Interprétabilité des réseaux de neurones profonds et segmentation faiblement supervisée des lésions cérébrales sur IRM

Valentine Wargnier-Dauchelle

N° d'ordre NNT : 2023ISAL0110

THESE de DOCTORAT DE L'INSA LYON, membre de l'Université de Lyon

Ecole Doctorale N° 205
Interdisciplinaire Sciences Santé

Spécialité/discipline de doctorat :
Ingénierie biomédicale

Soutenue publiquement le 08/12/2023, par :
Valentine Wargnier Dauchelle

Interprétabilité des réseaux de neurones profonds et segmentation faiblement supervisée des lésions cérébrales sur IRM

Devant le jury composé de :

Dolz, Jose	Professeur	ETS Montréal	Rapporteur
Hudelot, Céline	Professeure	CentraleSupélec	Rapporteure
Mateus, Diana	Professeure	Ecole Centrale Nantes	Rapporteure
Garcia, Christophe	Professeur	INSA Lyon	Examinateur
Petitjean, Caroline	Professeure	Université de Rouen	Examinatrice
Cotton, François	Professeur praticien hospitalier	Université Lyon 1	Directeur de thèse
Sdika, Michaël	Ingénieur de recherche HDR	CNRS	Co-directeur de thèse
Grenier, Thomas	Maître de conférences HDR	INSA Lyon	Encadrant

INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Département de la Formation par la Recherche et des Études Doctorales (**FEDORA**)

Bâtiment INSA direction, 1er étage
37, av. J. Capelle
69621 Villeurbanne Cédex
fedora@insa-lyon.fr

Référence : TH1041_WARGNIER Valentine

L'INSA Lyon a mis en place une procédure de contrôle systématique via un outil de détection de similitudes (logiciel Compilatio). Après le dépôt du manuscrit de thèse, celui-ci est analysé par l'outil. Pour tout taux de similarité supérieur à 10%, le manuscrit est vérifié par l'équipe de FEDORA. Il s'agit notamment d'exclure les auto-citations, à condition qu'elles soient correctement référencées avec citation expresse dans le manuscrit.

Par ce document, il est attesté que ce manuscrit, dans la forme communiquée par la personne doctorante à l'INSA Lyon, satisfait aux exigences de l'Etablissement concernant le taux maximal de similitude admissible.

Département FEDORA – INSA Lyon - Ecoles Doctorales

SIGLE	ECOLE DOCTORALE	NOM ET COORDONNEES DU RESPONSABLE
CHIMIE	**CHIMIE DE LYON** **https://www.edchimie-lyon.fr** Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage secretariat@edchimie-lyon.fr	**M. Stéphane DANIELE** C2P2-CPE LYON-UMR 5265 Bâtiment F308, BP 2077 43 Boulevard du 11 novembre 1918 69616 Villeurbanne **directeur@edchimie-lyon.fr**
E.E.A.	**ÉLECTRONIQUE, ÉLECTROTECHNIQUE, AUTOMATIQUE** **https://edeea.universite-lyon.fr** Sec. : Stéphanie CAUVIN Bâtiment Direction INSA Lyon Tél : 04.72.43.71.70 secretariat.edeea@insa-lyon.fr	**M. Philippe DELACHARTRE** INSA LYON Laboratoire CREATIS Bâtiment Blaise Pascal, 7 avenue Jean Capelle 69621 Villeurbanne CEDEX Tél : 04.72.43.88.63 **philippe.delachartre@insa-lyon.fr**
E2M2	**ÉVOLUTION, ÉCOSYSTÈME, MICROBIOLOGIE, MODÉLISATION** **http://e2m2.universite-lyon.fr** Sec. : Bénédicte LANZA Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 secretariat.e2m2@univ-lyon1.fr	**Mme Sandrine CHARLES** Université Claude Bernard Lyon 1 UFR Biosciences Bâtiment Mendel 43, boulevard du 11 Novembre 1918 69622 Villeurbanne CEDEX **sandrine.charles@univ-lyon1.fr**
EDISS	**INTERDISCIPLINAIRE SCIENCES-SANTÉ** **http://ediss.universite-lyon.fr** Sec. : Bénédicte LANZA Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 secretariat.ediss@univ-lyon1.fr	**Mme Sylvie RICARD-BLUM** Institut de Chimie et Biochimie Moléculaires et Supramoléculaires (ICBMS) - UMR 5246 CNRS - Université Lyon 1 Bâtiment Raulin - 2ème étage Nord 43 Boulevard du 11 novembre 1918 69622 Villeurbanne Cedex Tél : +33(0)4 72 44 82 32 **sylvie.ricard-blum@univ-lyon1.fr**
INFOMATHS	**INFORMATIQUE ET MATHÉMATIQUES** **http://edinfomaths.universite-lyon.fr** Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage Tél : 04.72.43.80.46 infomaths@univ-lyon1.fr	**M. Hamamache KHEDDOUCI** Université Claude Bernard Lyon 1 Bât. Nautibus 43, Boulevard du 11 novembre 1918 69 622 Villeurbanne Cedex France Tél : 04.72.44.83.69 **hamamache.kheddouci@univ-lyon1.fr**
Matériaux	**MATÉRIAUX DE LYON** **http://ed34.universite-lyon.fr** Sec. : Yann DE ORDENANA Tél : 04.72.18.62.44 yann.de-ordenana@ec-lyon.fr	**M. Stéphane BENAYOUN** Ecole Centrale de Lyon Laboratoire LTDS 36 avenue Guy de Collongue 69134 Ecully CEDEX Tél : 04.72.18.64.37 **stephane.benayoun@ec-lyon.fr**
MEGA	**MÉCANIQUE, ÉNERGÉTIQUE, GÉNIE CIVIL, ACOUSTIQUE** **http://edmega.universite-lyon.fr** Sec. : Stéphanie CAUVIN Tél : 04.72.43.71.70 Bâtiment Direction INSA Lyon mega@insa-lyon.fr	**M. Jocelyn BONJOUR** INSA Lyon Laboratoire CETHIL Bâtiment Sadi-Carnot 9, rue de la Physique 69621 Villeurbanne CEDEX **jocelyn.bonjour@insa-lyon.fr**
ScSo	**ScSo*** **https://edsciencessociales.universite-lyon.fr** Sec. : Mélina FAVETON INSA : J.Y. TOUSSAINT Tél : 04.78.69.77.79 melina.faveton@univ-lyon2.fr	**M. Bruno MILLY** Université Lumière Lyon 2 86 Rue Pasteur 69365 Lyon CEDEX 07 **bruno.milly@univ-lyon2.fr**

*ScSo : Histoire, Géographie, Aménagement, Urbanisme, Archéologie, Science politique, Sociologie, Anthropologie

Résumé

L'imagerie médicale est un outil fondamental pour diagnostiquer les maladies, suivre leur évolution mais aussi comprendre leur fonctionnement afin de mieux les soigner. L'imagerie par résonance magnétique est une méthode de choix pour visualiser le cortex cérébral et ses pathologies comme la sclérose en plaques, une maladie auto-immune inflammatoire et démyélinisante qui est la première cause de handicap non traumatique chez les jeunes adultes, ou encore les gliomes, qui sont les tumeurs primitives cérébrales les plus courantes.

Pour analyser ces images de manière automatique, les méthodes basées sur l'apprentissage profond présentent de très bonnes performances pour différents types de tâches comme la classification ou la segmentation. Ces méthodes automatiques apportent aux cliniciens une pré-analyse très utile dans leurs études ou diagnostics. Cependant, elles nécessitent beaucoup de données pour leur entraînement. Dans le cas des méthodes de segmentation supervisées, les annotations manuelles nécessaires pour chaque image sont très coûteuses. Le développement de méthodes faiblement ou non-supervisées performantes, ne nécessitant pas ou peu d'annotations manuelles, est donc nécessaire. En outre, dans un domaine critique comme celui de la médecine, il est important que les décisions des réseaux soient explicables et s'appuient sur les signes radiologiques de la pathologie présents dans l'image et utilisés par les cliniciens. Or, les réseaux de neurones profonds sont, de par leur grand nombre de paramètres et les interconnexions non linéaires dont ils sont composés, difficiles à expliquer. Proposer des réseaux explicables et interprétables est donc une problématique forte pour l'analyse d'images médicales par apprentissage profond.

Dans cette thèse, nous avons abordé ces deux thématiques. En nous focalisant sur une tâche de classification entre des images de sujets sains et des images de patients (notamment atteints de sclérose en plaques ou de gliomes), nous avons montré que la décision des classifieurs de l'état de l'art n'est pas forcément pertinente et en accord avec les aprioris médicaux. Cela peut avoir de lourdes conséquences : pour du diagnostic, l'utilisation de tels classifieurs biaisés n'est pas raisonnable et lorsqu'ils sont utilisés au sein d'autres modèles, comme les modèles génératifs, cela peut faire chuter les performances. Nous avons donc proposé des classifieurs plus interprétables avec une décision davantage basée sur les signes radiologiques de la pathologie considérée. Trois solutions ont été proposées. Tout d'abord, nous avons normalisé l'entrée des réseaux de neurones afin d'éliminer les biais présents dans l'image et qui peuvent être utilisés par les réseaux classiques pour prendre leur décision. Ensuite, nous avons contraint les classifieurs au cours de leur entraînement en utilisant les cartes d'attributions, des méthodes de l'état de l'art permettant d'identifier les zones de l'image d'entrée utilisées par le réseau pour prendre sa décision. Enfin, nous avons utilisé des réseaux intrinsèquement explicables : les réseaux monotones. Nous avons notamment proposé une méthode pour transformer n'importe quelle architecture en réseau monotone alors que les réseaux monotones de l'état de l'art étaient limités à des architectures de très faible profondeur. Avec ces réseaux de classification interprétables ne disposant que du label de l'image à l'entraînement, nous pouvons réaliser une segmentation faiblement supervisée des lésions cérébrales.

Mots-clés : Apprentissage profond, Apprentissage faiblement supervisé, Interprétabilité des réseaux de neurones, Segmentation, Lésions cérébrales, IRM

Abstract

Medical imaging is essential to diagnose diseases, follow their progress and understand how they work, in order to treat them more effectively. Magnetic resonance imaging is the method of choice to visualize the cerebral cortex and its pathologies, such as multiple sclerosis, an inflammatory demyelinating autoimmune disease which is the leading cause of non-traumatic disability in young adults, or gliomas, the most common primary brain tumors.

To analyze these images automatically, deep learning methods perform very well for different types of tasks, such as classification or segmentation. These automatic methods provide clinicians with a very useful pre-analysis for their studies or diagnoses. In the case of supervised segmentation methods, the manual annotations required for each image are very costly. This calls for developments of weakly or unsupervised segmentation methods requiring few or no manual annotation. In addition, in a field as critical as medicine, it is important for networks decision to be explainable and based on the same radiological signs of the pathology as the signs used by clinicians. However, deep neural networks are difficult to explain as they are composed of non-linear interconnections with a huge number of parameters. Providing explainable and interpretable networks is therefore a second key issue for medical image analysis using deep learning.

In this thesis, we addressed both these issues. Focusing on a classification task between images of healthy subjects and images of patients (especially with multiple sclerosis or gliomas), we have shown that the decision of state-of-the-art classifiers is not necessarily relevant and in accordance with medical knowledge. The consequences can be very serious : for diagnostics, the use of such biased classifiers is not desirable and when used within other models, such as generative models, it can lead to a drop in performances. Therefore, we proposed more interpretable classifiers, with a decision more based on the radiological signs of the pathology. Three solutions were proposed. First, we normalized the input of the neural networks in order to eliminate the biases present in the image and which can be used by classical networks to make their decision. Next, we constrained the classifiers during their training using attribution maps, a state-of-the-art method enabling the localization of the input image areas contributing the most in the network decision. Finally, we used intrinsically explainable networks : monotonic networks. We proposed a method to transform any deep network architecture into a monotone network, whereas state-of-the-art monotone networks were limited to very shallow architectures. With these interpretable classification networks only trained with image label, we can perform weakly supervised segmentation of brain lesions.

Keywords : Deep learning, Weakly supervised learning, Neural networks interpretability, Segmentation, Brain lesions, MRI

REMERCIEMENTS

Tout d'abord, je voudrais remercier Caroline Petitjean et Christophe Garcia pour avoir accepté d'évaluer mon travail au cours de cette thèse, à travers les comités de suivi et en tant qu'examiteurs pour conclure, je l'espère, en beauté. Je remercie également mes rapporteurs, Céline Hudelot, Diana Mateus et Jose Dolz, pour le temps consacré à la lecture et l'évaluation de ce manuscrit. Je me sens très chanceuse d'avoir un jury de si bonne qualité.

Ensuite, je souhaite remercier les membres du laboratoire CREATIS pour l'acceuil, la bonne humeur, les échanges et les conseils (merci Carole et Odyssée d'avoir écouté aux portes pour mieux me conseiller). Parmi eux, je remercie tout particulèrement mes encadrants, François Cotton, Michaël Sdika et Thomas Grenier. Merci Thomas d'avoir apporté un regard nouveau lorsqu'il était difficile d'y voir clair. Et surtout, merci Michaël, pour... tout ? Ton soutien et ton écoute dans les moments difficiles, ton esprit brillant (je dois le reconnaître), ton humour pas toujours de haut niveau (là c'est toi qui dois le reconnaître), ton style vestimentaire immuable et réconfortant, l'augmentation de mon taux de glycémie et plein d'autres choses. Merci aux doctorants d'un jour (Anne-Lise, Antoine F, Antoine N, Antonio, Arthur, Benoît V, Emile, Gaël, Julia, Juliette, Louise, Ludmilla, Matthis, Maxime, Mom, Morgane, Nathan, Paul, Pierre, Pierre-Elliott, Pilar, Raoul, Robin, Romain, Sophie, Thomas, Valentin, etc) et notamment ceux qui sont venus régulièrement jusqu'à mon bureau. Merci à mes co-bureaux, Frank et Benoît, de m'avoir acceptée dans *mon* bureau et de l'avoir égayé par leur présence (parfois).

Enfin, je remercie ma famille pour leur soutien et leur amour au cours de ces (longues ?) études. Sans vous, rien ne serait pareil...

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLEAUX

LISTE DES ABRÉVIATIONS

AE Auto-encodeur

AUPRC Area under precision recall curve

AUROC Area under receiver operating characteristic

BraTS Brain tumor segmentation challenge

CSF Cerebrospinal fluid / Liquide cérébrospinal

EG Expected gradient

FC Fully connected

GAN Generative adversarial network / Réseau adversaire génératif

Gd Gadolinium

GM Grey matter / Substance grise

GPU Graphics Processing Unit

G Gradient (attributions)

HCP Human connectum project

IBC Individual brain charting

IEG Mélange d'Integrated et Expected gradient

IG Integrated gradient

IRM Imagerie par résonance magnétique

IXI Information extraction from images

LIME Local interpretable model-agnostic explanations

LOP STAPLE Logarithmic opinion pool based simultaneous truth and performance level estimation

LRP Layer-wise Relevance Propagation

MILP Mixed-integer linear programming

MLP Multi-layers perceptron / Perceptron multi-couches

MNI Montreal neurological institute-hospital

MSSEG Multiple sclerosis segmentation challenge

NAM Neural additive models / Modèle neuronaux additifs

OFSEP Observatoire français de la sclérose en plaques

OMS Organisation mondiale de la santé

RMN Résonance magnétique nucléaire

ROC Receiver operating characteristic

SEP Sclérose en plaques

TE Temps d'écho pour la séquence spin écho en IRM

TI Temps d'inversion pour la préparation à l'aimantation en IRM

TNR True negative rate / Taux de vrais négatifs

TPR True positive rate / Taux de vrais positifs

TR Temps de répétition pour la séquence spin écho en IRM

VAE Variational auto-encoder / Auto-encodeur variationel

VIP Virtual imaging platform

WM White matter / Substance blanche

INTRODUCTION

L'imagerie médicale est essentielle dans la prise en charge des patients. Elle permet de visualiser les différents organes du corps humain et d'identifier les possibles pathologies ou dysfonctionnements. Parmi les différentes modalités, l'imagerie par résonance magnétique permet de visualiser les tissus mous avec un très bon contraste et de manière non-invasive. Elle est donc souvent la méthode d'imagerie la plus recommandée pour visualiser le cerveau et ses pathologies comme les lésions de sclérose en plaques ou les gliomes.

Ces dernières années, l'apprentissage profond a révolutionné la façon d'aborder l'analyse d'images médicales avec des modèles toujours plus performants que ce soit pour des problèmes de classification, de régression, de segmentation, etc. Ces méthodes automatiques peuvent aider le radiologue dans sa routine en apportant une deuxième analyse ou en réduisant le temps nécessaire pour étudier une image.

Néanmoins, des limites à leur utilisation subsistent. Tout d'abord, ces méthodes demandent de grandes bases de données pour leur apprentissage. Parmi elles, les méthodes supervisées nécessitent, en plus, des annotations manuelles pour chacune de ces données. Dans le cadre de la segmentation, une annotation manuelle peut demander plusieurs heures au médecin pour un seul volume 3D. Développer des méthodes non-supervisées ou faiblement supervisées, qui ne nécessitent donc pas de segmentations manuelles, les plus performantes possibles reste donc un défi. Un autre questionnement fort en apprentissage profond est l'explicabilité et l'interprétabilité des méthodes développées. En effet, les réseaux neuronaux sont généralement constitués de nombreuses couches reliées par des relations non linéaires entrelacées. Même en inspectant toutes les couches et en décrivant leurs relations, il est impossible de comprendre entièrement comment le réseau de neurones est parvenu à sa décision. Proposer des modèles explicables, dont la décision est en accord avec les experts médicaux, est donc une question importante. Dans les domaines sensibles comme le médical, cela est d'autant plus vrai car les praticiens ont besoin d'avoir confiance dans la décision des méthodes qui leur sont proposées.

Dans ce manuscrit, nous nous sommes intéressés à ces deux problématiques en ayant pour objectif, d'une part, de développer des réseaux de neurones interprétables, dont la décision est principalement basée sur les signes radiologiques présents dans les images utilisées en entrée du réseau, et d'autre part, de les utiliser pour faire de la segmentation faiblement supervisée de deux types de lésions cérébrales : les lésions de sclérose en plaques et les gliomes.

Ce manuscrit est organisé en six chapitres. Le premier chapitre présente le contexte à savoir l'imagerie par résonance magnétique, les pathologies cérébrales considérées et les bases de données utilisées, et établit la problématique abordée dans cette thèse. Le chapitre 2 propose un état de l'art du fonctionnement des approches d'apprentissage profond no-

tamment pour la segmentation faiblement et non-supervisée et l'explicabilité des réseaux de neurones. Les chapitres suivants correspondent à nos différentes contributions. Dans le chapitre 3, en essayant de segmenter les lésions de sclérose en plaques avec un réseau adversaire génératif, nous avons identifié le problème des classifieurs d'images de sujets sains vs pathologiques qui n'utilisent pas que la présence d'une pathologie pour prendre leur décision. Dans le chapitre 4, nous avons proposé une normalisation des images pour améliorer l'interprétabilité d'un classifieur d'images de sujets sains et de patients mais aussi identifier certains des biais utilisés par le classifieur pour prendre sa décision. Dans le chapitre 5, nous avons proposé de contraindre l'entraînement du classifieur pour améliorer son interprétabilité. Enfin, le chapitre 6 propose d'utiliser des réseaux intrinsèquement explicables mais connus pour être difficiles voire impossibles à entraîner, les réseaux monotones, pour améliorer l'interprétabilité de la décision.

CHAPITRE

1

CONTEXTE MÉDICAL

Introduction

Dans ce chapitre, nous présentons le contexte de cette thèse : l'imagerie de résonance magnétique (IRM) pour les pathologies cérébrales. Nous décrirons rapidement le principe de fonctionnement de l'IRM ainsi que les phénomènes physio-pathologiques des pathologies cérébrales que nous avons considérées au cours de nos travaux. Ensuite, nous détaillerons les bases de données d'IRM cérébrales utilisées. Finalement, nous dégagerons les problématiques propres à ce contexte médical que nous avons cherchées à résoudre au cours de cette thèse.

I L'imagerie par résonance magnétique pour le cerveau

I.1 Imagerie par résonance magnétique

L'Imagerie par Résonnance Magnétique (IRM) est une technique d'imagerie médicale très utilisée pour étudier de manière non-invasive le cerveau. Elle permet d'obtenir des volumes 3D des tissus mous avec une bonne résolution et un bon contraste. En outre, elle repose sur une technologie non-irradiante. Pour cela, un champ magnétique de forte intensité (en contexte clinique généralement de 1.5T ou 3T) est appliqué grâce à un aimant supraconducteur. La seule contre indication est la présence de métal dans le corps imagé. Son fonctionnement repose sur la Résonance Magnétique Nucléaire (RMN).

I.1.1 Résonance magnétique nucléaire

La RMN exploite le fait que certains noyaux atomiques possèdent un spin et donc un moment magnétique non nul. C'est, par exemple, le cas des atomes d'hydrogène très présents dans la matière vivante et principalement utilisés pour l'IRM. Ce moment peut être défini comme $\vec{\mu} = \gamma \vec{S}$ où $\vec{S}$ est le moment cinétique intrinsèque du noyau et γ est son rapport gyromagnétique.

Si on regarde au niveau macroscopique, nous sommes en présence d'un grand nombre d'atomes et l'on peut considérer un vecteur d'aimantation comme la somme des moments magnétiques de chaque noyau $\vec{\mu}_i$ par unité de volume V : $\vec{M} = \frac{1}{V} \sum_i \vec{\mu}_i$. En l'absence de champ magnétique, à l'équilibre, les moments magnétiques de chaque noyau peuvent être orientés dans toutes les directions de l'espace. L'aimantation résultante est nulle. Lorsqu'on soumet ces noyaux à un champ magnétique extérieur $\vec{B}_0$ (selon l'axe $\vec{z}$ d'un repère orthonormé $(0, \vec{x}, \vec{y}, \vec{z})$), les moments magnétiques se polarisent parallèlement (état d'énergie faible) ou anti-parallèlement (état excité) à $\vec{B}_0$, en effectuant un mouvement de rotation autour de $\vec{B}_0$, la précession de Larmor. La vitesse angulaire ω_0 correspondante est proportionnelle à la force du champ magnétique imposé : $\omega_0 = \gamma B_0$. L'aimantation macroscopique résultante est faible mais bien présente et colinéaire au champ avec la même direction (les spins parallèles au champ sont majoritaires car dans l'état d'énergie minimale du fait de la statistique de Boltzman) : $\vec{M} = M_{z0}\vec{B}_0$ (Figure 1.1). La composante transverse M_{xy} au champ est nulle du fait des déphasages des moments magnétiques des noyaux entre eux.

