

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université des Sciences et de la Technologie Houari Boumediene



FACULTÉ D'INFORMATIQUE

Master Informatique Visuelle

M É M O I R E

En vue de l'obtention du Diplôme de Master Informatique Visuelle

Thème

**INTELLIGENCE ARTIFICIELLE
ET DIAGNOSTIC AUTOMATIQUE
DE L'ALZEIHMER**

Présenté par : KABECHE MAZGHA
MAAMRA ABDELFETAH

Encadré par : Dr. FERROUKHI MERZAK
Pr. OUAHABI ABDELDJALIL

Membres de jurys : Mr. BABA ALI AHMED RIADH (président)
Mme DJIDEL FAIZA (examinatrice)
Mme ABDELLI Dalila (invitée)

Remerciements

Tout d'abord nous remercions ALLAH, le miséricordieux et le clément.

Nous exprimons toute notre gratitude à nos promoteurs, Docteur FERROUKHI et Professeur OUAHABI, pour les critiques et les conseils hautement fructueux, pour la confiance et le soutien qu'ils nous ont accordé toute au long de ce mémoire.

Nous adressons également nos remerciements, à tous nos enseignants, qui nous ont donné les bases de la science.

Nos remerciements chaleureux aux jurys pour avoir accepter d'examiner notre modeste travail.

Nous tenons à remercier aussi notre invitée Madame ABDELLI Dalila présidente de l'association 'Ma tensanich'.

Dédicaces

A nos parents et grands parents les plus chères au monde qui nous ont éclairés le chemin, qui nous ont toujours montrés leur générosité et qui nous ont soutenus tout au long de nos études.

A tous nos fidèles frères et sœurs qui nous ont encourager.

A tous nos proches et nos amis.

Et à toute personne qui a participé de près ou de loin pour l'accomplissement de ce modeste travail.

MAZGHA ET ABDELFETAH

ملخص

تعتبر الأمراض المرتبطة بالدماغ من بين الأمراض الخطيرة التي تصيب الإنسان ويجب اكتشافها في أسرع وقت ممكن من أجل علاجها إذا كان لديهم علاج أو على الأقل يكونون قادرين على ابطئها.

لهذا بالضبط ، اقترحنا نموذج $HCAE(HyperConnectomesAuto - Encoder)$ الذي يعتمد على $HGNNS(HypergraphNeuralNetwork)$ لاكتشاف مرض الزهايمر قبل فوات الأوان .

لدينا خطوة أولى ، إنشاء اتصال تشعبي خاص بالموضوع. في هذه الخطوة ، يحتوي نموذجنا على بيانات رقمية مدخلة تمثل $ROIs$ (مناطق الاهتمام). بعد ذلك ، تم مقارنة هذه المناطق مع بعضها البعض وفقاً لأربعة أشكال ويتم تخزين النتائج في أربع مصفوفات متغيرة. بمجرد تصميم المصفوفات ، يتم تكوين التوصيات الفائقة باستخدام ك أقرب الجيران $KNNs$. يتم وضع كل اتصال تشعبي في متوجه يتم دفعه بعد ذلك في متوجه واحد بالحجم ($140 * 35$) لذلك نحصل على متوجه واحد يحتوي على جميع معلومات العائد على الاستثمار المستخرج من التوصيات الفائقة.

بالنسبة للخطوة الثانية ، لدينا بناء المشفير التلقائي للوصلة الفائقة الذي تم الحصول عليه كنتيجة للخطوة الأولى. تتكون هذه المرحلة نفسها من جزأين ، المشفير ومفكك التشفير. يمر المشفير المتوجه ، الذي ينفذ من الخطوة الأولى ، أولاً عبر طبقة تلافيفية متبوعة بوظيفة التنشيط $ReLU$ متبوعة بطبقة تلافيفية أخرى مرة أخرى ، وبالتالي الحصول على متوجه متكامل. يتم إرسال هذا المتوجه بدوره إلى أداة التمييز التي تميز العينات الصحيحة والخطأ ، إلى مفكك الشفرة الذي يعيد إنتاج المتوجه في توصيات فائقة وإلى آلة المتوجه الداعمة (الذي يمثل المرحلة الأخيرة من عملنا) SVM .

الخطوة الأخيرة من عملنا هي تقييم التمييز مع SVM كما ذكرنا سابقاً. هذا يصنف نتائج الناقل المتكامل ويخبرنا ما إذا كان المريض في مرض الزهايمر أو فئة الضعف الإدراكي المعتدل.

في النهاية ، نحصل على نتيجة ثانية ، إما أن المريض يعاني من مرض الزهايمر أو أنه يعاني من ضعف إدراكي خفيف.

الكلمات الرئيسية : الزهايمر، التشفير التلقائي، التشفير، فك التشفير، الطبقة التلافيفية، المتوجه المتكامل، المميز، تقييم التمييز، التصنيف، النتيجة الثنائية، الضعف الإدراكي المعتدل.

Résumé

Les maladies reliées au cerveau sont parmi les maladies dangereuses pour les êtres humains et celles ci, nécessitent d'être détectées le plus vite possible afin de les traiter s'elles ont un traitement ou au moins, pouvoir les ralentir.

Pour exactement cela, nous avons proposé le modèle HCAE (HyperConnectomes Auto-Encoder) qui se base sur les HGNNs (Hypergraph Neural Network) pour détecter la maladie d'Alzheimer avant que ça soit trop tard.

Nous avons comme 1^{ère} étape, la génération de l'hyperconnectome spécifique au sujet. Dans cette étape, notre modèle a pour entrée des données numériques qui représentent les ROIs (Regions Of Interest). Ensuite, ces régions sont comparées ensemble par ensemble entre elles selon quatres morphologies et les résultats sont garder dans quatre matrices d'adjacence. Une fois les matrices sont conçus, les hyperconnectomes sont formés en utilisant les KNNs. Chaque hyperconnectome est mis dans un vecteur qui par la suite est fusionnée dans un seul vecteur de taille (140x35) donc nous obtenons un seul vecteur contenant toutes les informations des ROIs extraits des hyper-connectomes.

Pour une 2^{ème} étape, nous avons la construction de l'auto-encodeur de l'hyperconnectome obtenue comme résultat dans la 1^{ère} étape. Cette étape elle même est constituée de deux parties, l'encodeur et le décodeur. L'encodeur fait passer le vecteur, résultant de l'étape une, en premier lieu par une couche convolutionnelle qui est suivie par la fonction d'activation ReLu qui est suivie encore une fois d'une autre couche convolutionnelle, nous obtenant par la suite un vecteur intégré. Ce vecteur à son tour est envoyé au discriminateur qui fait la distinction des échantillons réels et faux, au décodeur qui reproduit le vecteur en hyperconnectome et au SVM (Support Vector Machine) qui représente la dernière étape de notre travail.

La dernière étape de notre travail est l'évaluation de la discriminativité avec le SVM comme déjà mentionné. Celui-ci fait la classification des résultats du vecteur intégré et nous dis si le patient fait partie de la catégorie AD ou MCI.

A la fin, nous obtenons un résultat binaire, soit le patient est atteint de la maladie d'Alzheimer ou il a une déficience cognitive légère.

Mots clés : HCAE, HGNN, Alzheimer, ROI, KNN, Auto-Encodeur, Encodeur, Décodeur, Couche convolutionnelle, ReLu, Vecteur intégré, Discriminateur, SVM, Évaluation de la discriminativité, Classification, Resultat binaire, Déficience cognitive légère.

Abstract

Diseases related to the brain are among the dangerous diseases for human beings and these need to be detected as soon as possible in order to treat them if they have a cure or at least be able to slow them down.

For exactly this, we have proposed the HCAE (HyperConnectomes Auto-Encoder) model which is based on HGNNs (Hypergraph Neural Network) to detect the Alzheimer disease before it is too late.

We have as a 1st step, the generation of the subject-specific hyperconnectome. In this step, our model has as an input, numerical data that represents the ROIs (Regions Of Interest). Then, these regions are compared together with each other according to four morphologies and the results are stored in four adjacency matrices. Once the matrices are designed, the hyperconnectomes are formed using the KNNs. Each hyperconnectome is put in a vector which is then merged into a single vector of size (140x35) so we obtain a single vector containing all the information of the ROIs extracted from the hyperconnectomes.

For a 2th step, we have the construction of the auto-encoder of the hyperconnectome obtained as a result in the 1st step. This step itself is made up of two parts, the encoder and the decoder. The encoder passes the vector, resulting from step one, first through a convolutional layer which is followed by the activation function ReLu which is followed again by another convolutional layer, we obtain afterwards an embedded vector. This vector in its turn is sent to the discriminator which distinguishes real and fake samples, to the decoder which reproduces the vector in hyperconnectome and to the SVM (Support Vector Machine) which represents the last step of our work.

The last step of our work is the evaluation of the discriminativeness with the SVM as already mentioned. This one does the classification of the results of the integrated vector and tells us if the patient is in the AD or MCI category.

At the end, we get a binary result, either the patient has Alzheimer disease or he has mild cognitive impairment (MCI).

Key words : HCAE, HGNN, Alzheimer, ROI, KNN, Auto-Encoder, Encoder, Decoder, Convolutional layer, ReLu, embedded vector, Discriminator, SVM, Evaluation of discriminativeness, Classification, binary result, MCI.

Table des matières

Introduction générale	11
1 Présentation de la maladie	13
1.1 Introduction	13
1.2 Démences neurodégénératives	13
1.3 Maladie d'Alzheimer (MA)	14
1.4 Symptômes et approche diagnostique	16
1.5 Imagerie et maladie d'Alzheimer	17
1.6 Conclusion	18
2 État de l'art	19
2.1 Introduction	19
2.2 Intelligence artificielle	19
2.3 Apprentissage automatique	19
2.4 Support Vector Machine	20
2.5 Apprentissage profond	23
2.6 Réseau neuronal graphique	23
2.6.1 K-plus proches voisins (KNN)	25
2.6.2 Réseau convolutif spectral	26
2.7 HGNN : Hypergraph Neural Network	27
2.8 Auto-Encodeur	28
2.8.1 Auto-encodeur incomplet	30
2.8.2 Auto-encodeur régularisé	30
2.9 Conclusion	31
3 Conception et architecture du modèle	32
3.1 Introduction	32
3.2 Position du problème	33
3.3 Proposition d'Hyper-Connectome Auto-Encoder (HCAE) pour la classification de l'état du cerveau	33
3.3.1 Définition et construction d'hyper-connectome multi-vues	33
3.3.2 Architecture d'hyper-connectome spécifique au sujet	35
3.3.3 Les hyperparamètres du HCAE	35
3.3.4 Classification par la méthode SVM	37
3.4 Conclusion	37

TABLE DES MATIÈRES

4 Quantification et mesure de l'étendue de la maladie	38
4.1 Introduction	38
4.2 Environnement matériel et logiciel	38
4.2.1 Python	38
4.2.2 Tensorflow	39
4.2.3 NumPy	40
4.2.4 Sklearn (Scikit-learn)	40
4.2.5 SciPy	41
4.2.6 Streamlit	41
4.3 Environnement de développement	41
4.3.1 Visual Studio Code	41
4.3.2 Google Colab	42
4.4 Données et discussion	43
4.5 Présentation de l'interface et fonctionnalités	44
4.5.1 Page de présentation	44
4.5.2 Page de visualisation des fMRIs du patient	45
4.6 Tests expérimentaux	48
4.6.1 Quelques problèmes rencontrés	51
4.7 Conclusion	51
Conclusion Générale	52
Bibliographie	54

Table des figures

1.1	Répartition diagnostique des démences : Centre Mémoire Lille-Bailleul (1992-2001) (échantillon de 4212 Patients) [3]	14
1.2	Première patiente de 51 ans, Auguste Deter	14
1.3	IRM et la représentation du tau de fluorodésoxyglucose (18F) sur un sujet normal, un patient souffre de mild cognitive impairment MCI et un patient souffrant de maladie alzheimer [5]	15
1.4	Progression de la MA [6]	16
1.5	Une coupe transversale du cerveau effectuée en son centre, entre les oreilles, d'un cerveau atteint d'Alzheimer	17
1.6	Les différents régions d'intérêts visualiser à l'aide de l'imagerie médicale [8]	18
2.1	Les familles de machine learning	20
2.2	Classes séparées par hyperplan	21
2.3	Jeu de données avec deux classes, bleu et rouge non linéairement séparable	21
2.4	Données projetées dans une dimension supérieure et séparées par un plan	22
2.5	Architecture d'un réseau de neurones	23
2.6	Un graphe orienté et un graphe non orienté	24
2.7	Un graphe non orienté et sa représentation en matrice d'adjacence	24
2.8	Visualisation de k-Plus Proches Voisins avec deux classes (cercles bleus et triangles rouges) avec $k = 3$ et $k = 7$	25
2.9	Exemple de graphique avec une caractéristique attribuée à chaque nœud	26
2.10	Exemple de matrice de contiguïté et de matrice de caractéristiques, les lignes et colonnes représente les noeuds	27
2.11	Exemple de convolution de graphe par multiplication matricielle	27
2.12	Comparaison entre graphe et Hypergraphe	28
2.13	Auto-encoder avec goulot d'étranglement "Bottleneck"	29
2.14	Auto-encoder sans goulot d'étranglement	30
3.1	Connectome cérébral	33
3.2	Architecture d'auto-codage de l'hyper-connectome (AEHC) pour la classification des états du réseau du cerveau	34
4.1	Python logo	39
4.2	Tensorflow logo	39
4.3	NumPy logo	40
4.4	Scikit-learn logo	40
4.5	SciPy logo	41
4.6	Streamlit logo	42

4.7	Visual Studio Code logo	42
4.8	Google Colab logo	43
4.9	Page d'accueil de l'interface	45
4.10	Page de visualisation des fMRIs du patient	45
4.11	Page de visualisation des données csv	46
4.12	Page de visualisation de l'analyse des données csv	47
4.13	page de mesure des performances d'un modèle auto-encodeur hyper-connectomes	48
4.14	Aperçu de la base de données sur les morphologies des ROI de chaque patient	49
4.15	Courbe ROC du SVM de notre modèle avec noyau RBF sur les données ADNI GO de l'Alzheimer	49
4.16	Courbe de validation du SVM de notre modèle avec noyau RBF sur les données ADNI GO de l'Alzheimer	50
4.17	Courbe de perte d'entraînement ACE	51

Introduction générale

De toutes les données médicales, les images priment en termes de volume de données, d'informations exploitées et de connaissances extraites. L'imagerie médicale permet de visualiser l'intérieur d'un corps humain (ou animal) sans intervention chirurgicale. Elle est utilisée d'une part à des fins cliniques, c'est-à-dire dans la recherche d'un diagnostic ou d'un traitement de certaines pathologies, mais aussi pour la recherche scientifique dans le but d'étudier les réactions du corps de l'être vivant face aux maladies (Cancer, Autisme, Alzheimer ... etc) qui ont préoccupé récemment les médecins et les chercheurs dans le domaine médical.

La production systématique sous forme de données médicales numérisées fait que l'analyse et la synthèse informatiques ont connu un développement très spectaculaire ces vingt dernières années, conduisant ainsi à l'automatisation du diagnostic de diverses maladies et de nombreuses études nécessitant de multiples traitements d'images médicales. Ainsi, les outils informatiques sont devenus indispensables, voire incontournables, face à cette structure de données.

Parmi les nombreux défis que l'imagerie peut exploiter en recherche médicale, figure la maladie d'Alzheimer, la forme la plus courante de démence.

