ARN - Laboratoire 3

Apprentissage par réseau de neurones

Francesco Monti

12.04.2022

Contents

Introduction

Dans ce laboratoire on va explorer la manière d'apprendre d'un réseau de neurones. On va utiliser un dataset composé d'extraits de voix humaines et synthétisées, appartenant à des femmes, des hommes et des enfants de 3, 5 et 7 ans. Les extraits sont des voyelles prononcées par ces individus.

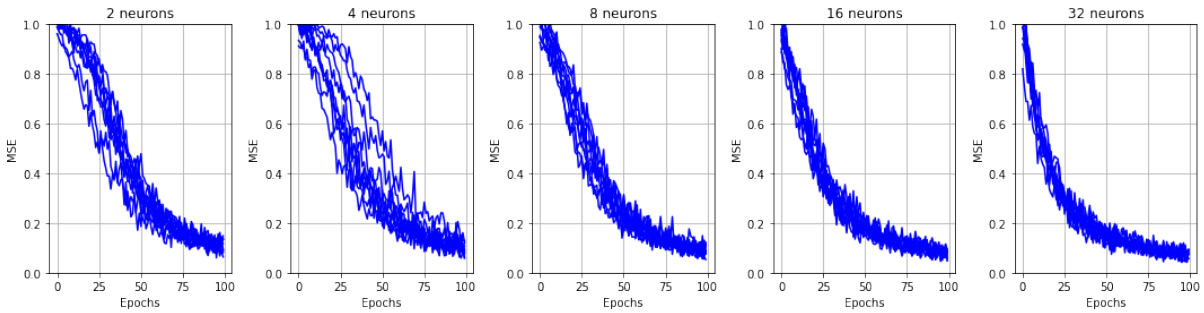
Le but va être de séléctionner 3 modèles permettant de catégoriser ces voix selon une métrique spécifique, les MFCCs. Celles-ci définissent 13 paramètres qui composent une voix et qui sont assez caractéristiques pour les différents types d'individus.

Hommes vs Femmes

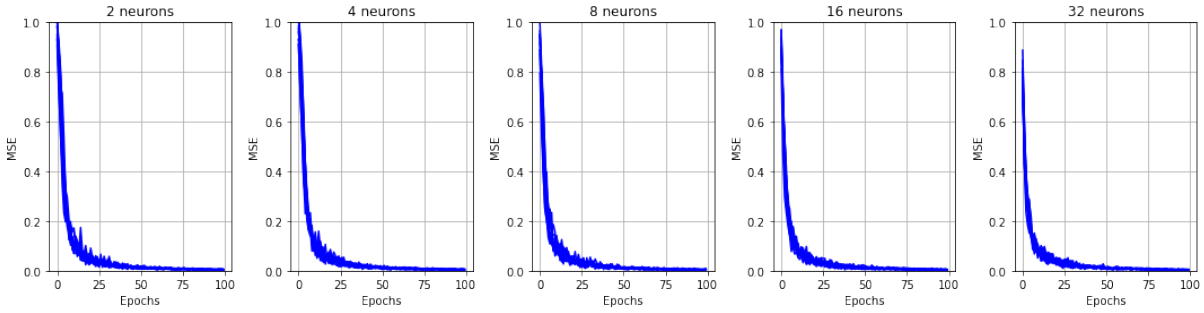
Le dataset est composé de 36 voix d'hommes et de 36 voix de femmes, ce qui le rend donc équilibré. On définit un seul neurone de sortie, qui devrait valoir -1 si c'est une femme et 1 si c'est un homme.

Analyse exploratoire

On commence par regarde avec un learning rate de *0.001* et un momentum de *0.5*, ce qui nous donne une analyse exploratoire suivante :



