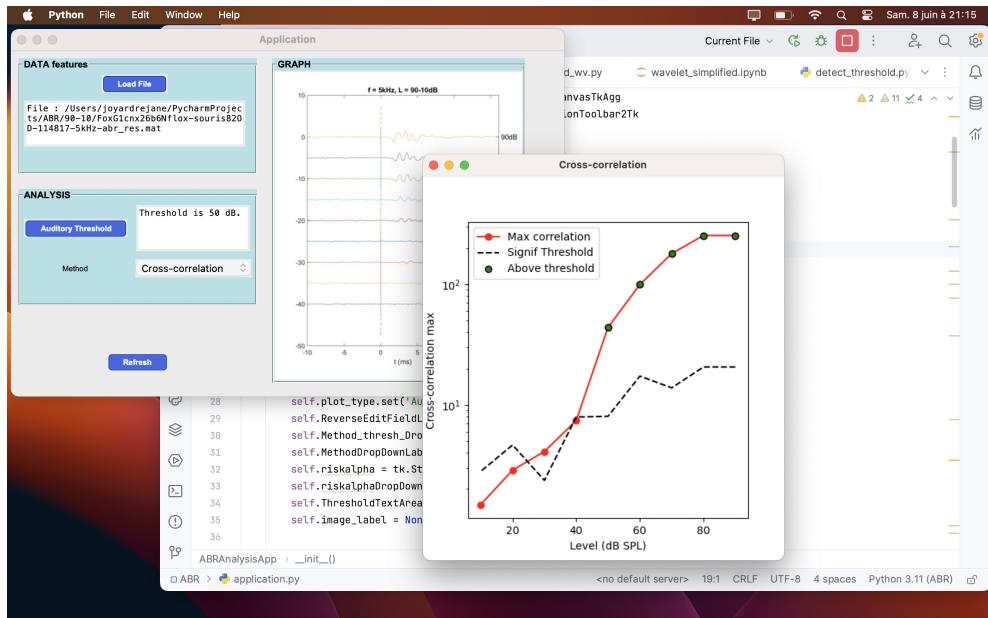


Figure 35 — Exemple : corrélation croisée

De la même manière lorsque la méthode *wavelet* est sélectionnée, le seuil sera affiché ainsi qu'une nouvelle fenêtre permettant de visualiser les scalogrammes associés à la donnée.

Les codes de ce projet sont regroupés sur https://github.com/rejanej/IP_auditory_threshold.

TABLE DES FIGURES

1	Système auditif	2
2	Schéma de la cochlée selon les fréquences (kHz)	2
3	Réponses de la membrane basilaire selon les fréquences	2
4	Réalisation des mesures d'ABR sur une souris	3
5	Système nerveux auditif ; morphologie d'un ABR	4
6	Enregistrements ABR d'une souris pour une fréquence de 40kHz et d'intensité variable de 90 à 20 dB	5
7	Distribution normale centrée réduite avec un intervalle de confiance de 99,999%	6
8	Graphe corrélation croisée avec différents seuils de significativité	7
9	Seuil exacte récupéré avec l'interpolation	7
10	Comparaison Transformée de Fourier/Transformée en ondelettes pour un signal (90 dB)	7
11	Morlet Wavelet	9
12	Complex Morlet Wavelet	9
13	Frequency B-Signal Wavelet	9
14	Mexican Hat Wavelet	9
15	Shannon Wavelet	10
16	Différents types d'ondelettes	10
17	Variations des paramètres bandwidths et center frequencies	10
18	Visualisation des ondelettes selon les variation bandwidths et center frequencies	11
19	Scalogramme (90 dB) avec placement des boxes	11
20	Croisement des moyennes	12
21	Comparaison des résultats avec $\alpha = 10^{-3}$	13
22	Comparaison des méthodes donnée par donnée $\alpha = 10^{-3}$	14
23	Comparaison des résultats avec $\alpha = 10^{-5}$	14
24	Comparaison des méthodes donnée par donnée $\alpha = 10^{-5}$	15
25	Donnée d'entrée du réseau de neurones : transformation de l'image vers un vecteur	16
26	Architecture du réseau de neurones	16
27	Structure réseau de classification avec ajout des pooling	17
28	Graphe du loss d'entraînement et du test	18
29	Graphe de l'accuracy d'entraînement et du test	18
30	Matrice de prédictions sur la partie test	19
31	Graphes de comparaison des 4 méthodes étudiées	20
32	Visualisation des seuils de chaque méthode pour chaque donnée	20
33	Application	22
34	Choix de la méthode	22
35	Exemple : corrélation croisée	23

RÉFÉRENCES

- [1] McKearney Richard (2023), *Improving Objective Analysis of the Auditory Brainstem Response.*
https://eprints.soton.ac.uk/476506/1/Final_Thesis_Richard_McKearney.pdf
- [2] Haoyu Wang, Bei Li, Yan Lu, Kun Han, Haibin Sheng, Jialei Zhou, Yumeng Qi, Xueling Wang, Zhiwu Huang, Lei Song and Yunfeng Hua (2021), *Real-time threshold determination of auditory.*
[https://www.cell.com/iscience/pdf/S2589-0042\(21\)01254-2.pdf](https://www.cell.com/iscience/pdf/S2589-0042(21)01254-2.pdf)
- [3] Alexander J.E. Kell, Daniel L.K. Yamins, Erica N. Shook, Sam V. Norman-Haignere, Josh H. McDermott (2018), *A Task-Optimized Neural Network Replicates Human Auditory Behavior, Predicts Brain Responses, and Reveals a Cortical Processing Hierarchy.*
<https://www.sciencedirect.com/science/article/pii/S0896627318302502?via%3Dihub>
- [4] Christopher Torrence and Gilbert P. Compo (1998), *A Practical Guide to Wavelet Analysis.*
https://journals.ametsoc.org/view/journals/bams/79/1/1520-0477_1998_079_0061_apgtwa_2_0_co_2.xml?tab_body=pdf
- [5] Nurettin Acir, Özcan Özdamar, Cüneyt Güzlis (2005), *Automatic classification of auditory brainstem responses using SVM-based feature selection algorithm for threshold detection.*
<https://www.sciencedirect.com/science/article/pii/S0952197605001028>
- [6] M. A. Chesnaye, S. L. Bell, J. M. Harte and D. M. Simpson (2018), *Objective measures for detecting the auditory brainstem response: comparisons of specificity, sensitivity and detection time.*
<https://www.tandfonline.com/doi/full/10.1080/14992027.2018.1447697?needAccess=true>
- [7] Wikipédia, *Cross-correlation.*
<https://en.wikipedia.org/wiki/Cross-correlation>
- [8] Wikipédia, *Wavelet Transform*
https://en.wikipedia.org/wiki/Wavelet_transform
- [9] Stéphane Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, 3rd edition, 2009.
<https://www.sciencedirect.com/book/9780123743701/a-wavelet-tour-of-signal-processing>
- [10] François Cholet, *Deep Learning with Python*, Manning, 2nd edition, 2018.
<https://www.manning.com/books/deep-learning-with-python>