COMPTE RENDU

PROJET THS/IA/JAVA
N3

COLARD JULIEN
COSSU ALEXANDRE
MOREL EVA
PHILIPPE ANTHONY
VILLENEUVE LÉO

ISEN
ALL IS DIGITAL!
MÉDITERRANÉE

yncréa

INFORMATIONS

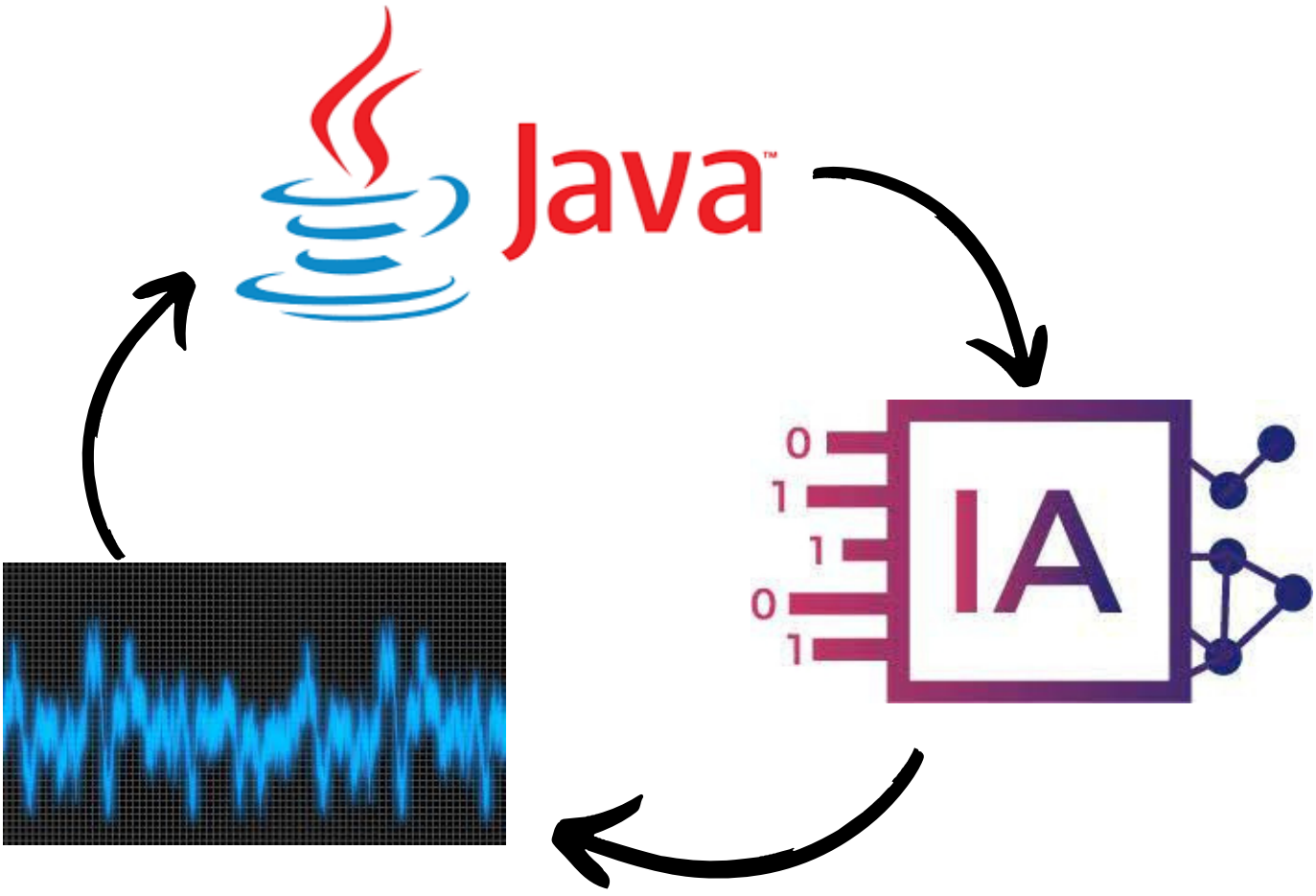
Sujet	Reconnaissance de chants d'oiseaux
Type de document	Compte rendu
Date	06/06/2023 au 07/06/2023
Auteurs	COLARD Julien COSSU Alexandre MOREL Eva PHILIPPE Anthony VILLENEUVE Léo

