# M2 MALIA Deep learning
# rendu n°2 : cas d'étude

Julien Velcin

2024-2025

Cette étude consiste à réaliser un outil de classification basé sur un algorithme basé sur les réseaux de neurones artificiels. Il ne s'agit pas nécessairement de proposer une architecture complexe, mais de montrer que vous êtes capable de développer toute la méthodologie nécessaire pour résoudre le problème de manière convaincante.

## Partie 1 : Jeu de données

**1.1** Il s'agit d'un extrait du jeu de données Spotify mis à disposition sur Kaggle https://www.kaggle.com/code/jgabrielsb/spotify-songs-music-genre-predictor-part-i/notebook. La tâche principale traitée consiste à identifier le genre musical d'un morceau à partir de ses caractéristiques. **Attention :** Vous devez travailler sur l'extrait qui vous est fourni et non sur les données en ligne. Par exemple, il ne contient que 5 des 6 classes initiales, la 6ème étant fournie à part pour les questions ultérieures.

**1.2** Réalisez quelques statistiques descriptives simples sur vos données afin de mieux les appréhender. Le choix de ce que vous souhaitez montrer, et les méthodes employées, sont laissés à votre appréciation.

**1.3** En utilisant uniquement les données quantitatives, testez quelques algorithmes standards de classification (par ex. SVM, arbres de décision, etc.) afin d'établir une base de comparaison.

## Partie 2 : Développement d'une architecture de classification sur la base des caractéristiques sonores

**2.1** Réfléchissez à une architecture de réseau de neurones artificiels. Celle-ci n'a pas besoin d'être très complexe, du moins pour commencer. Précisez clairement quelle est votre fonction d'erreur et comment vous allez évaluer les résultats obtenus.

**2.2** Réalisez vos premières expérimentations en faisant varier votre solution de différentes manières :

- taille des *batchs*,
- nombre de couches et nombre de neurones par couche,
- algorithme d'optimisation (par ex. contrôle du pas d'apprentissage),

Veillez bien à optimiser ces hyper-paramètres à l'aide d'un jeu de validation. Pensez aussi à monitorer les courbes d'apprentissage afin de faire les meilleurs choix.

**2.3** Ajoutez des heuristiques pour contrôler l'apprentissage des poids : batch normalization, gradient clipping, dropout...

**2.4** Finalement, vous devez utiliser votre meilleur modèle pour prédire le genre des morceaux fournis dans le fichier de test *test.csv*. La prédiction doit être ajoutée dans une colonne `playlist_genre` et le tableau exporté dans une fichier format `.csv` avec votre nom : `NOM_prediction_base.csv`.

## Partie 3 : Enrichissement des données à l'aide des champs textuels (option 1)

**3.1** Récupérez les informations textuelles (par ex. titre du morceau) et transformez-les en vecteurs numériques à l'aide d'encodeurs adaptés (par ex. dérivés de sentenceBERT).

**3.2** Développez et testez une architecture de réseau simple pour résoudre la tâche de classification uniquement sur la base de ces informations sémantiques.

**3.3** Essayez de combiner les deux types d'information (numérique et textuelle) pour voir si vous pouvez obtenir de meilleurs résultats.

**3.4** Vous pouvez réaliser de nouvelles prédictions et les exporter dans un fichier `NOM_prediction_full.csv`.

## Partie 4 : Vers une évolution du modèle (option 2)

Les poids qui ont été appris sur les 5 genres de musique peuvent peut-être être utilisés (transférés) pour ajouter un 6ème genre sans avoir à tout ré-entraîner depuis le départ.

**4.1** Chargez le nouveau jeu de données `data_EDM_full.csv` et modifiez votre architecture de réseau pour traiter le nouveau problème de classification.

**4.2** Entraînez le modèle depuis le départ afin de créer une base de comparaison.

**4.3** Créez une copie du modèle mais, cette fois, copiez les poids de l'ancien réseau dans les matrices de poids du nouveau. Vous pouvez vous contenter des premières couches (qui ont les mêmes dimensions) en utilisant la fonction `load_state_dict`.

**4.4** Continuez ensuite l'entraînement du modèle avec les nouveaux exemples. Comparez et commentez les résultats.

## Partie 5 : Rédaction de vos résultats

En adoptant la forme d'un article scientifique (limité à 8 pages dans le format Springer: https://preview.springer.com/gp/livingreviews/latex-templates), vous écrirez un rapport qui relate le déroulement de votre étude :

**5.1** une introduction qui présente la problématique en motivant le choix du jeu de données et donne un plan du travail réalisé;

**5.2** une section qui détaille le jeu de données et la tâche de classification, en donnant s'ils existent, les résultats précédemment obtenus;

**5.3** une section qui décrit votre architecture, avec les variantes envisagées, ainsi que votre protocole d'apprentissage et d'évaluation;

**5.4** une section qui présente les résultats de manière synthétique, par l'emploi de tableaux et de figures, en donnant des éléments d'appréciation sur la qualité de l'apprentissage (convergence, sur/sous apprentissage, etc.);

**5.5** une conclusion qui résume l'étude réalisée.

## Partie 6 : Votre rendu

Vous devez rendre deux éléments :

1. le rapport sous la forme d'un article de maximum 8 pages (format Springer)
2. le code que vous avez utilisé, abondamment commenté
3. le(s) fichier(s) contenant vos prédictions au format demandé

L'utilisation de ChatGPT, ou autre technologie approchant, est toléré à condition que vous indiquiez clairement les endroits où vous l'avez utilisé et comment.