La maladie d'Alzheimer est une affection neurodégénérative du cerveau pour laquelle il n'existe actuellement aucun remède. Le diagnostic précoce de cette maladie par apprentissage profond (Deep Learning) appliqué à l'imagerie par résonance magnétique fonctionnelle (IRMf) du cerveau et spécifiquement par les réseaux de neurones sur les graphes (GNN : Graphe neural network) et les réseaux de neurones sur les Hypergraphes (HGNN : Hypergraphes neural network), attire un intérêt croissant en raison de leurs hautes performances en matière de classification, de détection et de segmentation en imagerie médicale.

De plus, le cerveau s'appuie naturellement sur les approches de la théorie des graphes, car il peut être facilement représenté sous la forme d'un réseau (graphe) d'éléments (nœuds ou sommets) et de leur association par paires (connections ou arêtes). L'intégration des graphes est une méthode puissante pour représenter les données neurologique par exemple les connectomes cérébraux, cependant les algorithmes d'intégration existant ont deux limitations majeurs. Première, ils se concentrent principalement sur

la préservation un par un relations topologiques entre les noeuds, mais ils ont pour la plupart ignoré plusieurs à plusieurs relations(c'est-à-dire ensemble à ensemble).

Seconde, les techniques d'intégration de graphes existantes ne peuvent pas être facilement adaptées à multi-vue représenter graphiquement des données avec distributions hétérogènes. Pour remédier à ces limitations, un auto-encodeur hypergraphique géométrique, nommé HyperConnectome AutoEncoder (HCAE)[1], pour intégrer des hyper-connectomes cérébraux multi-vue dérivés de connectomes cérébraux muti-vues (vue du cerveau dissimilitude de la profondeur des sillons) ou chaque individu est représenté par un ensemble de connectomes cérébraux, chacun capturant un élément particulier.

Cette représentation peut fournir de nouvelles informations importantes sur la structure et la fonction des systèmes cérébraux en réseau, y compris leur architecture, leur évolution, leur développement et leurs troubles cliniques. Comprendre quelles régions du cerveau sont impliquées dans des troubles neurologiques ou des stimuli cognitifs spécifiques est un domaine important en neuroimagerie où les réseaux de neurones graphiques (GNN) aident à l'analyse de la fonction d'imagerie par résonance magnétique fonctionnelle (IRMf) et explorent les neurobiomarqueurs.

Notre travail se concentre sur l'analyse d'images médicales contenant des couches composites de sélection dites régions d'intérêt (ROI) qui met en évidence les ROI saillants (noeuds du Hypergraphe), de sorte qu'il peut être déduit quel ROI est important pour la prédiction et en particulier le diagnostic précoce de la maladie d'Alzheimer en d'autres termes les ROI qui caractérisent le patient malade.

Le travail a été concrétisé par ce document qui est organisé comme suit :

En premier lieu, nous avons une introduction générale pour avoir une vue d'ensemble sur la problématique et la méthode adaptée pour notre problème.

Dans le chapitre 1, intitulé présentation de la maladie, nous allons parler en général les différentes formes de démence. et en détails, de la maladie d'Alzheimer. De plus, nous allons mentionner la relation de l'imagerie avec la démence neurodégénérative.

Ensuite dans le chapitre 2, nous allons parler théoriquement sur les méthodes et approches appliquées dans le diagnostic précoce de la maladie d'Alzheimer.

La partie pratique du chapitre 2 va être traitée dans le chapitre 3 où on va présenter le modèle utilisé. Par la suite, nous allons voir en détails dans le chapitre 4, les résultats obtenus en appliquant notre modèle.

Et pour finir, nous allons clôturer notre mémoire par une conclusion générale qui résume notre travail et propose des améliorations.

Chapitre 1

Présentation de la maladie

1.1 Introduction

Le syndrome de démence touche environ 6,4% de la population mondiale de plus de 65 ans. La prévalence de cette classe, où l'âge est l'un des principaux facteurs de risque, augmente régulièrement à mesure que l'espérance de vie augmente.

En France, le nombre de personnes atteintes du syndrome démentiel était estimé à 1,2 million en 2014. Le nombre de cas de démence reconnus par l'Organisation mondiale de la santé (OMS)[2] comme une priorité mondiale de santé publique Global pourrait doubler d'ici 2030 et tripler d'ici 2050.

La démence est la principale cause des maladies neurodégénératives. Parmi elles, la maladie d'Alzheimer (MA) qui est prédominante. Seulement dans plus de 70% des cas de syndromes de démences, la maladie neurodégénérative la plus courante qui survient après la MA est la Démence à Corps de Lewy (10-26% des cas), dégénérescence Lobaire Fronto-Temporale (FTLD : 10-15% des cas), suivi d'autres syndromes rares.

1.2 Démences neurodégénératives

Les maladies neurodégénératives sont des maladies à progression lente caractérisées par une perte de la fonction ou de la structure neuronale conduisant à la mort des cellules neuronales, elles sont souvent associées à un dépôt anormal de protéines.

La présentation clinique de la démence était très variable (diagnostic du syndrome) ainsi que les lésions histologiques retrouvées (diagnostic de lésions, correspondant à un dépôt protéique anormal). Il n'y avait pas de corrélation forte entre le diagnostic du syndrome et le diagnostic de la lésion, en revanche, les lésions histologiques peuvent entraîner des différentes manifestations cliniques.

En pratique, la classification des troubles cognitifs s'établit en intégrant non seulement la présentation clinique mais aussi des facteurs intégrant les anomalies histologiques lorsque cela est possible.

Par exemple, le diagnostic de la maladie d'Alzheimer repose actuellement sur une combinaison de présentation clinique(symptômes et signes associés à un déclin cognitif) et de biomarqueurs positifs (marqueurs reflétant, du vivant du patient, les lésions cérébrales caractéristiques de la maladie) sachant que 100 % des patients ont des biomarqueurs positifs.

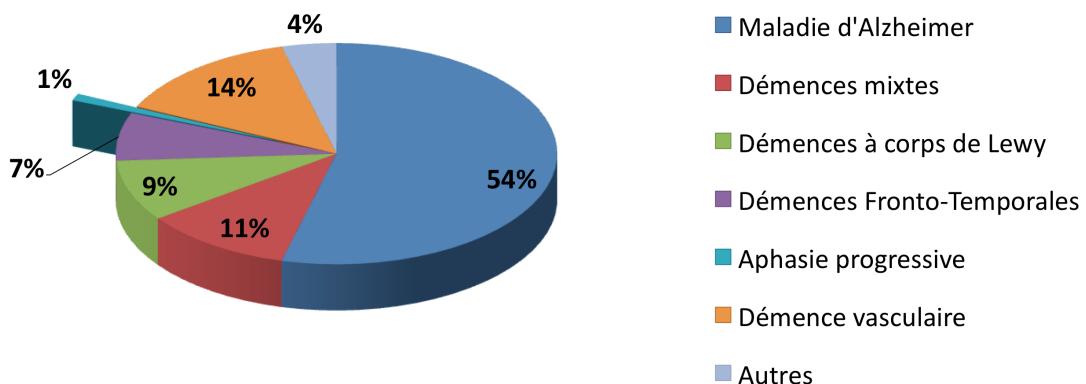


FIGURE 1.1 – Répartition diagnostique des démences : Centre Mémoire Lille-Bailleul (1992-2001) (échantillon de 4212 Patients) [3]

1.3 Maladie d'Alzheimer (MA)

La maladie d'Alzheimer est décrite pour la première fois en 1901 par Alois Alzheimer (1864-1915), un psychiatre et neuropathologiste allemand. Il a identifié le premier cas de la maladie, qui porte son nom, chez sa patiente de 51 ans, Auguste Deter. Il a suivi son cas jusqu'à la mort de la patiente en 1906 et a décrit les altérations anatomiques observées sur le cerveau de cette dernière.[9]

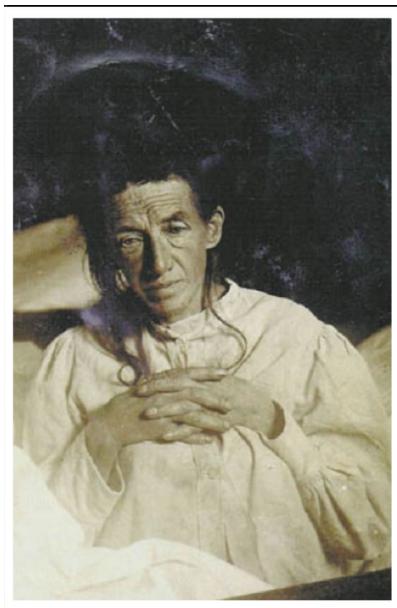


FIGURE 1.2 – Première patiente de 51 ans, Auguste Deter

Cette maladie provoque une perte progressive des cellules nerveuses dans les zones du cerveau qui gèrent certaines capacités, telles que la mémoire, le langage, le raisonnement ou l'attention. Les cellules perdent leur fonction puis meurent. Lorsqu'elles disparaissent, les neurones ne peuvent plus programmer efficacement certaines actions, et des pertes de mémoire, des troubles du comportement et d'autres symptômes apparaissent alors.[4]

La maladie d'Alzheimer est souvent associée à la perte de mémoire. En effet, ce sont les neurones localisés dans la région de l'hippocampe, siège de la mémoire, qui sont atteints en premier par le phénomène de dégénérescence. Mais ce n'est pas tout. Au fur et à mesure de son évolution, la maladie touche d'autres zones du cerveau, compliquant la capacité à communiquer, à réaliser plusieurs choses en même temps ou les actes de la vie quotidienne.

La MA entraîne deux types de lésions du système nerveux central, le dysfonctionnement d'une protéine essentielle aux neurones appelée "Tau" et l'apparition de plaques dites « séniles » due à une autre protéine "bêta amyloïde".

Peu à peu, ces lésions (Atrophie de certaines régions cérébrales) se multiplient et envahissent les régions supérieures du cerveau. La maladie devient alors de plus en plus visible. C'est généralement à ce stade que le diagnostic est posé et certaines facultés sont altérées et réduisent peu à peu l'autonomie du cerveau de la personne.

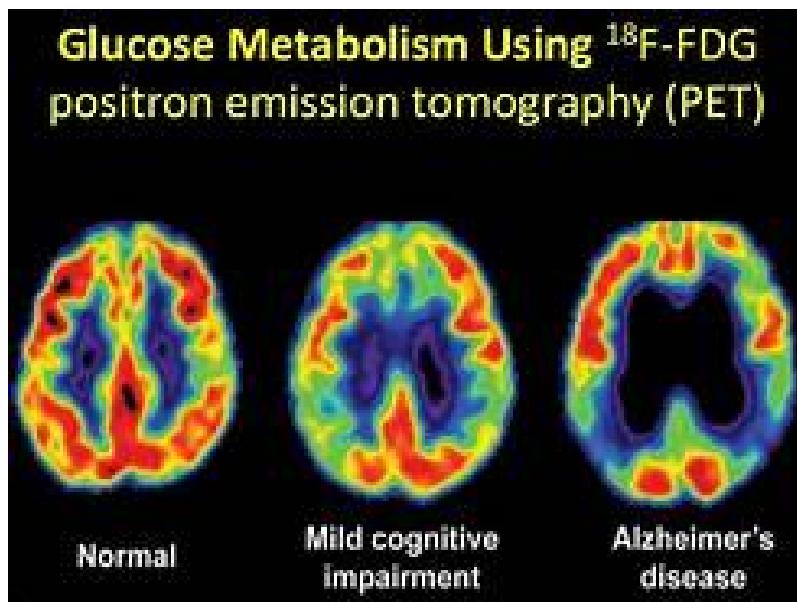


FIGURE 1.3 – IRM et la représentation du tau de fluorodésoxyglucose (18F) sur un sujet normal, un patient souffre de mild cognitive impairment MCI et un patient souffrant de maladie alzheimer [5]

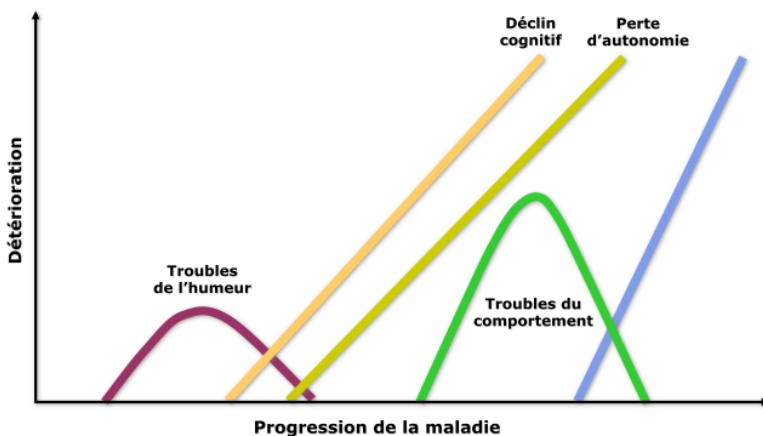


FIGURE 1.4 – Progression de la MA [6]

La maladie d’Alzheimer ne peut pas être la cause directe de la mort mais les complications indirectes, notamment les infections réduisent l’espérance de vie des personnes qui en sont atteintes. Les troubles de la déglutition augmentent par exemple le risque de développer une pneumonie qui est l’une des causes les plus fréquentes de décès chez les personnes atteintes de la maladie d’Alzheimer.[7]

1.4 Symptômes et approche diagnostique

La corrélation entre l’évolution clinique de la maladie d’Alzheimer et la découverte de marqueurs biologiques mesurables ont conduit à la révision des anciens critères diagnostiques et à la définition de la MA comme une entité biologique clinique.

Sur le plan clinique, la MA évolue généralement en 3 phases :

- La phase pré-démentielle (début de la phase symptomatique) : comprend l’apparition de troubles de la mémoire (pertes de mémoire).
- La phase d’état ou de démence (légère à sévère) : L’apparition d’un dysfonctionnement instrumental plus ou moins sévère (un syndrome appelé « aphaso-apraxo-agnosique » qui entraîne une perte d’autonomie(l’incapacité d’assurer seul certains actes de la vie).
- La phase de démence très sévère : perte complète d’autonomie avec altération sévère des fonctions cognitives, troubles du comportement (agitation ou apathie, hallucinations, déambulation, troubles du sommeil et de l’appétit, dysrégulation sphinctérienne) et autres symptômes neurologiques (épilepsie, myoclonies, syndromes pyramidaux et extrapyramidaux...etc). La mort résulte le plus souvent par des complications courantes de l’alimentation ainsi un changement sur l’anatomie du cerveau.

Sur le plan biologique et morphologique, trois conséquences de la disparition massive de cellules dans tout le cerveau au stade avancé de la maladie d’Alzheimer apparaissent :

- Le cortex se recroqueille et endommage les régions associées à la pensée, à la planification et à la mémoire.

- Le rétrécissement est particulièrement marqué dans l'hippocampe, une région du cortex qui joue un rôle essentiel dans la formation de nouveaux souvenirs.
- Les ventricules (des espaces remplis de fluide à l'intérieur du cerveau) grossissent.