TABLE DES MATIERES

1 - INTRODUCTION

Ce compte rendu fait état de notre projet pluridisciplinaire de fin de semestre, mêlant **THS**, **IA** et Java. Il porte sur l'applications des connaissances en transformée de Fourier, neurones et réseaux de neurones, ainsi que programmation orientée objet en Java.

Ce projet vise à explorer les liens entre ces différents domaines et à mettre en pratique les notions théoriques acquises au travers des cours.



2- SUJET

La **reconnaissance des chants d'oiseaux** est un domaine d'étude qui associe la science du son, les techniques de traitements de signal et l'intelligence artificielle.

L'objectif principal de ce projet est de **modéliser et de formaliser les données** d'entrée des chants d'oiseaux après leur transformation par la **FFT** (**F**ast **F**ourier **T**ransformation). En utilisant ces données prétraitées, nous chercherons à développer une **architecture de réseau de neurones** capable de reconnaitre les signatures sonores de chaque espèces.

Par la suite, on entrainera le réseau de neurones à reconnaitre les chants d'oiseaux par la mise en place de l'**algorithme d'apprentissage par correction d'erreur**. Nous examinerons les performances du réseau en modifiant la vitesse ainsi que le choix des échantillons d'apprentissage. Cette analyse nous permettra de comprendre **comment ces paramètres influencent** la capacité du réseau à reconnaître avec précision les chants des espèces.

Nous appliquerons également **nos connaissances** en programmation orientée objet dans le langage Java en mettant en oeuvre les différents algorithmes nécessaires pour le traitement du signal, la création et l'entrainement du réseau de neurones, ainsi que l'analyse des résultats.

Enfin, ce rapport sera le témoin de la **démarche scientifique** que nous adopterons tout au long du projet par la rédaction de rapports détaillés sur nos expérimentations, nos observations et nos conclusions.

3 - CAHIER DES CHARGES

A partir de l'analyse du sujet, nous avons noté des points remarquables auxquels nous allons porté une attention particulière.
Ainsi, cela nous permettra de vérifier tout au long de notre travail si ils sont correctement effectués.

I) Exigences fonctionnelles

Le système de reconnaissance devra respecter les exigences suivantes :

1. Prétraitement des données

- **Afficher la transformée de Fourier** sur les enregistrements audio des chants d'oiseaux pour obtenir une représentations fréquentielles
- **Modéliser** et **formaliser** les données d'entrées

2. Modélisation du réseau de neurones

- Concevoir et mettre en place une **architecture de neurones** complétement connecté capable de reconnaitre les chants d'oiseaux

3. Apprentissage par correction d'erreur

- **Implémenter l'algorithme** d'apprentissage par correction d'erreur pour entrainer le réseau
- Utiliser des données étiquetées pour l'apprentissage supervisé et **ajuster les poids synaptiques** en fonction de l'erreur

4. Analyse et évaluation du système

- **Evaluer les performances du système** en utilisant un ensemble de données de test distinct de l'ensemble d'apprentissage
- **Analyser les résultats** en matière de précision de reconnaissance, de taux de faux positifs et de faux négatifs

II) Exigences non-fonctionnelles

1. Performance

- Capable de **traiter efficacement des enregistrements audio** de chants d'oiseaux de différentes durées et qualités
- **Temps de calcul** doivent être **raisonnables** pour permettre une reconnaissance quasi instantanée

2. Convivialité

- Fonctionnalité permettant la **visualisation des résultats** de reconnaissances
- Le code doit être **clair et compréhensible**. De plus, il devra être **annoté** de commentaires.

