université
PARIS-SACLAY

Méthodes de classification

Classification de patients par utilisation de données encéphalographiques pour la détection précoce de la maladie d'Alzheimer

---

Réalisé par :

Dorra BENNOUR

Hadrien BOULANGER

Khalil BOUGUERRA

Année universitaire 2020-2021

Table des matières

Table des figures

Liste des tableaux

Introduction

Il est difficile de détecter la maladie d'Alzheimer à un stade précoce. L'électroencéphalographie présente de nombreux avantages par rapport à d'autres technologies concernant le monitoring de l'activité cérébrale. Nous disposons de données d'électroencéphalogrammes de 50 personnes dont 28 présentent la maladie d'Alzheimer (individus AD), et 22 ne présentent aucune pathologie au niveau de leur activité cérébrale (individus SCI). Nous tenterons ici de réaliser un classifieur qui permettra à partir de ce type de données de repérer si un individu est atteint de la maladie d'Alzheimer ou non. Nous tenterons donc de répondre à la question :

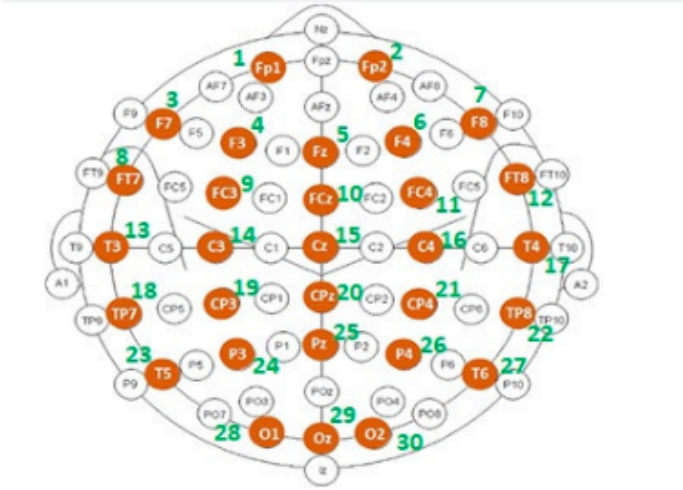
Comment classifier efficacement les individus AD et SCI à l'aide des données d'élécroencéphalogramme sur les 4 fréqunces Alpha Beta Delta Theta ?

Chapitre 1

Données

1.1 Présentation des données

Pour chacun des 50 patients, on dispose de 4 matrices (une à chaque fréquence) donnant les connexions entre 30 électrodes disposées sur la surface du cerveau, comme le montre le schéma ci-dessous.



On a donc pour chaque patient un ensemble de 900*4 soit 3600 données.

1.2 Prétraitement des données

1.2.1 Eliminer la redondance

Les données présentent une redondance importante dans l'information. En effet, les diagonales qui présentent la connexion entre une électrode et elle même n'apporte aucune information (que des 0 sur la diagonnale), et de plus les matrices sont symétriques. On fait donc le choix de travailler sur les matrices strictement inférieures, après avoir fait le traitement ci-dessous.

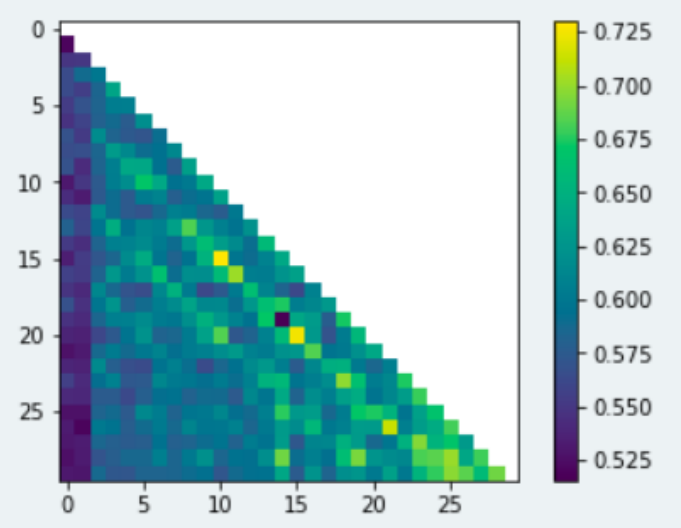


FIGURE 1.1 – Une des matrices en question après supression des données redondantes

On passe donc de 3600 données par patient à 1740.