Cette aimantation, dans cet état, n'est pas mesurable car elle est infime par rapport au champ. Il est possible d'augmenter la population de spins en état excité (anti-parallèle au champ) en envoyant une onde radiofréquence à une fréquence bien particulière : la fréquence de Larmor. Cela correspond à appliquer un champ $\vec{B}_1$ oscillant sur l'axe transverse $\vec{x}$. Du point de vue macroscopique, sous l'effet de cette onde, l'aimantation va s'écarter de sa position d'équilibre et tourner à la fois autour de $\vec{B}_0$ (à la fréquence de Larmor) et à la fois autour de $\vec{B}_1$. Cette fois, la composante transversale est "en cohérence" car l'énergie apportée le permet. L'aimantation va former une trajectoire spirale comme montré en Figure 1.2a. Si

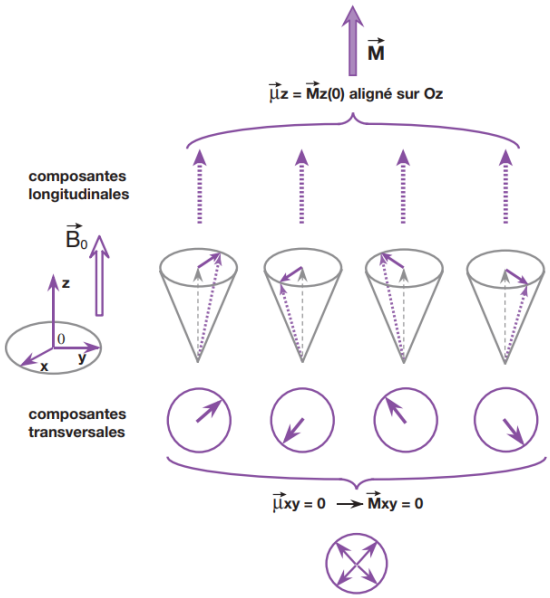


FIGURE 1.1 – *Composantes de l'aimantation macroscopique en présence d'un champ $\vec{B}_0$ selon $\vec{z}$. La composante longitudinale $\vec{M}_z$ est non nulle due à l'orientation parallèle ou anti-parallèle des spins le long de $\vec{B}_0$. La composante transversale $\vec{M}_{xy}$ est nulle à cause du déphasage des spins. Image tirée de [Kastler et Vetter, 2018].*

on se place dans un référentiel tournant suivant la rotation de $\vec{B}_1$, cette excitation conduit à une bascule de l'aimantation vers le plan perpendiculaire à $\vec{B}_0$ (Figure 1.2b). Plus l'excitation sera longue, plus l'angle entre l'aimantation et le champ $\vec{B}_0$ sera grand. Les angles de bascule les plus largement utilisés sont la bascule de 90°, pour laquelle l'aimantation longitudinale sera nulle et l'aimantation transversale maximale, et la bascule de 180°, pour laquelle l'aimantation transversale sera nulle et l'aimantation longitudinale sera opposée à celle sous $\vec{B}_0$.

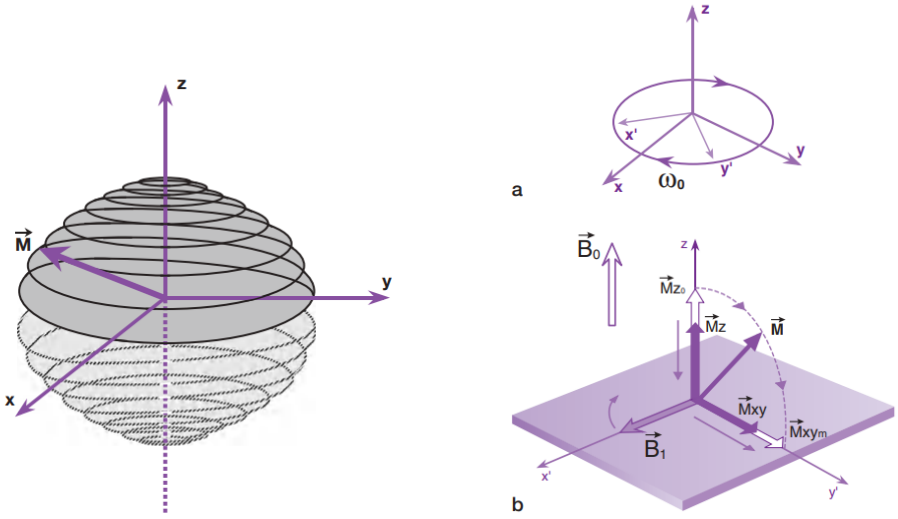
A l'arrêt de l'excitation, le vecteur d'aimantation va retourner dans sa position d'équilibre : il s'agit de la relaxation. Ce retour à l'équilibre se fait dans un mouvement de spirale autour de l'axe de $\vec{B}_0$. La relaxation est en réalité composée de deux phénomènes : la relaxation spin-réseau et la relaxation spin-spin. La première correspond au retour à l'aimantation longitudinale d'origine M_{z0}. Au niveau quantique, le retour à l'équilibre correspond à un retour à l'état d'énergie le plus bas. La relaxation spin-spin correspond au retour à zéro des composantes transverses de l'aimantation. Du point de vue quantique, l'excitation a aligné les phases des spins et le retour à l'équilibre correspond à un déphasage de ces derniers. Au niveau macroscopique, la relaxation peut être décrite à travers les équations de Bloch [Bloch, 1946] :

$$M_z(t) = M_{z0}\left(1 - e^{-t/T1} + M_z(0)e^{-t/T1}\right) \tag{1.1}$$

$$M_x(t) = e^{-t/T2}\left(M_x(0)\cos(\omega_0 t) + M_y(0)\sin(\omega_0 t)\right) \tag{1.2}$$

$$M_y(t) = e^{-t/T2}\left(M_y(0)\cos(\omega_0 t) - M_x(0)\sin(\omega_0 t)\right) \tag{1.3}$$

où T1 est le temps de relaxation longitudinale et T2 est temps de relaxation transversale. Ces temps sont caractéristiques des tissus observés et permettent donc une différenciation de ces derniers. Ils constituent la base des contrastes en IRM. On peut mesurer ces valeurs, de façon indirecte, en mesurant l'aimantation transverse par le biais d'une bobine qui est pondérée par ces constantes de temps.



(a) Dans le repère du laboratoire. (b) Dans le repère tournant : bascule.

FIGURE 1.2 – *Mouvement de l'aimantation sous l'influence de $\vec{B}_1$ formant une spirale autour de $\vec{B}_0$. Si on se place dans le repère tournant autour de $\vec{z}$ à la vitesse ω_0, le mouvement correspond à une bascule de l'aimantation dans le plan transversal. Image tirée de [Kastler et Vetter, 2018].*

En pratique, on ne peut pas avoir un signal directement pondéré en T2 car le champ $\vec{B}_0$ n'est pas parfaitement homogène, les aimantations au retour à l'équilibre ne précessent pas toutes exactement à la même fréquence de Larmor. Certaines auront un mouvement de précession plus rapide car le champ $\vec{B}_0$ est localement plus intense et d'autres seront plus lentes à cause d'un champ localement plus faible. On mesure sur le signal recueilli un T2*, plus petit que le T2, et qui peut être défini selon la relation : $\frac{1}{T2^*} = \frac{1}{T2'} + \frac{1}{T2}$ où T2' correspond uniquement à la partie liée aux inhomogénéités. Pour accéder au vrai T2, on utilise une séquence de type spin-écho (Figure 1.3). Cette séquence commence par une onde RF avec un angle de 90° qui permet de basculer l'aimantation dans le plan transversal. Les aimantations vont alors précesser à des vitesses légèrement différentes du fait des inhomogénéités de champ $\vec{B}_0$. Ensuite, une onde RF de 180° est envoyée au bout d'un temps TE/2 (TE est appelé le temps d'écho). Elle permet d'inverser les phases des aimantations dans le plan transversal. Ainsi les populations de spins qui tournaient à une vitesses angulaire plus faible vont passer devant celles aux aimantations les plus rapides. La vitesse de précession pour chacune des populations de spins reste la même car le champ est supposé inhomogène mais stable. Les populations de spins les plus rapides vont donc rattraper les plus lents. Ainsi, au bout d'un temps TE après l'impulsion à 90°, les phases des aimantations vont s'aligner : un écho est alors perceptible sur l'aimantation transversale. Cet écho de signal a une amplitude directement pondérée par T2, en ayant effacé les déphasages liés aux inhomogénéités.

I.1.2 Encodage spatial

Nous avons vu comment générer et acquérir un signal RMN. Pour former une image, il est nécessaire de connaître l'emplacement spatial de chaque signal émis. Pour cela, des encodages dit en phase et en fréquence des aimantations sont utilisés. Ils utilisent l'application temporaire d'un gradient de champ, superposé au champ $\vec{B}_0$, qui modifie linéairement selon l'axe choisi, la force du champ magnétique appliqué. Ce gradient est généré grâce à des bobines placées dans l'IRM. Pour rappel, les aimantations amenées dans le plan transverse ont un mouvement de précession de vitesse angulaire directement liée à la force du champ magnétique appliqué. Il est nécessaire de savoir localiser le signal mesuré par le principe de

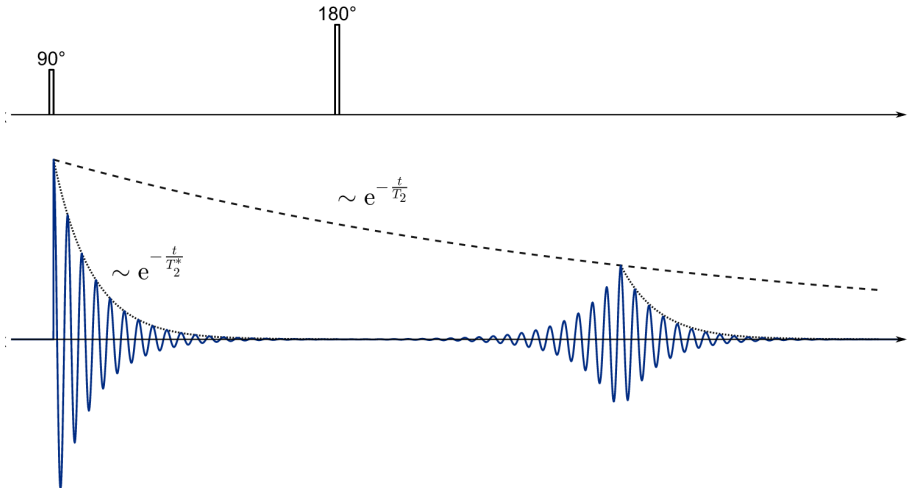


Figure 1.3 – *Séquence spin-echo (en haut) pour l'estimation du T2 réel à partir du signal reçu (en bas). La bascule à 90° ne permet d'estimer que T2*. En ajoutant la bascule à 180°, on peut estimer T2. Image issue de Wikimedia Commons.*

RMN dans les 3 dimensions. On ajoute donc ces gradients selon un axe choisi à des moments stratégiques de la séquence spin-écho présentée dans le paragraphe précédent. La nouvelle séquence est présentée en Figure 1.4.

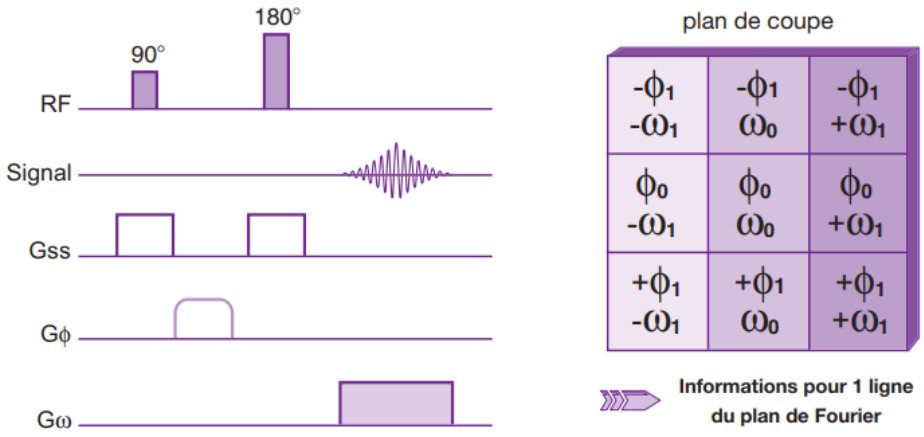


Figure 1.4 – *Séquence de spin-écho pour un encodage de phase et encodage spatial associé pour une coupe. Le gradient Gss permet de sélectionner la coupe, le gradient G_ϕ permet l'encodage en phase et le gradient G_ω permet l'encodage en fréquence. Image tirée de [Kastler et Vetter, 2018].*

La première étape consiste à sélectionner une coupe axiale selon l'axe $\vec{z}$ du champ $\vec{B}_0$, la séquence devant être appliquée pour chacune des coupes souhaitées. Pour cela, un gradient est appliqué selon cet axe au moment de l'excitation du système par une ondes RF (Gss dans la Figure 1.4). Avec l'application de ce gradient de champ magnétique, les fréquences de résonance seront directement liées à la position selon cet axe. Pour n'exciter qu'une coupe selon cet axe, l'onde radiofréquence 90° (et de façon facultative 180°) sera adaptée pour ne sélectionner que les fréquences correspondant à la coupe choisie. Par cette technique, on peut donc sélectionner la coupe voulue selon l'axe $\vec{z}$.

Il est ensuite nécessaire de se repérer dans l'espace 2D de cette coupe. Un gradient de champ est ajouté selon l'axe $\vec{x}$ pendant la lecture de l'écho (G_ω dans la Figure 1.4). Sous

l'effet de ce gradient, les fréquences de précession varient linéairement le long de l'axe. A la lecture, le signal reçu sera porteur de ces fréquences (voir Figure 1.5). Il sera possible de séparer les signaux liés à chacune des fréquences grâce à une transformée de Fourier.

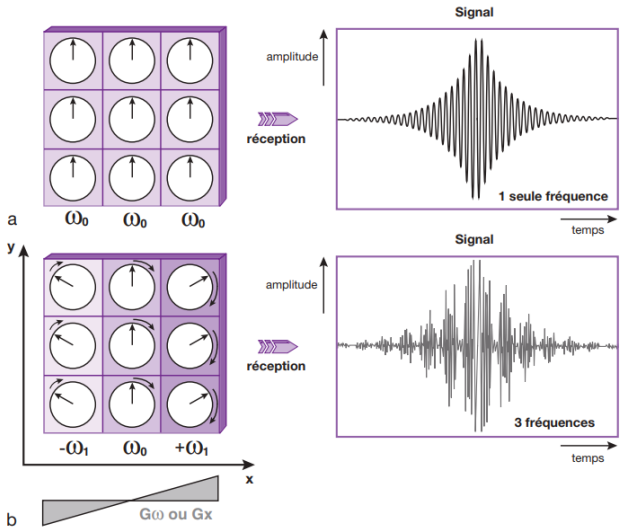


FIGURE 1.5 – *Encodage en fréquence et signaux de RMN associés pour une coupe. En haut, on visualise le signal reçu sans gradient de champ et en bas, avec un gradient. Image tirée de [Kastler et Vetter, 2018].*

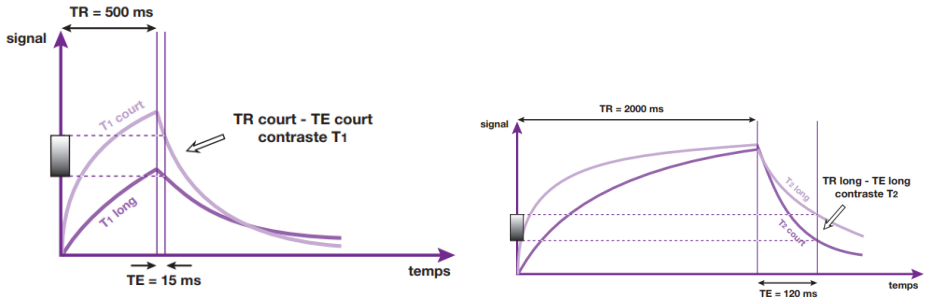
Enfin, pour récupérer la dernière dimension, un encodage en phase est utilisé. En effet, il est impossible d'utiliser un gradient selon l'axe $\vec{y}$ au moment de la lecture car le gradient résultant serait la somme des deux gradients. En modifiant la vitesse de précession des aimantations de façon momentanée par l'ajout d'un gradient de champ temporaire, un déphasage de ces derniers est créé. Une fois l'application du gradient arrêté, ce déphasage persiste et est donc observable a posteriori. On applique donc un gradient selon l'axe $\vec{y}$ en dehors de l'application des autres gradients ou de la lecture (G_ϕ dans la Figure 1.4). Dans ce cas, le passage dans le domaine de Fourier ne permet pas d'extraire chacune des phases mais l'ensemble des phases pour une fréquence donnée. L'opération doit donc être effectuée en faisant varier la phase et donc le gradient associé, pour chaque valeur de y c'est-à-dire pour chaque ligne de notre image axiale. Le temps séparant deux répétitions du schéma d'acquisition est appelé le temps de répétition (TR).

L'acquisition avec cette séquence est très longue. En pratique des méthodes plus rapides ont été développées comme la séquence de spin-écho rapide (turbo spin echo). Et de façon générale, de nombreuses variantes de ces principes ont été proposées.

I.2 Modalités IRM utilisées pour le cerveau

En changeant le temps d'écho (TE) et le temps de répétition (TR), on peut changer la pondération de l'image RM obtenue.

T1 En imposant un TR court, l'aimantation de tous les tissus n'aura pas le temps de revenir à l'équilibre entre deux bascules de 90°. On aura donc un fort signal pour les tissus au T1 le plus court et au contraire un signal faible pour les tissus au T1 plus long (Figure 1.6a). En imposant un TE court, on limite la pondération liée à la relaxation T2 dans le signal. Ainsi, avec des TE et TR courts (respectivement entre 10 et 20ms et 400 et 600ms), on obtiendra une image dite pondérée T1 (avec un signal de plus forte intensité pour les T1 plus courts et moins intense pour les T1 longs). Cette image est utilisée pour visualiser



(a) Pour deux tissus au T1 différents. (b) Pour deux tissus au T2 différents.

FIGURE 1.6 – *Aimantation transverse au cours du temps. Image tirée de [Kastler et Vetter, 2018].*

l'anatomie du cerveau. En effet, elle permet de différencier le liquide cérébrospinal (faibles intensités dans l'image car possédant des T1 très longs) dans lequel baigne le cerveau, de la substance grise et blanche (intensités moyennes et fortes) qui sont respectivement composées des corps cellulaires et des axones des neurones (plus de détails en Section II.1.1). Elle est également utilisée après l'injection d'un produit de contraste : le Gadolinium. Ce produit est injecté par intraveineuse et peut traverser la paroi des vaisseaux sanguins pour certaines parties du cerveau permettant de discriminer certaines pathologies en marquant les tissus inflammatoires. L'intégration dans les tissus du gadolinium est très visible avec cette pondération, car il possède de faibles T1 et apparaît donc en hypersignal.

T2 A l'inverse, en imposant un TR long, la contribution T1 sera effacée. En choisissant un TE long, on accentue les différences entre les tissus au long T2 et ceux au T2 plus court (Figure 1.6b). On obtient donc une image pondérée T2 avec des TE et TR longs (respectivement supérieur à 80ms et supérieur à 2000ms). Les T2 les plus longs auront un signal plus fort. Cette modalité permet souvent de distinguer les tissus pathologiques des tissus sains. En effet, leur T2 est souvent plus long et ces tissus apparaissent donc en hypersignal.

FLAIR Enfin, la modalité FLAIR est également très utilisée pour le cerveau. Il s'agit d'une séquence dite d'inversion-récupération. Elle se différencie de la séquence écho-spin par l'ajout d'une préparation à l'aimantation, c'est-à-dire de l'ajout d'une bascule de 180° avant la séquence écho-spin qui inverse, sur l'axe $\vec{z}$, l'aimantation longitudinale. Le délai entre la bascule de 180° et le lancement de la séquence écho de spin (avec la bascule de 90°) est appelé temps d'inversion (TI). Cette préparation permet de jouer sur la différence de T1 entre les tissus. En choisissant un TI long (2000ms) et une pondération T2, on s'affranchit du signal de l'eau libre et donc du liquide cérébrospinal (aux T1 longs). En effet, avec ce TI, la plupart des aimantations sont revenues à l'équilibre alors que celles liées aux liquides ne font qu'arriver au plan transversal au moment de la bascule à 90°. Lorsque la séquence écho de spin est lancée, les aimantations de ces composantes aux T1 longs ne sont pas excitées comme les autres. Elles apparaissent alors en hyposignal par rapport à la pondération T2. Les tissus pathologiques, de la même façon que pour les images uniquement pondérées en T2, apparaissent souvent en hypersignal. Cette modalité est l'une des plus utilisée [Hajnal *et al.*, 1992].

II Pathologies cérébrales considérées

Dans ce manuscrit, nous considérerons deux pathologies pour lesquelles l'IRM est utilisée dans le diagnostic et le suivi : la sclérose en plaques (SEP) et les gliomes.

II.1 Sclérose en plaques (SEP)

La sclérose en plaques est une maladie auto-immune inflammatoire et démyélénisante du système nerveux central (cerveau, moelle épinière et nerfs optiques). Il s'agit de la deuxième cause de handicap acquis chez les jeunes adultes en France après les traumatismes. Les atteintes et symptômes sont variées : visuels, moteurs, urinaires, sensitifs ou cognitifs. Elle touche plus de 2,5 millions de personnes dans le monde dont environ 120000 en France. Cela représente environ 3000 nouveaux cas diagnostiqués par an en France. Les femmes sont les plus touchées puisqu'elles représentent environ 75% des cas. Il y a également plus de cas dans les pays du nord. Les premiers symptômes apparaissent généralement entre 25 et 35 ans.

Les causes exactes de la sclérose en plaques sont encore méconnues. Des facteurs génétiques ou encore environnementaux ont été identifiés comme la faible synthèse de la vitamine D, le tabagisme, une infection au virus d'Epstein-Barr, etc. Cependant, le caractère multi-factoriel avec une prédisposition génétique et des facteurs environnementaux importants ne permet de conclure sur la cause réelle de la maladie.

II.1.1 Physiopathologie

La sclérose en plaques est une maladie auto-immune dont tous les processus ne sont pas encore connus. Elle se présente majoritairement sous la forme de poussées (forme récurrente-rémittente). Lors de ces dernières, les cellules immunitaires (lymphocytes B et T) traversent la paroi des vaisseaux sanguins du système nerveux central créant une rupture de la barrière hémato-encéphalique et s'attaquent aux neurones et aux cellules myélinisantes présents dans le système nerveux central. Cela se traduit par une inflammation focale : la plaque.

Les neurones assurent la réception et la transmission de l'influx nerveux, un signal bioélectrique, avec d'autres cellules nerveuses, musculaires ou encore glandulaires. Les neurones du système nerveux central intègrent ainsi les informations sensorielles (comme le toucher) reçues des neurones sensitifs et ordonnent une activité motrice via les neurones moteurs (comme la contraction d'un muscle). Dans chaque neurone, le signal électrique est reçu aux niveaux des dendrites puis il est traité par le corps cellulaire et repart par l'axone (Figure 1.7). L'information est alors transmise à d'autres cellules nerveuses par le biais des synapses où des échanges chimiques permettent à l'information de passer d'un neurone à l'autre.