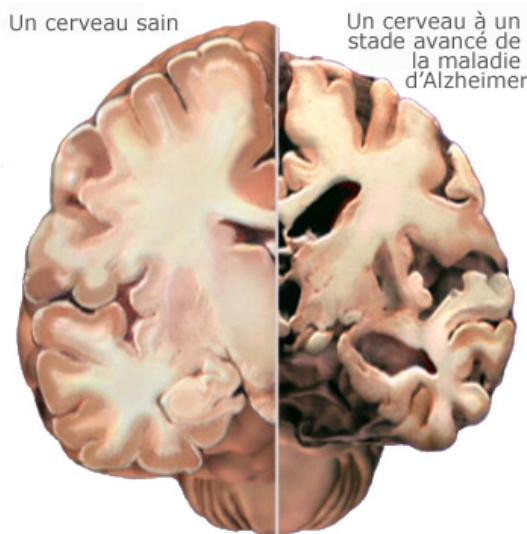


FIGURE 1.5 – Une coupe transversale du cerveau effectuée en son centre, entre les oreilles, d'un cerveau atteint d'Alzheimer

Pour les trois phases, il existe plusieurs méthodes pour le diagnostic de la maladie d'Alzheimer et parmi elles, on trouve :

- Bilan neuropsychologique.
- L'Imagerie par Résonnance Magnétique(IRM).
- La Tomographie par émission de positons (TEP).
- La Ponction Lombaire(PL).
- Examens laboratoire.

1.5 Imagerie et maladie d'Alzheimer

Actuellement, pour dépister la maladie d'Alzheimer, l'atrophie cérébrale représente un marqueur topographique par sa bonne corrélation avec les dégénérescences neuro-fibrillaires. L'IRMf et L'IRM, par leurs résolution spatiale, permettent l'évaluation la plus réaliste des zones atrophiques.

Les zones classiquement observées sont le cortex entorhinal, l'hippocampe et le cortex cingulaire postérieur, associés à une lésion hippocampique (mémoire ou circuits papes). Dans des formes plus évoluées, l'atrophie touche le néocortex des lobes frontal, temporal et pariétal.

Par les méthodes de l'intelligence artificielle, le traitement de ces images médicales et ces analyses récupérées par l'IRMf et l'IRM nous permettent de diagnostiquer la maladie à une phase de déclin modéré.

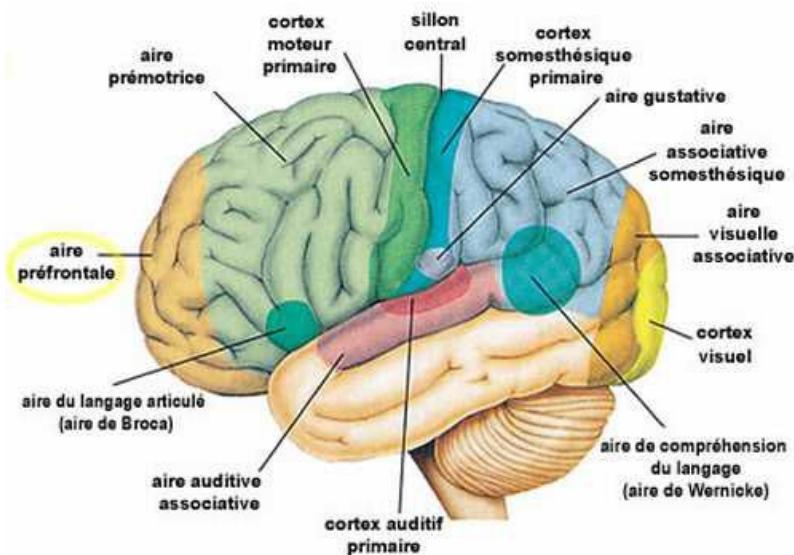


FIGURE 1.6 – Les différents régions d'intérêts visualiser à l'aide de l'imagerie médicale [8]

1.6 Conclusion

Aujourd’hui, les moyens mis à notre disposition, grâce aux avancées de la recherche et de la technologie, permettent d’établir un diagnostic non équivoque de la maladie d’Alzheimer du vivant du patient. L’intelligence artificielle aide déjà les patients à être plus autonome dans leur parcours de soins et à améliorer leur qualité de vie au quotidien.

Chapitre 2

État de l'art

2.1 Introduction

Ce chapitre présente des notions fondamentales et les méthodes de l'intelligence artificielle comme il décrit le rôle important de cette dernière durant le diagnostic et les différentes méthodes utilisées pour faire un diagnostic précoce de la maladie d'Alzheimer et la modélisation du cerveau humain.

2.2 Intelligence artificielle

L'intelligence artificielle (IA) est l'étude des algorithmes qui donnent aux machines la capacité de raisonner, d'exécuter des fonctions et de reproduire des comportements liés au humains. Cette dernière a été appliquée à divers aspects de la médecine, allant d'applications largement diagnostiques en radiologie et en pathologie à des applications plus thérapeutiques et interventionnelles en cardiologie et en chirurgie.

L'IA fait partie de notre vie quotidienne et parmi les exemples qui existent, nous citons : les achats en lignes, la recherche en lignes, la traduction automatique, les maisons connectées, le transport, la détection des maladies...etc.

Grâce à l'IA, plusieurs tâches qui étaient impossibles auparavant sont possible maintenant.

2.3 Apprentissage automatique

L'apprentissage automatique (ou machine learning) est une sous-catégorie de l'intelligence artificielle. Elle consiste à laisser des algorithmes découvrir des motifs modèle "patterns" et à savoir des motifs récurrents dans les ensembles de données.

Pour donner à un ordinateur la capacité d'apprendre, on utilise des méthodes d'apprentissage qui sont fortement inspirées de la façon dont nous, les êtres humains, ap-

prenons à faire des choses. Parmi ces méthodes, on compte :

- L'apprentissage supervisé (Supervised Learning). [10]
- L'apprentissage non supervisé (Unsupervised Learning). [11]

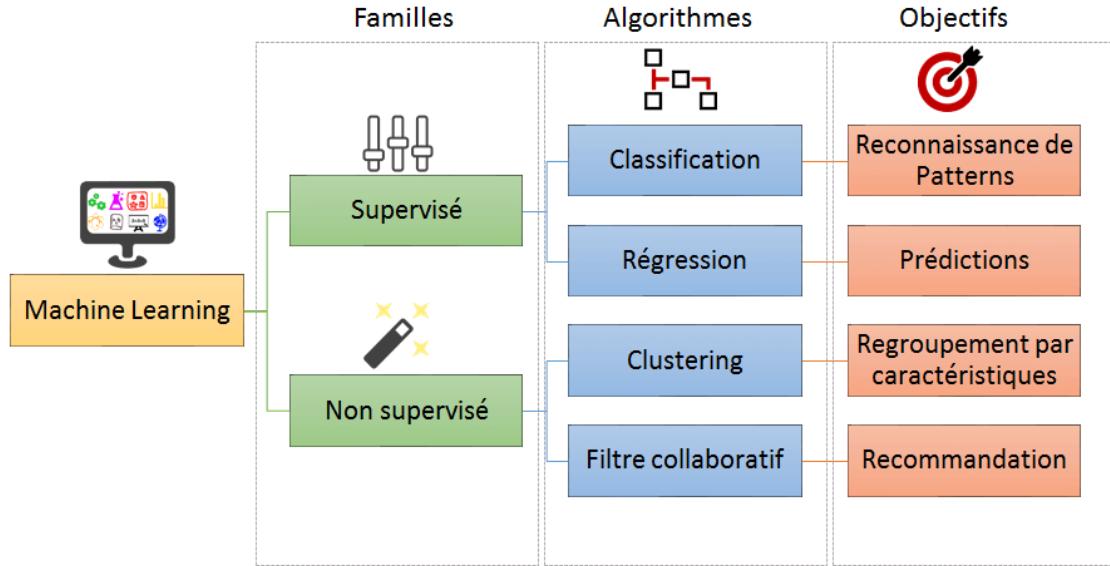


FIGURE 2.1 – Les familles de machine learning

2.4 Support Vector Machine

Le Support Vector Machine (SVM) est un ensemble de techniques d'apprentissage supervisé destinées à résoudre des problèmes de discrimination et de régression et c'est un algorithme puissant qui trouve un plan multidimensionnel qui sépare des classes distinctes. Dans la figure 2.2, nous avons deux classes désignées par des couleurs rouge et bleue. La ligne noire qui les sépare est l'hyperplan. Le SVM trouve l'hyperplan tel que la marge (la distance entre les deux lignes pointillées) soit maximale. Les points de données situés près des lignes pointillées sont appelés les vecteurs de support. Ils sont fortement responsables de la formation de l'hyperplan. Nous utilisons la méthode d'optimisation des multiplicateurs de Lagrange pour trouver cet hyperplan.[15]

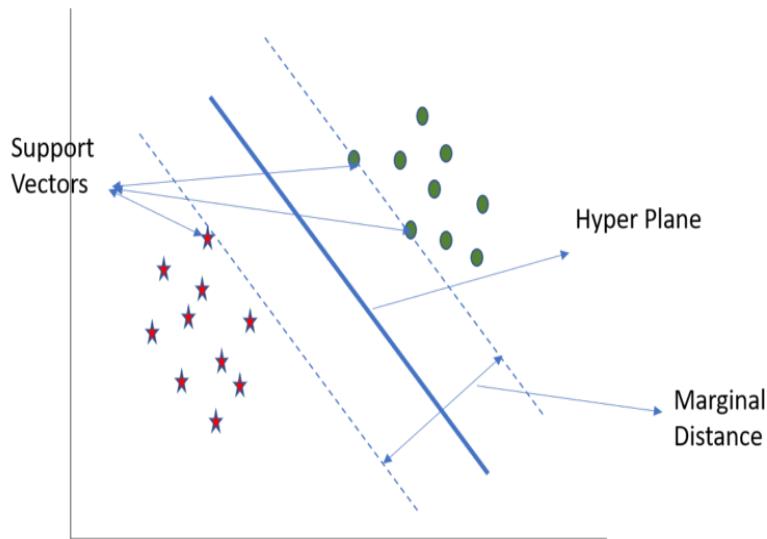


FIGURE 2.2 – Classes séparées par hyperplan

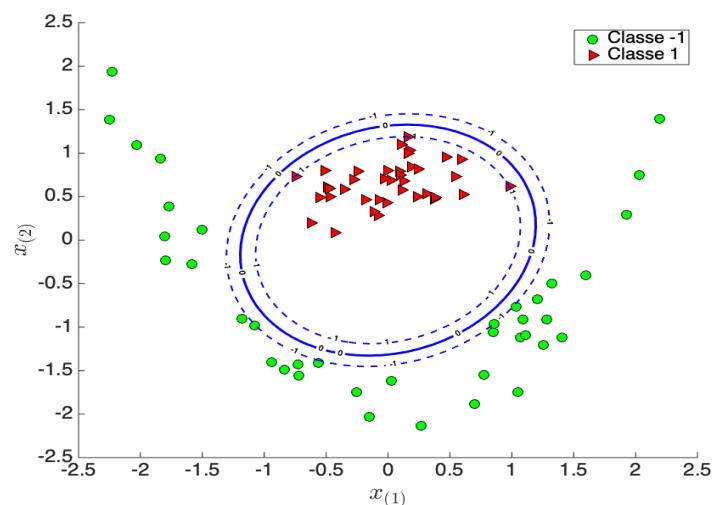


FIGURE 2.3 – Jeu de données avec deux classes, bleu et rouge non linéairement séparable

Généralement, les ensembles de données ne sont pas linéairement séparables (voir la figure 2.3).

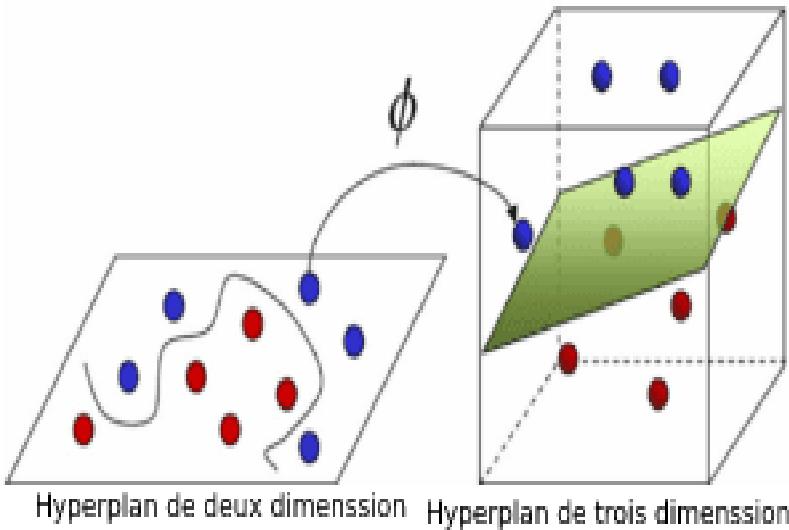


FIGURE 2.4 – Données projetées dans une dimension supérieure et séparées par un plan

Nous projetons, dans ce cas, les données dans une dimension supérieure (trois dimensions, figure 2.4) et nous appliquons la méthode SVM pour trouver le plan qui sépare les deux classes.

Les différents hyperparamètres de la méthode SVM sont :

- Noyau : Un noyau permet d'obtenir la bonne fonction de mappage. Dans cette étude nous utilisons le noyau de la fonction de base radiale (RBF).
- C : C'est un paramètre de régularisation. Il fait un compromis entre la précision de l'entraînement et la largeur de la marge. Une diminution de C entraîne des marges plus grandes et une précision d'entraînement plus faible, et vice versa.
- Gamma : Gamma (γ) définit l'influence des points d'entraînement. Ceci établit une relation intéressante entre les hyperparamètres C et Gamma. Généralement, nous recherchons les valeurs de Gamma et C sur une grille logarithmique de 10^{-3} à 10^3 en utilisant une validation croisée 5-folds.
- Degré : cet hyperparamètre est utilisé uniquement dans les noyaux polynomiaux ; un degré plus élevé signifie une limite de décision plus flexible. Le degré 1 donnerait un noyau linéaire.

2.5 Apprentissage profond

L'apprentissage profond (ou deep learning) est un sous-domaine de l'intelligence artificielle. Ce dernier, désigne l'ensemble des techniques d'apprentissage automatique (machine learning), autrement dit, une forme d'apprentissage fondée sur des approches mathématiques, utilisées pour modéliser des données. Cette réflexion va donner naissance à une machine qui communique et se comporte en fonction des informations stockées.

Le deep learning est un système avancé basé sur le cerveau humain, qui comporte un vaste réseau de neurones artificiels (voir figure 2.5). Ces neurones sont inter-connectés pour traiter et mémoriser des informations, comparer des situations quelconques avec des situations similaires passées, analyser les solutions et résoudre le problème avec la meilleure façon possible.

Comme chez les êtres humains, le deep learning consiste à apprendre des expériences vécues ou, dans le cas des machines, des informations enregistrées.

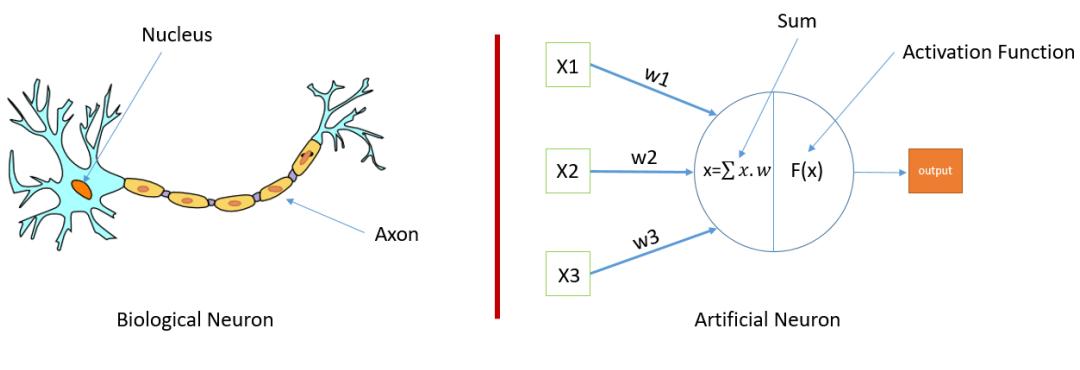


FIGURE 2.5 – Architecture d'un réseau de neurones

2.6 Réseau neuronal graphique

La partie la plus fondamentale du réseau neuronal graphique[12] (ou en anglais, GNN : Graph Neural Network) est un graphe.