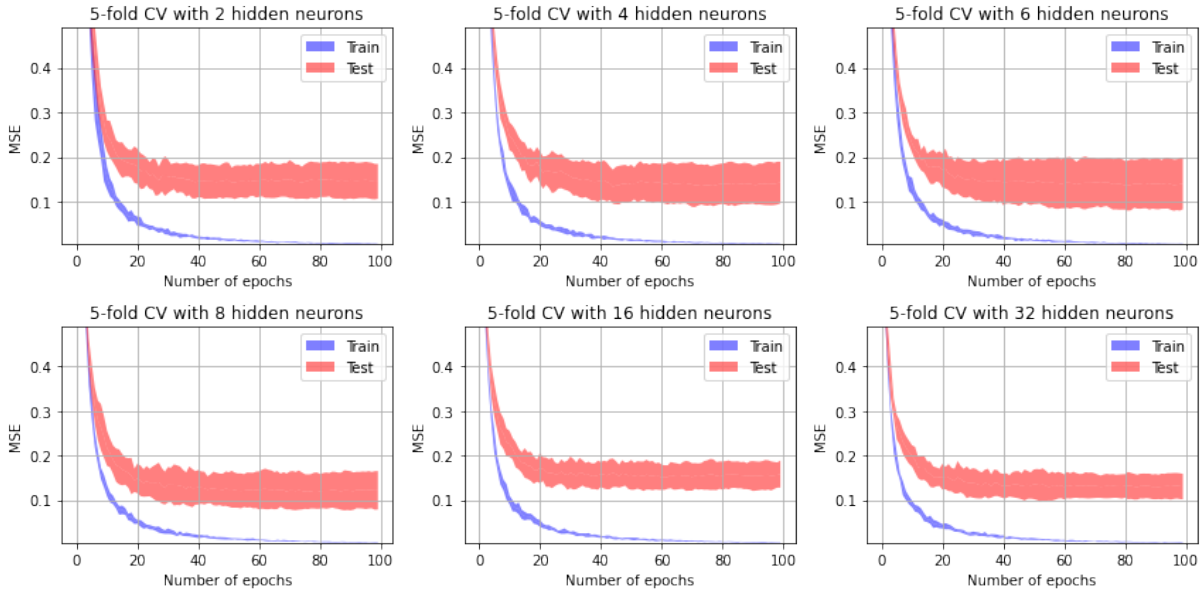
Ce n'est pas très glorieux, on va donc essayer plusieurs valeur jusqu'à avoir quelquechose de plus décent. Après de multiples essais, on arrive à la conclusion que avec un learning rate de *0.01* et un momentum de *0.55* on obtient de meilleurs résultats :



On remarque que après 50 itérations l'erreur ne diminue pas tant que ça, quelque soit le nombre de neurones. Si on tient compte uniquement de l'erreur d'entrainement il nous suffirait d'une trentaine d'itérations seulement.

Entrainement du modèle

Après entrainement du modèle avec ces paramètres là on observe les résultats suivants :



On peut voir que le modèle est quand même assez overfitté, mais vu la quantité de données à disposition c'est assez normal. On remarque que la courbe du MSE a tendance à stagner entre 0.1 et 0.2. On peut également remarquer que, comme dit précédement, on arrive vite à un plateau de performances. On peut donc prendre une valeur d'environ 30 époques pour arriver à un résutat correct. En observant ces résultats on remarque que il n'y a pas de gain massif à augmenter le nombre de neurones dans la couche cachée. De ce fait on va se fixer à *4* neurones, ce qui semble être suffisant.

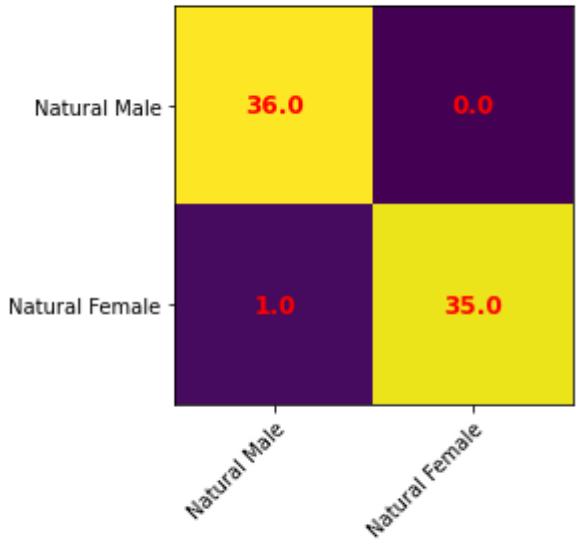
Au final nos hyper-paramètres seront les suivants :

```
EPOCHS = 30
N_NEURONS = 4
LEARNING_RATE = 0.01
MOMENTUM = 0.55
```

Modèle final

Après l'entrainement du modèle final on obtient la matrice de confusion suivante :

```
MSE training:  0.033719573471790776
MSE test:  0.13163292264669962
Confusion matrix:
```




```
F1 score     : 0.9863
Accuracy     : 0.9861
```

Avec un f-score de 0.97 on peut en déduire que notre modèle n'est pas trop mal.