4 - OUTILS UTILISÉS

Nous avons utilisé plusieurs outils ainsi que les documents fournis pour mener à bien ce projet.

Notamment des **cours sur la Transformation de Fourier rapide** et **les réseaux de neurones**. La FFT est essentielle dans le traitement des signaux audio, car elle permet d'analyser les fréquences présentes dans un signal. Quant aux réseaux de neurones, ils sont utilisés pour la **classification de données**, y compris dans le domaine de la reconnaissance d'objets. L'assemblement de ces 2 outils nous a été utile dans le cadre de notre projet qui consiste à **reconnaitre les sons**.

Secondement, un **algorithme de FFT en Java** nous a été fourni. Cet algorithme nous a permis d'**analyser les fichiers audios** que nous avons utilisés pour notre projet. Également une archive était fourni composé de divers **enregistrements de chants d'oiseaux**. Ces fichiers audios ont été utilisés comme **jeu de données** pour entraîner et tester notre modèle de **reconnaissance d'oiseaux**.

Quant aux outils que nous avons utilisé pour l'utilisation de l'algorithme de FFT et pour le développement de la **fonction d'apprentissage**. Nous avons choisi d'utiliser **IntelliJ IDEA**, qui nous a été introduit cette année en **cours de WEBDEV** ainsi qu'en **Kotlin**, nous maitrisions donc le mieux ce logiciel. Le langage **JAVA** était imposé, cependant il nous avait été enseigné durant l'année, nous avions donc toutes les connaissances nécessaires pour mener à terme le projet.

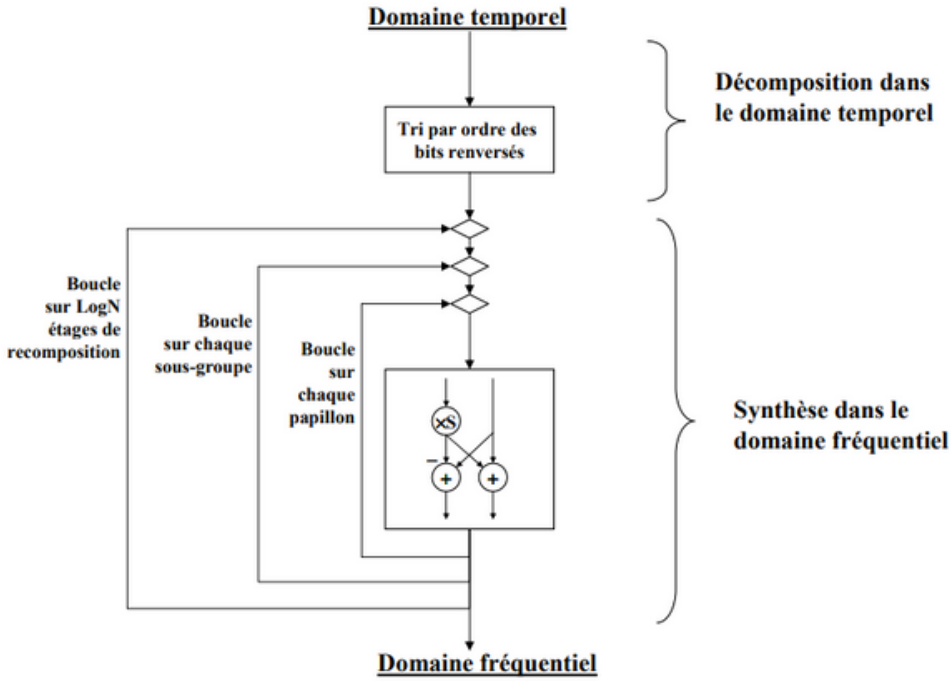
5 - RELATION DE LA FFT

Nous avons examiné la **relation entre l'entrée et la sortie** de la FFT, ainsi que les conséquences de la **variation de la fréquence** du signal d'entrée.