1.2.2 Cibler les valeurs faibles

Pour la classification, il nous a semblé intéressant de nous concentrer sur les valeurs faibles des connexions. On s'est demandé si le fait de ne travailler que sur ces valeurs permet de mieux discriminer les individus entre les deux classes AD et SCI. C'est le choix de la résolution.
Dans la suite nous effectuerons la classification sur plusieurs résolutions différentes, pour ensuite comparer les performances et voir si l'une d'elle est plus pertinente :

- résolution 100% : on travaille sur toutes les valeurs.
- résolution 70% : on travaille sur les 70% des valeurs les plus faibles.
- résolution 50% : on travaille sur les 50% des valeurs les plus faibles.

— résolution 30% : on travaille sur les 30% des valeurs les plus faibles.

Mais comment "oublier" les valeurs les plus fortes puisque l'on ne peut pas supprimer purement et simplement une valeur ? En effet, notre classifieur n'acceptera pas qu'il y ait des données manquantes surtout qu'elles correspondront peut être à des connexions différentes selon l'individu.
Il faut donc remplacer les valeurs fortes par une seule et même valeur tout le temps pour éliminer la variabilité entre ces valeurs fortes et ainsi faire en sorte que le classifieur s'y intéresse au minimum. Il n'a pas de sens de leur affecter la valeur 0 puisqu'elles passeraient de valeurs fortes au départ, à la valeur la plus faible du tableau. A la place, on calcule pour chaque tableau la moyenne entre les valeurs fortes à éliminer (par exemple pour une résolution de 70% cela correspond aux 30% valeurs les plus fortes), et on remplace chacune des valeurs fortes à éliminer par cette moyenne. En voici une illustration :

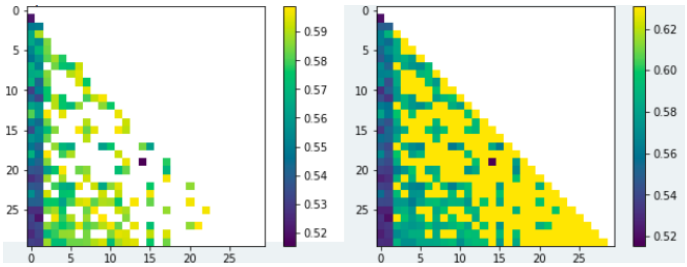


FIGURE 1.2 – remplacement des valeurs fortes par leur moyenne (résolution 50%)

1.2.3 Sélection des variables

Classement des variables

A présent, on a toujours 1740 connexions par individus. Ce qui veut dire que l'on se lance dans une classification avec seulement 50 individus et 1740 variables. On a beaucoup trop de variables, il est préférable d'en sélectionner une petite partie des plus pertinentes avant de procéder à la classification. On souhaite donc selectionner un ensemble de variables (jusqu'à 30 variables de préférence) qui contient le maximum d'information et surtout le maximum d'information discriminante entre les deux classes.

Chapitre 2

Méthodologie

2.1 Méthodes deployées

Afin de classifier les différents patients on procède sur 3 étapes (voir figure 2.1) :

1. **XGBoost** : Etape de sélection des 30 variables les plus importantes
2. **Wrapping** : Etape de sélection du nombre de variables
3. **Classification** : Etape de classification en "AD" : personnes atteintes de l'Alzheimer, "SCI" : personnes saines.