Hommes vs Femmes vs Enfants

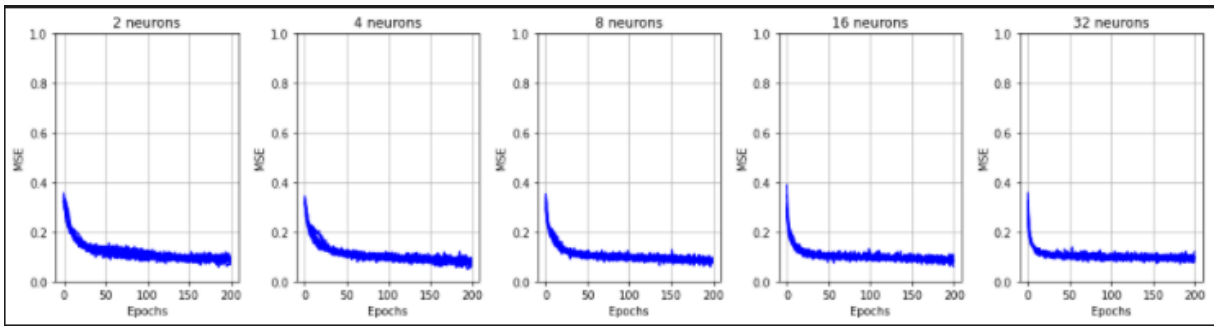
Cette fois on rajoute les voix d'enfants au dataset. Les voix d'enfants sont formées de 3 groupes de 36 voix, pour des enfants de 3, 5 et 7 ans. Pour ne pas déséquilibrer le dataset on ne va prendre qu'un tiers des données d'enfants, de manière a avoir assez de samples différents.

On a choisi d'avoir 3 neurones de sortie, un pour chaque classe. Initialement, les classes étaient annotée de la manière suivante : - Hommes : (1, -1, -1) - Femmes : (-1, 1, -1) - Enfants : (-1, -1, 1)

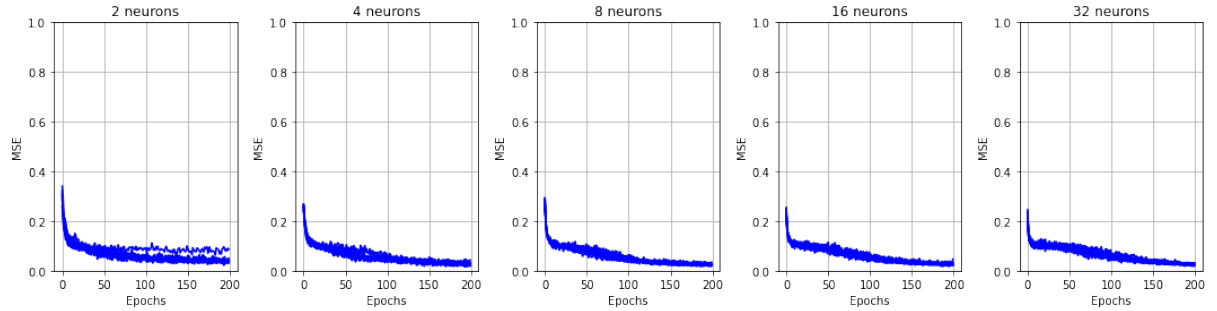
Analyse exploratoire

Comme auparavant, on va effectuer une analyse exploratoire pour essaier de trouver des hyperparamètres satisfaisants. On a choisis de mettre *-1* aux neurones qui ne sont pas de la classe mais après plusieurs essais infructueux le choix à été de remplacer les *-1* par des *0* (Hommes -> (1, 0, 0), …).