En informatique, un graphe est une structure de données composée de deux éléments : les nœuds (sommets) et les arêtes. Un graphe G peut être défini comme $G = (V, E)$ où V est l'ensemble des nœuds et E sont les arêtes entre eux. S'il existe des dépendances directionnelles entre les nœuds, les arêtes sont dirigées, sinon les arêtes sont non dirigées.

On peut modéliser plusieurs problèmes de recherche complexe et de pathologie à l'aide des graphes tel que : le cerveau humain, les réseaux sociaux et les molécules.



FIGURE 2.6 – Un graphe orienté et un graphe non orienté

Cela permet de prendre en compte la complexité des interactions entre des données hétérogènes.

Un graphe est souvent représenté par A , une matrice d'adjacence. Si un graphe comporte n noeuds, A a une dimension de $(n \times n)$. Parfois, les noeuds possèdent un ensemble de caractéristiques (par exemple un profil d'utilisateur), si le noeud possède un nombre f de caractéristiques, la matrice X des caractéristiques du noeud a une dimension de $(n \times f)$.

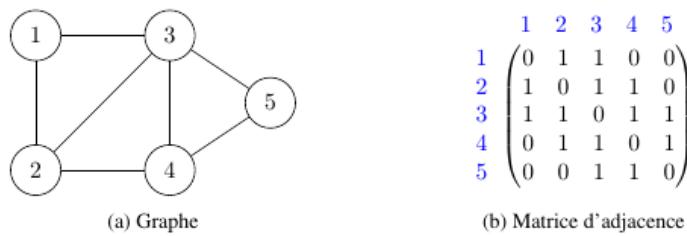


FIGURE 2.7 – Un graphe non orienté et sa représentation en matrice d'adjacence

Un GNN est une classe de réseaux de neurones pour le traitement de données mieux représentées par des structures de données de graphes. Ils ont été popularisés par leur utilisation dans l'apprentissage supervisé sur les propriétés de diverses molécules.

L'intuition des ces GNNs est que les nœuds sont naturellement définis par leurs voisins et leurs connexions. Pour comprendre cela, nous pouvons simplement imaginer que si nous supprimons les voisins et les connexions autour d'un nœud, alors le nœud perdra toutes ses informations. Par conséquent, les voisins d'un nœud et les connexions aux voisins définissent le concept de nœud.

Ayant cela à l'esprit, nous donnons ensuite à chaque nœud un état (x) pour représenter son concept. Nous pouvons utiliser l'état du nœud (x) pour produire une sortie (o), c'est-à-dire une décision sur le concept. L'état final (x_n) du nœud est normalement appelé noeud integre "node embedding". La tâche de tous les GNNs est de détermi-

ner le "node embedding" de chaque nœud, en regardant les informations sur ses nœuds voisins et cela en utilisant les KNNs.

2.6.1 K-plus proches voisins (KNN)

Le K-plus proches voisins (KNN : K Nearest Neighbours) est un algorithme d'apprentissage supervisé qui est une partie de machine learning. Il sert pour la regression et aussi la classification , la raison de son utilisation dans les CNNs (Convolutional Neural Network).¹ Son comportement peut être comparé à l'analogie suivante : *'dis moi qui sont tes voisins, je te dirai qui tu es.'*

Le KNN n'a pas besoin de créer un modèle prédictif, ainsi, pour ce dernier, il n'existe pas de phase d'apprentissage proprement dite. Pour cette raison, il est parfois classé dans la catégorie du apprentissage paresseux '*Lazy learning*'.

Pour pouvoir effectuer une prédiction, le KNN se base sur le jeu de données pour produire un résultat.

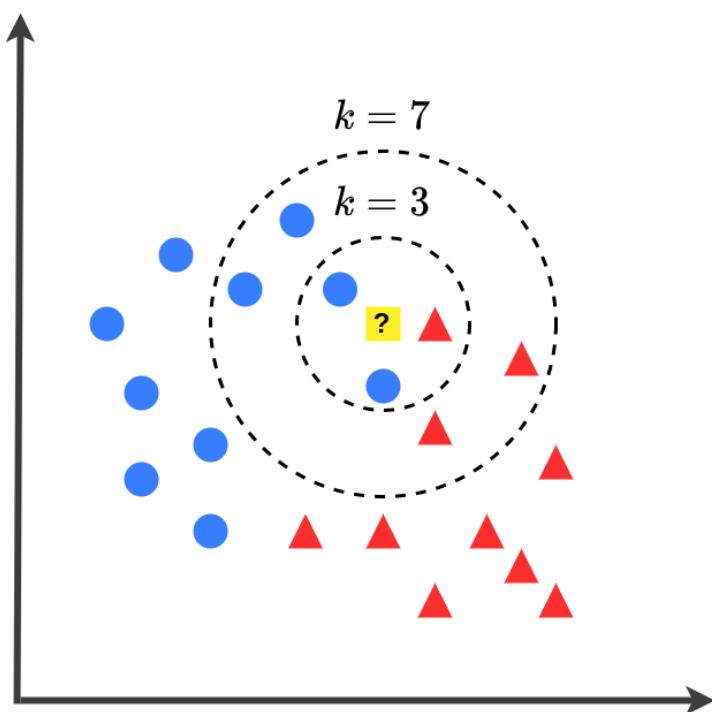


FIGURE 2.8 – Visualisation de k-Plus Proches Voisins avec deux classes (cercles bleus et triangles rouges) avec $k = 3$ et $k = 7$.

1. Un Cnn est un réseau de neurones d'apprentissage profond conçu pour traiter des tableaux structurés de données telles que des images. Couramment utilisés en vision par ordinateur tel la classification d'images.

2.6.2 Réseau convolutif spectral

Par rapport à d'autres types de GNNs, ce type de réseau de convolution de graphes a une base mathématique très solide. Le réseau convolutif spectral (ou en anglais, Spectral Convolutional Network) est construit sur la théorie du traitement du signal graphique et, par simplification et approximation de la convolution du graphe.

Par approximation polynomiale de Chebyshev [13], la convolution du graphe peut être simplifiée sous la forme suivante :

$$g_\theta * x \approx \sum_{k=0}^k \theta_k T_k(\Lambda) \quad (2.1)$$

Après une simplification supplémentaire, l'article GCN[14] suggère une structure de réseau neuronal à 2 couches, qui peut être décrite dans une équation comme ci-dessous :

$$Z = f(X, A) = \text{softmax}(\hat{A} \text{Relu}(\hat{A} X W^{(0)}) W^{(1)}) \quad (2.2)$$

où \hat{A} est le laplacien prétraité de la matrice de contiguïté du graphe d'origine A.

Cette formule est une structure à deux couches entièrement connectées qui est couramment utilisée mais il sert en effet de convolution de graphe.

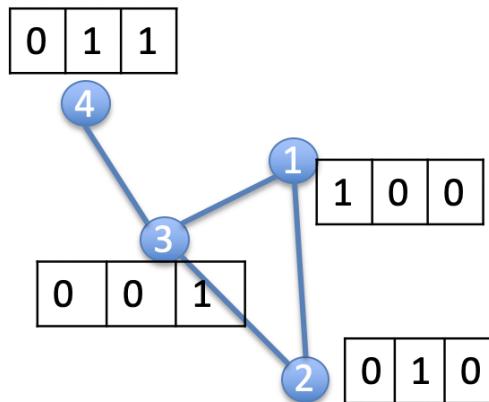


FIGURE 2.9 – Exemple de graphique avec une caractéristique attribuée à chaque nœud

Considérons que nous avons un graphe simple avec 4 noeuds. Chacun de ces noeuds se voit attribuer une matrice de fonctionnalités, comme illustré dans la figure 2.9. Il est facile de créer une matrice d'adjacence graphique et une matrice de caractéristiques comme indiqué ci-dessous :

La diagonale de la matrice d'adjacence est délibérément changée en '1' pour ajouter une boucle automatique pour chaque noeud. Il s'agit d'inclure la fonctionnalité de

<table border="1" style="border-collapse: collapse; width: 100px; height: 100px;"> <tr><td>1</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>1</td></tr> </table>	1	1	1	0	1	1	1	0	1	1	1	1	0	0	1	1	<table border="1" style="border-collapse: collapse; width: 100px; height: 100px;"> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> </table>	1	0	0	0	1	0	0	0	1	0	1	1
1	1	1	0																										
1	1	1	0																										
1	1	1	1																										
0	0	1	1																										
1	0	0																											
0	1	0																											
0	0	1																											
0	1	1																											
Adjacency matrix (A)	Feature matrix (X)																												

FIGURE 2.10 – Exemple de matrice de contiguïté et de matrice de caractéristiques, les lignes et colonnes représentent les noeuds

chaque nœud lui-même lorsque nous effectuons l'agrégation de fonctionnalités. Nous effectuons ensuite $A \times X$

$$A \times X = H$$

$$\begin{array}{|c|c|c|c|} \hline
 1 & 1 & 1 & 0 \\ \hline
 1 & 1 & 1 & 0 \\ \hline
 1 & 1 & 1 & 1 \\ \hline
 0 & 0 & 1 & 1 \\ \hline
 \end{array}
 \times
 \begin{array}{|c|c|c|} \hline
 1 & 0 & 0 \\ \hline
 0 & 1 & 0 \\ \hline
 0 & 0 & 1 \\ \hline
 0 & 1 & 1 \\ \hline
 \end{array}
 =
 \begin{array}{|c|c|c|} \hline
 \textcolor{yellow}{1} & \textcolor{yellow}{1} & \textcolor{yellow}{1} \\ \hline
 1 & 1 & 1 \\ \hline
 1 & 2 & 2 \\ \hline
 0 & 1 & 2 \\ \hline
 \end{array}$$

$$\begin{aligned}
 h_{1,1} &= A_{1,1}X_{1,1} + A_{1,2}X_{2,1} + A_{1,3}X_{3,1} + A_{1,4}X_{4,1} \\
 h_{1,2} &= A_{1,1}X_{1,2} + A_{1,2}X_{2,2} + A_{1,3}X_{3,2} + A_{1,4}X_{4,2} \\
 h_{1,3} &= A_{1,1}X_{1,3} + A_{1,2}X_{2,3} + A_{1,3}X_{3,3} + A_{1,4}X_{4,3}
 \end{aligned}$$

FIGURE 2.11 – Exemple de convolution de graphe par multiplication matricielle

Le résultatat de la multiplication matricielle est affiché dans la matrice la plus à droite. Regardons la fonctionnalité résultante du premier nœud à titre d'exemple, il n'est pas difficile de remarquer que le résultatat est une somme de la multiplication de toutes les fonctionnalités du [nœud 1] de la matrice A avec les fonctionnalités du [nœud 1] de la matrice X . Les fonctionnalités de [nœud 4] ne sont pas incluses car ce n'est pas le voisin du [nœud 1].

Mathématiquement, la matrice d'adjacence du graphe vaut '1' uniquement lorsqu'il y a une arête, et '0' sinon, cela fait de la multiplication matricielle la somme des caractéristiques des nœuds connectés au nœud de référence.

2.7 HGNN : Hypergraph Neural Network

Étant donné le défi d'apprendre à représenter réellement des données complexes, les hypergraphes qui sont des graphes complexes, sont suggérés. Un hypergraphe est un ensemble de sommets qui sont reliés entre eux par une arête complexe nommée 'hyperarête' (l'hyperarête apparaît dans la figure 2.12). Ce graphe est défini par $G = (V, E, W)$

où V sont les sommets, chaque hyper-arête se voit attribuer un poids par W et E sont les hyper-arêtes. Le degré de chaque noeud est défini comme suit :

$$d(v) = \sum_{e \in \epsilon} w(e)h(v, e) \quad (2.3)$$

Le degré de chaque hyper-arête est défini comme suit :

$$\Theta(e) = \sum_{v \in V} h(v, e) \quad (2.4)$$

tel que h est la matrice d'incidence.

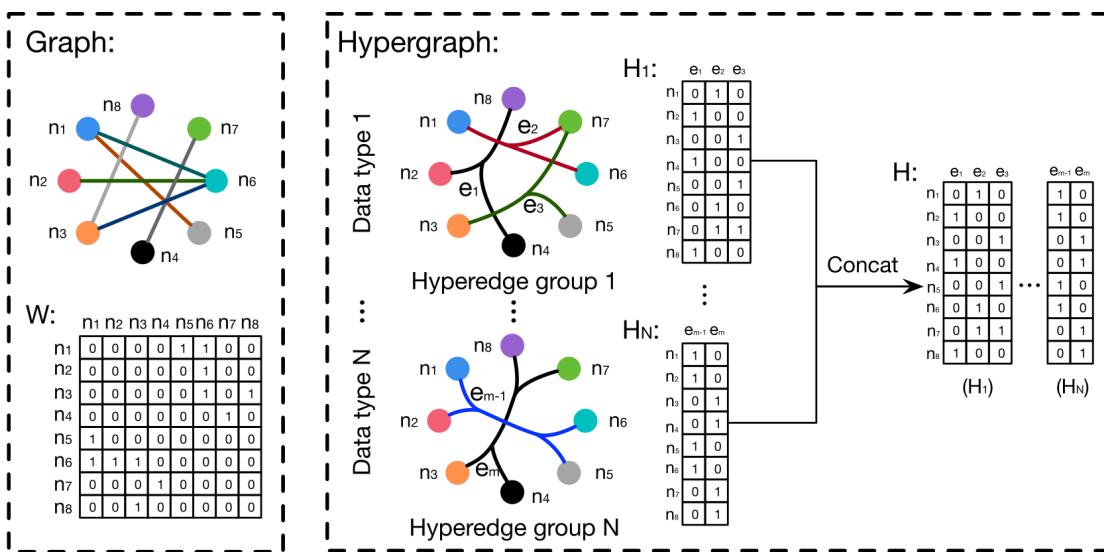


FIGURE 2.12 – Comparaison entre graphe et Hypergraphe

Les hypergraphes sont plus flexibles dans la modélisation des données, surtout lorsqu'il s'agit de données complexes et irrégulières. Par conséquent, les HGNNs sont la méthode utilisée car ils peuvent gérer des données complexes, contrairement aux GNNs parce que les HGNNs peuvent comparer plusieurs sommets en même temps ce qui n'est pas possible pour les GNNs qui comparent les sommets un par un.

2.8 Auto-Encodeur

Un auto-encodeur peut être considéré comme un réseau de neurones qui essaie de reproduire l'entrée comme sortie. Plus précisément, il a une représentation cachée intermédiaire , qui décrit un code pour indiquer l'entrée. L'auto-encoder se compose de deux composants :

- Un codeur $h = f(x)$ qui code l'entrée x en un code h .