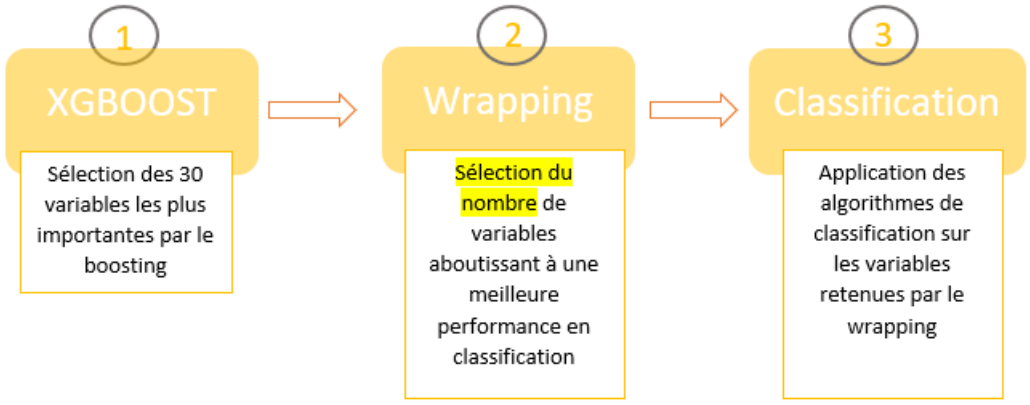


FIGURE 2.1 – Schéma simplifié de la méthodologie déployée

2.1.1 XGBoost

Cette étape consiste en une étape de sélection des variables. On souhaite exclure les méthodes qui combinent les variables de départ pour en créer de nouvelles par souci de conserver une facilité d'interprétation de cette sélection. On privilégiera donc une méthode de sélection des variables qui les conserve telles qu'elles. L'algorithme de classification XGBoost nous fournira directement un classement des connexions les plus pertinentes à analyser pour répartir les individus entre les deux classes. Partant d'une base de données à 1740 variables, nous visons à les réduire en tenant compte de leur ordre d'importance. La méthode de boosting a été choisie en se basant sur la rapidité de son exécution et ses performances. Comme il s'agit d'un algorithme d'ensembles, souvent utilisé notamment pour la sélection de variables dans le domaine génétique [2], cette approche pourrait alors s'étendre pour s'appliquer à notre problème. Le choix a été aussi fait par élimination, puisqu'on ne peut pas utiliser les mêmes algorithmes de classification pour la sélection de variables afin que les résultats ne soient pas biaisés.

Rappel du principe

Le Boosting de Gradient est un algorithme d'apprentissage supervisé dont le principe et de combiner les résultats d'un ensemble de modèles plus simple et plus faibles afin de fournir une meilleur prédiction. Pour décrire succinctement le principe, l'algorithme travaille de manière séquentielle. Contrairement par exemple au Random Forest. Cette façon de faire va le rendre plus lent bien sur mais il va surtout permettre à l'algorithme de s'améliorer par capitalisation par rapport aux exécutions précédentes. Il commence donc par construire un premier modèle qu'il va bien sur évaluer (apprentissage supervisé). A partir de cette première évaluation, chaque individu va être alors pondérée en fonction de la performance de la prédiction.

2.1.2 Wrapping

Après avoir appliqué quelques expérimentation, on a constaté que la sélection des variables fournie par l'algorithme de boosting n'est pas nécessairement optimale. Nous sélectionnerons donc le nombre de variables optimal grâce à la méthode du wrapping sur les variables classées par XGBoost. En effet, le nombre de variables retenu pour une meilleure performance du boosting n'aboutit pas forcément à une meilleure performance en classification. Cela consiste à tester le classifieur (RandomForest puis K nearest neighbours) obtenu avec la première variable retenue par le boosting, puis les deux premières, puis les trois premières et cela jusqu'aux 30 premières. On retiendra ainsi le nombre de variables qui donne les meilleures

performances en cross validation leave one out sur nos 50 individus. En résumé cela consiste à retenir le nombre de variables donnant la meilleure performance en classification (Voir Figure 2.2) :