La FFT est un algorithme utilisé pour **convertir un signal** du domaine **temporel** au domaine **fréquentiel**. Il est largement utilisé dans le traitement du signal pour **analyser les composantes fréquentielles** d'un signal donné.

La classe **FFTCplx**:

Elle fournie implémente la FFT pour des signaux composés de **nombres complexes**. Elle utilise une approche **récursive** pour diviser le signal en **sous-groupes** et applique ensuite le **regroupement "papillon"** pour obtenir le résultat.



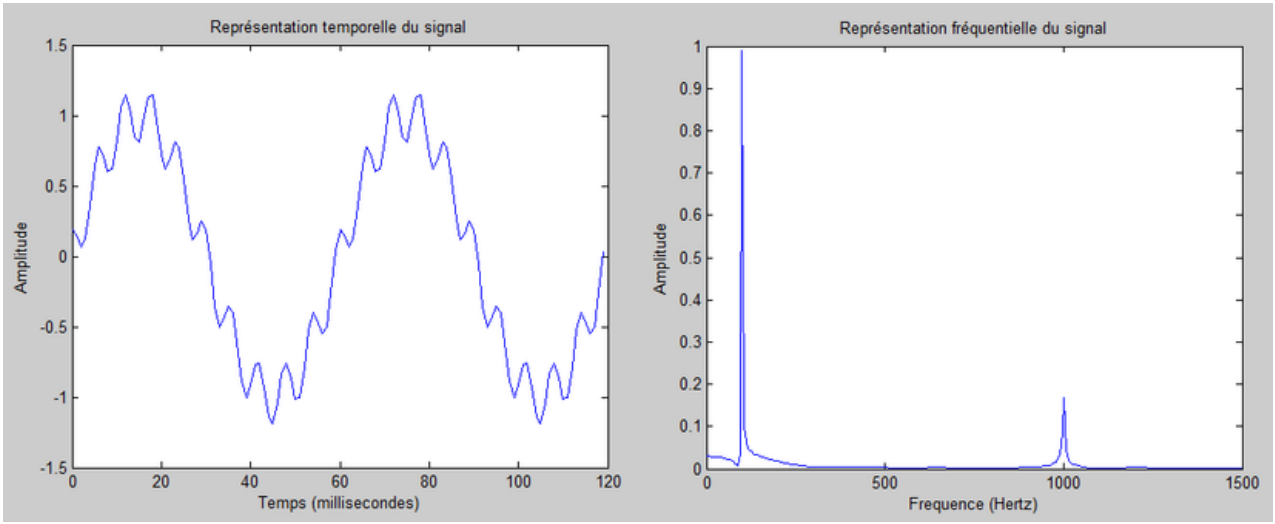
Avec l'exemple de test fourni dans la méthode main, qui utilise un signal de test simple généré à partir d'une **fonction cosinus**, où la taille du **signal** et la **période** du signal peuvent être configurées.

Nous obtenons les valeurs **réelles**, **imaginaires**, **modules** et **arguments** de chaque composante du résultat. Nous pouvons observer que la sortie représente les **différentes composantes fréquentielles** du signal d'entrée.

L'effet de la **variation de la fréquence** du signal d'entrée:

Elle est lié au nombre de périodes du signal générées par le code. En augmentant le nombre de périodes, la fréquence du signal d'entrée augmente, ce qui se traduit par une plus **grande densité de composantes fréquentielles** en sortie. Et inversement en diminiuant le nombre de périodes.

Nous avons remarqué que la résolution fréquentielle **dépend** également de la **taille du signal d'entrée**. Plus elle est grande, plus la résolution fréquentielle est fine, ce qui signifie que des composantes fréquentielles plus proches **peuvent être distinguées dans la sortie** de la FFT.



En conclusion, la FFT permet d'analyser les composantes fréquentielles d'un signal en le transformant du domaine **temporel** au domaine **fréquentiel**. La relation entre l'entrée et la sortie est telle que chaque composante fréquentielle du signal d'entrée est représentée par un pic dans la sortie.