- Un décodeur qui reconstruit x à partir du code h . Le décodeur peut être représenté par $\hat{x} = g(h)$

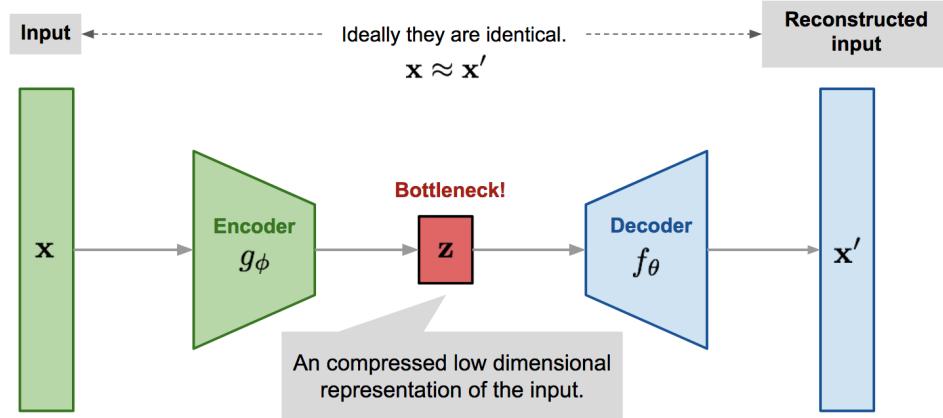


FIGURE 2.13 – Auto-encodeur avec goulot d'étranglement "Bottleneck"

Si un auto-encodeur fonctionne parfaitement pour reproduire l'entrée, il n'est pas particulièrement utile car au lieu de cela, les auto-encodeurs doivent reproduire approximativement l'entrée en incluant certaines restrictions ; en d'autres termes, ils doivent compresser les informations nécessaires de l'entrée dans le code caché pour reproduire une sortie satisfaisante.

Comme la figure 2.13 le montre, l'entrée x est poussée à travers un 'goulot d'étranglement' (Bottleneck)², celui-ci contrôle les informations qui peuvent être conservées dans le code h . Le réseau décodeur utilise ensuite le code h pour sortir \hat{x} où \hat{x} reconstruit l'entrée x .

La conception du 'goulot d'étranglement' est importante pour les auto-encodeurs car sans ce dernier, un auto-encodeur peut simplement apprendre à mémoriser l'entrée et à la transmettre au décodeur pour la reproduire, ce qui peut rendre l'auto-encodeur inutile.

2. A bottleneck est un module qui contient les représentations de connaissances compressées et qui est donc la partie la plus importante du réseau.

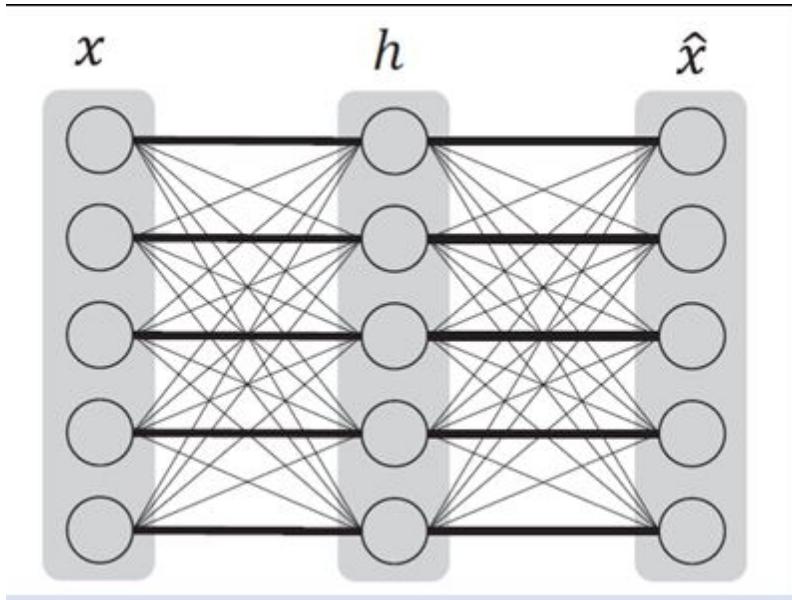


FIGURE 2.14 – Auto-encoder sans goulot d'étranglement

Il existe plusieurs façons de concevoir le goulot d'étranglement et cela en ajoutant des contraintes à l'auto-encodeur. La manière naturelle est de limiter le nombre de dimensions dans le code h , résultant en un auto-encodeur incomplet. Nous pouvons également ajouter un terme de régularisation pour empêcher la mémorisation entre l'entrée et la sortie, résultant en un auto-encodeur régularisé.

2.8.1 Auto-encodeur incomplet

Limiter le nombre de dimensions du code h à l'entrée x est un moyen simple et naturel de concevoir le goulot d'étranglement. Un auto-encodeur dont la dimension du code est inférieure à la dimension de l'entrée est appelé un auto-encodeur incomplet.

La figure 2.13 est un exemple illustrant un auto-encodeur incomplet où l'encodeur et le décodeur ne contiennent qu'une seule couche du réseau et la couche cachée a moins d'unités que la couche d'entrée. En minimisant l'erreur de reconstruction, le modèle peut conserver les caractéristiques les plus importantes de l'entrée dans le code caché.

2.8.2 Auto-encodeur régularisé

Nous pouvons rendre l'auto-encodeur plus profond en empilant plus de couches pour l'encodeur et le décodeur. Pour les auto-encodeurs profonds, il faut veiller à limiter leur capacité car si l'encodeur et le décodeur sont trop volumineux, les auto-encodeurs peuvent ne rien apprendre d'utile.

Pour empêcher l'auto-encodeur d'apprendre une fonction d'identité, nous pouvons insérer le terme de régularisation suivant dans la fonction de perte de l'auto-encodeur :

$$l(x; g(f(x))) + \eta \Omega(h)$$

Où $\Omega(h)$ est le terme de régularisation appliqué au code h et η est un hyperparamètre qui contrôle l'impact du terme de régularisation.

2.9 Conclusion

Au long de ce chapitre, nous avons vue c'est quoi l'intelligence artificielle, le machine learning et le deep learning comme nous avons vue l'utilité des graphes et comment ils servent à la détection d'Alzheimer.

Chapitre 3

Conception et architecture du modèle

3.1 Introduction

L'intégration de graphes en imagerie médicale est un moyen puissant de représenter des données neurologiques (dans notre étude, celles du connectome cérébral)¹ dans un espace de faible dimension pour cartographier, prédire et classer la connectivité cérébrale. Cependant, les algorithmes d'intégration existants ont deux limites principales.

Premièrement, pour les graphes, l'accent est mis sur le maintien des relations topologiques un à un entre les nœuds (régions d'intérêt (ROI)), alors que les relations entre les nœuds plusieurs à plusieurs sont plus significatives en imagerie médicale (ensemble à ensemble) ; ce qui nous a poussé à associer la notion des hypergraphes enregistrables (Hyper Connectome).

Deuxièmement, les techniques d'intégration de graphes existantes ne peuvent pas être facilement adaptées aux données des graphes à vues multiples avec une distribution non uniforme.

Dans ce chapitre, nous étudions l'intégration potentielle profonde des graphes cérébraux multi-vues spécifiques à un sujet, croisant l'apprentissage profond antagoniste² et la théorie des hypergraphes, et donnant une technique d'apprentissage profond géométrique permettant de réaliser la classification multi-vues.

1. Un connectome cérébral est une carte complète des connexions cérébrales, il est nécessaire pour interpréter les mesures des signaux transmis entre les neurones. Dans le cerveau humain, ces signaux circulent dans un réseau complexe de dizaines de milliards de neurones, dont chacun est relié à 10 000 autres, via les axones qui constituent les faisceaux de fibres de la matière blanche.

2. L'apprentissage profond antagoniste est une méthode d'apprentissage automatique qui vise à tromper les modèles d'apprentissage profond en fournissant des informations trompeuses. Par conséquent, il inclut à la fois la génération et la détection d'exemples contradictoires, qui sont des entrées spécialement créées pour tromper les classificateurs.

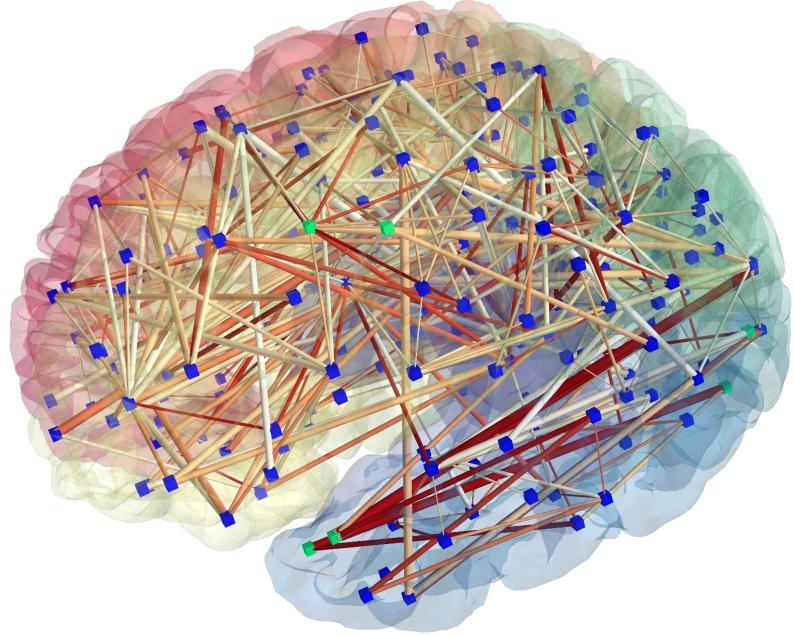


FIGURE 3.1 – Connectome cérébrale

3.2 Position du problème

Soit $G^s = \{G_1, \dots, G_M\}$ un ensemble de connectomes cérébraux multi-vues du sujet s dans la population G , comprenant des graphes cérébraux M entièrement connectés où G_m représente le graphe cérébral dérivé de la mesure m (par exemple, corrélation dans l'activité neuronale ou similitude dans la morphologie).

Chaque graphe cérébral $G_m = (V, E, X_m) \in G^s$ capture une vue connexionnelle du câblage cérébral, où V désigne un ensemble de N ROI cérébrales, E un ensemble d'arêtes reliant des paires de nœuds et $X_m \in R^{N \times N}$ désigne une matrice de connectivité cérébrale symétrique codant la relation par paires entre les ROI cérébrales.

Notre objectif est d'apprendre la représentation latente discriminante et représentative de chaque ensemble G_s en capitalisant sur la structure de l'hypergraphe et le champ naissant de l'apprentissage géométrique profond.

3.3 Proposition d'Hyper-Connectome Auto-Encoder (HCAE) pour la classification de l'état du cerveau

3.3.1 Définition et construction d'hyper-connectome multi-vues

Étant donné un graphe cérébral $G_m = (V, E, X_m) \in G^s$ du sujet s , nous définissons un hyper-connectome de vue m $H_m^s = (V, E_m, W_m^s)$ comme un ensemble de nœuds V et un ensemble d'hyper-arêtes E_m où chaque hyper-arête se voit attribuer un poids

unitaire dans la matrice de poids diagonal

$$W_m \in R^{|E_m| \times |E_m|}.$$

Fondamentalement, nous construisons l'hypergraphe en connectant chaque nœud v dans V à ses k nœuds voisins les plus proches, définissant ainsi une hyper-arête $e \in E_m$. Ensuite, nous créons une matrice d'incidence d'hyper-connectome spécifique à la vue $H_m^s \in R^{|V| \times |E_m|}$, dont les entrées sont définies comme suit :

$$H_m^s(e, v) = \begin{cases} 1, & \text{si } v \in e \\ 0, & \text{si } v \notin e \end{cases} \quad (3.1)$$

pour $v \in V$ et $e \in E_m$.

De plus, chaque nœud stocke un ensemble d'entités qui représente la i_{me} ligne de la matrice de connectivité graphique X_m^s pour préserver la relation entre les différents nœuds puisque notre matrice d'incidence est binaire.

En fin de compte, en construisant un hyper-connectome pour chaque vue m , nous créons un hyper-connectome multi-vues H^s empilant horizontalement des matrices d'incidence à partir de différentes vues (voir figure 3.1) :

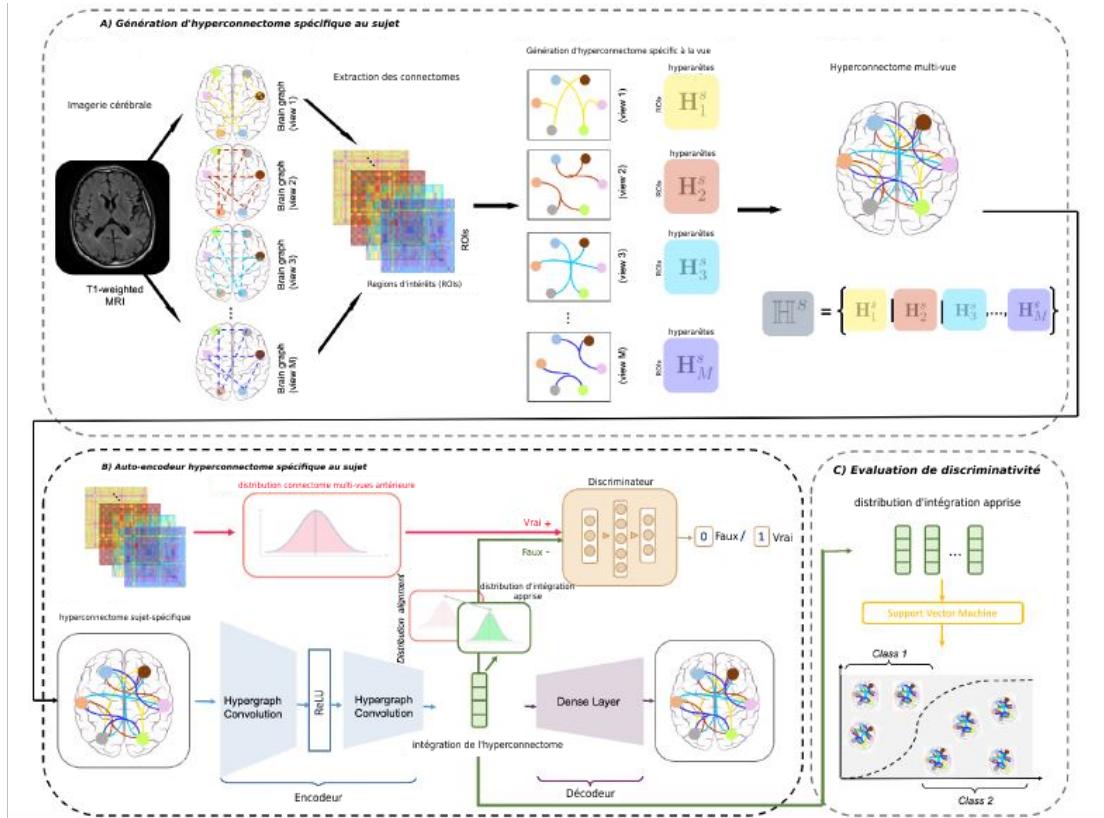


FIGURE 3.2 – Architecture d'auto-codage de l'hyper-connectome (AEHC) pour la classification des états du réseau du cerveau

$$H^s = H_1^s | H_2^s | H_M^s | \quad (3.2)$$

En parallèle, nous empilons également les matrices de connectivité de toutes les vues comme suit :

$$X_s = X_1^s | X_2^s | X_M^s \quad (3.3)$$

3.3.2 Architecture d'hyper-connectome spécifique au sujet

Pour chaque sujet,

- A) Nous générerons un ensemble de connectomes cérébraux multi-vues, chacun capturant une vue connexionnelle du cerveau (par exemple, la fonction ou la morphologie). À partir de chaque connectome cérébral, nous construisons un hyper-connectome spécifique à la vue qui se compose d'hyper-arêtes, construits en connectant chaque nœud (c'est-à-dire une région d'intérêt (ROI)) à ses nœuds voisins les plus proches (en utilisant les KNNs). En fin de compte, nous empilons horizontalement les matrices d'incidence à partir de vues distinctes pour créer un hyper-connectome cérébral multi-vues, qui capture la connectivité d'ordre élevé à partir de différentes vues cérébrales complémentaires.
- B) HCAE est formé d'une manière spécifique à la matière, c'est-à-dire pour chaque individu, indépendamment. Ensuite, nous construisons et entraînons le modèle HCAE afin d'extraire des représentations de faible dimension de notre hyper-connectome. L'architecture se compose de trois composants principaux :
 - 1) L'encodeur
 - 2) Le décodeur
 - 3) Le discriminateur
- C) Évaluer la discrimination des intégrations apprises dans la distinction entre les états cérébraux à l'aide de Support Vector Machine (SVM) avec noyau de la fonction de base radiale [16]. Nous utilisons ici les hyper-connectomes multi-vues codés pour entraîner un classificateur SVM à classer un cerveau en deux états (par exemple, la maladie d'Alzheimer et les troubles cognitifs léger).