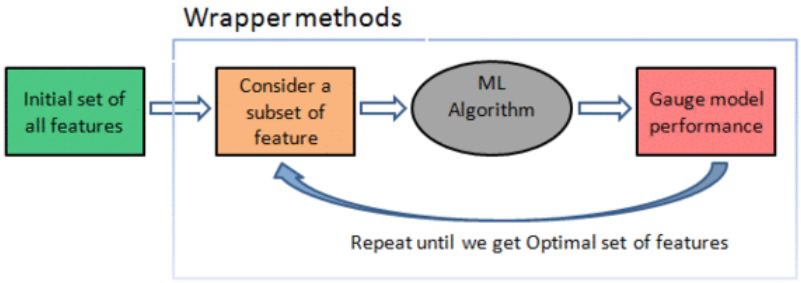


FIGURE 2.2 – Schéma simplifié de la méthode des wrappers

2.1.3 Classification

Finalement, pour la partie classification, nous avons choisi de comparer deux algorithmes de classification supervisée qui sont souvent utilisés :

- **Random forests**
  Rappel du principe :
  La forêt aléatoire, comme son nom l'indique, est constituée d'un grand nombre d'arbres de décision individuels qui fonctionnent comme un ensemble. Chaque arbre individuel de la forêt aléatoire crache une prédiction de classe et la classe ayant le plus de votes devient la prédiction de notre modèle.
- **K-plus proches voisins**
  Rappel du principe :
  Le principe de l'algorithme est le suivant : à partir d'une base de données étiquetées, on peut estimer la classe d'une nouvelle donnée en regardant quelle est la classe majoritaire des k données voisines les plus proches (d'où le nom de l'algorithme). Le seul paramètre à fixer est k, le nombre de voisins à considérer .

Les résultats de cette méthodologie sont présentés dans les parties qui suivent.

Chapitre 3

Résultats : Classification AD vs SCI

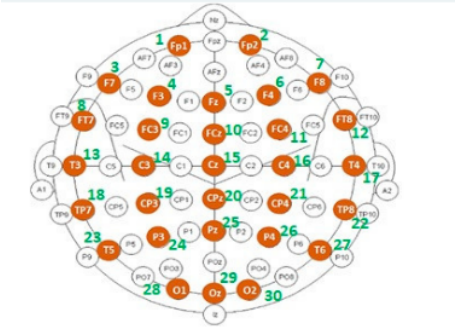
3.1 Interprétation des variables retenues par XGBoost

XGBoost a retenu majoritairement certaines connexions comme étant plus discriminantes entre les patients atteints d'Alzheimer et les patients sains :
Pour une résolution de 100% des valeurs :

- Fréquence Alpha (repos) electrode 12 et electrode 1 : frontal-temporal/préfrontal
- Fréquence Alpha (repos) electrode 4 et electrode 16 : frontal/central

Valeurs faibles :

- Fréquence Alpha (repos) electrode 8 et electrode 2 : frontal-temporal/préfrontal
- Fréquence Alpha (repos) electrode 4 et electrode 10 : frontal/frontal-central
- Fréquence Theta (somnolence) electrode 11 et electrode 10 : frontal-central/frontal-central



3.2 Métriques d'évaluation

Pour évaluer les modèles mis en œuvre dans les parties précédentes on opte pour différentes métriques d'évaluation : accuracy, précision, F1-score, Rappel et AUC, qui sont définis comme suit :

- **Accuracy :** correspond au taux de bon classement total des individus. Comme notre base de données est bien équilibrée pour les deux classes "AD" et "SCI".
- **Précision :** correspond à la proportion des vrais positifs.
- **Rappel :** cette métrique correspond à la proportions des vrais positifs qui ont été bien classé
- **F1-score :** Le score F1 est un nombre compris entre 0 et 1 et représente la moyenne harmonique de la précision et du rappel.

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

- **AUC :** AUC signifie "aire sous la courbe ROC". Cette valeur mesure l'intégralité de l'aire à deux dimensions située sous l'ensemble de la courbe ROC (par calculs d'intégrales) de (0,0) à (1,1).

On a évalué les deux modèles ; Forêts aléatoires (Random Forests) et K-plus proches voisins pour les différentes résolutions. Ayant un petit nombre de données (50) nous avons opté pour présenter des résultats obtenus en Leave-one-out afin d'éviter de diviser notre base en un ensemble d'entraînement et un ensemble de test.