La variation de la fréquence du signal d'entrée a un impact direct sur le **nombre de pics fréquentiels** dans la sortie de la FFT. Une fréquence plus élevée entraîne plus de pics, tandis qu'une fréquence plus faible entraîne moins de pics. La taille du signal d'entrée influe également sur la résolution fréquentielle de la FFT.

4 - DÉMARCHE SCIENTIFIQUE

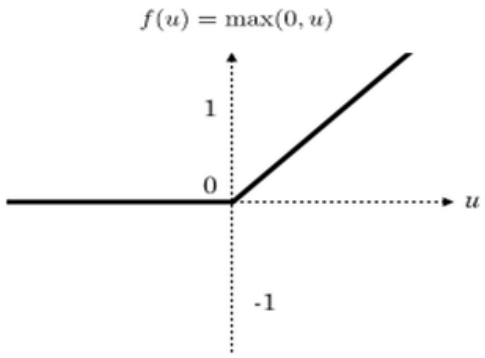
I) Expérimentation

L'expérimentation joue un rôle **crucial** dans la démarche scientifique. Dans notre cas, elle consiste à **étudier** la reconnaissance de **chants d'oiseaux** en utilisant la **FFT** (Transformée de Fourier rapide) et les **réseaux de neurones.** Nous allons appliquer la FFT sur les enregistrements sonores de chants d'oiseaux pour **analyser** leur **composition fréquentielle.**

1. Fonction d'activation

La fonction d'activation est un **élément clé** des réseaux de neurones. Elle **détermine** la **sortie d'un neurone** en fonction de la **somme pondérée** de ses **entrées**. Dans notre étude, nous allons **explorer** différentes fonctions d'activation utilisées dans les réseaux de neurones pour la **reconnaissance** de chants d'oiseaux. Nous aborderons des fonctions telles que **ReLU**, **Heavyside** et **Sigmoïde** et analyserons leur **impact** sur les performances du réseau.

ReLU - Rectified Linear Unit



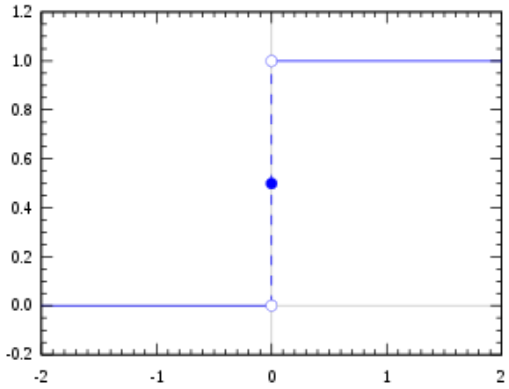

```java
public class NeuroneReLU extends Neurone {
    public NeuroneReLU(int nbEntrees) {
        super(nbEntrees);
    }
    @Override
    protected float activation(float valeur) {

        float valeurRetour=0;
        if (valeur<0){
            valeurRetour=0;
        }else if (valeur>=0){
            valeurRetour=valeur;
        }
        return valeurRetour;
    }
}
```

C'est une fonction d'activation couramment utilisée dans les réseaux neuronaux et les modèles d'apprentissage profond. Elle consiste en une fonction mathématique simple qui induit la non-linéarité.

Heavyside

H(x) =
{ 0, si x < 0
1, si x ≥ 0 }



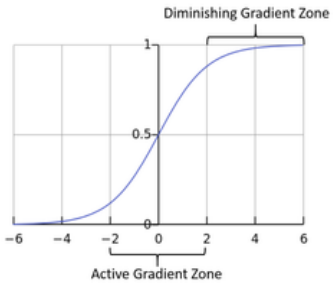

```java
public class NeuroneHeavyside extends Neurone {
    // Fonction d'activation d'un neurone (peut facilement être modifiée par héritage)
    protected float activation(final float valeur) {return valeur >= 0 ? 1.f : 0.f;}

    // Constructeur
    public NeuroneHeavyside(final int nbEntrees) {super(nbEntrees);}
}
```

Egalement appelée fonction d'échelon unitaire, c'est une fonction mathématique qui prend une valeur réelle en entrée et renvoie une valeur binaire en sortie.