L'axone est entouré d'une gaine de myéline discontinue composée majoritaire de lipides mais aussi de protéines. Les zones très minces sans myéline sont appelées les nœuds de Ranvier. Cette gaine permet d'augmenter la vitesse de propagation de l'influx nerveux. En effet, sans cette gaine, la vitesse de propagation dans l'axone conducteur est entre 10 et 75 m/s. Lorsque la gaine est présente, le potentiel d'action électrique passe de nœud de Ranvier en nœud de Ranvier plutôt que de suivre l'axone en continue. Cette conduction saltatoire permet d'accélérer la conduction électrique avec une vitesse maximale de 120 m/s.

Dans la sclérose en plaques, les cellules lymphocytaires s'attaquent à la myéline et aux cellules qui la produisent, les oligodendrocytes. Quand la couche de myéline est endommagée, les axones affectés ne peuvent transmettre les signaux que de manière limitée, voire pas du tout. En fonction des axones touchés et de leur rôle dans le système nerveux central, les symptômes seront différents. L'atteinte de ces axones est visible à l'IRM et se caractérise par des lésions dans la substance blanche, appelées plaques. En effet, la substance blanche représente les axones entourés de leur gaine de myéline alors que la substance grise représente les corps cellulaires des neurones. Dans les formes les plus avancées, des lésions de la substance grise peuvent également apparaître.

En parallèle, les oligodendrocytes, s'ils ne sont pas complètement détruits par la ré-

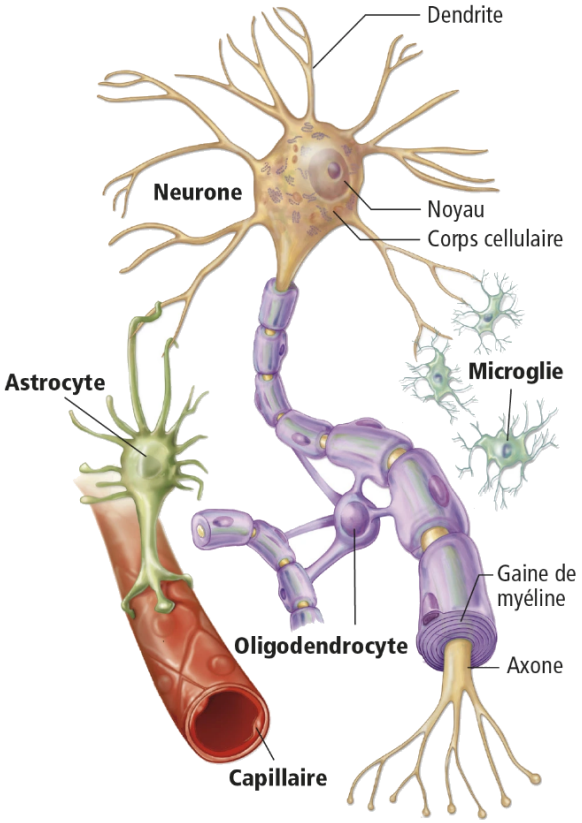


FIGURE 1.7 – *Schéma d'un neurone et des cellules gliales (oligodendrocyte, astrocyte, microglie). Image issue de [Mader et al., 2009].*

ponse immunitaire, sont capables de repousser et de produire de nouveau de la myéline pour réparer les axones endommagés comme montré en Figure 1.8. Ainsi, en dehors des poussées, les patients peuvent voir leurs symptômes disparaître ou diminuer grâce à cette remyélinisation totale ou partielle. Les lésions dites fantômes correspondent à une remyélinisation d'un axone lésé. La gaine de myéline est alors plus fine sans qu'un processus actif des lymphocytes ne soit encore présent sur ces axones (flèches rouges de la Figure 1.8).

Néanmoins, ce processus de remyélinisation est très hétérogène entre les patients et lorsque cette dernière n'a pas lieu, les neurones privés de leur gaine protectrice dégénèrent et meurent. Ainsi en fonction de cette remyélinisation, de la charge lésionnelle globale et de la localisation des lésions, il peut ne pas y avoir de récupération du patient. En outre, la perte myélinique, axonale et neuronale peut conduire à terme à une atrophie cérébrale, c'est-à-dire une diminution du volume cérébral.

Une gliose atrocytaire, c'est-à-dire du tissu cicatriciel, peut également apparaître sur les axones touchés par les cellules immunitaires.

II.1.2 Formes de la sclérose en plaques

L'évolution de la maladie peut prendre trois formes représentées en Figure 1.9.

85% des patients présentent une forme dite récurrente-rémittente. Les phases d'apparition de nouveaux symptômes, autrement dit les phases de poussées, sont séparées par des phases de rémission durant lesquelles les symptômes s'estompent ou disparaissent. Les poussées peuvent durer quelques jours à un mois et doivent être espacées d'au moins un mois.

Dans 50% des cas, la maladie évolue, dans les 5 à 20 ans après le diagnostic, vers une forme progressive où le handicap persiste et augmente avec le temps, indépendamment de

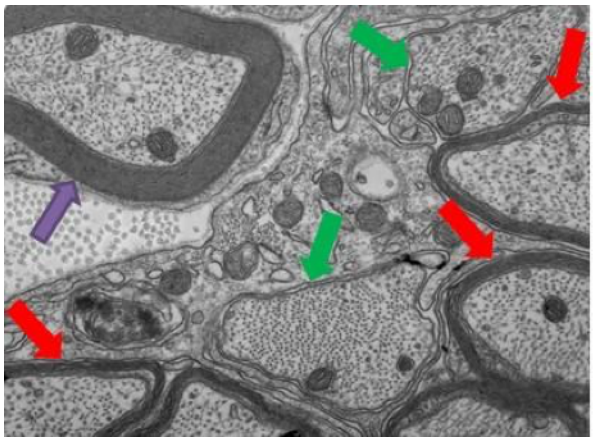


FIGURE 1.8 – *Différence entre un axone sain (flèche violette), un axone démyélinisé (flèches vertes) et un axone remyélinisé (flèches rouges). Acquisition par microscope électronique. Image issue du site de l'Institut du cerveau.*

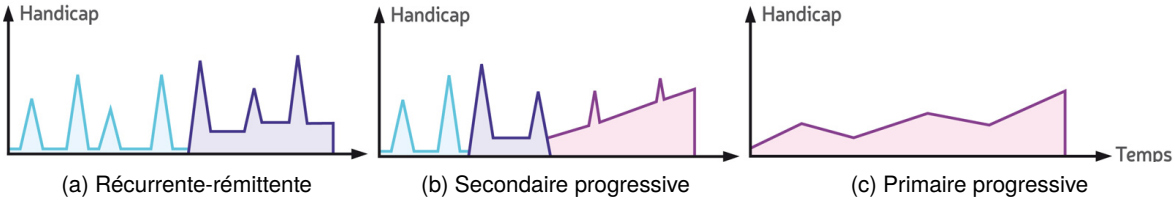


FIGURE 1.9 – *Handicap en fonction du temps pour les différentes formes de SEP. Images tirées du site de Notre sclérose.*

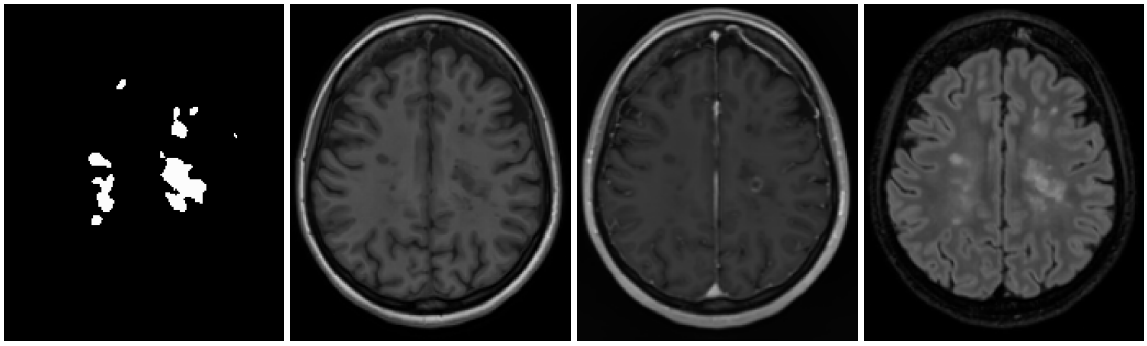
la survenue de nouvelles poussées. Cette forme est appelée forme secondaire progressive. Des poussées avec des symptômes plus intenses et temporaires peuvent s'ajouter mais elles sont moins nombreuses.

Enfin, une forme plus rare est progressive dès le début de la maladie : il s'agit de la forme primaire progressive. Elle touche des personnes plus âgées (autour de 50 ans). Les symptômes et le handicap augmentent avec le temps sans rémission et avec presque aucune poussée.

II.1.3 Diagnostic

Les symptômes de la SEP étant très variés, il est difficile d'identifier cette maladie ou de la distinguer d'autres pathologies à partir de ces derniers. Les critères pour diagnostiquer la maladie ont évolué depuis l'établissement des premiers critères par Allison et Millar en 1954 avec l'avancée des connaissances sur la maladie et celle des technologies utilisées pour le diagnostic. Aujourd'hui, le diagnostic repose sur le critère de McDonald révisé en 2017 [Thompson *et al.*, 2018]. Il repose sur la preuve d'une dissémination spatiale et temporelle des atteintes et donc des lésions, c'est-à-dire que les lésions doivent apparaître dans plusieurs territoires du système nerveux central et à différents moments. En outre, il est nécessaire d'avoir écarté les autres pathologies. Il est donc possible de diagnostiquer la sclérose en plaques sur les symptômes à travers un interrogatoire : si plusieurs poussées ont eu lieu avec des symptômes impliquant des zones du système nerveux central différentes, le diagnostic peut être posé. Néanmoins, la preuve clinique que plusieurs zones sont touchées peut être difficile à obtenir du fait de la complexité du système nerveux central.

L'IRM est le moyen le plus sensible pour diagnostiquer la sclérose en plaques. En effet,



(a) Lésions FLAIR (b) T1 (c) T1 Gd (d) FLAIR

FIGURE 1.10 – *Exemple d'IRM (vue axiale) d'un patient SEP (base MSSEG).*

elle permet de visualiser les lésions du cerveau, de la moelle épinière et des nerfs optiques. La dissémination spatiale peut donc être facilement établie en identifiant le nombre de lésions dans les différentes zones. Pour cela, l'IRM FLAIR est principalement utilisée pour le cerveau et l'IRM T2 pour la moelle osseuse En effet, les lésions de sclérose en plaques y apparaissent en hyperintensités. L'IRM T1 peut également être utilisée comme séquence anatomique, pour quantifier l'atrophie cérébrale [Rocca *et al.*, 2017] et évaluer les "trous noirs" témoins d'une atteinte axonale sévère. Sur cette modalité, les lésions apparaissent plutôt en signal moyen ou hyposignal. Plus le signal est en hypointensités, plus la densité axonale est faible. La dissémination temporelle peut également être établie grâce à l'IRM avec l'apparition de nouvelles lésions entre deux IRM prises à des moments différents ou bien avec l'injection d'un produit de contraste, le Gadolium (Gd). En effet, l'injection de cet agent permet de discriminer une partie les lésions actives pour lesquelles il y a une rupture de la barrière hémato-encéphalique. Dans ce cas, l'agent de contraste peut passer du sang au système nerveux central. On observe alors une lésion rehaussée sur l'IRM T1. A noter que les lésions dont la taille augmente sont aussi considérées comme actives, même sans rehaussement avec l'agent de contraste. La présence conjointe d'une lésion ne prenant pas le contraste (lésion ancienne) et d'une le prenant (lésion récente) sur une même IRM confirme la dissémination temporelle de la maladie. Un exemple des modalités IRM pour un patient SEP est donné en Figure 1.10. En France, l'Observatoire Français de la Sclérose En Plaques (OFSEP) a établi un protocole IRM détaillant les modalités à utiliser ainsi que le suivi à mettre en place [Brisset *et al.*, 2020].

Une ponction lombaire peut également être effectuée pour rechercher des signes d'inflammation dans le liquide céphalo-rachidien à travers le dosage d'anticorps. Elle n'est pas utilisée dans le suivi mais plutôt pour le diagnostic initial, notamment pour écarter d'autres pathologies.

II.1.4 Traitements

Il n'existe aucun traitement pour guérir la sclérose en plaques mais d'énormes progrès ont été faits pour stabiliser la maladie. L'objectif des traitements symptomatiques est d'améliorer le confort de vie des patients en agissant pendant les poussée et sur les symptômes. Un traitement de fond a pour but de réduire la fréquence des crises et donc la progression du handicap. Ainsi, chaque symptôme est traité individuellement quand cela est possible. Un traitement anti-douleur et de la rééducation peuvent notamment être mis en place. Pendant les crises, des anti-inflammatoires sont administrés pour limiter l'impact de ces dernières. Enfin, un traitement de fond est mis en place en continu. Ce traitement est basé sur des immunomodulateurs ou des immunosuppresseurs dans les formes les plus graves. L'objec-

tif est de moduler, voire de réduire les défenses immunitaires responsables de la sclérose en plaques. L'utilisation des immunosuppresseurs a des conséquences importantes puisque tout le système immunitaire est affaibli et le patient peut être plus sensible aux infections.

La recherche de nouveaux traitements permettant de réguler le système immunitaire ou encore de favoriser la reconstruction de la myéline pour une meilleure récupération des patients est en cours.

L'IRM constitue un atout dans le suivi des patients sous traitement. En effet, elle permet de visualiser l'impact de ce dernier sur l'apparition de lésions (même asymptomatiques).

II.2 Gliomes

Les gliomes sont des tumeurs du système nerveux central issues des cellules gliales notamment des astrocytes et des oligodendrocytes (Figure 1.7). Ils correspondent à un amas anormal suite à une multiplication anarchique de ces cellules. L'apparition de ces tumeurs est rare mais elles représentent plus de la moitié des tumeurs cérébrales primitives, c'est-à-dire qui ne sont pas des métastases d'un cancer dans une autre partie du corps. En France, environ 3000 cas sont diagnostiqués tous les ans avec une augmentation notable des cas depuis une dizaine d'années. Ces tumeurs touchent tous les âges avec une prédominance masculine. Les causes sont multi-factorielles et encore méconnues : des facteurs environnementaux comme l'exposition à des champs électromagnétiques induits par exemple par les téléphones portables ou encore une prédisposition génétique pourraient être impliqués.

Les gliomes peuvent être bénins, c'est-à-dire sans récidive en cas de résection, ou malins mais sans produire de métastases. Néanmoins, dans les deux cas le pronostic vital est engagé du fait de leur emplacement rendant les traitements difficiles et les conséquences graves. La présence de cette masse augmente la pression dans le cerveau pouvant aller jusqu'au coma. Leur caractère souvent infiltrant, où les cellules tumorales et saines se mélangent, pose également problème. La gravité de ces tumeurs peut être classée selon leur grade. Les symptômes sont variés (troubles moteurs, cognitifs ou comportementaux) en fonction de la localisation de la tumeur dans le système nerveux central ou encore sa taille. Certains symptômes peuvent également être dû à une hypertension intracrânienne liée à la compression du cerveau par la masse. Cette pathologie peut être également asymptomatique pendant longtemps entraînant un diagnostic tardif.

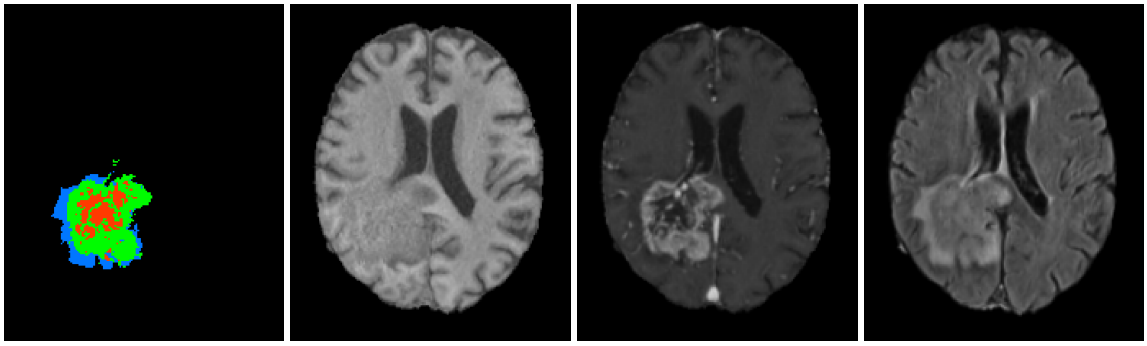
II.2.1 Grade

L'Organisation Mondiale de la Santé (OMS) à mis en place une échelle basée sur l'histologie de ces tumeurs gliales. On y retrouve quatre grades. Les deux premiers représentent des gliomes de bas grade alors que les deux derniers regroupent les gliomes de haut grade.

Le grade I correspond à des tumeurs bénignes, souvent des astrocytomes pilocytiques. Ils représentent environ 5% des gliomes et touchent principalement les enfants. Ils ont une croissance lente et des bords bien définis.

Le grade II regoupe les oligoastrocytomes (qui peuvent être des tumeurs dérivées des astrocytes ou oligodendrocytes), les oligodendrogliomes et les astrocytomes diffus. Ils représentent environ 15% des gliomes et touchent des sujets plutôt jeunes (en moyenne 35 ans). Leur évolution est plutôt lente mais ils peuvent évolués en des gliomes de grade III. Il peut s'agir de tumeurs diffuses.

Les gliomes de grade III sont des gliomes anaplasiques. L'anaplasie correspond à l'absence de différenciation complète des cellules d'un tissu. Elles sont donc plus proches des cellules immatures et ont perdu certains caractères de la différenciation cellulaire. Ces tumeurs sont souvent issues de gliomes de stade II. Elles sont diagnostiquées chez l'adulte entre 40 et 50 ans. Leur développement est rapide et elles envahissent les tissus environnants. La masse est hétérogène. La survie moyenne est de l'ordre de 2 à 4 ans. Elles peuvent



(a) Tumeur (b) T1 (c) T1 Gd (d) FLAIR

FIGURE 1.11 – *Exemple d'IRM (vue axiale) d'un patient atteint d'un gliome de haut grade (base BraTS). Pour le masque de la tumeur, le rouge correspond à une zone sans prise de contraste ou de nécrose, le vert correspond à la zone avec prise de contraste et le bleu correspond à l'œdème.*

dégénérées en un gliome de stade IV.

Les gliomes de stade IV sont les glioblastomes multiformes. Ce sont les gliomes les plus agressifs et les plus fréquents chez l'adulte (60%). L'âge moyen de diagnostic est 65 ans. Ce sont des astrocytomes présentant des tissus nécrosés à la différence des astrocytomes de stade III. Un œdème se forme également. Ils sont très vascularisés et infiltrants. La survie est de située entre 2 mois et 2 ans.

II.2.2 Diagnostic

En cas de suspicion d'un gliome suite à l'examen clinique et à la présence des différents symptômes, il est nécessaire d'explorer le système nerveux central. Un scanner CT peut être fait en première attention, il permet de repérer 80% des tumeurs cérébrales. Cependant, l'IRM reste l'examen le plus précis. En effet, comme le scanner, l'IRM permet de localiser la tumeur et d'apprécier sa taille mais aussi de visualiser des tumeurs invisibles au scanner et d'obtenir plus d'information sur la tumeur. L'IRM FLAIR est utilisée pour visualiser en partie l'infiltration ainsi que l'œdème et la nécrose qui apparaissent en hypersignal. En IRM T1, la tumeur est plutôt de faible intensité. La prise de contraste suite à l'injection du Gadolinium met en avant un processus d'anaplastie permettant d'identifier les grades supérieurs. Des exemples d'IRM pour un patient atteint d'un glioblastome sont donnés en Figure 1.11.

La classification de l'OMS repose sur une analyse histologique. Cette analyse est donc réalisée suite à une biopsie de la tumeur. Néanmoins, selon la zone prélevée, le résultat peut fortement varié. En effet, plusieurs grades peuvent être présents en parallèle. Il est donc important de prélever à différents endroits pour une analyse complète de la tumeur. L'IRM peut aider à localiser les différentes zones de la tumeur et donc où les prélèvements doivent être réalisés. En outre, l'analyse histologique prend plusieurs jours. L'échelle de Sainte-Anne utilise l'IRM, en plus de l'analyse histologique, pour différencier les gliomes et contrer les faiblesses d'une seule analyse histologique.

II.2.3 Traitements

Le traitement dépend du grade, de l'infiltration et de la localisation de la tumeur. En effet, si elle est opérable, il est recommandé de retirer chirurgicalement un maximum de tumeur. Dans ce cas, l'IRM aura permis de délimiter la zone tumorale à retirer. Le contrôle post-opératoire se fait également par IRM. Il permet de vérifier qu'il n'y a pas de complication et que toute la zone tumorale (possible) a été retirée.

Une radiothérapie peut également être réalisée mais elle est compliquée car il ne faut pas endommager les tissus environnants au risque de créer de lourds handicaps. Encore une fois, l'IRM permet de délimiter la zone à traiter. La chimiothérapie peut également être utilisée mais la barrière hémato-encéphalique réduit le passage des médicaments dans le système nerveux central. Des techniques utilisant des ultrasons peuvent être utilisées pour ouvrir cette barrière de manière temporaire [Idbaih *et al.*, 2019]. Ces deux traitements sont souvent couplées avec l'exérèse.

Comme pour la sclérose en plaques, des traitements et soins palliatifs sont apportés aux patients pour réduire les symptômes et améliorer leur confort de vie.

III Données et prétraitements

III.1 Bases de données

Pour nos expériences, nous disposons de plusieurs bases de données d'IRM cérébrales : des bases de données de sujets sains et des bases de données de patients atteints de sclérose en plaques ou de gliomes. Ces bases de données nous permettent notamment de travailler sur deux modalités d'IRM d'intérêt : l'IRM FLAIR ou T1.

III.1.1 Bases de données de sujets sains

IXI La base de données publique IXI [1] (Information eXtraction from Images) est composée d'IRM pour 581 sujets sains. Nous avons utilisé les IRM T1. Parmi ces sujets, 45% sont des hommes et la moyenne d'âge est de 50 ± 17 ans. Les données ont été collectées dans 3 hôpitaux londoniens sur 3 machines différentes : un scanner Philips de 3T (32% des images), un scanner Philips de 1.5T (52%) et un scanner General Eletric de 1.5T. La taille des voxels est d'environ 1mm^2 et les images sont de taille 150×256.

HCP La base de données publique HCP [2] (Human Connectum Project) - Young Adult est composée de 1113 IRM de sujets sains dont nous avons utilisé les IRM T1. Toutes les IRM ont été réalisées sur un unique scanner Siemens de 3T à l'Université de Washington. 59% des sujets sont des femmes. Les sujets sont jeunes : entre 22 et 35 ans avec une moyenne de 29 ± 4 ans. Les images sont de taille $260 \times 311 \times 260$ pour des voxels de taille 0.7mm^2

IBC La base de données publique IBC (Individual Brain Charting) [Pinho *et al.*, 2018] est composée de 12 sujets sains dont nous avons utilisé les IRM FLAIR. Il y a 83% d'hommes. La moyenne d'âge est de 34 ± 5 ans. Les voxels sont environ de taille $0.5\text{mm} \times 0.5\text{mm} \times 1\text{mm}$ pour une image $512 \times 512 \times 160$. La base provient d'un unique scanner Siemens 3T.

kirby21 La base de données publique kirby21 [Landman *et al.*, 2011] est composée de 21 sujets avec chacun 2 IRM prises à une heure d'intervalle. Nous avons utilisé l'IRM FLAIR. Parmi ces sujets, nous en avons retiré 5 car ils présentaient des lésions cérébrales importantes et ne pouvaient donc pas être considérés comme radiologiquement sains. Un exemple de sujet retiré est donné en Figure 1.12. 55% des sujets utilisés sont des hommes et l'âge moyen est de 31 ± 7 ans. Les images proviennent d'un unique scanner Philips de 3T. Les images ont des voxels d'une taille d'environ 0.5mm^3 pour une image $327 \times 576 \times 576$.