3.3.3 Les hyperparamètres du HCAE

L'encodeur se compose de deux couches convolutives hypergraphiques empilées. Les deux couches utilisent l'opération convolutionnelle hypergraphique traditionnelle [17]. Une fonction d'activation d'unité linéaire rectifiée (ReLU)³ est utilisée après la première couche, et une fonction d'activation linéaire est ajoutée après la seconde.

³. ReLU (Rectified Linear Units) désigne la fonction réelle non-linéaire définie par : $\text{ReLU}(x)=\max(0,x)$. La couche de correction ReLU remplace donc toutes les valeurs négatives reçues en entrée par des zéros, cette dernière agit comme une fonction d'activation.

$$Y^{(l)} = \varphi(D_v^{-\frac{1}{2}} H^s W_m D_e^{-1} (H^s)^T D_v^{-\frac{1}{2}} Y^{l-1} \Theta^l) \quad (3.4)$$

Où

- $d(v) = \sum_{v \in V} W_m H_m^s(e, v)$ et $\delta(v) = \sum_{v \in V} H_m^s(e, v)$ sont respectivement le degré du sommet v et du degré d'arête e .
- $D_v \in R^{|V| \times |v|}$ et $D_e \in R^{|R| \times |E|}$ sont respectivement les matrices diagonales des degrés de sommet et des degrés d'arête.
- W_m est la matrice de poids de l'hyper-arête.
- $\Theta^{(l)}$ représente le filtre appris (c'est-à-dire les poids appris des couches de convolutions hypergraphiques) appliqué à tous les nœuds hyper-connectome pour extraire les caractéristiques connexions d'ordre élevé.
- Y^{l-1} représente l'entrée produite par la couche précédente. $Y^{(0)} = X^s$ désigne la matrice de connectivité cérébrale multi-vues du sujet.
- φ représente la fonction d'activation (ReLU).

La fonction perte est définie comme suit :

$$L_0 = E_{q(Z|X^s, H^s)}[\log P(\tilde{H}^s|Z)] \quad (3.5)$$

En nous inspirant de [18], nous utilisons la régularisation contradictoire pour forcer les intégrations latentes de faible dimension à correspondre à la distribution des réseaux cérébraux multi-vues d'entrée (Fig. 3.1). Plus précisément, nous intégrons un discriminateur accusatoire sous la forme d'un perceptron multi-couches (MLP) comprenant plusieurs couches denses empilées.

Un discriminateur D , qui est principalement utilisé dans les réseaux antagonistes génératifs (GAN) est généralement formé pour faire la distinction entre les échantillons réels et les faux échantillons générés par un autre réseau connu sous le nom de générateur G . Dans notre cas, nous intégrons la perte du discriminateur dans le modèle afin de forcer l'intégration apprise en basse dimension d'un hyper-connectome pour mieux s'aligner sur la distribution antérieure des réseaux cérébraux multi-vues d'entrée d'un seul sujet :

$$-\frac{1}{2} E_{z \sim P_z}[\log D(Z)] - \frac{1}{2} E_{X^s}[\log(1 - D(G(X^s, H^s)))] \quad (3.6)$$

En tant que tel, nous créons une forme de régularisation qui propulse un meilleur auto-encodage en résolvant le problème d'optimisation min-max suivant :

$$\min_G \max_D E_{Z \sim P_z}[\log D(Z)] + E_{Z \sim P_z}[\log(1 - D(G(X^s, H^s)))] \quad (3.7)$$

Le décodeur reconstruit l’hyper-connectome cérébral multi-vues spécifique au sujet à partir de l’intégration apprise à travers une seule couche dense. Le discriminateur qui est utilisé pour la régularisation de la distribution antagoniste est composé de couches denses.

3.3.4 Classification par la méthode SVM

la méthode machine à vecteurs de support (SVM : support vector machine) est utilisée pour évaluer la discriminativité des incorporations apprises dans la distinction entre les états cérébraux, connus pour leurs solides garanties théoriques, leur grande flexibilité ainsi que leur simplicité d’utilisation même sans grandes connaissances de data mining.

La perte que nous calculons à partir des données de validation sera sensible au choix des données dans l’ensemble de validation. Ceci est particulièrement une problématique si cet ensemble est petit. La validation croisée est une technique qui nous permet d’utiliser plus efficacement les données qui nous permettent de déterminer les hyperparamètres qui caractérisent la méthode SVM (C, Gamma).

La validation croisée K-plis[19] divise les données en K blocs de tailles égales (ou aussi proche que possible) de tailles. Chaque bloc devient à son tour un ensemble de validation pour un ensemble d’apprentissage composé des autres blocs K-1. La moyenne des K valeurs de perte résultantes nous donne notre valeur de perte finale. Un cas extrême de validation croisée K-fold est quand $K = N$ où N est le nombre d’observations dans notre ensemble de données, chaque observation de données est conservée à son tour et utilisée pour tester un modèle formé sur les $N-1$ autres objets.

3.4 Conclusion

HCAE fonctionne avec plusieurs connectomes cérébraux, il utilise l’apprentissage contradictoire pour l’auto-codage des hyperconnectomes à l’aide de la couche de convolution hypergraphique et il s’est avéré expérimentalement identifiable dans les connexions cérébrales d’ordre supérieur.

Cette architecture fait progresser le domaine des neurosciences des réseaux à la pointe de l’hyperconnectivité dans le but de présenter une représentation globale des différents aspects du connectome cérébral.

Chapitre 4

Quantification et mesure de l'étendue de la maladie

4.1 Introduction

Après présentation et explication du modèle suivi, nous allons voir les outils et environnements qui ont contribué à la codification de ce dernier.

4.2 Environnement matériel et logiciel

Processeur : Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz 2.71 GHz

Mémoire RAM : 8,00 Go (7,88 Go utilisable)

Type du système : Système d'exploitation 64 bits, processeur x64

Carte graphique : Nvidia Getforce 940MX

En ce qui concerne l'environnement logiciel, nous le présentons comme suit :

4.2.1 Python

Python est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est aussi le langage de programmation open source le plus utilisé par les informaticiens. [20]



FIGURE 4.1 – Python logo

Les principales utilisations de Python par les développeurs sont :

- la programmation d'applications.
- la création de services web.
- la génération de code.
- la métaprogrammation.

4.2.2 Tensorflow

TensorFlow est un outil open source d'apprentissage automatique (machine learning) développé par Google, il est l'un des outils les plus utilisés en intelligence artificielle (IA).



FIGURE 4.2 – Tensorflow logo

Tensorflow permet de développer et d'exécuter des applications de machine learning et du deep learning. De plus, il permet de résoudre des problèmes mathématiques extrêmement complexes avec aisance. [21]

4.2.3 NumPy

NumPy est une bibliothèque open source pour le langage de programmation Python. Cette bibliothèque est destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.[22]



FIGURE 4.3 – NumPy logo

4.2.4 Sklearn (Scikit-learn)

Sklearn est une bibliothèque d'apprentissage automatique Python libre. Elle fournit dans son framework une grande bibliothèque d'algorithme d'implémentation prête à l'emploi dans le cadre. Ces bibliothèques sont particulièrement accessibles aux data scientists.[23]



FIGURE 4.4 – Scikit-learn logo

En particulier, Sklearn comprend des fonctions d'estimation de forêts aléatoires, de régression logistique, d'algorithmes de classification et de machines à vecteurs de support. Elle est conçue pour fonctionner en harmonie avec d'autres bibliothèques Python libre, comme NumPy et SciPy.

4.2.5 SciPy

SciPy est un logiciel open source pour les mathématiques, les sciences et l'ingénierie. [24]

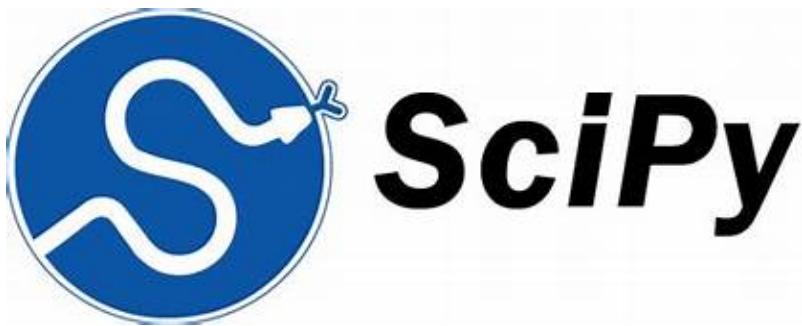


FIGURE 4.5 – SciPy logo

La bibliothèque SciPy dépend de NumPy, qui à son tour permet une manipulation pratique et rapide des tableaux à N dimensions. Conçue pour fonctionner avec les tableaux NumPy, cette dernière fournit de nombreuses routines numériques faciles à utiliser et efficaces.

4.2.6 Streamlit

Streamlit est une bibliothèque open source Python. Cette bibliothèque est un outil utilisé en machine learning qui vise à représenter graphiquement et à visualiser les données. Dans notre projet, streamlit a été utilisée pour construire l'interface graphique.

4.3 Environnement de développement

Pour concevoir le code, nous avons utilisé deux environnements de développement : Visual Studio Code et Google Colab.

4.3.1 Visual Studio Code

Visual Studio Code est un éditeur de code multiplateforme open source développé par Microsoft. Ce dernier a été développé sous Electron et utilise les fonctionnalités d'édition avancées du projet Monaco Editor.



FIGURE 4.6 – Streamlit logo

Conçu principalement pour le développement d'applications utilisant JavaScript, TypeScript et Node.js, le système d'extension bien construit permet à l'éditeur de s'adapter à d'autres types de langage.



FIGURE 4.7 – Visual Studio Code logo

Souvent, les gens confondent entre visual code studio est visual studio. Visual studio est un IDE (Integrated development environment ou environnement de développement en français) de microsoft.

4.3.2 Google Colab

Google Colab est un environnement de notebook Jupyter open source qui s'exécute entièrement dans le cloud. De plus, aucune configuration n'est requise et les notebook créés peuvent être modifiés par les membres de l'équipe en même temps, un peu comme la modification d'un document dans Google Docs.

Colab prend en charge de nombreuses bibliothèques de machine learning populaires qui peuvent être facilement chargées sur votre ordinateur portable comme scikit-learn.



FIGURE 4.8 – Google Colab logo

L'avantage de l'utilisation de Google Colab est qu'il offre un service Cloud gratuit avec un GPU gratuit. Cela veut dire que même si votre machine n'a pas de GPU (Graphical Processing Units ou unité de traitement graphique en français), le code s'exécutera sans problèmes bien qu'il contient des outils comme 'Tensorflow' car 'Tensorflow' nécessite la présence d'un GPU.

L'inconvénient de Colab est la nécessité d'une connexion internet pour l'utiliser et aussi l'obligation d'importer votre projet à chaque fois que vous reconnectez à ce dernier.

4.4 Données et discussion

Dans notre expérience, nous avons utilisé les ensembles de données ADNI GO¹ publics de l'Université de southern california (USC) avec 7020 sujets (3520 MCI² et 3500 AD³) pour évaluer le modèle à l'aide des images structurelles T1-w-MRI⁴. Chaque sujet possède quatre réseaux morphologiques corticaux dérivés tel que décrit. Chaque réseau morphologique cortical est codé dans une matrice d'adjacence symétrique de taille (35 x 35) générée à l'aide d'attributs corticaux spécifiques :

- Courbure principale maximale .
- Épaisseur corticale .
- Profondeur de rainure .
- Courbure moyenne.

1. ADNI GO est le nom de la base de données

2. MCI : mild cognitive impairment (en français, trouble cognitif léger)

3. AD : Alzheimer Disease (en français, maladie d'Alzheimer)

4. T1 weighted image (également appelée T1WI ou the "spin-lattice" relaxation time) est l'une des séquences d'impulsions de base en IRM et montre des différences dans les temps de relaxation T1 des tissus.)

Nous avons comparé notre cadre proposé avec l'intégration de connectome antagoniste (ACE)[25], une architecture d'auto-encodeur géométrique, qui capture les relations de région une à une dans une structure de graphe. Il utilise des couches convolutives graphiques conventionnelles pour apprendre les intégrations et le réseau de régulation contradictoire pour l'alignement de la distribution codée d'origine. Nous avons évalué ACE et HCAE sur des connectomes à vue unique et à vue multiple. Étant donné qu'ACE ne fonctionne pas naturellement sur des données multi-vues, nous avons fait la moyenne des connectomes multi-vues pour les évaluer sur des données multi-vues. HCAE est formé sur chaque sujet, indépendamment, en utilisant 30 époques.

Nous avons aussi fourni une visualisation du comportement des régions d'intérêts selon les morphologies corticales pour une analyse pour :

- Explorer la relation entre les mesures des morphologies corticales des régions d'intérêts.
- Mesures morphologiques moyennes par type de patient (malade Alzheimer ou non).
- Explorer la distribution des mesures des morphologies corticales des régions d'intérêts .
- Explorer la concentration des mesures morphologiques du cortical.

4.5 Présentation de l'interface et fonctionnalités

Notre application fut conçue dans le but de garantir une bonne communication entre l'utilisateur et l'interface, et une facilité d'utilisation. Elle permet de uploader les données médicale des patients sous forme d'un fichier sous l'extension 'csv' "Comma separated values", le calcul et l'analyse des données soumis et de visualiser les résultats de la classification. Dans ce qui suit, nous allons présenter notre interface et ses principales fonctionnalités :

4.5.1 Page de présentation

La page de présentation de notre projet est un élément clé pour que l'utilisateur se met dans le contexte et aussi c'est un guide pour l'utilisateur.

Celle-ci contient :

- Un menu qui contient différentes fonctionnalités du projet.
- Un hamburger bouton qui contient les différents paramètres en relation avec l'interface.
- Un main qui contient la description de notre projet.



FIGURE 4.9 – Page d'accueil de l'interface

4.5.2 Page de visualisation des fMRIs du patient

La page de visualisation des résultats de l'image par résonances magnétiques fonctionnelles (fMRIs) permet à l'utilisateur de visualiser et agrandir pour plus de détails.