3.3 Random Forests

3.3.1 Résolution 100%

En appliquant le modèle des forêts aléatoires sur toute la base de données, tout en retenant que le nombre de variables optimale séléctionné (comme décrit lors du troisième chapitre), on trouve les résultats suivants :

classe	**Précision**	**Rappel**	**F1-score**	**Support**
SCI	0.83	0.86	0.84	22
AD	0.89	0.86	0.87	28
Accuracy=0.86				
AUC=0.86				

TABLE 3.1 – Résultats du classifieur "Random forest" pour une résolution de 100% des données

3.3.2 Résolution 70%

classe	**Précision**	**Rappel**	**F1-score**	**Support**
SCI	0.83	0.86	0.84	22
AD	0.89	0.86	0.87	28
Accuracy=0.86				
AUC=0.86				

TABLE 3.2 – Résultats du classifieur "Random forest" pour une résolution de 70% des données

3.3.3 Résolution 50%

classe	**Précision**	**Rappel**	**F1-score**	**Support**
SCI	0.83	0.86	0.84	22
AD	0.89	0.86	0.87	28
Accuracy=0.86				
AUC=0.86				

TABLE 3.3 – Résultats du classifieur "Random forest" pour une résolution de 50% des données

3.3.4 Résolution 30%

classe	**Précision**	**Rappel**	**F1-score**	**Support**
SCI	0.79	0.86	0.83	22
AD	0.88	0.82	0.85	28
Accuracy=0.86				
AUC=0.86				

TABLE 3.4 – Résultats du classifieur "Random forest" pour une résolution de 30% des données

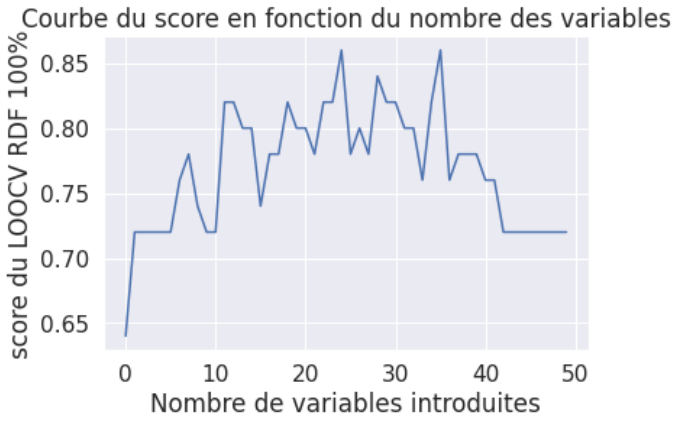
3.3.5 Discussion

En regardant de plus près les résultats de la classification du modèle des forêts aléatoires par rapport à chaque résolution, on ne ne remarque pas une grande grande différence. En effet on a toujours un taux de bon classement de 86%. La différence réside cependant au niveau des métriques de rappel et de précision pour une résolution 30%, où la précision pour la classe SCI est égale à 0.79. Dans ce cas, à 21% des cas on prédit des personnes saines comme malades, ce qui peut être assez dangereux. Une résolution supérieure ou égale à 50% semble être à privilégier.

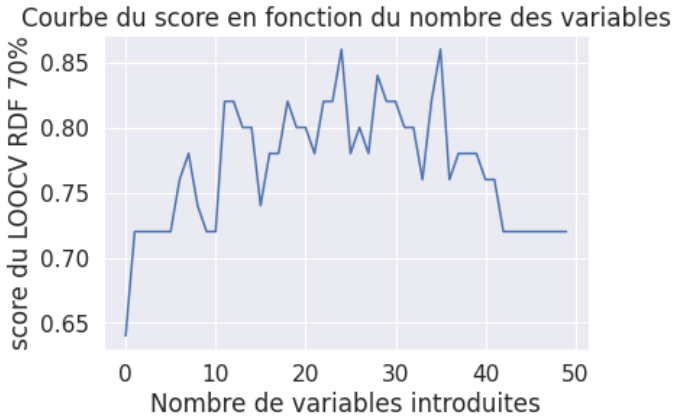
3.3.6 Réduire le biais des résultats obtenus

Nous sommes conscients que les résultats présentés sont obtenus en choisissant le nombre de variables qui les maximise (wrapping). On donne donc en annexe l'Accuracy obtenue pour les autres choix de nombre de variables entre 1 et 30 (plus de variables n'est pas pertinent).