1. https://brain-development.org/ixi-dataset
2. https://www.humanconnectome.org/study/hcp-young-adult

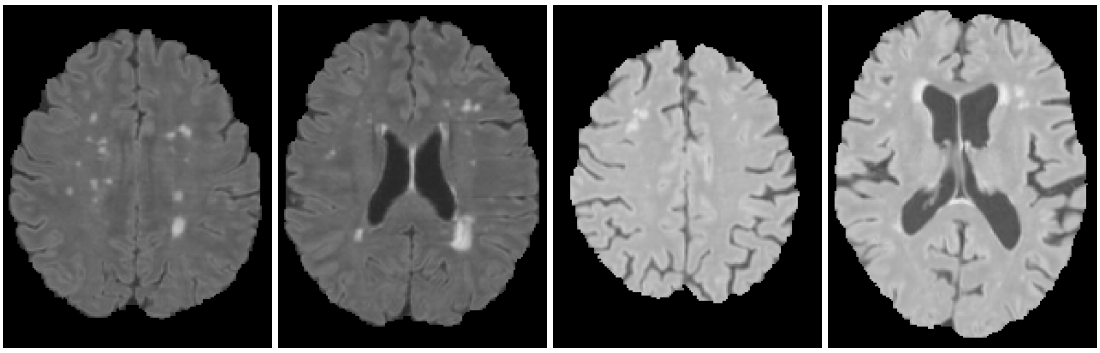


FIGURE 1.12 – *Exemples d'IRM FLAIR (coupe axiale) retirées des bases de données saines kirby21 et MPI car présentant des lésions cérébrales (apparaissant en hyperintensités). Les deux images de gauche correspondent à un même patient de la base kirby21 alors que celles de droites sont issue d'un seul patient de la base MPI.*

MPI La base de données publique MPI [Babayan *et al.*, 2019] est composée de 318 sujets sains. Nous n'avons retenu que les IRM FLAIR de haute résolution et 3D. Cela a réduits le nombre de sujets à 117. Ensuite, nous avons, comme pour la base kirby21, retiré les sujets présentant des lésions (Figure 1.12). Après ce filtrage, la base comporte 94 images. Elle est constituée de 19% de femmes. L'âge moyen est de 31 ± 14 ans. Les images font $192 \times 512 \times 512$ pour des voxels de taille $1\text{mm} \times 0.5\text{mm} \times 0.5\text{mm}$.

III.1.2 Bases de données de patients SEP

OFSEP1 La base de données OFSEP nous a été fournie par l'Observatoire Français de la Sclérose en Plaques [1] [Vukusic *et al.*, 2020, Confavreux *et al.*, 1992]. Cet organisme collecte les images de patients atteints de SEP dans 36 centres français. Nous avons ainsi récupéré 445 patients totalisant 3766 IRM de différentes modalités. Après avoir retirer les images mal étiquetées ou de trop mauvaise qualité mais aussi en ne conservant que les patients ayant une IRM T1 (que nous utilisons pour le recalage : voir Section III.2), nous obtenons une base finale de 512 IRM T1 et 501 IRM FLAIR correspondant à 427 patients car 62 patients ont plusieurs IRM prises à moments différents. Les images sont de taille variable avec une résolution voxelique d'au moins 1mm^3. Il s'agit d'une base très hétérogènes car multi-centrique. Plusieurs marques de scanner ont été utilisées : Siemens (71%), Philips (17%) et General Electric. En outre, pour une même marque, différents modèles et pour un même modèle des paramètres différents pour l'acquisition sont utilisés. Parmi les scanners, 42% sont en 3T. L'âge moyen est de 43 ± 12 ans. Il y a 68% de femmes. 82% des images correspondent à des patients ayant une forme récurrente-rémittente, 11% une forme secondaire progressive et 7% une forme primaire progressive

OFSEP2 Par la suite, nous avons obtenu d'autres données de l'Observatoire Français de la Sclérose en Plaques avec les mêmes caractéristiques que la base OFSEP1. Dans cette base, il y a 463 patients atteints de sclérose en plaques qui totalisent 595 IRM T1 et 572 IRM FLAIR.

MSSEG La base de données publique MSSEG (Multiple Sclerosis SEGmentation challenge) [Commowick *et al.*, 2021] est issue d'un challenge de la conférence MICCAI 2016. Les données ont été fournies par l'OFSEP. Elle est composée de 53 patients avec notamment les IRM T1 et FLAIR. Les images ont été acquises dans 4 centres différents avec 4 scanners : 28% des images ont été acquises avec un scanner Siemens de 3T, 28% avec un scanner Siemens

1. http://www.ofsep.org

de 1.5T, 28% avec un scanner Philips de 3T et le reste avec un scanner General Electric de 3T. La moyenne d'âge est de 45 ± 10 ans. Les femmes sont représentées à 72%. Comme pour la base OFSEP, la taille des images est variable avec des voxels d'une taille maximale de $1mm^3$. Cette base de données possède les annotations manuelles des lésions SEP annotées sur les IRM FLAIR. Ces annotations ont été réalisées par 7 radiologues juniors entraînés et issus des mêmes centres que les images. Certaines lésions ont été retirées de la segmentation comme les lésions de moins de $3mm^3$, les lésions punctiformes et les lésions périventriculaires évocatrices de leucoaraiose. Un consensus entre les experts a ensuite été généré en utilisant l'algorithme LOP STAPLE (Logarithmic Opinion Pool Based Simultaneous Truth And Performance Level Estimation) [Akhondi-Asl *et al.*, 2014] qui minimise les déviations individuelles.

III.1.3 Base de données de patients avec gliome

BraTS La base de données publique BraTS (Brain Tumors Segmentation challenge) [Bakas *et al.*, 2017, Bakas *et al.*, 2018, Menze *et al.*, 2014] est issue d'un challenge MICCAI. Deux versions seront utilisées dans ce manuscrit : celle de 2019 et celle de 2020, de nouvelles images étant ajoutées chaque année. La base de données 2020 est composée de 369 patients (contre 336 en 2019) atteints d'un gliome, avec notamment l'IRM FLAIR et T1 des patients. Les données proviennent de 11 centres américains utilisant des scanners de plusieurs marques (Philips, Siemens, General Electric, Hitachi) de 1.5T et 3T. La moyenne d'âge est de 60 ± 9 ans. 79% des cas sont des gliomes de haut grade. A la différence des autres bases, les images ont subi un prétraitement [1]. Il est constitué d'une correction des inhomogénéités de champ avec l'algortithme N4 [Tustison *et al.*, 2010], puis d'un recalage rigide entre les modalités et un recalage de la modalité T1 (post injection de Gadolinium) sur l'atlas SRI-24 [Rohlfing *et al.*, 2010] en utilisant FSL FLIRT [Jenkinson et Smith, 2001, Jenkinson *et al.*, 2002]. Le recalage calculé est alors appliqué sur les images sans correction N4. Enfin, le crâne est retiré pour ne garder que le cerveau en utilisant le réseau de neurones de [Thakur *et al.*, 2020]. Plus de détails sur les prétraitements pour l'IRM du cerveau seront donnés dans la Section III.2. La taille des images est de $240 \times 240 \times 155$ pour des voxels de taille $1mm^3$. La segmentation en 3 classes est fournie : la zone tumorale prenant le contraste après injection de Gadolinium, la zone nécrosée et la zone ne prenant pas le contraste, regroupées en une seule classe, et enfin l'œdème. Les segmentations ont été faites par 4 radiologues experts. Un consensus a par la suite été établi en utilisant le vote majoritaire. Un masque de l'ensemble de la masse tumorale peut être obtenu en combinant toutes les zones tumorales.

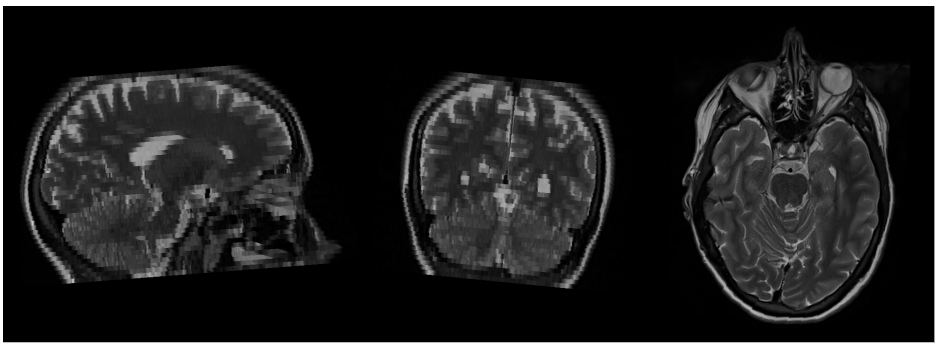
III.2 Prétraitements

Il est important de prétraiter les données IRM afin d'uniformiser le mieux possible les IRM acquises par différents centres avec des scanners et des paramètres d'acquisition très différents. Il est également nécessaire de supprimer certains artefacts qui sont propres à l'IRM. En normalisant les images, on diminue ainsi les biais et informations non pertinentes pour l'analyse. Pour cela, nous proposons la chaîne de prétraitements suivante :

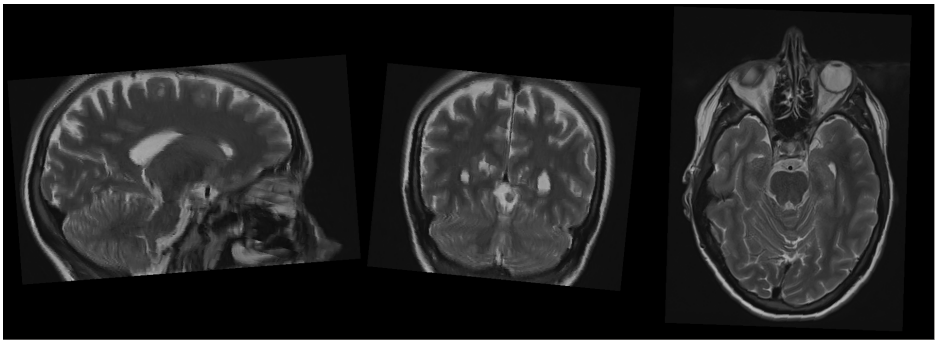
Interpolation entre les coupes axiales Dans les volumes IRM, il est très fréquent que la résolution en $\vec{z}$ soit plus faible que celle dans le plan axial afin de réduire le temps d'acquisition de la séquence. Pour obtenir des images isotropes, nous avons utilisé une interpolation inter-coupes basée sur du recalage [Sdika, 2013]. Chaque coupe est recalée sur la suivante et les coupes intermédiaires sont obtenues par interpolation entre la déformation géométrique obtenue et la déformation identité. Le processus est rendu symétrique en effectuant la même opération entre chaque coupe et la précédente et en moyennant les deux

1. https://cbica.github.io/CaPTk/preprocessing_brats.html

coupes intermédiaires obtenues. Pour assurer la continuité 3D du recalage et gagner en temps de calcul, tous les recalages 2D d'une coupe à l'autre sont obtenus avec un unique recalage 3D [Sdika, 2008] du volume initial sur le volume décalé d'une coupe. Un exemple d'avant-après cette interpolation est donné Figure 1.13.



(a) Avant interpolation



(b) Après interpolation

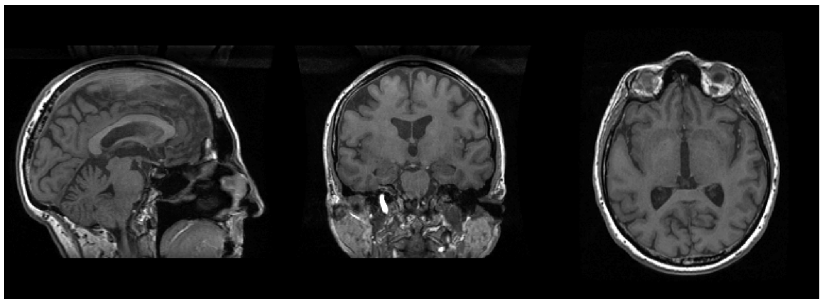
FIGURE 1.13 – *Exemple de l'interpolation inter-coupes sur une image T2.*

Recalage sur l'atlas du MNI Il est également nécessaire que les images se situent dans le même espace. Pour cela, un recalage sur un atlas doit être fait. Nous avons choisi l'atlas MNI152 qui correspond à la moyenne de 152 IRM T1 recalée dans l'espace du MNI (Montreal Neurological Institute-Hospital). Le calcul de la transformation pour passer dans cet espace doit donc se faire une IRM T1. Pour recaler les IRM T1 sur cet atlas, nous avons utilisé le recalage affine de FSL FLIRT [Jenkinson et Smith, 2001, Jenkinson *et al.*, 2002]. Un exemple est donné Figure 1.14.

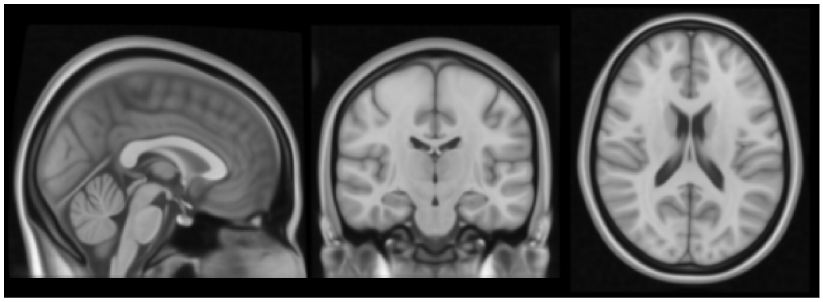
Lorsque d'autres modalités IRM sont utilisées, un recalage de ces modalités sur l'IRM T1 doit être fait au préalable. S'agissant du même sujet, un recalage rigide est utilisé pour placer les autres modalités dans l'espace du T1 du même sujet. Nous avons utilisé Elastix [Klein *et al.*, 2009, Shamonin *et al.*, 2014]. L'exemple d'une IRM FLAIR recalée sur l'IRM T1 du même sujet est donné en Figure 1.15. Une fois ce recalage fait, on peut appliquer la transformation calculée pour passer de l'IRM T1 au MNI.

Selon le choix de l'atlas utilisé, à savoir un atlas avec une résolution voxelique de 1mm^3 ou 2mm^3 dans notre cas, la résolution de l'image de sortie sera imposée. En outre, ce recalage permet d'obtenir des images de même taille à savoir $109 \times 91 \times 109$ pour une résolution de 2mm^3 et $182 \times 218 \times 182$ pour 1mm^3. A noter que ce recalage sur le MNI a également été fait sur le base déjà prétraitée BraTS.

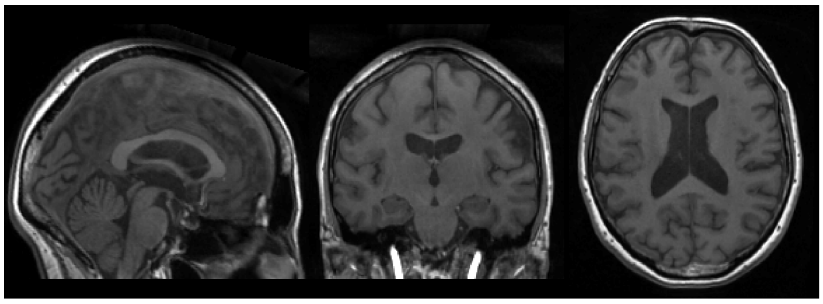
Suppression du crâne Le crâne ne nous apporte aucune information sur les pathologies considérées et la base BraTS proposent uniquement les images sans ce dernier. Pour ne



(a) T1 avant recalage



(b) Atlas T1 du MNI



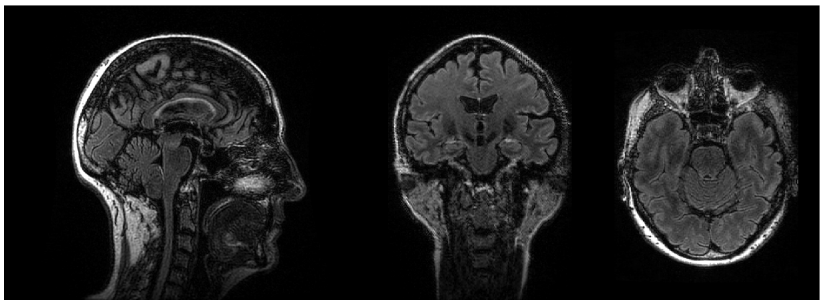
(c) T1 après recalage

FIGURE 1.14 – *Exemple de recalage d'une IRM T1 sur l'atlas du MNI.*

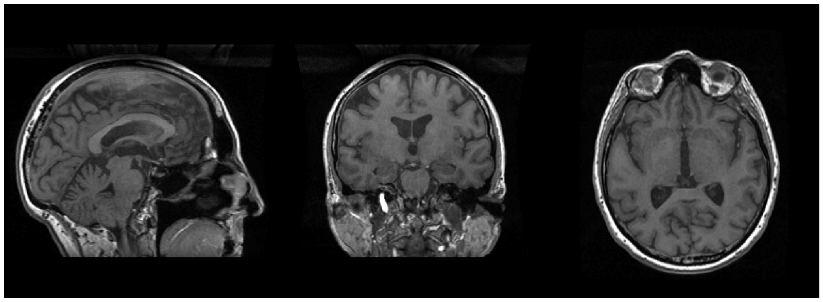
garder que l'information pertinente et uniformiser nos bases de données, nous retirons donc le crâne des IRM en ne gardant que le cerveau. Pour le retirer, nous avons utilisé HD-BET [Isensee *et al.*, 2019], un réseau de neurones entraîné à segmenter le cerveau. En appliquant le masque trouvé, on peut aisément retirer le crâne. Un exemple de segmentation est donné Figure 1.16.

Correction des inhomogénéités de champ Il est difficile de maintenir des champs magnétiques constants : les champs $\vec{B_0}$ et $\vec{B_1}$ peuvent donc varier localement. L'antenne de réception du signal peut également avoir un profil de sensibilité inhomogène. Cela crée des artefacts dans l'image acquise qui peuvent être représentés comme un champ de basse fréquence appliqué à l'image. L'image acquise peut être vue comme : $I_a = I_r B + N$ où I_r est l'image réelle, B le champ recherché et N du bruit. En supposant le bruit nul et en passant au logarithme, on a donc en posant $\hat{I} = log(I)$: $\hat{I_a} = \hat{I_r} + \hat{B}$. Nous avons utilisé la méthode [Tustison *et al.*, 2010], pour estimer et supprimer ce champ. Cette méthode estime itérativement le champ sachant le champ estimé à l'itération précédente : $\hat{B}^n = \hat{B}^{n-1} - S\left(\hat{B}^{n-1} - E(\hat{B}|\hat{B}^{n-1})\right)$ où S est un estimateur de B-Splines. Un exemple de correction est donné en Figure 1.17.

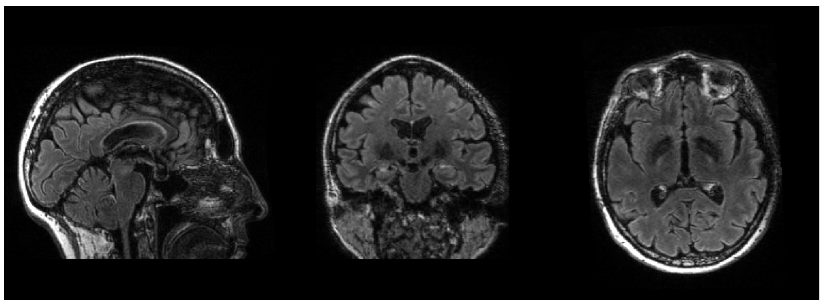
Cette correction peut poser problème dans le cadre de larges pathologies comme les tumeurs. En effet, la correction des basses fréquences peut fortement diminuer le contraste entre la tumeur et les tissus sains. Des exemples sont donnés Figure 1.18. Il est donc préférable



(a) FLAIR avant recalage



(b) T1



(c) FLAIR après recalage

FIGURE 1.15 – *Exemple de recalage d'une IRM FLAIR sur l'IRM T1 du même sujet.*

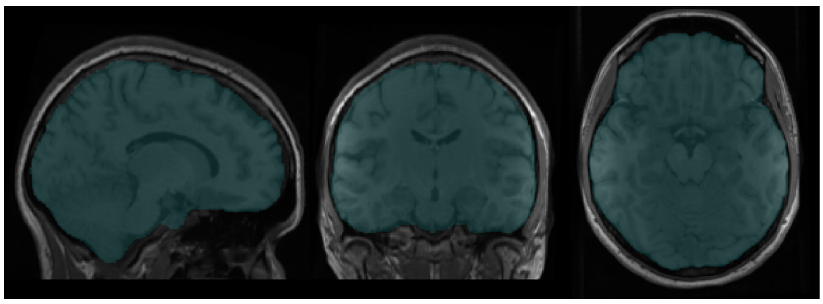
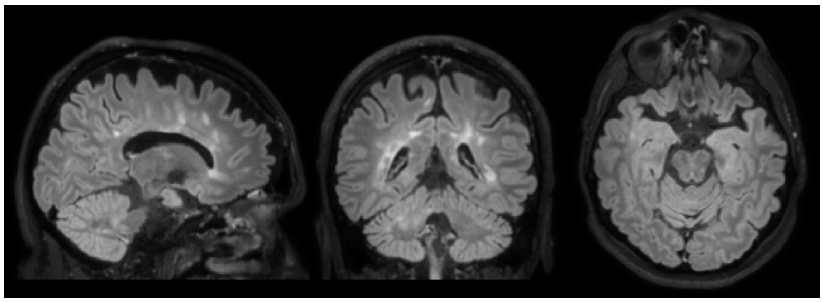
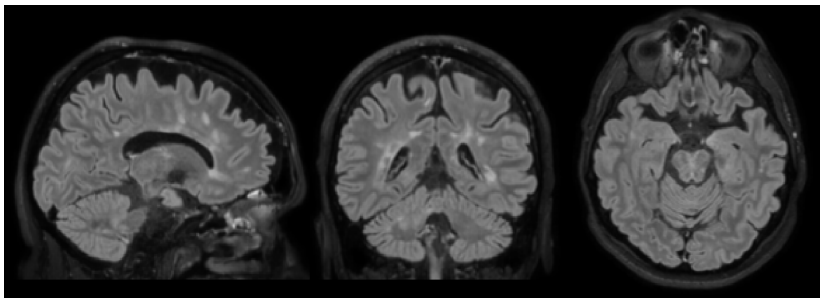


FIGURE 1.16 – *Exemple de segmentation du cerveau par HD-BET sur une IRM T1.*

de ne pas appliquer cette correction dans ce cas : la chaîne de prétraitements proposée pour le challenge BraTS n'utilise d'ailleurs pas les images après correction. Sans correction, il peut subsister des artefacts dus aux inhomogénéités de champ. L'utilisation de la correction peut également servir à créer une base de données de mauvaise qualité avec de faibles contrastes et donc plus difficile à analyser par les méthodes automatiques.