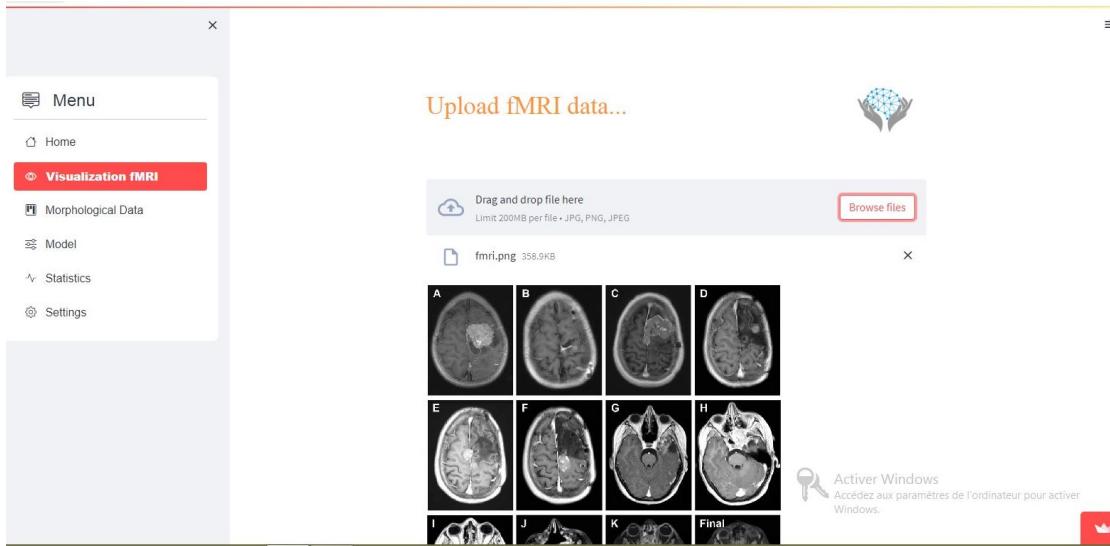


FIGURE 4.10 – Page de visualisation des fMRIs du patient

Page de visualisation du fichier csv des données

La page de visualisation des données du fichier csv permet à l'utilisateur de visualiser le contenu et les attributs de la base de données des patients.

The screenshot shows a user interface for data analysis. On the left, a sidebar menu lists "Home", "Visualization fMRI", "Morphological Data" (which is highlighted in red), "Model", "Metrics", and "Settings". The main area features a "Drag and drop file here" input field with a "Limit 200MB per file • CSV" note, a "Browse files" button, and a table titled "UCSFFSX6_05_02_22.csv 4.3MB". The table contains 26 rows of data with the following columns: OVERALLQC, ST102CV, ST102SA, ST102TA, ST102TS, ST103CV, and ST103SA.

	OVERALLQC	ST102CV	ST102SA	ST102TA	ST102TS	ST103CV	ST103SA
2187	MCI	4,372.0000	1,568.0000	2.5730	0.5040	1,933.0000	762.0000
2191	MCI	4,120.0000	1,654.0000	2.3760	0.4250	2,213.0000	677.0000
2201	MCI	3,707.0000	1,665.0000	2.1110	0.5330	2,094.0000	613.0000
2219	AD	4,260.0000	1,520.0000	2.5050	0.5950	1,992.0000	589.0000
2220	MCI	3,722.0000	1,477.0000	2.2970	0.5720	1,720.0000	634.0000
2234	MCI	3,166.0000	1,248.0000	2.3410	0.4410	1,998.0000	559.0000
2238	MCI	4,004.0000	1,758.0000	2.1970	0.5340	1,694.0000	586.0000
2239	MCI	3,041.0000	1,464.0000	2.0340	0.5570	2,055.0000	703.0000
2245	AD	3,104.0000	1,214.0000	2.3490	0.4170	1,609.0000	555.0000
2263	MCI	3,366.0000	1,445.0000	2.2610	0.6290	2,413.0000	686.0000

FIGURE 4.11 – Page de visualisation des données csv

Page de visualisation des analyses de données

La page de visualisation de l'analyse des données permet à l'utilisateur de visualiser la cohérence des données et leurs relations et cela selon quatre visualisations :

- Diagramme de dispersion : Explorer la relation entre les mesures.
- Diagramme à barres : Mesures moyennes du retour sur investissement par phase Alzheimer.
- Histogram : Explorer la distribution des mesures (Explore Distribution of Measurements).
- Tableau d'hexagones : Explorer la concentration des mesures

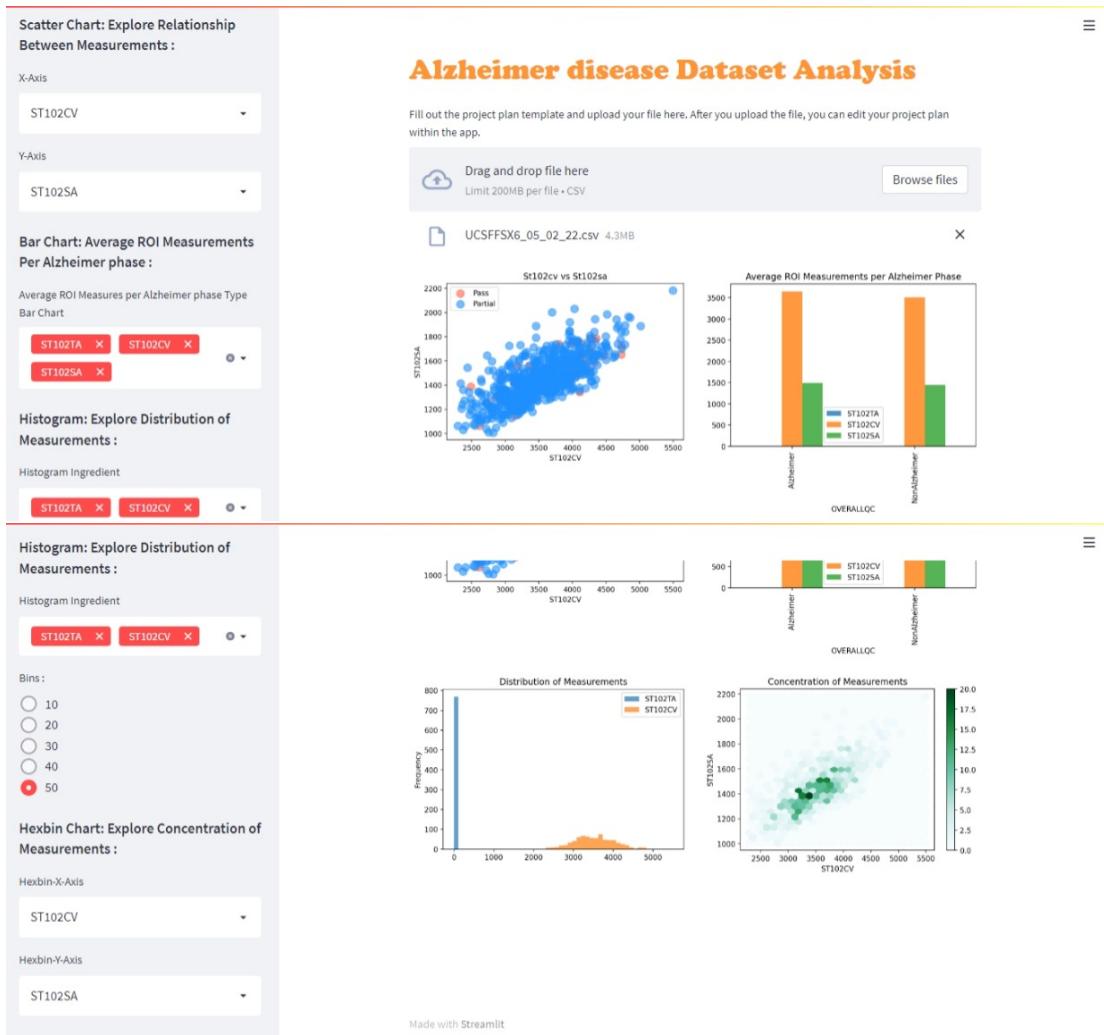


FIGURE 4.12 – Page de visualisation de l’analyse des données csv

Page de mesure des performances du modèle

La page de mesure des performances permet de visualiser les divers métriques pour l’évaluation de notre modèle et de visualiser un point de référence pour comparer entre nos résultat du modèle HCAE et ACE.

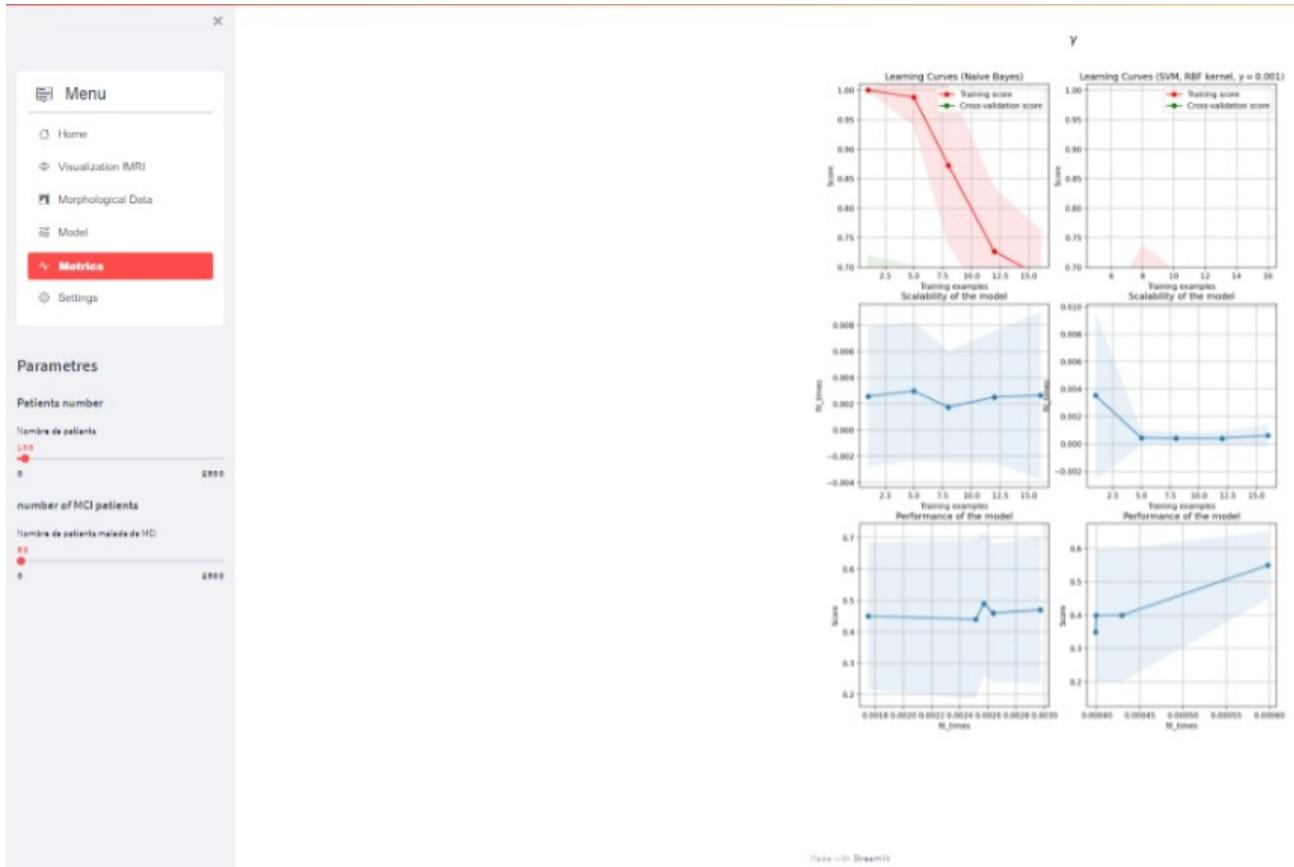


FIGURE 4.13 – page de mesure des performances d'un modèle auto-encodeur hyper-connectomes

4.6 Tests expérimentaux

Dans cette partie de ce chapitre, nous allons exposer quelques résultats des tests expérimentaux, les commenter et donner une analyse de ces derniers.

Cette base regroupe un ensemble de 7020 patients dans un contexte de mesures morphologiques corticales des régions d'intérêts du cerveau et leurs diagnostics. Comme nous travaillons sur des données quantitatives, nous avons pensé que comparer et visualiser les mesures morphologiques corticales donnerait de bonnes analyses sur les régions d'intérêts aux médecins.

	OVERALLQC	ST102CV	ST102SA	ST102TA	ST102TS	ST103CV	ST103SA	ST103TA	ST103TS	ST104CV	ST104SA	ST104TA	ST104TS	ST105CV
2187	MCI	4,372.0000	1,568.0000	2.5730	0.5040	1,933.0000	762.0000	2.3930	0.4380	5,135.0000	1,749.0000	2.6790	0.4390	3,053.0000
2191	MCI	4,120.0000	1,654.0000	2.3760	0.4250	2,213.0000	677.0000	2.8680	0.6880	3,633.0000	1,181.0000	2.7250	0.3600	2,127.0000
2201	MCI	3,707.0000	1,665.0000	2.1110	0.5330	2,094.0000	613.0000	3.0620	0.9390	3,442.0000	1,240.0000	2.4550	0.4910	2,307.0000
2219	AD	4,260.0000	1,520.0000	2.5050	0.5950	1,992.0000	589.0000	2.9000	0.8700	3,679.0000	1,295.0000	2.4960	0.4220	2,940.0000
2220	MCI	3,722.0000	1,477.0000	2.2970	0.5720	1,720.0000	634.0000	2.4430	0.6610	3,076.0000	1,163.0000	2.3090	0.4210	2,321.0000
2234	MCI	3,166.0000	1,248.0000	2.3410	0.4410	1,998.0000	559.0000	3.0150	0.9220	3,064.0000	1,115.0000	2.4390	0.4030	2,312.0000
2238	MCI	4,004.0000	1,758.0000	2.1970	0.5340	1,694.0000	586.0000	2.5920	0.6430	2,880.0000	1,173.0000	2.3920	0.5430	2,670.0000
2239	MCI	3,041.0000	1,464.0000	2.0340	0.5570	2,055.0000	703.0000	2.5500	0.6070	2,745.0000	1,110.0000	2.3010	0.3930	1,960.0000
2245	AD	3,104.0000	1,214.0000	2.3490	0.4170	1,609.0000	555.0000	2.6040	0.8010	3,316.0000	1,239.0000	2.3930	0.4600	2,850.0000
2263	MCI	3,366.0000	1,445.0000	2.2610	0.6290	2,413.0000	686.0000	2.9800	0.8940	4,426.0000	1,812.0000	2.2610	0.6080	3,279.0000
2301	MCI	3,420.0000	1,289.0000	2.4670	0.4590	2,014.0000	643.0000	2.7580	0.6350	3,766.0000	1,372.0000	2.5210	0.3620	3,295.0000
2304	MCI	3,898.0000	1,689.0000	2.2120	0.4830	2,087.0000	589.0000	2.9970	0.9070	3,190.0000	1,250.0000	2.3840	0.4510	2,873.0000
2315	MCI	3,551.0000	1,457.0000	2.2960	0.4380	2,015.0000	622.0000	2.7920	0.7460	3,038.0000	1,209.0000	2.2750	0.3800	1,751.0000
2332	AD	2,613.0000	1,066.0000	2.3030	0.5080	1,714.0000	541.0000	2.7720	0.7840	2,500.0000	1,096.0000	2.1520	0.4620	2,588.0000
2333	MCI	3,593.0000	1,693.0000	2.0500	0.5870	1,674.0000	582.0000	2.4620	0.6650	2,655.0000	1,098.0000	2.2730	0.4160	2,662.0000
2374	MCI	3,675.0000	1,429.0000	2.3080	0.6790	1,537.0000	577.0000	2.3550	0.5790	3,571.0000	1,371.0000	2.3510	0.4350	2,534.0000
2379	MCI	3,739.0000	1,636.0000	2.2080	0.4440	1,542.0000	631.0000	2.3220	0.5870	3,046.0000	1,299.0000	2.2740	0.3270	2,314.0000
2380	MCI	3,698.0000	1,729.0000	2.1230	0.5030	1,518.0000	608.0000	2.2780	0.6090	3,528.0000	1,546.0000	2.2420	0.5270	2,696.0000
2389	MCI	3,035.0000	1,343.0000	2.1830	0.4820	1,839.0000	616.0000	2.6510	0.6670	2,996.0000	1,081.0000	2.5930	0.3400	2,333.0000
2392	MCI	3,449.0000	1,481.0000	2.2660	0.4740	2,056.0000	568.0000	3.0900	0.8000	3,666.0000	1,272.0000	2.5990	0.5340	2,171.0000

FIGURE 4.14 – Aperçu de la base de données sur les morphologies des ROI de chaque patient

La méthode de l'auto-encodeur donne d'une manière générale de bons résultats, néanmoins nous avons remarqué que les données de très grandes tailles (grand nombre de patients) donnent de meilleurs résultats paradoxalement, ça prend plus de temps de calcul.