Sigmoide

$$A = \frac{1}{1+e^{-x}}$$




```java
public class NeuroneSigmoide extends Neurone {

    public NeuroneSigmoide(int nbEntrees) { super(nbEntrees); }
    @Override

    protected float activation(float valeur){
        double valeurDouble = 1/(1+Math.exp(-valeur));
        float valeurFloat = (float)valeurDouble;
        return valeurFloat;
    }
}
```

Egalement appelée fonction logistique, c'est une fonction d'activation couramment utilisée dans les réseaux de neurones et l'apprentissage automatique qui prend une valeur réelle et renvoie une valeur entre 0 et 1, représentant une probabilité. Elle est souvent utilisée pour modéliser des relations de croissance ou de décroissance limitées .

2. Fonction d'apprentissage

La fonction d'apprentissage est responsable de l'ajustement des poids et des biais du réseau de neurones pour qu'il puisse reconnaître les signatures sonores spécifiques des chants d'oiseaux. Nous étudierons les algorithmes d'apprentissage par correction d'erreur, tels que la rétropropagation, et leur application à notre problème de reconnaissance de chants d'oiseaux. Nous examinerons également l'importance du choix des échantillons d'apprentissage et de la vitesse d'apprentissage dans le comportement du réseau.

```java
public int apprentissage(final float[][] entrees, final float[] resultats) {
    int compteurEchecs = 0;
    boolean apprentissageFini = false;
    while (!apprentissageFini) {
        apprentissageFini = true;
        for (int i = 0; i < entrees.length; i++) {
            metAJour(entrees[i]);
            float erreur = resultats[i] - sortie();
            if (Math.abs(erreur) > ToleranceSortie) {
                for (int j = 0; j < synapses().length; j++) {
                    synapses()[j] += eta * erreur * entrees[i][j];
                }
                fixeBiais( nouveauBiais: biais() + eta * erreur);
                apprentissageFini = false;
                compteurEchecs++;
            }
        }
    }
    return compteurEchecs;
}
```

3. Nos tests

Pour évaluer les performances de notre approche, nous avons effectué divers tests. Nous avons utilisé des enregistrements sonores de chants d'oiseaux préalablement étiquetés pour entraîner nos réseaux de neurones. Nous avons analysé le comportement du réseau en modifiant la vitesse d'apprentissage et en utilisant différents échantillons d'apprentissage.

II) Nos résultats

Nos expérimentations ont donné des résultats **prometteurs**. Les réseaux de neurones entraînés avec la combinaison de la **FFT**, des **fonctions d'activation** et **d'apprentissage** ont montré une efficacité à reconnaître les chants d'oiseaux. Les performances varient en fonction des **paramètres** choisis, tels que la **vitesse d'apprentissage** et les échantillons d'apprentissage.

```
Synapses : [0.98173606, 0.98051405]
Biais : -0.971853
Entree 0 : 0.0
Entree 1 : 0.008661032
Entree 2 : 0.009883046
Entree 3 : 0.9903971
```

Résultats fonction ReLU

Résultats fonction Heavyside

```
Synapses : [0.23110127, 0.10590624]
Biais : -0.29763472
Entree 0 : 0.0
Entree 1 : 0.0
Entree 2 : 0.0
Entree 3 : 1.0
```

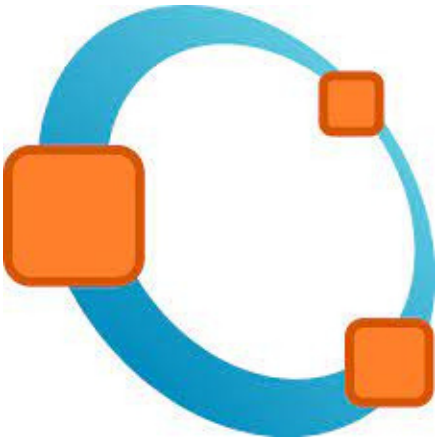
```
Synapses : [9.191556, 9.191341]
Biais : -13.787668
Entree 0 : 1.0282325E-6
Entree 1 : 0.009988058
Entree 2 : 0.00999018
Entree 3 : 0.9900011
```