(a) Avant correction



(b) Après correction

FIGURE 1.17 – *Exemple de correction des inhomogénéités de champ sur une IRM FLAIR.*

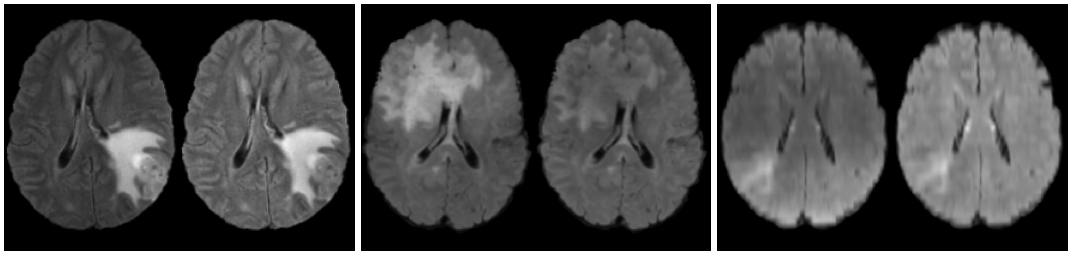
Standardisation Une fois ces étapes effectuées, l'image est centrée réduite en ne tenant compte que des intensités à l'intérieur du cerveau. Le fond est fixé égal à une constante.

La chaîne de prétraitement proposée est robuste : elle a été appliquée à de nombreuses bases de données avec succès. Elle a également été intégrée sur VIP [1] (Virtual Imaging Platform), un portail Web permettant de réaliser des calculs sur des données médicales grâce aux ressources disponibles dans l'organisation virtuelle biomed de l'e-infrastructure EGI [2]. Elle a été utilisée sur plusieurs projets au sein du laboratoire CREATIS.

III.3 Augmentation de données

Pour simuler une base de données plus grande, de l'augmentation de données peut être utilisée. Nous avons choisi d'utiliser 3 techniques appliquées de manière aléatoire. Tout d'abord, nous pouvons changer la luminosité des images en multipliant ces dernières par un facteur (entre 0.5 et 2). Nous pouvons également inverser les hémisphères du cerveau avec un effet miroir par rapport au plan sagittal. Enfin, nous pouvons procéder à des déformations élastiques pour modifier la forme du cerveau [Isensee *et al.*, 2020].

1. urlhttps ://www.creatis.insa-lyon.fr/vip/
2. https://www.egi.eu/



(b) Avant (c) Après (e) Avant (f) Après (h) Avant (i) Après

(c) Perte de contraste sur une partie de la tumeur (f) Perte de contraste sur toute la tumeur (i) Artefact de champ important

Figure 1.18 – *Exemples de correction d'inhomogénéités de champ sur des IRM FLAIR avec tumeur.*

IV Problématique

Nous avons vu que l'IRM est une méthode d'imagerie très utilisée pour le diagnostic et le suivi des pathologies cérébrales comme la sclérose en plaques ou les gliomes. En effet, dans le cas de sclérose en plaques, le critère de diagnostic se base sur la présence de lésions afin d'évaluer leur dissémination spatiale et temporelle. La charge lésionnelle et la position des lésions sont également des facteurs importants dans le suivi de la maladie et la réponse aux traitements mais également pour l'étude de la physiopathologie de cette maladie [Gourraud *et al.*, 2013]. Dans le cas des gliomes, il est nécessaire de déterminer précisément la zone tumorale pour pouvoir procéder à une exérèse ou de la radiothérapie. L'IRM donne également des informations sur le grade de la tumeur et permet d'évaluer la réponse aux traitements que ce soit la chirurgie, la radiothérapie ou encore la chimiothérapie. La segmentation des lésions de sclérose en plaques et des gliomes est donc une tâche importante dans le processus de diagnostic, le suivi et l'étude de la maladie.

Des méthodes d'analyse d'images peuvent remplir cette tâche de manière automatique. Les méthodes d'apprentissage profond sont les plus utilisées car elles présentent souvent des performances remarquables. Parmi elles, on retrouve les méthodes supervisées nécessitant une base de données avec des masques de segmentation annotés manuellement par des experts. Cette tâche étant longue et fastidieuse, les bases de données sont souvent petites ou inexistantes. Or les méthodes d'apprentissage profond nécessitent beaucoup de données pour apprendre mais aussi pour assurer une bonne robustesse à des données extérieures. Les méthodes non ou faiblement supervisées sont une alternative intéressante car elles ne nécessitent pas de segmentation manuelle. Au cours de cette thèse, nous avons développé des méthodes de segmentation de pathologies (comme les lésions de sclérose en plaques ou les gliomes) entraînées de manière faiblement supervisée.

Une autre limite des méthodes d'apprentissage est leur manque d'interprétabilité. Ces dernières sont souvent décrites comme des "boîtes noires" dont la décision est difficilement explicable. Or, dans un contexte médical, il est primordial que la décision des réseaux ne soit pas seulement correcte mais qu'elle soit basée sur des éléments pertinents. Pour l'imagerie, la décision des réseaux doit tenir compte des signes radiologiques de la pathologie présents dans l'image et ne pas se baser sur des biais quelconques. Or l'utilisation de l'IRM et de données multicentriques peut considérablement augmenter le nombre de biais utilisables par les réseaux de neurones. En effet, l'IRM n'est pas une imagerie quantitative : les contrastes y sont relatifs. Ce n'est par exemple pas le cas pour d'autre méthode d'imagerie comme la tomodensitométrie pour laquelle l'échelle des intensités peut être reliée à une réalité physique à travers le système d'unité de Hounsfield. En réalisant deux IRM avec des scanners

différents pour un même patient, les images seront différentes. En utilisant et en agrégeant des données provenant de plusieurs centres et donc de machines différentes (différentes marques, intensités de champ, protocoles et paramètres d'acquisition comme le temps d'écho ou de répétition, etc), la signature de l'IRM avec laquelle les images ont été acquises peut être utilisée par les méthodes automatiques pour la prise de décision. C'est pourquoi au cours de cette thèse, nous avons proposé des réseaux explicables et interprétables avec une décision basée sur les signes radiologiques présents dans les images. Dans ce manuscrit, nous considérerons que la méthode est **explicable** s'il est possible d'expliquer à partir de quels éléments la décision a été prise. Une méthode sera dite **interprétable** si ces éléments sont pertinents connaissant les aprioris médicaux.

Les deux axes traités dans cette thèse, la segmentation faiblement supervisée et l'interprétabilité, se rejoignent sur certains aspects. En effet, comme nous le verrons, l'identification des structures radiologiques utilisées par un réseau de neurones pour prendre sa décision conduira à leur détection, voire à leur segmentation, de manière faiblement supervisée.

Conclusion

Dans ce chapitre, nous avons présenté le contexte médical autour des pathologies cérébrales que sont la sclérose en plaques et les gliomes mais également l'utilisation de l'imagerie par résonance magnétique pour leur diagnostic. Nous avons également présenter les thématiques abordées au cours de cette thèse et qui seront décrites avec plus de détails dans ce manuscrit à savoir la segmentation faiblement supervisée de ces pathologies à partir des IRM cérébrales mais aussi l'amélioration de l'interprétabilité des réseaux de neurones pour la classification de ces pathologies.

CHAPITRE

2

ETAT DE L'ART

Introduction

Dans ce chapitre, nous proposons un état l'art sur les thématiques d'apprentissage profond abordées dans ce manuscrit. Pour cela, nous ferons un rapide rappel sur les réseaux de neurones convolutifs et les mécanismes d'apprentissage. Puis nous aborderons la classification et la segmentation par apprentissage profond. Nous détaillerons notamment les méthodes actuelles de segmentation faiblement supervisée qui nous intéressent. Enfin, nous discuterons des méthodes de l'état de l'art pour l'explicabilité des réseaux qui nous permettront d'évaluer l'interprétabilité des modèles proposés par la suite.

I Réseaux de neurones convolutifs

Les réseaux de neurones artificiels ont été développés à partir des années 1960 avec le perceptron [Rosenblatt, 1958]. Ce modèle, inspiré des neurones biologiques, présente déjà les couches linéaires, les fonctions d'activation ainsi que l'optimisation de poids par apprentissage qui caractérisent les réseaux de neurones d'aujourd'hui. En 1969, [Minsky et Papert, 1969] soulèvent le problème des activations linéaires qui restreignent les tâches solvables aux problèmes linéaires, ce qui ralentit la recherche autour de ces modèles. Néanmoins, les travaux sur la rétropropagation du gradient pour l'apprentissage des réseaux [LeCun, 1985, Rumelhart *et al.*, 1986, Werbos, 1974, Parker, 1985] ainsi que le développement des GPU (*Graphics Processing Unit*) ont propulsé ces méthodes parmi les plus populaires et les plus compétitives. Aujourd'hui, les réseaux de neurones convolutifs [LeCun *et al.*, 1998] ont démontré leurs performances dans de nombreux domaines et notamment en imagerie médicale.

I.1 Composition d'un réseau de neurones à propagation avant

Ces réseaux sont généralement composés d'un enchaînement de couches linéaires suivies par une fonction d'activation. Des couches de normalisation peuvent être ajoutées afin de stabiliser l'apprentissage du réseau. Enfin, la dimension spatiale des couches peut être modulée à travers des sur- ou sous- échantillonnages. Chaque couche reçoit la sortie de la couche précédente comme entrée. L'entrée de la première couche et donc du réseau de neurones peut être un vecteur, une image, un volume, etc. L'information est extraite de cette entrée par le réseau pour résoudre la tâche voulue. La sortie, quant à elle, dépend de cette tâche : il peut s'agir d'un masque de segmentation, d'une probabilité de classification, etc.

I.1.1 Couches linéaires

Les couches linéaires sont les principales couches d'un réseau. Elles peuvent être modélisées par $y = Wx + b$ où y est la sortie de la couche, x est l'entrée, W et b sont respectivement les poids et les biais à optimiser lors de l'apprentissage. Elles permettent d'extraire, petit à petit, les caractéristiques de l'entrée du réseau pour résoudre la tâche d'entraînement. Il existe deux types de couches linéaires : les couches dites denses ou entièrement connectées et les couches convolutives.

Couche entièrement connectée. Ces couches, utilisées dans le perceptron, connectent chaque neurone d'entrée à tous ceux de la sortie. Les entrées sont vectorisées si nécessaires, par exemple dans le cas d'images, comme le montre la Figure 2.1.

Ainsi la matrice des poids W pour une couche de ce type sera de taille $n_i \times n_o$ où $n_i = |x|$ est le nombre de neurones en entrée et $n_o = |y|$ est le nombre en sortie. Le vecteur des biais b lui est de taille n_o. Chacun de ces poids et biais devra être optimisé lors de l'apprentissage. Ajouter ces couches dans un réseau augmente donc considérablement le

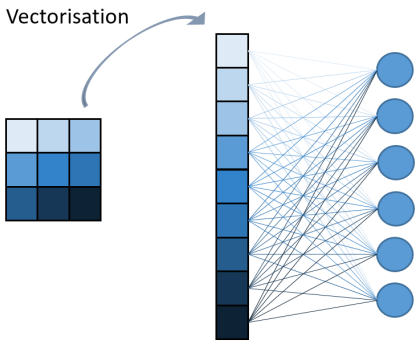


FIGURE 2.1 – *Couche entièrement connectée avec 9 neurones en entrée et 6 en sortie*

nombre de paramètres, c'est-à-dire de valeurs à optimiser. Ces couches denses sont surtout utilisées en fin de réseaux de classification pour obtenir la probabilité d'appartenir à chaque classe à partir des informations extraites par le réseau.

Couche convolutive. Pour les couches de convolution, un filtre, ou noyau de convolution, de taille donnée est partagé sur les dimensions spatiales de l'entrée. Ce filtre est déplacé sur ces dimensions spatiales en appliquant un produit de corrélation croisée avec l'entrée afin d'obtenir une sortie de dimension spatiale identique (Figure 2.2). Ces couches permettent d'extraire des caractéristiques de l'image de manière efficace et sont l'essentiel des réseaux convolutifs. La taille du filtre donne le champ récepteur de la couche, c'est-à-dire la taille de la zone de l'entrée vue par chaque neurone de sortie. Une dimension supplémentaire est ajoutée : les entrées et sorties peuvent avoir plusieurs cartes de caractéristiques (*feature maps*). Ainsi la matrice W des poids à optimiser est de taille $k^d \times f_o \times f_i$ où k est la taille du noyau, d la dimension du noyau, f_i et f_o le nombre de cartes de caractéristiques respectivement en entrée et en sortie. Le vecteur de biais est quant à lui de taille f_o. Le champ récepteur peut être augmenté sans augmenter la taille du filtre, et donc le nombre de paramètres, en considérant des voisins plus éloignés pour l'opération de convolution : il s'agit de la dilatation. Enfin, des techniques de remplissage (*padding*) sont utilisées pour résoudre le problème des bords et conserver la taille de l'entrée en sortie. Pour cela, des zéros ou une copie de l'entrée peuvent être ajoutés sur les bords.

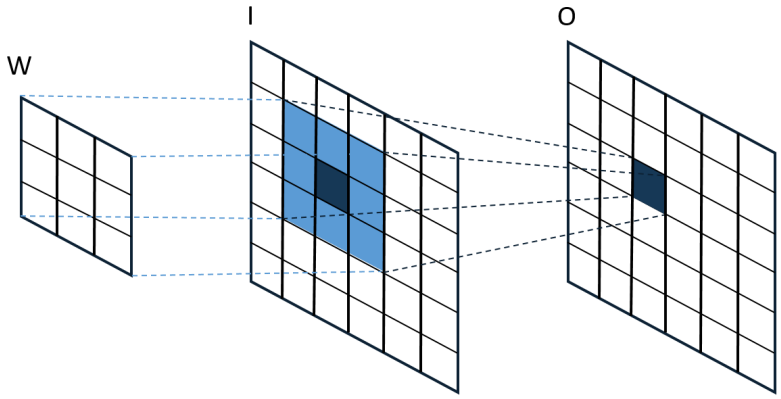
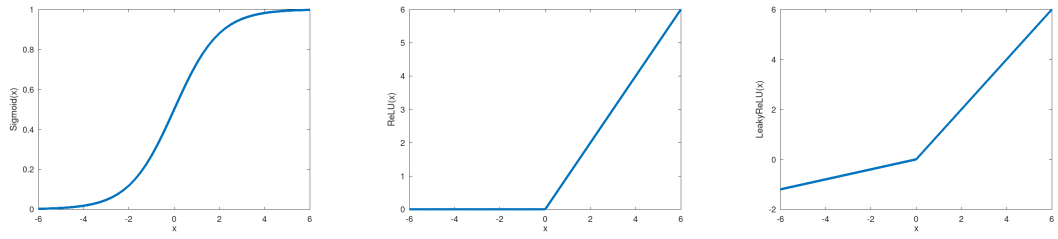


FIGURE 2.2 – *Convolution pour un pixel avec un noyau* 3×3 *et une carte de caractéristiques en sortie.* W *correspond au noyau,* I *à l'entrée et* O *à la sortie.*



(a) Sigmoïd (b) ReLU (c) Leaky ReLU

FIGURE 2.3 – *Exemples de fonctions d'activation utilisées pour les réseaux de neurones.*

.

I.1.2 Fonctions d'activation

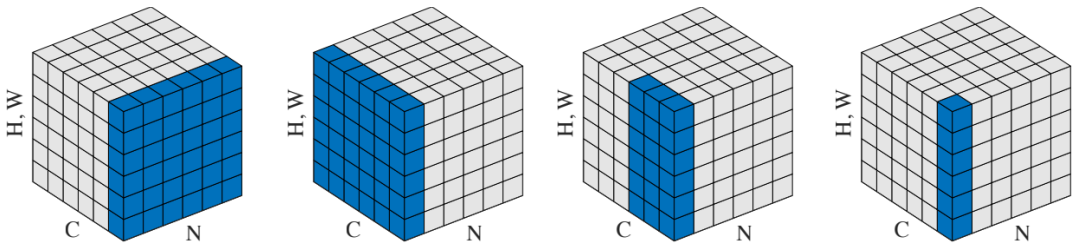
Les fonctions d'activation non-linéaires sont appliquées entre les différentes couches linéaires. Elles permettent d'adresser le problème soulevé par [Minsky et Papert, 1969] à savoir la résolution de problèmes non linéaires par des réseaux de neurones. En plus d'introduire de la non-linéarité, elles peuvent aussi servir à normaliser les données à travers le réseau. Les fonctions les plus utilisées sont la ReLU [Hahnloser *et al.*, 2000], la LeakyReLU, le softmax ou encore la sigmoïde (présentées Figure 2.3). Les avantages de la fonction ReLU, la plus utilisée, sont sa parcimonie (les sorties négatives d'un neurone sont mises à zéro) et le calcul rapide de sa dérivée. Elle réduit les problèmes d'évanescence du gradient rencontré avec les sigmoïdales utilisées auparavant. Néanmoins, du fait d'une dérivée nulle sur sa partie négative, les neurones ayant une sortie négative n'évoluent pas. Pour résoudre ce problème, la fonction LeakyReLU autorise une légère fuite avec une faible pente sur la partie négative. Le softmax et la sigmoïd sont couramment utilisés comme dernière activation pour des problèmes de classification (au niveau de l'objet ou du pixel pour la segmentation) respectivement sous forme d'encodage 1 parmi n (*one-hot*) ou de classification binaire.

I.1.3 Sous- et sur- échantillonnage

Pour augmenter le champ réceptif tout en limitant le nombre de paramètres et en gardant uniquement l'information pertinente, il est possible de procéder à un sous-échantillonage. Ce dernier peut être fait de différentes manières. Le plus souvent, on représente un ensemble de pixels voisins (la taille du voisinage étant fixée à la construction du réseau) par son maximum (*max pooling*) ou sa moyenne (*average pooling*). Il est également possible d'utiliser une couche de convolution où le filtre ne se déplace pas avec un pas (*stride*) de un, en ne considérant pas chaque neurone comme centre du filtre. Les dimensions spatiales de la sortie de la convolution sont alors divisées par ce pas, par rapport à l'entrée. L'opération inverse, le sur-échantillonage, permet de retrouver les dimensions spatiales originelles après un sous-échantillonage. Il peut être fait classiquement par interpolation ou en recopiant la valeur des neurones sur leur voisinage. Il existe également des convolutions transposées (*transposed convolution*). Pour ces dernières, des neurones nuls sont introduits entre les neurones d'origine afin d'augmenter artificiellement les dimensions spatiales. Puis, une convolution classique de pas unitaire est appliquée.

I.1.4 Couches de normalisation

La donnée d'entrée d'un réseau de neurones est souvent centrée et réduite (voir Section III.2). Afin de maintenir le caractère centré-réduit pour les différentes cartes de caractéristiques du réseau, des couches de normalisation peuvent y être ajoutées. Pour cela, la moyenne et l'écart-type d'un ensemble de neurones sont calculés. Puis chaque neurone est



(a) *Batch normalization* (b) *Layer normalization* (c) *Group normalization* (d) *Instance normalization*

FIGURE 2.4 – *Ensemble considéré selon le type de normalisation. C correspond aux canaux, H et W aux dimensions spatiales et N à la dimension du lot. Images tirées de [Wu et He, 2018]*

.

remplacé par sa valeur à laquelle on a soustrait la moyenne et qu'on a divisé par l'écart-type. L'ensemble considéré est défini par rapport à la dimension choisie (Figure 6.10) : cela peut être celle des lots d'entrées (*batch normalization*) [Ioffe et Szegedy, 2015], des cartes de caractéristiques (*layer normalization*) [Ba *et al.*, 2016], d'une sous partie de cette carte (*group normalization*) [Wu et He, 2018] ou encore une seule carte (*instance normalization*). Ces couches de normalisation peuvent avoir des paramètres à optimiser si on ne souhaite pas une distribution centrée-réduite.

I.1.5 Abandon

L'abandon (*dropout*) [Hinton *et al.*, 2012] consiste à mettre aléatoirement et de manière temporaire (le temps de l'itération courante) à zéro une partie des neurones. Le tirage aléatoire est refait à chaque itération changeant les neurones inactifs. Ainsi le réseau de neurone apprend à correctement accomplir sa tâche avec une partie des neurones en moins. Cela apporte de la robustesse et évite notamment le sur-apprentissage (*over-fitting*).

I.2 Apprentissage

L'apprentissage d'un réseau est un processus itératif qui permet d'optimiser les paramètres (poids et biais) afin de minimiser une fonction de coût à partir d'un jeu de données. On divise souvent ce jeu de données en trois. Une partie est utilisée pour l'apprentissage et servira pour l'optimisation des paramètres. Une partie de validation permettra de vérifier si le modèle est robuste à des données en dehors de celles sur lequel il apprend mais également à ajuster les hyperparamètres, c'est-à-dire les paramètres non appris et qui sont définis par l'utilisateur. Enfin, le jeu de test permet d'évaluer de manière impartiale la méthode une fois le réseau entraîné.

Pour l'optimisation des poids, à chaque itération, deux étapes sont répétées : une première étape où l'information va vers l'avant (*forward*) et une deuxième où l'information retourne en arrière (*backward*). Notons $x \in \mathbb{R}^n$ un échantillon issu de la base d'apprentissage utilisé comme entrée du réseau et $\hat{y} \in \mathbb{R}^m$ la sortie du réseau. Cette prédiction doit être comparée au résultat voulu et connu, la vérité terrain y. Pour cela, il est nécessaire de définir une fonction de coût (*loss function*) qui mesura l'erreur entre la prédiction $\hat{y}$ et le résultat attendu y. L'objectif est de minimiser cette erreur $L(\hat{y}, y)$ en modifiant les paramètres du réseau.

Pour cela, la dérivée de la fonction de coût par rapport à ces paramètres est calculée. On souhaite alors se diriger dans le sens où $L(\hat{y}, y)$ diminue le plus c'est-à-dire dans le direction opposée au gradient. L'optimisation des paramètres se fait donc selon la descente de gradient :

$$w' = w - \alpha \frac{\partial L(\hat{y}, y)}{\partial w} \tag{2.1}$$

où w est un paramètre (poids ou biais) à optimiser, w' sa nouvelle valeur après cette itération et α est le taux d'apprentissage (*learning rate*) qui pondère le gradient pour la mise à jour des poids.

Cette optimisation peut être légèrement modifiée en fonction de l'optimiseur choisi [Kingma et Ba, 2014, Zeiler, 2012, Zaheer *et al.*, 2018] ou par la plannification de la vitesse d'apprentissage choisie [Smith, 2017].

Le calcul du gradient se fait depuis la fin du réseau vers l'entrée par rétropropagation. En effet, le gradient peut être aisément calculé pour tout paramètre comme la dérivée des fonctions composées (*chain rule*) depuis la couche de ce paramètre jusqu'à la sortie du réseau.