Enfin nous allons présenter ci-dessous les métriques d'évaluations que nous nous sommes basé sur pour évaluer notre modèle.

La courbe de la fonction d'efficacité du récepteur 'ROC curve'

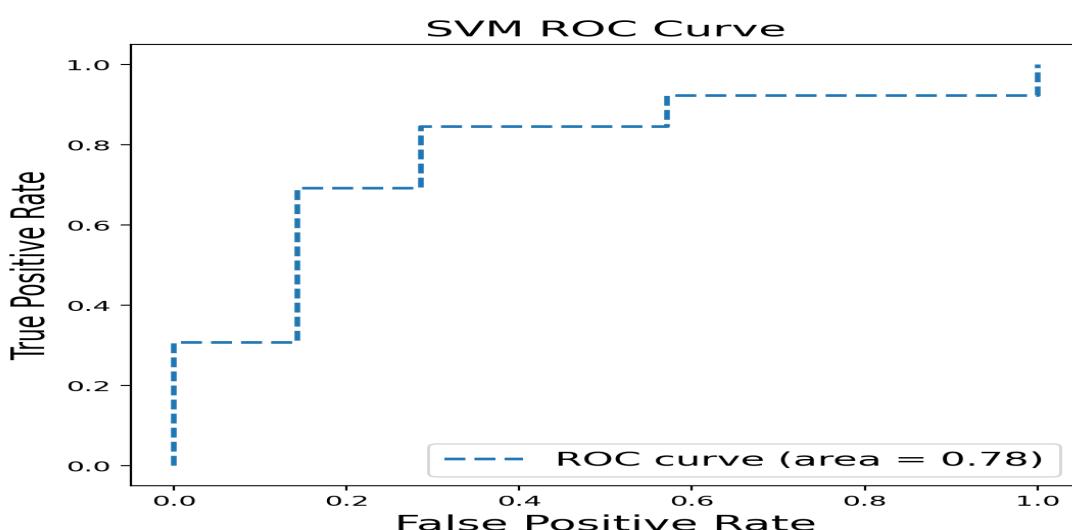


FIGURE 4.15 – Courbe ROC du SVM de notre modèle avec noyau RBF sur les données ADNI GO de l'Alzheimer

L'aire sous la courbe ROC (ou Area Under the Curve, AUC) est interprétée comme

la probabilité que, parmi deux sujets choisis au hasard, un malade et un non-malade, la valeur du marqueur soit plus élevée pour le malade que pour le non-malade. Par conséquent, une AUC de 0,7 (70%) indique que le marqueur est informatif.

Courbes de validation

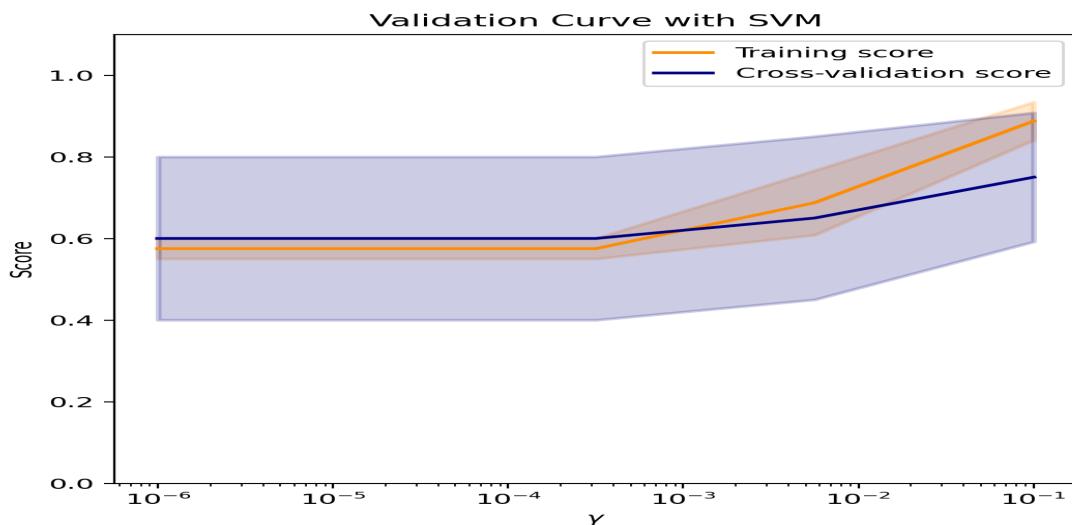


FIGURE 4.16 – Courbe de validation du SVM de notre modèle avec noyau RBF sur les données ADNI GO de l’Alzheimer

En regardant la courbe, nous pouvons déterminer si le modèle est sous-ajusté, surajusté ou juste pour une plage de valeurs d’hyperparamètres de max_depth. Dans le graphique, le score de précision de l’ensemble de train est marqué comme le "score d’entraînement" et le score de précision de l’ensemble de test est marqué comme le "score de validation croisée".

Idéalement, nous avons la courbe de validation et la courbe d’entraînement similaires pour un interval de 10^{-6} à 10^{-3} avec un score de 0,6 et une croissance par la suite jusqu’un score de 0,9 pour la courbe d’entraînement et un score de 0,75 pour la courbe de validation croisé.

perte d’entraînement du ACE

La courbe de perte d’entraînement est stable mais faible (0.69, 69 %) par rapport au modèle HCAE et cela n’est pas vraiment informatif et n’est pas pris en considération comme résultat.

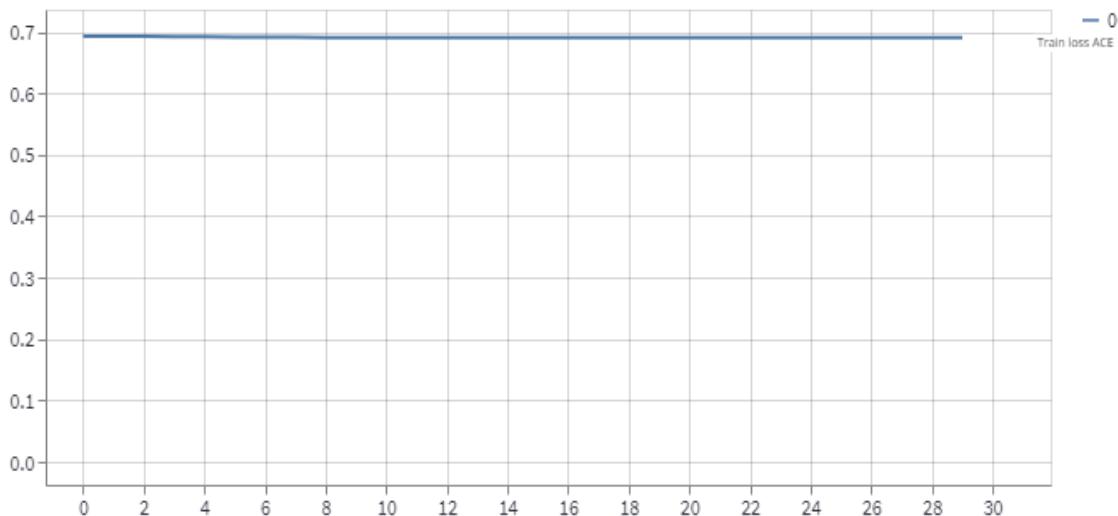


FIGURE 4.17 – Courbe de perte d’entraînement ACE

4.6.1 Quelques problèmes rencontrés

La méthode donne d’une manière générale de bons résultats, néanmoins nous avons remarqué que par rapport à la taille des données, ça prend plus de temps de calcul et aussi l’erreur d’entraînement stable est du discriminateur 0.69 (69 %) .

4.7 Conclusion

Parmi les outils et langages qui existent, nous avons présenté ceux qui nous ont été utiles pour réaliser notre projet de manière efficace et qui nous permettent d’avoir les résultats voulus d’une manière rapide. Nous avons aussi vue les résultats obtenus après l’exécution de notre code.

Conclusion Générale

Mener de telles recherches est une étape vers l'amélioration du domaine médicale basé sur l'imagerie. Vue que ces derniers temps, la maladie d'Alzheimer apparaît beaucoup, sa détection précoce sera sûrement bénéfique pour l'humanité.

La méthode avec les hyperconnectomes semble la plus efficace quand il s'agit de la détection des maladies par imagerie et cela revient à sa possibilité de comparer plusieurs régions d'intérêts et de classer les relations entre eux en utilisant les KNNs, aussi le fait qu'ils sont facilement adapté aux données graphiques multi-vues.

Nous avons vue les définitions des méthodes utilisées en général dans le chapitre 02 et nous avons discuter en détails sur notre modèle dans le chapitre 03.

Dans le chapitre 04, nous avons vue les résultats obtenus et après comparaisons des résultat du modèle HCAE avec ACE, nous avons conclue que ces derniers sont bon et que le modèle HCAE est efficace.

Même si nous avons eu des résultats satisfaisant, notre modèle a quelques limites qui peuvent être améliorées dans des travaux futures.

Premièrement, les prolongement appris sont appris de manière non supervisé et la solution que nous proposons et d'ajouter un autre discriminateur pour apprendre les imbrications discriminatoires supervisées par les états cérébraux. Comme il faut adapter le modèle HCAE à la formation axée sur la population.

Deuxièmement, dans notre modèle nous avons utilisé la multiplication multicanal pour opérer sur un hyperconnectome multi-vues. Pour améliorer cela, l'ajout des d'un bloc de fusion hyperconnectome pour intégrer les hyperconnectomes de manière pondérée dans toutes les vues est possible.

Finalement, tandis que notre modèle a l'avantage de comparer plusieurs ensembles (relation plusieurs-à-plusieurs) qui s'avère utile pour la classification de l'état du cerveau, ce dernier ne peut pas gérer les connectomes cérébraux multi-vues spatio-temporel qui dépendent du temps. Comme solution à cette limite, nous suggérons d'encoder automatiquement la trajectoire évolutive du connectome cérébral multi-vues en modélisant les relations temporelles entre les connectomes acquis à différents moments à l'aide des réseaux de neurones récurrents dans l'apprentissage profond géométrique.

Conclusion Générale

Pour finir, nous souhaitons que notre travail soit bénéfique pour l'ensemble des lecteurs.

Bibliographie

- [1] Alin Banka, Inis Buzi, and Islem Rekik[2020]. Multi-View Brain HyperConnectome AutoEncoder For Brain State Classification. <https://arxiv.org/abs/2009.11553>
- [2] DÉFINITION ET CHIFFRES DE LA MALADIE D'ALZHEIMER [en ligne]. Disponible sur : <https://alzheimer-recherche.org/la-maladie-alzheimer/question-maladie-dalzheimer/definition-et-chiffres/>
- [3] Dr M. WILLERY[2012]. DEMENCES ET SEMEIOLOGIE DES DEMENCES <http://entreide.free.fr/cours/2a/ue27s4/DEMENCES.pdf>
- [4] La maladie d'Alzheimer,ministère des solidarités et de la santé [en ligne]. Disponible sur : <https://solidarites-sante.gouv.fr/soins-et-maladies/maladies/maladies-neurodegeneratives/article/la-maladie-d-alzheimer#>
- [5] Center for Functional Imaging, Lawrence Berkeley National Laboratory.[2015]. initiative de neuroimagerie de la maladie d'Alzheimer (ADNI) <https://neurosciencenews.com/dementia-diabetes-mci-neurology-1790/>
- [6] Gauthier, Thal Rossor. In S. Gauthier[1996]The future diagnosis and management of Alzheimer's disease <http://ndl.ethernet.edu.et/bitstream/123456789/22069/1/111.pdf#page=395>
- [7] France Alzheimer Maladies appartées [2019]Meurt-on de la maladie d'Alzheimer? <https://www.francealzheimer.org/comprendre-la-maladie/la-maladie-dalzheimer/premiers-reperes-maladie/>
- [8] Mme DJEDIAT.R Mme BOUKROUFA.M [2019]Anatomie et histologie <https://slideplayer.fr/slide/10151991/>
- [9] La maladie d'Alzheimer [en ligne]. Disponible sur : <https://solidarites-sante.gouv.fr/soins-et-maladies/maladies/maladies-neurodegeneratives/article/la-maladie-d-alzheimer>
- [10] Apprentissage supervisé : Définition et explications... [en ligne]. Disponible sur : <https://www.jeveuxetredatascientist.fr/apprentissage-supervise/>
- [11] Apprentissage non supervisé — Wikipédia [en ligne]. Disponible sur : https://fr.wikipedia.org/wiki/Apprentissage_non_supervisé
- [12] A Gentle Introduction to Graph Neural Networks [en ligne]. Disponible sur : <https://distill.pub/2021/gnn-intro/>
- [13] Hammond et al [2011]. Wavelets on Graphs via Spectral Graph Theory <https://arxiv.org/abs/0912.3848>
- [14] footnotesizeYadati, N., Nimishakavi, M., Yadav, P., Nitin, V., Louis, A. and Talukdar, [2018]. HyperGCN : A New Method of Training Graph Convolutional Networks on Hypergraphs <https://arxiv.org/abs/1809.02589>

Bibliographie

- [15] Simon Rogers, Mark Girolami[2016]. a first course in machine learning Chapman amp; Hall/CRC
<https://doi.org/10.1201/9781315382159>
- [16] Noyau de la fonction de base radiale — Wikipédia [en ligne]. Disponible sur : https://en.wikipedia.org/wiki/Radial_basis_function_kernel
- [17] Feng, Y., You, H., Zhang, Z., Ji, R., Gao, Y[2018]. Hypergraph Neural Networks <https://arxiv.org/abs/1809.09401>
- [18] Pan, S., Hu, R., Long, G., Jiang, J., Yao, L., Zhang, C.[2018]. Adversarially Regularized Graph Autoencoder for Graph Embedding. <https://arxiv.org/abs/1802.04407>
- [19] Agarwal T[2020]. Hyperparameter optimization in machine learning. , India
- [20] Python-Wikipédia [en ligne]. Disponible sur : [https://fr.wikipedia.org/wiki/Python_\(langage\)](https://fr.wikipedia.org/wiki/Python_(langage))
- [21] TensorFlow : tout savoir sur la bibliothèque Machine Learning... [en ligne]. Disponible sur : <https://www.lebigdata.fr/tensorflow-definition-tout-savoir>
- [22] NumPy — Wikipédia [en ligne]. Disponible sur : <https://fr.wikipedia.org/wiki/NumPy>
- [23] Scikit-learn — Wikipédia [en ligne]. Disponible sur : <https://fr.wikipedia.org/wiki/Scikit-learn>
- [24] SciPy — Wikipédia [en ligne]. Disponible sur : <https://fr.wikipedia.org/wiki/SciPy>
- [25] Banka, A., Rekik, I[2019]. :Adversarial connectome embedding for mild cognitive impairment identification using cortical morphological networks. *International Workshop on Connectomics in Neuroimaging*.