On commence l'analyse par un learning rate à *0.003* et un momentum de *0.5* :



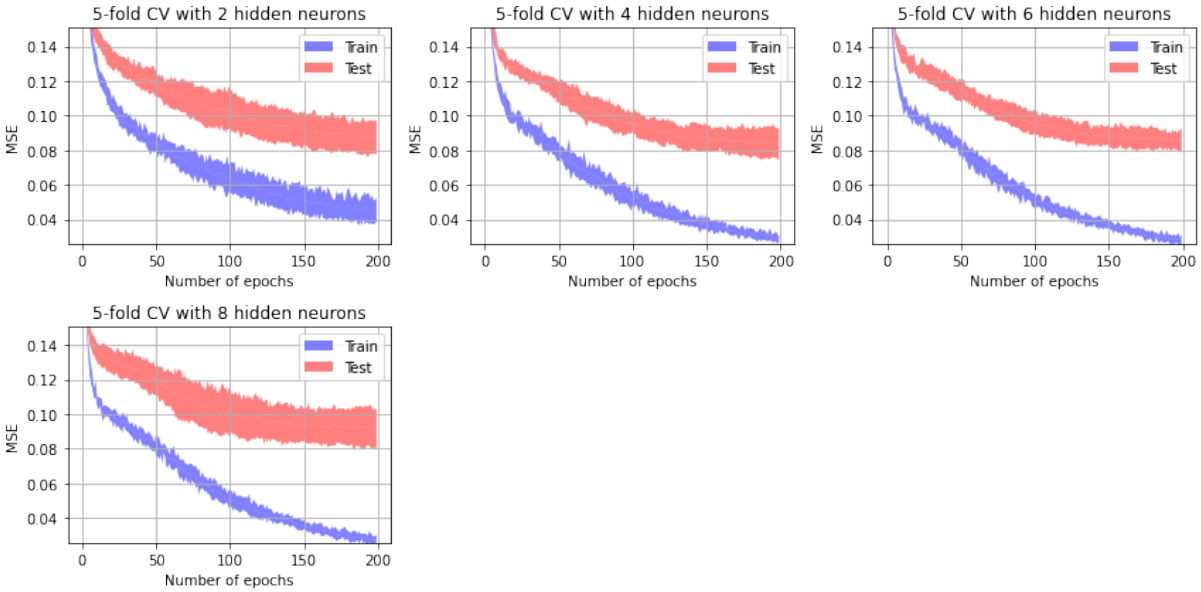
On peut voir que la courbe ne converge pas tellement et l'erreur reste assez élevée. On a également augmenté le nombre d'époques pour mieux visualiser l'évolution du MSE. Après de multiples tentatives nous avons trouvé que pour un learning rate de *0.003* et un momentum de *0.9* on obtenait des résultats satisfaisants :



On peut voir que cette fois la courbe converge plus et se stabilise autour des 150 époques. Comme pour l'expérience précédente on ne voit pas de grande amélioration à partir de 8 neurones.

Entrainement du modèle

Après entrainement du modèle avec ces hyper-paramètres, on obtient les résultats suivants :



Au premier regard ce modèle semble très overfitté, mais l'échelle varie entre 0.14 et 0.04, ce qui donc rend le modèle assez solide. Le modèle est toujours un peu overfitté, mais comme dit auparavant, la petite quantité de données influence pas mal ce résultat.

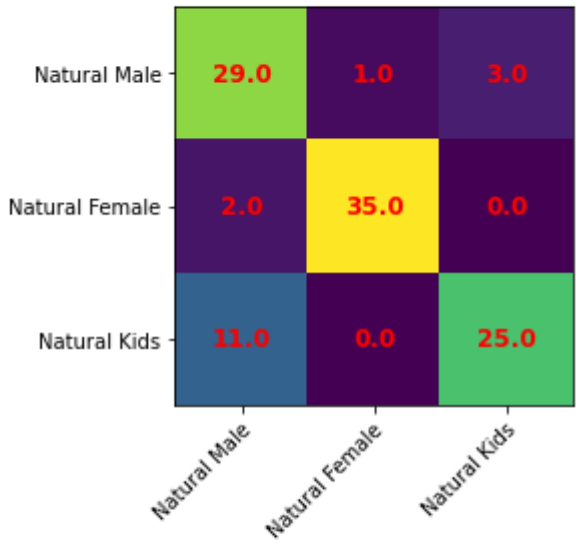
Après analyse des résultats on fixe les hyper-paramètres suivants :

```
EPOCHS = 130
N_NEURONS = 4
LEARNING_RATE = 0.003
MOMENTUM = 0.9
```