Résultats fonction Sigmoïde

III) Conclusion

La démarche scientifique est essentielle pour étudier et résoudre des **problèmes complexes** tels que la reconnaissance de chants d'oiseaux. L'utilisation de la **FFT** et des **réseaux de neurones** offre une approche prometteuse pour cette tâche. Les fonctions **d'activation** et **d'apprentissage** jouent un rôle crucial dans les performances du réseau. Nos résultats montrent que l'ajustement de ces paramètres peut **améliorer la précision** de la reconnaissance. Cependant, des recherches supplémentaires sont nécessaires pour **optimiser** davantage les performances du système.

6 - OCTAVE

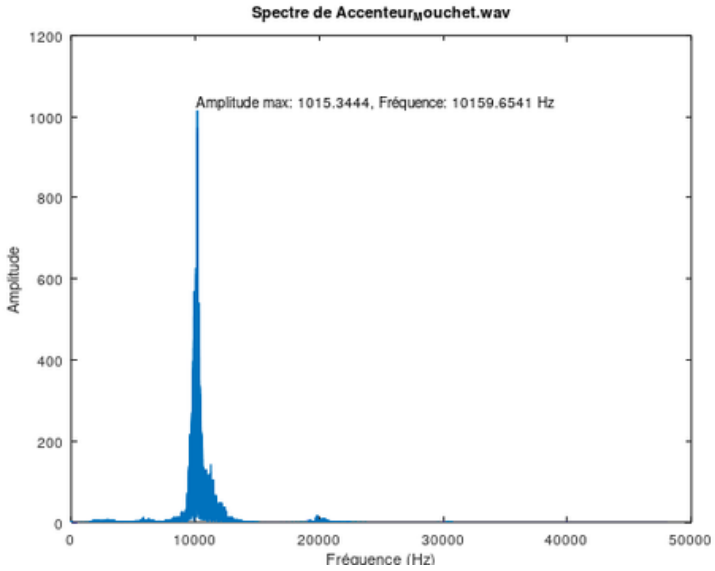


Pour aller plus loin dans notre projet, nous avons choisi d'utiliser un outil dont nous avons appris l'utilisation en cours de Traitement du Signal, le logiciel **Octave**.

Nous avons décidé de l'impliqué dans l'**analyse des signaux audio** notamment en analysant l'**amplitude** et la **fréquence** du champs d'oiseaux différents. Nous avons ensuite comparé les résultats entre-eux afin de voir des **différences de spectre**.

(Programme Octave fournis dans le dossier du projet)