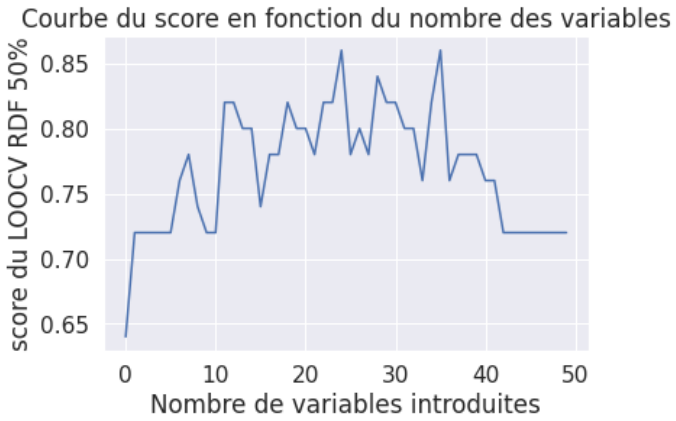
— Pour une résolution de 100% : on obtient une accuracy de 77.24% en moyenne



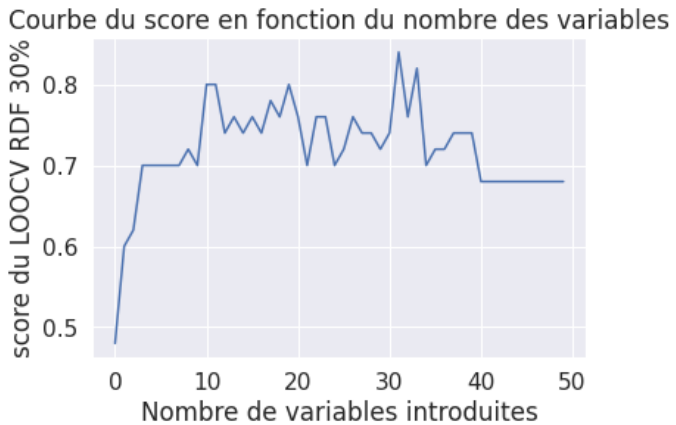
— Pour une résolution de 70% : on obtient une accuracy de 77.24% en moyenne



— Pour une résolution de 50% : on obtient une accuracy de 77.24% en moyenne



— Pour une résolution de 30% : on obtient une accuracy de 70.21% en moyenne



En moyenne, une résolution de 30% donne des résultats légèrement moins bons.

3.4 K-plus proches voisins

3.4.1 Résolution 100%

En appliquant le modèle des k-plus proches voisins sur toute la base de données (avec K=7), tout en retenant que le nombre de variables optimal séléctionné (comme décrit lors du troisième chapitre), on trouve les résultats suivants :

classe	**Précision**	**Rappel**	**F1-score**	**Support**
SCI	0.75	0.82	0.78	22
AD	0.85	0.79	0.81	28
Accuracy=0.80				
AUC=0.77				

TABLE 3.5 – Résultats du classifieur "K-plus proches voisins" pour une résolution de 100% des données

3.4.2 Résolution 70%

classe	**Précision**	**Rappel**	**F1-score**	**Support**
SCI	0.83	0.91	0.87	22
AD	0.92	0.86	0.89	28
Accuracy=0.88				
AUC=0.88				

TABLE 3.6 – Résultats du classifieur "K-plus proches voisins" pour une résolution de 70% des données

3.4.3 Résolution 50%

classe	**Précision**	**Rappel**	**F1-score**	**Support**
SCI	0.83	0.86	0.84	22
AD	0.89	0.86	0.87	28
Accuracy=0.86				
AUC=0.86				

TABLE 3.7 – Résultats du classifieur "K-plus proches voisins" pour une résolution de 50% des données