Modèle final

Après l'entrainement du modèle on obtient les résultats suivants :

```
MSE training:  0.0411144210156736
MSE test:  0.10130067646660372
Confusion matrix:
```




```
(108, 16)
F1 score natural male : 0.7733
F1 score natural female : 0.9589
F1 score natural kids : 0.7812

Weighted average f1 score : 0.8152
```

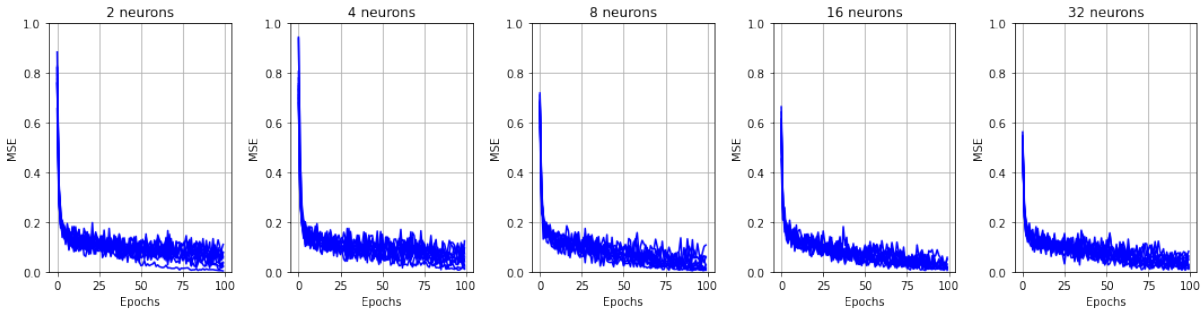
Dans cette expérience on remarque que les résultats et les scores F1 sont bons mais pas excellents. Ceci est sûrement au petit nombre de données à disposition. Quelques erreurs pèsent vite dans les résultats. Au final ce modèle se débrouille pas trop mal.

Voix naturelles vs Voix synthétiques

Dans cette expérience, on a choisi de traiter les voix naturelles et de les séparer des voix synthétiques. On va donc utiliser tous les fichiers audios `n*.wav` pour les voix naturelles et `s*.wav` pour les voix synthétiques. Il y a 180 enregistrements pour chaque type de voix ce qui rend le dataset équilibré.

Analyse exploratoire

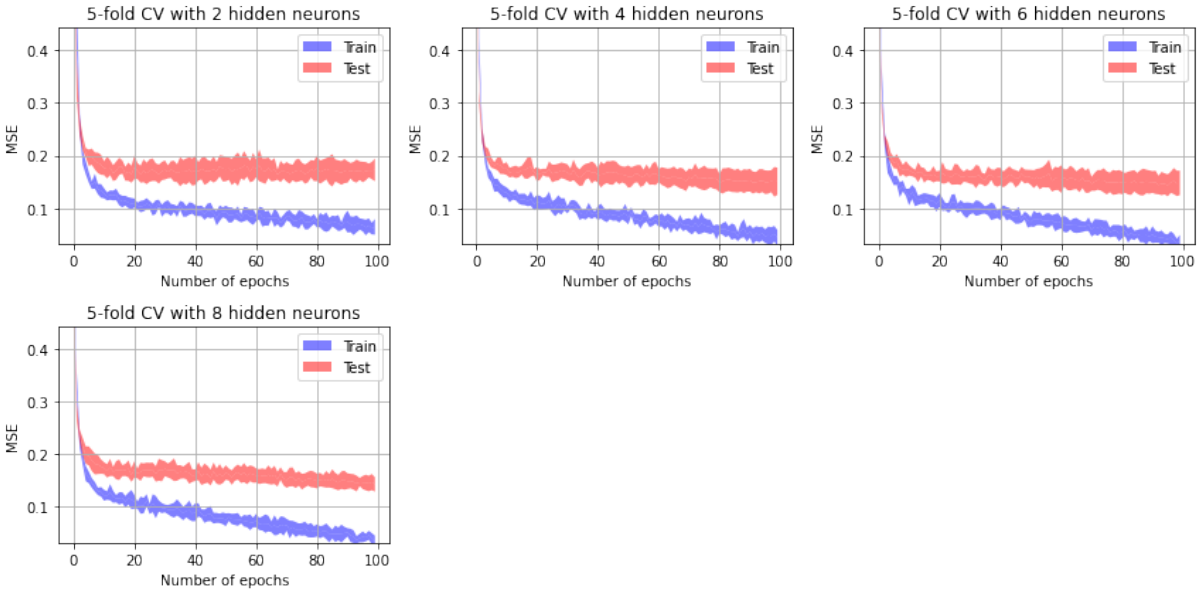
On commence par explorer les hyper-paramètres en utilisant les valeurs de l'expérience 1 (learning rate = *0.01* et momentum = *0.55*) :