I.3 Initialisation des poids

Avant d'entraîner un réseau de neurones, ces paramètres optimisables sont initialisés aléatoirement. Le choix de cette initialisation est important pour une convergence du modèle lors de l'apprentissage. Deux problèmes doivent être évités : l'évanescence et l'explosion du gradient. Or une bonne initialisation contribue à les prévenir.

L'explosion du gradient signifie que le gradient de la fonction de coût par rapport aux paramètres croît exponentiellement au fur et à mesure de la rétroprogation du gradient. En effet, si le gradient est très supérieur à 1 sur la dernière couche, d'après la règle de composition de la rétropropagation, celui de la couche précédente serait plus élevé, etc. Or si ce gradient est trop élevé, c'est-à-dire que $\frac{\partial L(\hat{y},y)}{\partial w_l} \underset{l\to 0}{\longrightarrow} +\infty$ (où l est le numéro de couche) dans l'Equation 2.1 alors les poids optimisés w' seront très différents de ceux à l'itération précédente w. Ce grand écart peut conduire l'apprentissage du modèle à osciller autour du minimum de la fonction de coût sans l'atteindre. Dans ce cas, le modèle aura de mauvaises performances. Dans le pire des cas, w' peut être tellement grand qu'il n'est plus possible de le le représenter numériquement (valeur NaN) : on a alors un modèle qui diverge et qui est donc inutilisable.

A contrario, l'évanescence du gradient signifie que le gradient devient proche de zéro dans les premières couches dû à un gradient initial (en fin de réseau) trop petit. Lorsque ce phénomène se produit, c'est-à-dire que $\frac{\partial L(\hat{y},y)}{\partial w_l} \underset{l\to 0}{\longrightarrow} 0$, les poids optimisés seront très proches des poids initiaux et le modèle n'apprendra plus.

Dans l'état de l'art, l'initialisation des poids des réseaux profonds vise à préserver la variance à travers les couches en tirant des poids aléatoires de telle sorte que si l'entrée suit une distribution centrée-réduite, il en va de même pour la sortie, sous réserve de certaines hypothèses. La sortie d'une couche linéaire (entièrement connectée ou convolution) peut être décrite comme suit :

$$y^l = W^l x^l + b^l \tag{2.2}$$

où l est l'indice de la couche, x_l est la carte des caractéristiques d'entrée, W_l est la matrice des poids de la couche et b_l est le vecteur de biais.

Les initialisations de Xavier [Glorot et Bengio, 2010] et de Kaiming [He *et al.*, 2015] fixent le biais à 0. Si nous considérons un canal k de la sortie de la couche, l'équation précédente devient :

$$y_k^l = \sum_{j=1}^{n^l} w_{kj}^l x_j^l \tag{2.3}$$

Ainsi, la variance peut être définie comme suit :

$$V[y_k^l] = V\left[\sum_{j=1}^{n^l} w_{kj}^l x_j^l\right] \tag{2.4}$$

où n^l est la taille du support des poids w^l.

Les initialisations de Xavier et de Kaiming supposent que les poids et les entrées sont indépendants et identiquement distribués et qu'ils sont mutuellement indépendants. L'équation devient donc :

$$V[y_k^l] = \sum_{j=1}^{n^l} V[w_{kj}^l x_j^l] \tag{2.5}$$

$$= \sum_{j=1}^{n^l} V[w_{kj}^l] E[(x_j^l)^2] + V[x_j^l] E[w_{kj}^l]^2 \tag{2.6}$$

Les deux initialisations supposent que w^l a une moyenne nulle, ce qui permet de conserver les hypothèses précédentes :

$$V[y^l] = n^l V[W^l] E[(x^l)^2] \tag{2.7}$$

La disjonction entre Xavier et Kaiming est que Xavier considère que la moyenne des x^l est nulle et donc $E[(x_j^l)^2] = Var[x_j^l]$ alors que Kaiming s'est intéressé au cas où l'activation précédente est l'activation non linéaire ReLU. Si $x^l = ReLU(y^{l-1}) = max(0, y^{l-1})$ la moyenne n'est plus nulle. Ainsi, une bonne initialisation sera $V[W^l] = 1/n^l$ pour Xavier et $V[W^l] = 2/n^l$ pour Kaiming afin de conserver la même variance à travers les couches.

II Classification

La classification automatique consiste à prédire le label attribué à une entrée. Elle est très utilisée dans le domaine de l'imagerie médicale pour l'aide au diagnostic : pour déterminer si une personne est atteinte d'une pathologie, pour déterminer le grade ou le type de la pathologie, etc. Nous nous intéresserons ici à la classification supervisée, c'est-à-dire lorsque nous disposons du label de chaque entrée pour l'entraînement du réseau de neurones.

II.1 Architectures

De nombreuses architectures de réseaux de neurones existent dans l'état de l'art pour la classification. Pour la classification d'image, on peut, par exemple, citer ResNet [He *et al.*, 2016], GoogLeNet [Szegedy *et al.*, 2015], DenseNet [Huang *et al.*, 2017], etc. Des architectures plus simples composées uniquement de quelques couches de convolution réduisant la dimension spatiale de l'entrée peuvent également être utilisées. Dans cette partie, nous ne présenterons que les architectures utilisées par la suite.

PatchGAN. PatchGAN [Isola *et al.*, 2017] a été proposé comme discriminateur d'un réseau adversaire dont le but est de déterminer si une image est une image réelle ou générée par un autre réseau (voir Section III.2 pour plus de détails sur ces approches). Il sert donc pour une classification binaire, c'est-à-dire avec deux classes (dénommées par la suite classe 0 et classe 1). Pour l'architecture dite 70x70, il s'agit d'un réseau convolutif composé de quatre couches de convolution (64, 128, 256 et 512 canaux) avec un sous-échantillonage divisant par deux la dimension spatiale à chaque convolution. Chaque convolution est suivi d'une activation de type ReLU et d'une normalisation par *batch* sauf pour la première couche. Une dernière convolution est appliquée afin de réduire le nombre de canaux à 1 .

Ainsi en sortie de cette architecture, nous obtenons une carte de caractéristiques ayant le même nombre de dimensions spatiales que l'entrée où chacun des neurones à un champ récepteur sur une partie de l'image d'où l'appellation de "patch". La taille de ce patch est, avec les paramètres originaux, de 70x70. Une probabilité d'appartenance à la classe 1 est donc donnée pour chacun de ces patchs, la probabilité pour l'autre classe étant la différence par rapport à 1. Pour obtenir la probabilité globale de l'entrée, une moyenne est faite sur tous les logits.

ResNet. L'une des différences entre ResNet [He *et al.*, 2016] et PatchGAN est la sortie du réseau et la représentation des classes. En effet, dans cette architecture, les probabilités d'appartenance aux différentes classes sont sous forme d'encodage 1 parmi n. Un vecteur de taille N (où N est le nombre de classes) donne cette probabilité pour chaque classe.

Mais ResNet se différencie surtout par l'introduction de couches résiduelles (*residual layers*). Ces couches peuvent être définies comme :

$$y = x + F(x) \tag{2.8}$$

où y est la sortie de cette couche, x est entrée et F est la sortie d'un ensemble de couches classiques (convolution, fonction d'activation et normalisation).

Cette couche a été introduite pour pallier les difficultés d'apprentissage des réseaux trop profonds. En effet, contre intuitivement, lorsque la profondeur du réseau augmente trop, on observe que les performances stagnent ou même diminuent. Or, on pourrait s'attendre à ce qu'augmenter la profondeur et donc le nombre de paramètres augmente la capacité de représentation du réseau et permette de résoudre plus facilement la tâche. Néanmoins, nous avons vu dans la Section III.2 le problème de l'évanescence du gradient qui est accentué par une trop grande profondeur.

En ajoutant ces connexions résiduelles (*skip connection*), le réseau peut aisément passer outre la couche F si cette dernière n'apporte rien en terme de performance. Pour cela, l'optimisation des poids conduira F à être une approximation de la fonction nulle et y sera presque égale à x. Reproduire ce résultat sans la couche résiduelle reviendrait à approximer la fonction identité ($y = x$) avec des couches classiques. Or cela n'est pas aisé de faire converger les poids vers cette configuration à partir des initialisations classiques comme le montre le problème de dégradation de performances par l'ajout de couches. En effet, si cette convergence vers l'identité était simple, ajouter des couches ne devrait pas changer les performances car l'optimisation rendrait les couches ajoutées équivalentes à l'identité. De plus, les expériences de [He *et al.*, 2016] montrent que $F(x)$ est faible. Il semble que la sortie optimale d'une couche soit proche de son entrée. Il est donc plus facile d'apprendre cette petite variation $F(x)$ plutôt que directement $F(x) + x$ avec des couches classiques. La rétropropagation est facilitée puisque même si les gradients sont faibles dans les couches intermédiaires, l'ajout du gradient par rapport au signal d'entrée permet de propager le gradient directement à la couche précédente, évitant ainsi l'évanescence.

Les réseaux ResNet sont ainsi constitués d'un enchaînement de couches résiduelles. Plusieurs profondeurs ont été proposées : 18, 34, 50, 121 et 152. Les réseaux de profondeur 18 et 34 utilisent des couches résiduelles dites classiques alors que les autres utilisent des couches résiduelles en goulot (*bottleneck*) qui diminuent puis réaugmentent le nombre de canaux. Ces deux types de couches sont présentés Figure 2.5.

II.2 Fonctions de coût

La fonction de coût utilisée pour apprendre la tâche de classification doit pénaliser par une augmentation la valeur de la fonction de coût, une mauvaise classification d'une donnée à son label (sa classe) connu.

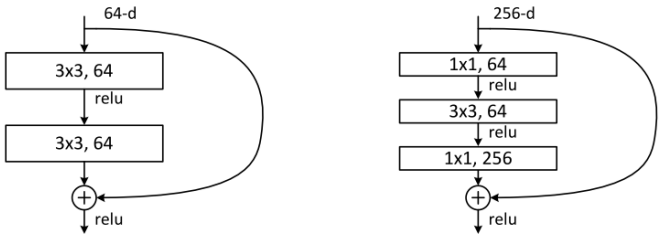


FIGURE 2.5 – *Exemple de couches résiduelles : classique (avec une entrée à 64 canaux) à gauche et en goulot à droite (avec une entrée à 256 canaux). La notation "AxB,C" correspond à une convolution de noyau AxB avec C canaux en sortie. Image tirée de [He et al., 2016].*

Classiquement, l'entropie croisée (*cross entropy*) est utilisée. Elle permet de comparer deux distributions de probabilités discrètes et la fonction de coût est définie comme :

$$L = -\sum_{l} \alpha_l y_l \log(\hat{y_l}) \tag{2.9}$$

où l est le label, $\hat{y_z}$ est la probabilité en sortie de réseau pour la classe z et y_z la sortie désirée.

Il est possible de pondérer l'importance de chaque classe dans la classification notamment lorsque certaines classes sont sous représentées dans la base d'entraînement. Pour cela, un poids α_l est ajouté pour chaque classe.

Dans le cas d'une classification multi-classe, la vraisemblance est utilisée en pratique :

$$L = -\alpha \log\left(\frac{\exp(\hat{y_l})}{\sum_{c=1}^{C} \exp(\hat{y_c})}\right) y_l \tag{2.10}$$

où l est le label, y_z est la probabilité en sortie de réseau pour la classe z et C est le nombre de classe.

II.3 Métriques

Différentes métriques peuvent être utilisées pour évaluer une tâche de classification.

Exactitude. L'exactitude (*accuracy*) est la métrique la plus utilisée. Elle peut être définie comme :

$$acc = \frac{\sum\limits_{c \in C} T_c}{\sum\limits_{c \in C} n_c} \tag{2.11}$$

où C est l'ensemble des classes, T_c est le nombre d'éléments bien classés de la classe c et n_c le nombre total d'éléments de la classe c.

Une version pondérée peut être utilisée en cas de classes déséquilibrées. Dans ce cas, une moyenne pondérée des exactitudes par classe est réalisée.

Spécificité et sensibilité. Dans le cas d'une classification binaire, la sensibilité et la spécificité peuvent être utilisées mais un compromis doit être trouvé entre ces deux métriques. Pour leur calcul, une classe est considérée comme "négative" et l'autre comme "positive". La spécificité (*specificity*) mesure la capacité à ne pas trop détecter de faux positifs et éviter les fausses alarmes. A l'inverse la sensibilité (*sensitivity*) mesure la capacité à détecter suffisamment de vrais positifs. Elles sont définies comme suit :

$$spe = \frac{TN}{N} \quad \text{et} \quad sen = \frac{TP}{P} \tag{2.12}$$

où TN est le nombre de vrais négatifs, TP le nombre de vrais positifs, N le nombre de négatifs et $P = TP + FN$ le nombre de positifs.

Ainsi en augmentant le seuil de décision, comme par exemple, le seuil sur la probabilité obtenue en sortie du réseau et permettant de définir si une entrée est classée positive, on produit un test plus spécifique mais moins sensible. L'inverse se produit en diminuant ce seuil.

AUROC - courbe ROC. Afin d'évaluer la méthode sans considération du seuil fixé, il est possible de tracer la courbe ROC (*receiver operating characteristic*), c'est-à-dire la sensibilité en fonction du taux de faux positifs ($1 - spe$) en faisant varier le seuil. L'aire sous cette courbe (AUROC) peut alors servir de métrique dans le cadre d'une classification binaire. La courbe peut aussi être utilisée pour trouver le seuil optimal du point de vue du compromis spécificité/sensibilité.

Néanmoins, l'AUROC n'est pas adaptée lorsque les classes sont déséquilibrées. En effet, imaginons un déséquilibre fort entre une classe négative nombreuse et une classe positive faiblement représentée : l'augmentation du taux de faux positifs dégrade rapidement la qualité de la classification. Pour le même nombre de vrais positifs et faux positifs, le taux de faux positifs (axe des abscisses de la ROC) sera $n = N/P$ (où N est le nombre de négatifs et P le nombre de positifs) fois plus petit que le taux de vrais positifs (axe des ordonnées).

Il est possible de ne considérer qu'une partie de la courbe pour limiter ce problème en considérant l'aire jusqu'à un certain taux de faux positifs p%. L'aire obtenue est alors souvent normalisée par rapport à l'aire maximale atteignable sous cette condition soit p%.

Ce problème étant récurrent en détection d'anomalies, davantage de détails seront donnés en Section III.1.2.

Précision et rappel. Plutôt que d'utiliser la classe négative (qui peut être déséquilibrée par rapport à la classe positive) pour quantifier les fausses alarmes, on peut utiliser la précision. Cette dernière peut être défini comme :

$$pre = \frac{TP}{PP} \tag{2.13}$$

où TP est le nombre de vrais positifs et $PP = TP + FP$ le nombre de prédictions pour la classe positive.

Cette métrique est souvent associée au rappel (*recall* (rec)) qui est un autre nom pour la sensibilité. De la même manière que la ROC, la courbe précision-rappel peut être tracée et son aire calculée (AUPRC).

Il est également possible de calculer une nouvelle métrique qui permet de tenir compte à la fois de la précision et du rappel. En effectuant la moyenne harmonique, on obtient le score F1 :

$$F_1 = 2\,\frac{pre \times rec}{pre + rec} = 2\,\frac{TP}{P + PP} \tag{2.14}$$

Comme ces métriques ne font pas appel à la classe négative, il est possible de généraliser ces dernières à la classification multi-classe. Dans ce cas, une moyenne des précisions et rappels pour chaque classe est effectuée.

III Segmentation

La segmentation d'images consiste à attribuer un label à chaque pixel de l'image. Elle est très utilisée dans le domaine médical [Liu *et al.*, 2021]. En effet, l'imagerie médicale est utilisée

au quotidien pour le diagnostic, le suivi du patient et de son traitement ou encore dans le cadre d'études cliniques basées images. La segmentation des structures pathologiques y représente une étape importante.

III.1 Segmentation supervisée

Une tâche de segmentation peut être apprise par un réseau de neurones de manière supervisée. Dans ce cas, on fournit lors de l'entraînement l'image et le masque de segmentation. Ce masque, de la même taille que l'image, attribue à chaque pixel un label numéroté de 0 à $N-1$ (N est le nombre de classes). Ce masque est souvent fait manuellement par des experts du domaine.

III.1.1 Architectures

L'architecture de référence pour la segmentation d'images médicales est U-Net [Ronneberger *et al.*, 2015, Du *et al.*, 2020]. Son architecture est présentée Figure 2.6.

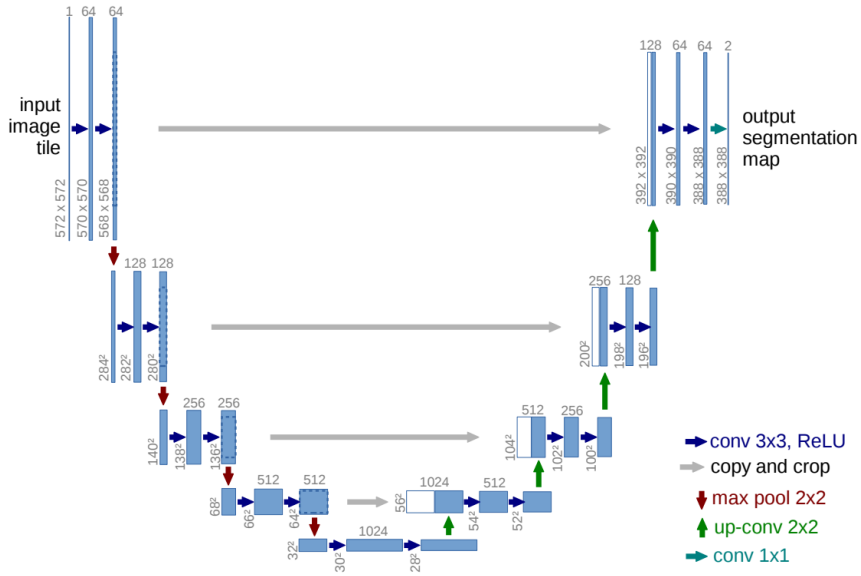


Figure 2.6 – *Exemple d'architecture type U-Net. Les rectangles bleus correspondent aux cartes de caractéristiques dont le nombre de canaux est indiqué au-dessus et les dimensions spatiales sur le côté. Les rectangles blancs représentent des cartes de caractéristiques copiées. Les flèches indiquent les différentes opérations. Image tirée de [Ronneberger et al., 2015].*

Elle est composée de deux parties : un encodeur et un décodeur. L'encodeur permet de réduire les dimensions spatiales de l'image tout en augmentant la dimension des caractéristiques. L'objectif est d'extraire les caractéristiques importantes de l'image et d'obtenir une représentation abstraite de l'image en bas du U-Net (*bottleneck*). Pour cela, des blocs de convolutions s'enchaînent et sont séparés par un sous-échantillonage réduisant la dimension spatiale par 2. Dans l'article original, le sous-échantillonage est réalisé par *max pooling* mais il est aussi possible d'utiliser une convolution avec un pas de 2. Chaque bloc est lui-même constitué de plusieurs couches de convolution suivies d'une activation type ReLU permettant l'introduction de non-linéarités dans le modèle. Généralement deux convolutions par bloc sont utilisées. Après chaque sous-échantillonnage, le nombre de canaux est multiplié par deux par les couches de convolution. La profondeur du U-Net correspond au nombre de répétitions de ces blocs convolutifs.

A partir de la représentation obtenue en bas du U-Net, une architecture inverse symétrique à l'encodeur est utilisée. Elle permet de remonter progressivement aux dimensions spatiales de l'image de départ et d'obtenir une carte de segmentation de cette taille. Pour cela, un enchaînement de sur-échantillonage et de blocs convolutifs est fait. Ainsi, à chaque étape, la dimension spatiale est multipliée par deux alors que le nombre de canaux est divisé par deux. Dans le papier original, des convolutions inverses sont utilisées pour le sur-échantillonage mais une interpolation est aussi classiquement utilisée. A la fin du processus, une convolution (avec un noyau de taille 1) est appliquée pour obtenir le bon nombre de canaux correspondant au nombre de classe à segmenter. Ainsi le $n^{\text{ième}}$ canal représentera la probabilité pour chaque pixel d'appartenir à la classe numéro n.

Cette structure encodeur-décodeur existait avant U-Net [Kalchbrenner et Blunsom, 2013]. L'ajout fondamental de U-Net est l'ajout de connexions résiduelles (*skip connection*) entre l'encodeur et le décodeur. Pour cela, la sortie de chaque sur-échantillonage est concaténée avec la carte de caractéristiques de l'encodeur ayant le même nombre de canaux. Ces connections permettent d'incorporer de l'information venant du début du réseau dans la segmentation finale. Cette segmentation est donc réalisée en ayant les caractéristiques fines obtenues lors de la phase d'encodage.

III.1.2 Métriques

Des métriques de classification peuvent également être utilisées au niveau du pixel. On peut ainsi utiliser le score F_1 appelé en segmentation le coefficient de Sørensen-Dice (ou Dice) [Sorensen, 1948]. Il s'agit probablement de la métrique la plus utilisée en segmentation. Il permet de mesurer le recouvrement entre la segmentation prédite et la segmentation utilisée comme vérité terrain. Il peut cependant présenter certains inconvénients. En effet, les grosses structures à segmenter sont prépondérantes pour son calcul. Dans le cas où de grosses et petites structures seraient à segmenter, une méthode segmentant mieux la grosse structure aura un meilleur Dice par rapport à une méthode segmentant un peu moins bien cette grosse strucure mais détectant les petites structures présentes (Figure 2.7 à gauche). De plus, sur des petites structures, le moindre pixel de différence entre deux segmentations aura un fort impact sur cette métrique (Figure 2.7 à droite). Le Dice n'est donc pas la métrique la plus appropriée si des structures de tailles très variées sont à segmenter. Il est également important de noter l'importance de la taille des structures pour cette métrique.

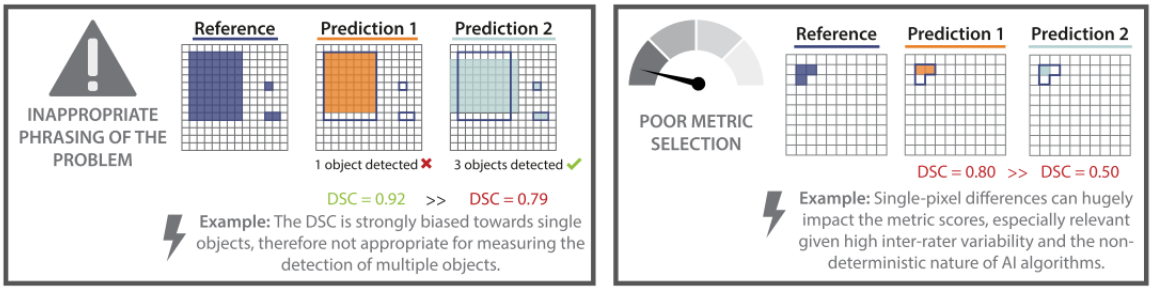


Figure 2.7 – *Problématique du Dice. Image tirée de [Maier-Hein et al., 2022].*

L'AUROC et l'AUPRC (vues Section en II.3) peuvent également être utilisées. Le problème du déséquilibre de classe est encore plus présent en segmentation. En effet, les objets à segmenter sont souvent petits par rapport à la taille de l'image. Par exemple, dans la Figure 2.8, on voit clairement qu'à partir de 30% de faux positifs, la segmentation n'est plus pertinente puisqu'il y a autant de faux positifs que de vrais positifs. Plus la zone à segmenter sera petite par rapport à la taille de l'image, plus le taux de faux positifs devra être réduit pour obtenir une segmentation correcte.