3.4.4 Résolution 30%

classe	**Précision**	**Rappel**	**F1-score**	**Support**
SCI	0.71	0.91	0.80	22
AD	0.91	0.71	0.80	28
Accuracy=0.80				
AUC=0.81				

TABLE 3.8 – Résultats du classifieur "K-plus proches voisins" pour une résolution de 30% des données

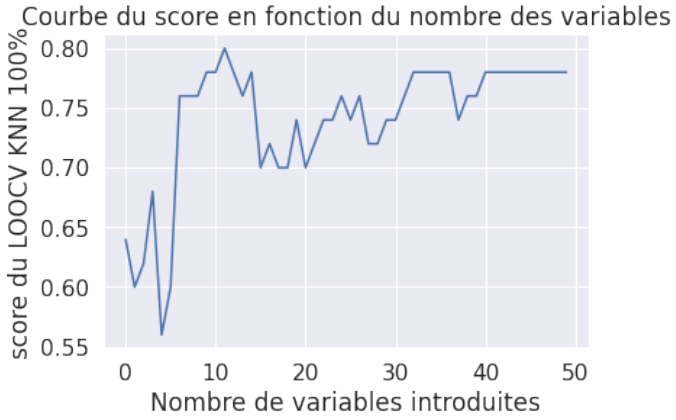
3.4.5 Discussion

Les résultats de la classification par les k-plus proches voisins sont équivalents pour toutes les résolutions. On a néanmoins une meilleure performance pour une résolution de 70%. En effet, on pour la classe "SCI" on s'intéresse plutôt à la métrique de précision puisqu'on considère la prédiction des personnes non malades comme malades comme étant pernicieuse. Concernant la classe "AD" relative aux personnes atteintes de la maladie d'alzheimer, on se focalise sur la métrique de "Rappel" car on a comme but de détecter le maximum possible des personnes atteintes de cette pathologie. Ces deux métriques pour une résolution de 70% sont alors assez satisfaisants. Les résolutions 100% et 30 %, quant à eux, nous donnent une plus faible précision c'est à dire on tend à mal prédire la classe des individus sans pathologie.

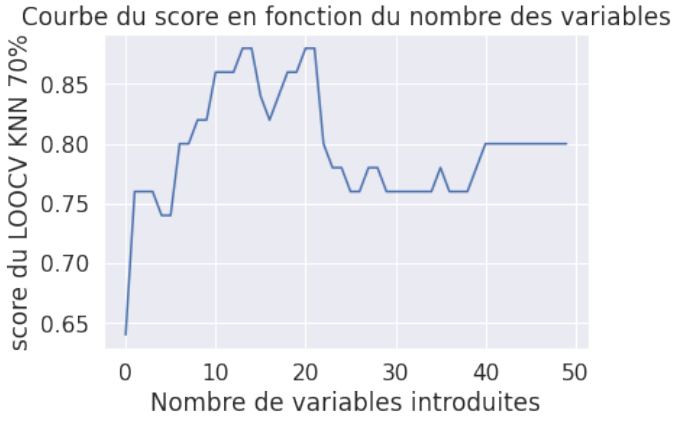
3.4.6 Réduire le biais des résultats obtenus

On donne donc en annexe l'Accuracy obtenue pour tous les choix de nombre de variables entre 1 et 30 (plus de variables n'est pas pertinent).

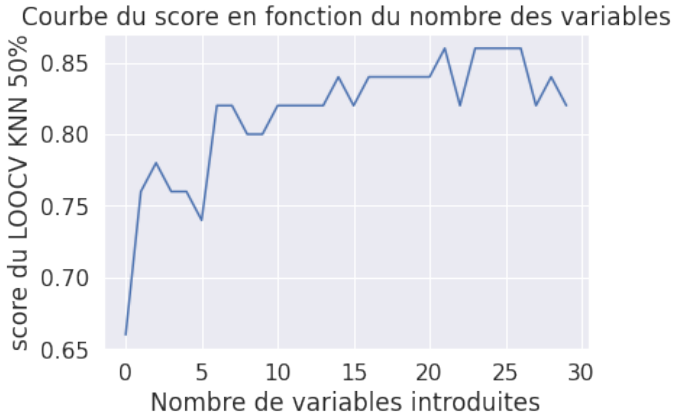
— Pour une résolution de 100% : on obtient une accuracy de 71.79% en moyenne