C'est déjà assez bien comme résultat. Après plusieurs tentatives, la meilleure version était celle là. On peut voir qu'il suffit d'environ 25 époques pour arriver à un bon résultat.

Entrainement du modèle

On va donc entrainer le modèle avec ces hyper-paramètres là :



Le modèle est un peu overfitté mais c'est encore acceptable au vu de la quantité de données à disposition. On voit aussi qu'on peut facilement prendre 4 neurones pour s'assurer un bon résultat.

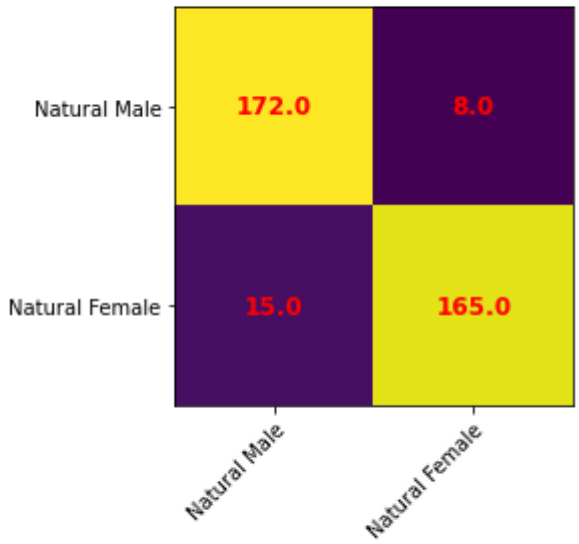
On va donc utiliser ces hyper-paramètres :

```
EPOCHS = 30
N_NEURONS = 4
LEARNING_RATE = 0.01
MOMENTUM = 0.55
```

Modèle final

Après l'entrainement du modèle on obtient les résultats suivants :

```
MSE training:  0.09595276232336707
MSE test:  0.18609850539104697
Confusion matrix:
```




```
F1 score      : 0.9373
Accuracy      : 0.9361
```

On peut remarquer que malgré que le modèle soit relativement overfitté, la matrice de confusion et le score F1 sont pas si mal. Peut-être qu'avec plus de données on aurait pu mieux entrainer notre modèle, et avoir un meilleur résultat.

Conclusion

On a pu observer différents modèles et datasets durant ce labo. On a pu comprendre comment faire une recherche d'hyper-paramètres et comment intérpréter les courbes de MSE.

Un point à améliorer dans notre démarche serait d'accélérer les exécutions de code. En effet, dans l'expérience 3, l'entrainement du modèle prenait facilement 10-12 minutes, ce qui commence à devenir long si on change tout le temps. On aurait plutôt du faire une analyse de plein de paramètres et lancer tout ça un soir, quitte à le laisser tourner toute la nuit, afin de nous éviter d'innombrables pertes de temps.

Nous avons également essayé d'implémenter ces fonctions pour utiliser un processeur graphique pour faire du multi-threading avec CUDA mais sans succès, dû à la difficulté de lier CUDA au fonctions du notebook. Pour une prochaine fois…