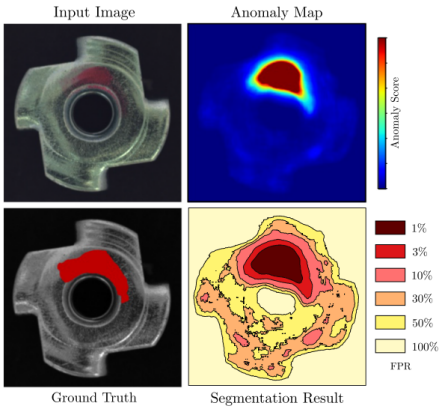


FIGURE 2.8 – *Influence du taux de faux positifs sur un exemple de segmentation. Image tirée de [Bergmann et al., 2021].*

Enfin, la distance de Hausdorff est également utilisée pour quantifier la qualité d'une segmentation [Huttenlocher *et al.*, 1993]. Elle permet de quantifier la dissimilarité entre deux ensembles et est définie comme :

$$HD = max\{\sup_{p \in P} d(p, M), \sup_{m \in M} d(m, P)\} \tag{2.15}$$

où P et M sont deux ensembles de points : par exemple, P est la segmentation prédite et M le masque de référence. Ici, la taille des structures n'a pas d'incidence sur le calcul de la métrique mais elle est très sensible aux structures non segmentées par la méthode et très éloignées des autres même si cette dernière est petite.

III.1.3 Fonctions de coût

La tâche de segmentation est équivalente à une tâche de classification pixel à pixel. On peut donc utiliser l'entropie croisée comme pour la classification (Section II.2) en attribuant un label à chaque pixel. Cette fonction de coût est locale à chaque pixel.

Il est également possible d'optimiser directement les métriques utilisées pour mesurer la qualité de la segmentation comme le Dice ou la distance de Hausdorff. Une version généralisée du Dice, dérivable et donc utilisable pour le calcul du gradient a été proposée par [Sudre *et al.*, 2017]. La fonction de coût peut être définie comme :

$$L_{Dice} = 1 - \frac{\sum_{l \in L} w_l \sum_n m_{ln} p_{ln}}{\sum_{l \in L} w_l \sum_{n \in N} m_{ln} + p_{ln}} \tag{2.16}$$

où L désigne l'ensemble des labels, N l'ensemble des pixels de l'image, w_l est la pondération attribuée à la classe l et m_{ln} et p_{ln} sont respectivement la valeur du masque vérité terrain et de la prédiction pour le label l au pixel n. Dans le papier original, $w_l = \frac{1}{(\sum_{n \in N} m_{ln})^2}$ pour prendre en compte le déséquilibre des classes.

Pour réduire la distance de Hausdorff, [Kervadec *et al.*, 2019, Karimi et Salcudean, 2019] ont proposé de l'intégrer à la fonction de coût. Pour cela, il est nécessaire de calculer la carte des distances par rapport à la vérité terrain. Cette carte de la même taille que l'image, vaut en chaque pixel la distance de ce pixel au masque. Un exemple de carte de distance est donné Figure 2.9

Une fois cette carte de distance calculée, on peut pondérer l'erreur de segmentation de chaque pixel par cette distance : une erreur de segmentation sur un pixel très éloigné de la zone à segmenter sera davantage pénalisée. La formulation est la suivante :

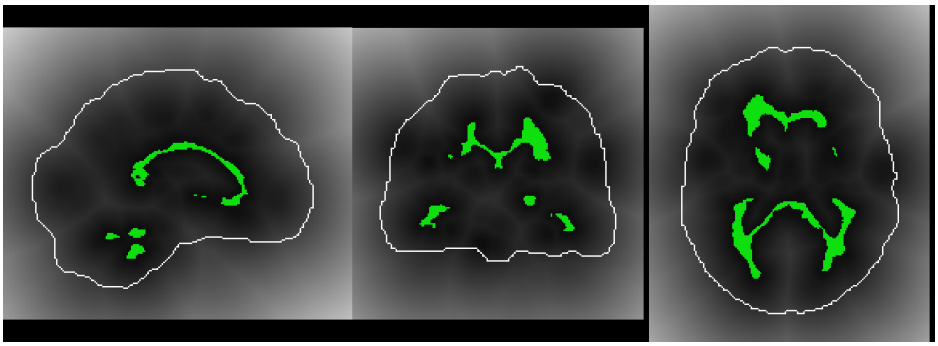


FIGURE 2.9 – *Exemple de carte de distance (vue sagittale, coronale et axiale). En vert, des lésions de sclérose en plaques (vérité terrain). Le cerveau est délimité par une ligne blanche. L'échelle des distances va du noir (proche) au blanc (éloigné).*

$$L_{HD} = \sum_{l \in L} \frac{1}{|N|} \sum_{n \in N} ((p_{ln} - m_{ln})^2 d_{ln}^{\alpha}) \tag{2.17}$$

où d est la carte de distance, α est un coefficient qui permet de moduler la pénalisation (fixé à 2 dans le papier) et les autres notations sont celles de l'équation précédente.

III.1.4 Limites

Les méthodes de segmentation supervisée sont les plus performantes mais présentent néanmoins quelques limites. Tout d'abord, il est difficile d'obtenir les annotations manuelles qui serviront pour l'entraînement des réseaux. En effet, ces annotations sont très coûteuses : elles mobilisent des experts médicaux pendant plusieurs heures pour chaque volume. Il existe donc peu de données annotées alors que les techniques d'apprentissage profond nécessitent de grandes bases d'apprentissage.

Une autre limite est sur la variabilité d'annotation intra et inter-expert. Il peut exister de grandes différences entre ces annotations dépendant du niveau de l'expert, de son niveau de fatigue, etc. En apprenant sur des annotations possiblement erronées, le réseau apprend donc les erreurs des experts. Des consensus sont parfois établis mais nécessitent davantage de temps et ne sont pas complètement véridiques non plus.

Prenons l'exemple de la base MSSEG d'IRM cérébrales de patients atteints de sclérose en plaques pour la segmentation de ces dernières. Nous avons optimisé un modèle supervisé de segmentation des lésions FLAIR sur cette base en prenant le consensus établi comme vérité terrain. Nous avons utilisé un U-Net appris avec les trois fonctions de coût présentées précédemment, de l'augmentation de données (déformations élastiques, variation de luminosité et retournement de l'image le long du plan sagittal) et avec l'ajout de module de compression et excitation (*squeeze and excitation* [Hu *et al.*, 2018]). Les résultats sont présentés dans le Tableau 2.1. On remarque que les performances de notre modèle sont proches du moins bon expert et que l'écart entre le meilleur et le moins bon expert est de plus de 10 points de Dice. Cette variabilité limite l'apprentissage des méthodes supervisées : elles peuvent apprendre des erreurs ou même des biais.

Les méthodes non-supervisées ou faiblement supervisées ne permettent pas d'obtenir d'aussi bonnes performances de segmentation mais sont nécessaires quand il n'est pas possible d'obtenir assez d'annotations.

III.2 Segmentation faiblement et non-supervisée

Nous nous intéresserons uniquement à la segmentation binaire de pathologies où l'objectif est de classer chaque pixel soit comme étant du tissu sain soit comme étant du tissu

Model	Dice
Gagnant du challenge MSSEG 2016 [McKinley *et al.*, 2016]	0.591 ± 0.212
Notre modèle	0.644 ± 0.187
Meilleure expert par rapport au consensus	0.782 ± 0.069
Pire expert par rapport au consensus	0.669 ± 0.146

Tableau 2.1 – *Dice sur la base de test de MSSEG 2016 en considérant le consensus.*

pathologique.

La segmentation faiblement supervisée peut être faite avec une annotation partielle : un point, une boîte, un trait sur l'objet à segmenter. Nous considérerons ici le cas où seul le label/la classe de l'image est disponible : image saine, image d'un patient atteint de sclérose en plaques, etc.

Les méthodes non-supervisées consistent à apprendre la représentation d'un sujet sain puis, à l'inférence, de détecter les anomalies des patients par rapport à cette référence. Pour cela, plusieurs architectures peuvent être utilisées : les auto-encodeurs (*autoencoder*, AE), les auto-encodeurs variationnels (*variational autoencoder*, VAE) et les réseaux antagonistes génératifs (*Generative adversarial network*, GAN [Goodfellow *et al.*, 2020]).

Auto-encodeurs. Les auto-encodeurs [Baur *et al.*, 2019] suivent une architecture d'encodeur-décodeur (Figure 2.10). La dimension de l'image d'entrée est réduite grâce à l'encodeur composé de couches convolutives et de sous-échantillonage. En résulte une représentation de faible dimension dans l'espace latent, c'est-à-dire dans le goulot de l'encodeur-décodeur. Le décodeur est composé de couches convolutives qui sur-échantillonnent l'espace latent afin d'obtenir une image en sortie de la même taille que l'image entrée. Dans le cas de l'auto-encodeur, l'objectif est de reconstruire l'image d'entrée. Pour cela, il est entraîné grâce à une fonction de coût de reconstruction :

$$L_{recons} = \|x - \hat{x}\|_k \tag{2.18}$$

où $\|.\|_k$ est la norme k (généralement 1 ou 2), x l'image d'entrée et $\hat{x}$ l'image reconstruite par le réseau.

Pour la détection d'anomalies, le réseau est appris uniquement sur une base d'images saines. Après entraînement, il doit donc être capable de reconstruire des images proches de celles utilisées pour l'entraînement. A contrario, il ne devrait pas être capable de reconstruire correctement des images qui sont trop différentes comme les images pathologiques. Ainsi, à l'inférence, les images pathologiques sont données en entrée du réseau. En sortie, la reconstruction doit être convenable dans les zones sans pathologie car elles sont proches des images saines utilisées pour l'entraînement. Les zones pathologiques, quant à elles, seront a priori mal reconstruites. On peut ainsi utiliser la carte d'erreur, ou résidu, c'est-à-dire la différence entre l'image d'entrée et l'image reconstruire par le réseau comme masque de segmentation de l'anomalie.

Auto-encodeurs variationnels. Les auto-encodeurs variationnels [Kingma et Welling, 2013, Zimmerer *et al.*, 2019] sont des auto-encodeurs pour lesquels on apprend une distribution représentative de l'image d'entrée dans l'espace latent (Figure 2.11). Dans un auto-encodeur classique, pour une entrée x, on obtient après encodage E, un vecteur de l'espace latent représentant cette image $z = E(x)$ et ce vecteur est donné en entrée d'un décodeur D pour retrouver une image qui doit être égale à l'image d'entrée dans le cas idéal $\hat{x} = D(z) \approx x$. Pour un auto-encodeur variationnel, l'encodeur permet de déterminer la distribution représentative $p(z|x)$. On impose que la loi suivie soit une loi normale dont il faut déterminer la

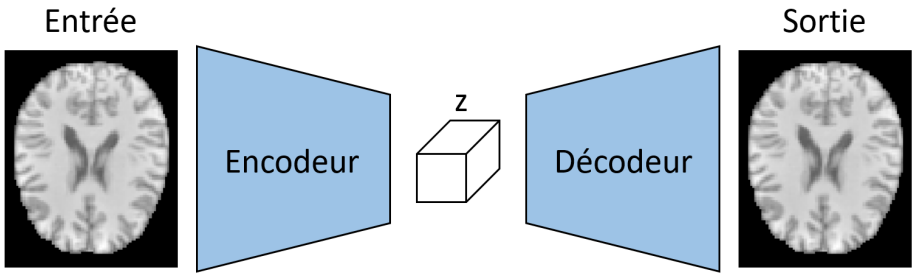


FIGURE 2.10 – *Architecture d'un auto-encodeur. L'entrée est réduite par l'encodeur pour obtenir un espace latent de faible dimension z. Le décodeur permet de récupérer les dimensions spatiales d'origine.*

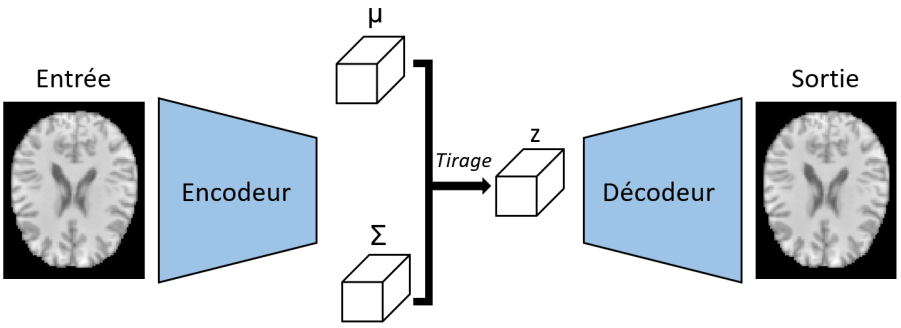


FIGURE 2.11 – *Architecture d'un auto-encodeur variationnel. L'entrée est réduite par l'encodeur pour obtenir les paramètres de la distribution μ et Σ. Une représentation latente z par tirage dans cette distribution. Le décodeur permet de récupérer les dimensions spatiales d'origine.*

moyenne μ et la matrice de covariance Σ (de dimension d) lors de l'encodage. Pour la phase de décodage, un vecteur suivant cette loi est choisi de manière aléatoire $z \sim p(z|x)$ et est passé au décodeur qui doit apprendre à correctement reconstruire l'image d'entrée à partir de ce vecteur $\hat{x} = D(z) \approx x$.

Un terme de régularisation est ajouté. Il a pour objectif de forcer la distribution de l'espace latent obtenu à être proche d'une loi $\mathcal{N}(0,1)$. Pour cela, une fonction de coût basée sur la divergence de Kullback-Leibler entre la loi estimée et $\mathcal{N}(0,1)$ est utilisée.

$$KL = E_{X\sim P}\left[ln\left(\frac{P(X)}{Q(X)}\right)\right] \tag{2.19}$$

où $P(x) = \frac{1}{\sqrt{(2\pi)^d}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}$ et $Q(x) = \frac{1}{\sqrt{(2\pi)^d}} e^{-\frac{1}{2}x^Tx}$

Cela permet d'avoir une continuité de cet espace. En effet, avec les auto-encodeurs classiques, rien ne garantit que toutes les valeurs de l'espace latent conduiront à une image semblable à celle du jeu d'entraînement. Ceci pose problème pour les tâches de génération de données. Avec le VAE, on contraint l'espace latent à suivre une distribution donnée, souvent $\mathcal{N}(0, I)$. En tirant un échantillon de l'espace latent suivant cette distribution, on ne sera donc pas "hors distribution", et on générera bien une image dans la distribution de la base d'apprentissage.

L'un des inconvénients des auto-encodeurs, qu'ils soient classiques ou variationnels, est qu'ils produisent souvent des images floues. Pour améliorer la reconstruction de l'image, une fonction de coût adversaire peut être ajoutée.

Réseaux génératifs adversaires Les GAN sont composés de deux architectures : un générateur et un discriminateur. L'objectif du générateur est de produire une image la plus réaliste possible à partir d'un bruit (souvent un bruit gaussien). On utilise pour cela une structure de décodeur permettant d'augmenter la dimension de l'entrée pour obtenir une image de taille voulue. Le discriminateur est un classifieur qui est entraîné à distinguer les images réelles, issues de la base d'entraînement, des images générées par le générateur. Il est entraîné avec une fonction de coût de classification classique comme l'entropie croisée binaire. On choisit généralement une petite architecture pour ce discriminateur afin de limiter ses capacités. En effet, le générateur est entraîné pour produire des images qui duperont le discriminateur, c'est-à-dire de telle sorte que les images générées soient classées comme images réelles. Il est donc nécessaire de bien équilibrer la capacité et l'entraînement des deux blocs adversaires qui sont optimisés en même temps. La fonction de coût utilisée pour entraîner le générateur traduit la classification de $D(G(z))$ (où D est le discriminateur, G le générateur et z le bruit d'entrée) comme étant de la classe "réelle" du discriminateur.

F-AnoGAN [Schlegl *et al.*, 2019] propose de réaliser de la détection d'anomalies en utilisant un GAN pour modéliser la normalité. Dans un premier temps, le GAN est entraîné de manière classique en générant un bruit aléatoire donné en entrée du générateur et en optimisant les poids du générateur et du discriminateur. Puis les poids de ces architectures sont fixés pour apprendre une architecture de type encodeur à produire un espace latent à partir des images saines. Pour cela, tout comme dans un auto-encodeur une fonction de coût de reconstruction est utilisée. Une fonction de coût perceptuelle est également ajoutée : avec le même objectif que la fonction de coût adversaire, qui est d'obtenir la même sortie du discriminateur pour les images réelles et fausses, on peut forcer cette égalité pour les cartes de caractéristiques du discriminateur (c'est-à-dire une sous-partie du discriminateur) [Salimans *et al.*, 2016]. La fonction de coût globale utilisée dans cette deuxième phase est donc :

$$L_{lat} = \frac{1}{n}\|x - G(E(x))\|_2 + \frac{1}{n_d}\|f(x) - f(G(E(x)))\|_2 \tag{2.20}$$

où n est le nombre de pixels dans l'image d'entrée x, $\|.\|$ est la norme Euclidienne, G est le générateur, E est l'encodeur ici appris et $f(.)$ représente une sous-partie du discriminateur dont la sortie est une carte de caractéristiques cachée de taille n_d du discriminateur.

De la même manière que cette fonction de coût, une moyenne entre les résidus de reconstruction de l'image et de reconstruction de la carte de caractéristiques du discriminateur est utilisée pour établir la carte de segmentation à l'inférence.

Des méthodes faiblement supervisées existent également. Elles nécessitent d'avoir des données pathologiques et saines. Elles reposent sur une classification "sain" contre "pathologique" puis sur l'explication de la prise de décision [Selvaraju *et al.*, 2017] puisque c'est la présence d'une pathologie qui devrait distinguer une image pathologique d'une image saine. Cela nécessite d'avoir des méthodes pour expliquer et interpréter la décision d'un réseau de neurones.

IV Explicabilité des réseaux de neurones

A cause du grand nombre de paramètres et de la non-linéarité introduite, il est difficile d'expliquer la décision d'un réseau de neurones. La décision peut, en effet, être juste sans avoir été prise à partir de caractéristiques pertinentes. Cela est très problématique dans certains domaines comme l'imagerie médicale où l'on souhaite que le réseau s'appuie sur des éléments radiologiques. Il existe, dans l'état de l'art, des méthodes pour expliquer la

décision d'un réseau de neurones et qui permettent de valider ou non une méthode pour son interprétabilité, c'est-à-dire la pertinence de sa prise de décision.

IV.1 Attributions

Les méthodes d'attributions permettent de définir les caractéristiques de l'image d'entrée du réseau les plus importantes pour la décision à partir d'un réseau entraîné. Il existe deux types d'attributions : celles basées sur le gradient et celles basées sur une perturbation.

IV.1.1 Attributions basées sur le gradient

Pour ces méthodes, pour une entrée donnée, le réseau est parcouru dans le sens avant (*forward*) puis à partir de la sortie, le réseau est parcouru en arrière (*backpropagation*) afin d'obtenir une carte de la même taille que l'entrée qui indique les zones qui ont contribué positivement ou négativement à la décision.

Dans le cas d'une classification, il est nécessaire de choisir la classe cible pour laquelle on veut une explication. On choisit alors le neurone de sortie de cette classe comme point de départ de la rétro-propagation. Dans le cas de PatchGAN ou des réseaux similaires pour lesquels la sortie pour une classe donnée n'est pas un seul logit mais une carte 2D/3D, une moyenne peut être faite entre les logits de sortie puis utilisée pour le calcul du gradient. Ici, $F(x)$ représentera la sortie du réseau F avec l'entrée x où on a sélectionné le point de départ selon le cas d'utilisation. $F(x)$ est donc bien le logit d'intérêt.

Dans la suite, nous présenterons les principales approches d'attributions basées sur le gradient.

Layer-wise Relevance Propagation (LRP). Pour remonter dans le réseau de neurones par rapport au logit choisi en sortie jusqu'à l'image d'entrée, il est nécessaire de suivre des règles. Dans le cas de LRP [Bach *et al.*, 2015], une pertinence (*relevance*) est donnée à chaque neurone du réseau de telle sorte que les sommes pondérées des pertinences de chaque couche soient égales. Prenons l'ensemble des neurones N et M de deux couches successives l et $l+1$, alors :

$$\forall n \in N, R_n = \sum_{m \in M} R_{n \leftarrow m} \tag{2.21}$$

$$\forall m \in M, R_m = \sum_{n \in N} R_{n \leftarrow m} \tag{2.22}$$

où R_x est la pertinence du neurone x et $R_{n \leftarrow m}$ est la proportion de R_m qui est redistribuée à R_n.

On a ainsi bien une conservation de la pertinence entre chaque couche puisque :

$$\sum_{n \in N} R_n = \sum_{n \in N} \sum_{m \in M} R_{n \leftarrow m} = \sum_{m \in M} \sum_{n \in N} R_{n \leftarrow m} = \sum_{m \in M} R_m \tag{2.23}$$

En supposant que l'activation d'un neurone vaut :

$$a_m = \sigma \left(\sum_{n \in N} a_n w_{nm} + b_m \right) \tag{2.24}$$

où a_x est l'activation du neurone x, w_{nm} est le poids reliant les neurones n et m et b_m le biais. Alors :

$$R_n = \sum_{m \in M} \left(\alpha \frac{a_m w_{nm}^+}{\sum\limits_{n \in N} a_m w_{nm}^+} - \beta \frac{a_m w_{nm}^-}{\sum\limits_{n \in N} a_m w_{nm}^-} \right) \tag{2.25}$$

où w^+_{nm} correspond au poids positif, w^-_{nm} au poids négatif et α et β sont des paramètres à régler pour pondérer l'impact des contributions positives et négatives tel que $\alpha - \beta = 1$ et $\beta > 0$. Une petite perturbation ϵ peut être ajoutée aux dénominateurs pour plus de stabilité. Cette méthode est appelée ϵ-LRP. En partant de la fin du réseau on peut ainsi remonter jusqu'à l'entrée, couche par couche.

Dans [Ancona *et al.*, 2017], il est prouvé que cette méthode est similaire à l'entrée multipliée par le gradient de la sortie par rapport à l'entrée. En effet, ils montrent que LRP revient à calculer les dérivées partielles successives où la dérivée des fonctions d'activations non-linéaires f' seraient remplacées par $g(x) = f(x)/x$. Ils montrent également que dans le cas des ReLU, les deux méthodes sont équivalentes.

Il est aisé d'accéder à ce gradient de manière efficace avec les libraires Python d'apprentissage profond. La formulation sous forme du calcul du gradient est donc plus efficace. Nous allons donc nous intéresser à ces méthodes.