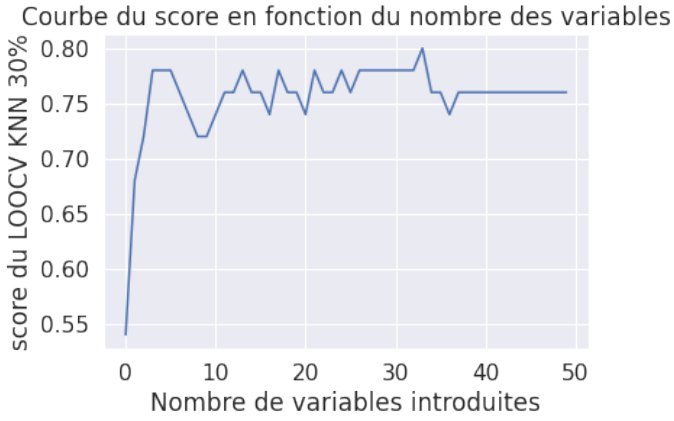
— Pour une résolution de 70% : on obtient une accuracy de 80.70% en moyenne



— Pour une résolution de 50% : on obtient une accuracy de 81.45% en moyenne



— Pour une résolution de 30% : on obtient une accuracy de 74.97% en moyenne



En moyenne, une résolution de 50% donne des résultats légèrement meilleurs. C'est mieux encore que pour Random Forest.

3.5 Comparaison des deux classifieurs

Les résultats obtenus par les deux classifieurs "Forêts aléatoires" et "k-plus proches voisins" sont équivalents lorsque l'on choisit un nombre de variables qui maximise les performances. En s'appuyant de plus sur les métriques les plus cohérentes à notre problème : "Precision" et "Rappel", on peut conclure que les résolutions à 50% et 70% fournissent une meilleure performance. De plus, le nombre de variables étant choisi pour maximiser les performances sur l'ensemble des 50 individus, rien ne nous dit que ce nombre de variable est optimal en généralisation. Les K plus proches voisins sont moins sensibles au nombre de variables retenues : en moyenne pour un nombre de variable entre 1 et 30 et pour des résolutions de 50 et 70% on obtient plus de 80% d'accuracy contre 77% environ pour les même résolutions avec Random Forest. De plus les K-plus proches voisins sont plus rapides d'exécution.
Les K plus proches voisins semble présenter quelques avantages sur RandomForest et semble assurer en généralisation une accuracy de plus de 80% en classification à condition de sélectionner au moins 7 variables (voir annexe). Les résolutions les plus intéressantes sont globalement de travailler sur les 50% ou les 70% des valeurs les plus faibles.

Conclusion

On a pu déterminer les zones du cerveau et les états (liés aux fréquences) qui permettent le mieux de discriminer les électroencéphalogrammes des personnes saines et des personnes atteintes d'Alzheimer. Les principales différences s'observent au repos et principalement dans les zones frontales et centrales du cerveau. Les réslutats en classification semblent confirmer ces hypothèses de manière assez fiable. Avec peu de données, en se concentrant sur les connexions les plus faibles, en utilisant un classifieur pour retenir les meilleures variables, en classifiant avec un autre classifieur et en calculant les performances en Leave One Out, on a réalisé un classifieur performant.

Annexe

Bibliographie

[1] N. Houmani ,F. Vialatte,E. Gallego-Jutglà,G. Dreyfus,V. Nguyen-Michel,J. Mariani, K. Kinugawa (2018) *Diagnosis of Alzheimer's disease with Electroencephalography in a differential framework.*

[2] J. Molina Mora1, F. Mata Ordoñez, D. Alexander Bonilla (2017) *Improvement of K-Means clustering algorithm performance in gene expression data analysis through pre-processing with principal component analysis and boosting.*