I) Analyse des spectres

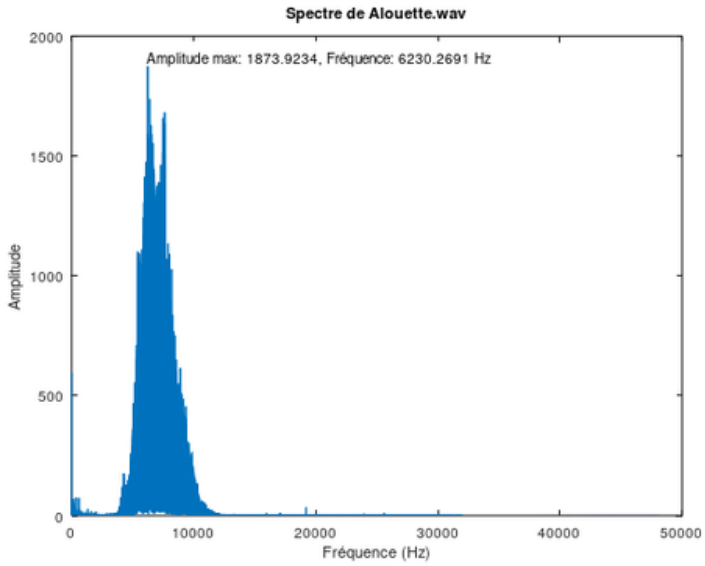


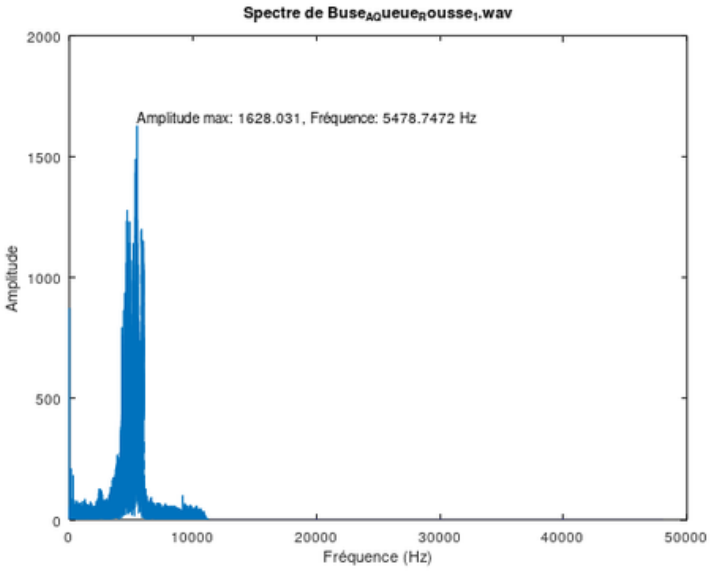
<u>Accenteur Mouchet :</u>

- Fréquence : **10 159** Hz
- Amplitude : **1015**
- Tonalité : **Elevée**

Alouette :

- Fréquence : **6 920** Hz
- Amplitude : **1873**
- Tonalité : **Moyenne**





Buse à queue rousse :

- Fréquence : **5 478** Hz
- Amplitude : **1629**
- Tonalité : **Basse**

Voici le classement des différents sons en fonction de leur tonalité, du plus aigu au plus grave:

Accenteur Mouchet > Alouette > Buse à Queue Rousse

On peut alors se demander pourquoi nous avons ce classement :

La fréquence d'un son est directement liée à sa tonalité perçue. Plus la fréquence est élevée, plus la tonalité est aiguë, et inversement, plus la fréquence est basse, plus la tonalité est grave.

7 - CONCLUSION

Ce projet de reconnaissance de chants d'oiseaux nous a permis d'approfondir nos connaissances en matière de traitement de signal, d'apprentissage automatique en intelligence artificielle et en programmation orientée objet.

Il nous a également sensibilisé à la démarche scientifique.

En plus des aspects techniques du projet, le travail d'équipe représente une caractéristique primordiale pour mener à bien un projet de recherche et de développement. Ainsi, l'engagement et la collaboration de plus ou moins chaque membre de l'équipe ont permis de réaliser les différentes tâches. Tout au long de notre travail, la plupart de l'équipe ont entretenu une communication ouverte sur l'accomplissement de leur avancée, proportionnellement à la charge de travail fournie.
Cela a favorisé les échanges d'idées, la résolution de problèmes et la prise de décision, nous rendant ainsi plus efficace.

Enfin, dans un soucis d'amélioration de chacun, nous avons suivi le processus d'évaluation 360 qui consistait à noter chaque membre de l'équipe selon son implication, sa prise de décision et sa communication avec autrui.

Nous sommes fiers des résultats obtenus. Nous pouvons aussi imaginer qu'un tel travail peut servir de base à un logiciel permettant à mieux comprendre et à protéger notre environnement naturel, ainsi que sa faune ou sa flore.

ISEN
ALL IS DIGITAL!
MÉDITERRANÉE

yncréa