Gradient. Le gradient de la sortie par rapport à l'entrée, ou saillance, est le moyen le plus simple d'évaluer la pertinence des pixels pour la sortie [Simonyan *et al.*, 2013]. Il consiste à dériver le logit de sortie choisi par chaque pixel de l'entrée donnant une carte de la même taille que l'entrée. La valeur de cette attribution au pixel i est :

$$G_i(x) = \frac{\partial F(x)}{\partial x_i} \tag{2.26}$$

où x est l'entrée que l'on veut expliquer et F est le réseau.

Intuitivement, le gradient de la sortie du réseau par rapport à l'image d'entrée indique quels pixels sont les plus critiques dans la décision du réseau : plus la dérivée par rapport à un pixel est importante, plus la modification de sa valeur changera la sortie du réseau.

Il est possible de multiplier (au sens de Hadamard) la carte obtenue en sortie par l'entrée. Cette méthode est connue sous le nom de EntréeXGradient (*InputXGradient*). La différence entre ces deux options est développée dans [Ancona *et al.*, 2017]. Lorsque la multiplication par l'entrée est faite, la méthode est dite globale alors qu'elle est considérée comme locale sans cette multiplication. Les attributions globales permettent d'identifier l'effet marginal sur la sortie de la présence d'une caractéristique dans l'entrée alors que les attributions locales visent à expliquer comment l'entrée doit être modifiée afin d'obtenir une variation souhaitée de la sortie.

Integrated Gradient. Integrated Gradient (IG) [Sundararajan *et al.*, 2017] est la moyenne des valeurs du gradient pour des entrées qui sont des interpolations linéaires entre l'image x dont on veut expliquer la décision et l'image x' dite de référence (*baseline*) pour laquelle la carte d'attributions sera par définition nulle. Cette référence doit être choisie par l'utilisateur comme "neutre" et le choix se porte souvent sur une image nulle (composée uniquement de zéros) de même taille que l'image d'entrée. La définition mathématique d'Integrated Gradient au pixel i est la suivante :

$$IG_i(x, x') = (x_i - x'_i) \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha(x - x'))}{\partial x_i} d\alpha \tag{2.27}$$

Sous cette forme, il s'agit d'une méthode d'attributions globale. Pour l'utiliser de manière locale, il suffit de ne pas multiplier l'intégrale par $(x_i - x'_i)$.

Il a été prouvé que IG possède plusieurs propriétés garanties : la sensibilité, l'invariance à l'implémentation, la complétude, la linéarité et la préservation de la symétrie. Si l'axiome de sensibilité est respecté alors, si pour un pixel i, la valeur de l'image et de la référence sont différentes ($x_i \neq x'_i$) et que les sorties du réseau pour l'image et pour la référence

sont différentes ($F(x) \neq F(x')$) alors la valeur de l'attribution pour ce pixel sera non nulle. IG est invariant à l'implémentation, c'est-à-dire que si deux réseaux F et G ont la même sortie pour toute même entrée, alors les attributions calculées avec IG seront identiques pour une même image pour ces deux réseaux. La complétude peut être définie comme : si F est différentiable presque partout alors $\sum_i IG_i(x, x') = F(x) - F(x')$. La définition de la linéarité est classiquement : si F est la combinaison linéaire de plusieurs réseaux, la carte d'attributions pour une entrée est la même combinaison linéaire des attributions obtenues avec ces réseaux. Enfin, la préservation de la symétrie implique que, si en échangeant deux pixels de l'image d'entrée, la sortie du réseau est la même alors pour une même référence, la valeur de la carte d'attributions pour ces pixels sera la même. En comparaison, G ne respecte pas l'axiome de sensibilité et de la complétude car il n'y a pas de référence.

Expected Gradient. Le résultat d'IG est fortement dépendant de la référence choisie et le choix de l'image nulle est discutable puisque cela contraint ces pixels à avoir des attributions nulles et donc à être considérés comme sans importance pour la décision dans l'explication générée. Dans [Sturmfels *et al.*, 2020] de nombreuses propositions de référence sont testées sans conclusion sur un meilleur choix ou un choix plus robuste .

Dans Expected Gradients (EG) [Erion *et al.*, 2021], ce problème est résolu en intégrant plusieurs références et notamment des images de la base de données :

$$EG_i(x) = \int_{x' \in X} IG_i(x, x')\, dx'. \tag{2.28}$$

où x est l'entrée, x' est la référence et X est la distribution des données.

Grad-Cam. GradCAM [Selvaraju *et al.*, 2017] se différencie des méthodes précédentes sur deux points. Tout d'abord, la carte d'attributions n'est pas un gradient : la moyenne du gradient est seulement utilisée comme pondération. Ensuite, ce n'est pas l'impact de l'entrée sur la sortie qui est visualisé mais celle d'une carte de caractéristiques intermédiaire. Ainsi, sa formulation est la suivante :

$$GC(x) = ReLU\left(\sum_{c \in C} A_c \frac{1}{|N|} \sum_{n \in N} \frac{\partial F(x)}{\partial A_{cn}}\right) \tag{2.29}$$

où C est l'ensemble des canaux de la carte de caractéristiques sélectionnée, F est le réseau, x est l'entrée, N est la dimension spatiale de la carte et A est la carte de caractéristiques (A_c est le canal c de A et A_{cn} est le canal c à l'indice spatial n.

Cette carte d'attributions a la dimension de la carte de caractéristiques sélectionnée. Pour obtenir une carte de mêmes dimensions que l'image et ainsi connaître les pixels de l'image d'entrée importants pour la décision, il est nécessaire de sur-échantillonner cette carte.

L'activation ReLU permet de récupérer uniquement les valeurs positives de l'attribution et donc les zones de l'image d'entrée qui contribuent positivement à la décision. Pour récupérer les contributions positives et négatives, il suffit de ne pas appliquer cette activation.

GradCam est ainsi une visualisation d'une carte de caractéristiques pour laquelle une moyenne pondérée des canaux a été faite selon leur importance pour la décision, cette importance étant traduite par le gradient. On visualise donc principalement l'information des cartes de caractéristiques. Il est préconisé d'utiliser la dernière couche avant les éventuelles couches totalement connectées pour calculer la carte d'attribution. En effet, les cartes de caractéristiques au début du réseau manqueront de sémantique car le champ récepteur est encore faible. Néanmoins, même si les dernières couches possèdent davantage d'information pertinente, la structure type encodeur utilisée en classification implique que la dimension spatiale des dernières couches sera beaucoup plus petite que celle de l'image nécessitant un

fort sur-échantillonage. La carte d'attributions sera donc moins précise qu'avec les méthodes basées sur le gradient par rapport à l'image.

IV.1.2 Attributions basées sur les perturbations

Les méthodes de perturbations modifient l'entrée et analyse l'impact de cette modification sur la sortie pour identifier les caractéristiques de l'entrée les plus importantes pour la décision.

La méthode de référence est celle des occlusions [Zeiler, 2012]. Ici, une partie de l'image, un patch, est remplacée par une valeur constante (généralement zéro). L'image modifiée est alors passée dans le réseau. La valeur de la carte d'attributions au niveau de la zone modifiée vaut alors la différence entre la sortie du réseau pour l'image originale et pour l'image modifiée. Ainsi, plus la différence, c'est-à-dire l'erreur, est grande, plus les pixels modifiés sont importants pour la décision. La position du patch à modifier varie pour parcourir l'image entière et avoir une carte d'attributions de la taille de l'image. Une illustration du fonctionnement est donnée en Figure 2.12.

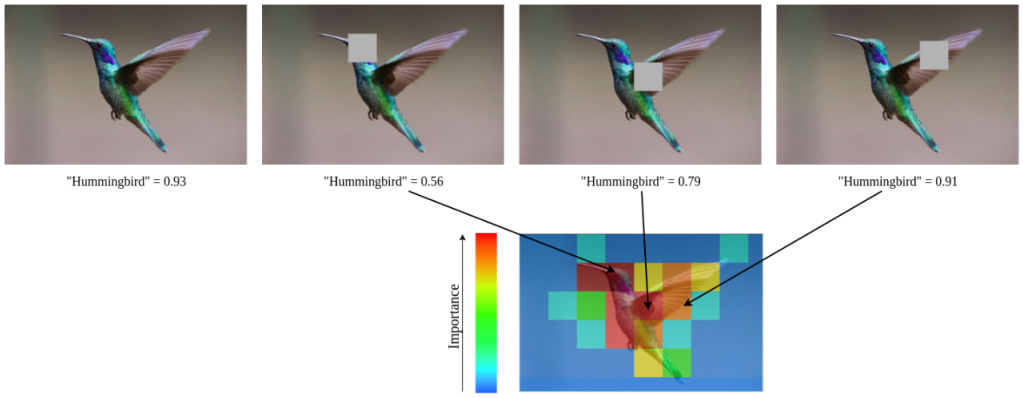


Figure 2.12 – *Principe des méthodes d'occlusion. Un patch gris est déplacé dans l'image et la différence de probabilité d'appartenir à la classe "Colibri" ("Hummingbird") pour l'image modifiée et l'image originale sert à construire la carte d'attribution. Image tirée de [Ras et al., 2022].*

Il est nécessaire de passer l'image modifiée dans le réseau à chaque déplacement du patch : cette méthode est donc très coûteuse notamment en fonction de la taille du patch et du pas de déplacement choisi. Un compromis doit donc être fait : pour un petit patch et un petit pas, la carte sera précise mais le temps et les ressources nécessaires pour produire la carte seront grands, à l'inverse un grand patch et un grand pas de déplacement donneront une carte peu précise avec de grandes zones de l'image ayant la même valeur d'attribution, pour un coût plus faible mais toujours largement supérieur aux méthodes de gradient. Le choix de la constante est également discutable au même titre que l'image référence d'Integrated Gradient.

Des variantes de cette méthode existent. On peut ainsi modifier l'image en bruitant ou floutant une région plutôt qu'en remplaçant ses valeurs par une constante [Fong et Vedaldi, 2017]. Il est également possible d'échanger la valeur de pixels entre eux [Fisher *et al.*, 2019] pour s'assurer que la valeur utilisée pour remplacer un pixel est dans la distribution vue par le réseau à l'entraînement.

IV.2 Génération d'exemples contrefactuels

Les méthodes de génération d'exemples contrefactuels (*counterfactual examples*) partent d'un réseau entraîné et d'une entrée à expliquer (par exemple une image). L'objectif est alors

de générer une entrée modifiée (une autre image) la plus proche de l'entrée originale qui modifie la sortie du réseau. En inspectant cette sortie et les différences avec l'image d'origine, on peut alors identifier les éléments de l'entrée qui sont décisifs. Dans [Guidotti, 2022], plusieurs propriétés sont évoquées pour qualifier une bonne méthode de génération d'exemples contrefactuels : la validité, le minimalisme, la similarité, la plausibilité, la discrimination, l'actionabilité, la causalité et enfin la diversité. La validité impose que la décision du réseau pour l'exemple contrefactuel soit différente de l'entrée d'origine. Le minimalisme veut que le moins d'éléments possible soit changé dans l'entrée pour générer l'exemple contrefactuel et la similarité, que la différence soit faible. La plausibilité correspond à un exemple proche de la distribution réelle des données. La discrimination repose sur une différence entre l'entrée originale et son exemple contrefactuel lisible par l'humain. L'actionnabilité et la causalité veulent reproduire des propriétés des données d'origine avec respectivement certaines caractéristiques non modifiables ou encore liées. La diversité intervient lorsque plusieurs exemples sont générés en imposant une grandes différences entre ces derniers.

La méthode la plus basique, proposée dans [Wachter *et al.*, 2017], est de trouver l'exemple contrefactuel x' de l'entrée x minimisant : $L_{cf} = \lambda(F(x') - y')^2 + d(x, x')$ où F est le réseau appris (les poids sont fixés), y' est la sortie désirée et d est une mesure de distance dans l'espace de x et x'. λ peut être fixé ou maximisé. Dans le cadre d'une classification, l'objectif peut être de passer d'une classe à une autre, y' est alors la classe désirée. L'optimisation peut être faite avec une descente de gradient si le modèle à expliquer est différentiable, comme c'est le cas pour les réseaux de neurones. Cette méthode se concentre sur les propriétés de validité, de minimalisme er de similarité. Pour améliorer la propriété de minimalisme, la distance entre x et x' peut être pondérée, permettant ainsi d'avoir une distance plus grande pour les caractéristiques fortement discriminantes comme proposé dans [Grath *et al.*, 2018]. Pour quantifier le caractère discriminant des caractéristiques, on peut, par exemple, utiliser une analyse de la variance (ANOVA).

Des modèles génératifs peuvent également être utilisés pour générer les exemples contrefactuels. Dans [Joshi *et al.*, 2019], un auto-encodeur variationnel déjà entraîné est utilisé pour encoder l'entrée pour laquelle on désire un exemple contrefactuel. Puis, l'espace latent est modifié jusqu'à ce que la sortie du décodeur donne un exemple contrefactuel sur le même principe que l'optimisation de [Wachter *et al.*, 2017]. En outre, pour augmenter l'actionabilité, certaines caractéristiques peuvent être fixées manuellement pour générer l'exemple contrefactuel. Dans [Zhao, 2020], un GAN est entraîné pour générer le résidu nécessaire au changement de l'entrée en exemple contrefactuel, c'est-à-dire que $x + G(x, y')$, où G est le générateur entraîné, serait classé selon la classe y'. Pour cela, de nombreuses fonctions de coût sont utilisées : des fonctions de coût adversaires sans et avec distinction des classes mises en jeux, des fonctions de coût modélisant l'optimisation présentée dans [Wachter *et al.*, 2017] pour assurer que le résidu est faible et permet de changer de classe par rapport à l'entrée et au classifieur entraîné dont on veut l'explication, et enfin une fonction de coût pour préserver la propriété de minimalisme à travers un cycle pour revenir à la classe d'origine. Dans [Charachon *et al.*, 2022], un GAN est également utilisé pour expliquer un classifieur d'images à travers la génération d'un exemple contrefactuel et d'un exemple dit stable de la même classe que l'entrée. L'explication visuelle du classifieur est alors la différence entre ces deux exemples. Le GAN est entraîné pour satisfaire des propriétés pour l'explication : la pertinence (les exemples contrefactuels et stables ne doivent différer que dans les zones de l'image pertinentes pour la décision du classifieur), la régularité (les processus de génération de l'exemple contrefactuel et stable doivent être comparables pour éviter des différences propres au processus génératif) et le réalisme (les images générées doivent être dans la distribution des images réelles). Cette méthode est notamment utilisée sur des images médicales. D'autres modèles génératifs comme les très récents modèles de diffusion profonds peuvent également être utilisés pour générer des exemples contrefactuels [Jeanneret *et al.*, 2022].

IV.3 Distillation

Les méthodes de distillation partent d'un modèle entraîné et d'une image pour laquelle on veut une explication de la décision. Pour cela, un modèle explicable est construit pour reproduire la décision du modèle à expliquer. Les deux méthodes les plus utilisées sont LIME et SHAP.

LIME. LIME (Local interpretable model-agnostic explanations) [Ribeiro *et al.*, 2016] propose de trouver un modèle explicable G reproduisant les résultats du modèle non explicable F localement, c'est-à-dire pour le voisinage d'une entrée donnée x. L'objectif est de construire le modèle G avec le moins de paramètres possible en minimisant $L(F, G, \Pi_x)$ qui mesure la capacité de G à reproduire les résultats de F dans le voisinage de x. Une pondération Π_x est ajoutée pour privilégier une bonne reproduction de G par rapport à F des données les plus proches de x.

Pour créer le voisinage de x, un système de perturbations est utilisé. Tout d'abord, il nécessaire d'extraire des représentations interprétables depuis l'image d'entrée x. Cela revient à extraire des zones de l'image (super-pixels) qui représentent sémantiquement la même chose. Ces zones peuvent être délimitées manuellement. Des méthodes automatiques pour extraire ces super-pixels existent également [Chen *et al.*, 2020]. Les représentations interprétables, notées x', sont donc des masques binaires permettant d'extraire une sous-partie de l'image correspondant à un super-pixel. A partir de ces représentations, des perturbations aléatoires z' sont générées. Ces perturbations sont la somme aléatoire de plusieurs éléments de x', modifiés en ne gardant qu'une partie aléatoire du masque (le reste étant mis à zéro). Les masques obtenus sont appliqués sur l'image originale pour obtenir les échantillons reconstruits z. Ces échantillons représentent donc une partie de l'image, le reste étant mis à zéro. Ces images sont alors passées en entrée du modèle non explicable F. On obtient pour chaque échantillon z, la sortie du réseau $F(z)$. Une mesure de similitude entre z et x est également introduite par une fonction $\Pi_x(z)$. Cela permet d'obtenir une base de données composées des perturbations aléatoires z' (des masques binaires) ayant pour labels $F(z)$. Le modèle G est entraîné sur cette base de données en minimisant $L(F, G, \Pi_x)$ tout en limitant son nombre de paramètres. A noter que cette base peut-être réduite en ne conservant que les perturbations aléatoires z' les plus significatives.

Une fois ce modèle trouvé et appris, étant explicable, il peut être utilisé pour produire une explication sous la forme de masque binaire représentant les zones importantes. Le modèle G est souvent linéaire $G(z') = W_g z'$ où W_g est la matrice des poids. La fonction L est la distance L2 entre $F(z)$ et $G(z')$ pondérée par $\Pi_x(z)$.

Cette méthode est coûteuse en temps et calculs. En outre, utiliser une méthode non explicable pour extraire les représentations interprétables peut être discutée et la délimitation manuelle par des experts pose également problème.

SHAP. Les valeurs de Shapley [Castro *et al.*, 2009] permettent de définir la pertinence d'un groupe de pixels (super-pixel) dans la décision d'un réseau. Elles sont définies comme :

$$\Phi_i = \sum_{S \subseteq A \setminus \{i\}} \frac{|S|!(|A| - |S| - 1)!}{|A|!} \left(F_{S \cup \{i\}}(x_{S \cup \{i\}} - F_S(x_S) \right) \tag{2.30}$$

où A est l'ensemble des super-pixels, $F_{S \cup \{i\}}$ est le modèle entraîné avec un sous-ensemble de super-pixels incluant le super-pixel i et F_S est le modèle entraîné avec un sous-ensemble de super-pixels n'incluant pas le super-pixel i.

La méthode SHAP (ou *Shapley Additive Explanations*) [Lundberg et Lee, 2017] propose de calculer les valeurs de Shapley en estimant le modèle linéaire :

$$g(z') = \Phi_0 + \sum_{i=1}^{M} \Phi_i z'_i \tag{2.31}$$

où M est le nombre de super-pixels.

Pour cela, ils montrent qu'il est possible d'utiliser LIME avec $L(F, G, \Pi_x)$ et Π_x adaptés. Cette méthode présente donc les mêmes inconvénients mais les expériences montrent que les explications fournies sont plus en accord avec les attentes humaines.

IV.4 Modèles intrinsèquement explicables

Dans les méthodes de distillations, l'objectif est de reproduire les résultats d'un modèle non explicable avec des modèles explicables de par leur construction. Ces modèles linéaires explicables sont souvent très simples et ne permettraient pas de résoudre la tâche directement. Il existe très peu de modèles profonds explicables ou partiellement explicables de manière intrinsèque, notamment pour traiter des images. Dans cette partie, nous allons présenter quelques réseaux de neurones qui sont, de par leur construction, partiellement explicables.

IV.4.1 Classification à partir de prototypes

Dans le cas de la classification à partir de prototypes, des prototypes sont choisis ou construits pour chaque classe comme représentatif de l'ensemble des données de cette classe. La classification est alors faite en fonction de la similitude entre la donnée d'entrée pour laquelle on veut prédire la classe et les différents prototypes, assignés à une classe. Les prototypes peuvent être dans l'espace des données d'entrée mais généralement on utilise plutôt une représentation de faible dimension (du type espace latent). Dans ce cas, la similitude est calculée entre le prototype et l'encodage de l'entrée dans le même espace que le prototype.

La classification à partir des prototypes est explicable. En effet, on calcule une mesure de similarité entre l'entrée et les différents prototypes (généralement avec une distance L2, dans l'espace de départ ou l'espace latent). Une couche linéaire permet alors d'obtenir les probabilités d'appartenance aux classes à partir de ces distances. Il est donc aisé d'obtenir les prototypes les plus prépondérants pour la classification. Dans le cas d'un prototype dans l'espace latent, il est nécessaire de le projeter dans l'espace des images pour une interprétation lisible par l'homme.

Prenons l'exemple de [Li *et al.*, 2018] : le réseau est composé d'un auto-encodeur et d'un classifieur de prototypes comme décrit précédemment. L'auto-encodeur permet de générer un espace latent, qui sera aussi celui des prototypes, et donc de passer de l'espace des entrées à celui des prototypes et vice-versa. Le modèle est ainsi classiquement entraîné avec une fonction de coût de classification entre la classe prédite en fin de réseau et la classe véridique et une fonction de coût de reconstruction entre l'entrée et la sortie de l'auto-encodeur. A cela s'ajoute deux régularisations des prototypes pour qu'ils soient représentatifs des données d'entrée :

$$R1(P, B) = \frac{1}{|P|} \sum_{p \in P} \min_{x \in B} \|p - E(x)\|_2^2 \tag{2.32}$$

$$R2(P, B) = \frac{1}{|P|} \sum_{x \in B} \min_{p \in P} \|p - E(x)\|_2^2 \tag{2.33}$$

où P est l'ensemble des prototypes, B est le lot d'entrée utilisé pour l'itération (*mini-batch*) et E l'encodeur.

La minimisation de R1 impose que chaque prototype soit proche d'au moins une représentation dans l'espace latent d'une image d'entraînement (du lot utilisé pour l'itération) alors que la minimisation de R2 impose que chaque représentation des données d'entrée (du lot utilisé pour l'itération) dans l'espace latent soit proche de l'un des prototypes.

D'autres régularisations peuvent être ajoutées comme des conditions d'orthogonalité pour éviter la redondance des prototypes dans [Rymarczyk *et al.*, 2022]. D'autres architectures peuvent également être utilisées pour construire l'espace latent comme des VAE [Gautam *et al.*, 2022].

La visualisation des prototypes appris peut se faire comme dans [Li *et al.*, 2018] grâce au décodeur ou encore, en prenant l'entrée de la base d'apprentissage dont la représentation dans l'espace latent est la plus proche de celle du prototype. Cette visualisation permet une explication partielle du réseau à l'inférence : la prédiction est liée à une proximité de l'entrée avec ces prototypes. Néanmoins, le nombre de prototypes est un paramètre difficile à choisir : un trop grand nombre rend difficile l'interprétation alors qu'un nombre trop faible peut ne pas suffire pour représenter convenablement les données. Dans le papier présenté, il est fixé à 15 pour une classification sur MNIST (10 classes) [Deng, 2012].

IV.4.2 Modèles neuronaux additifs

Les modèles neuronaux additifs (*Neural additive models* (NAM)) [Agarwal *et al.*, 2021] ont été introduits comme une méthode explicable pour les données tabulaires (c'est-à-dire des données sous la forme d'un vecteur 1D). Cette méthode consiste à décomposer le vecteur d'entrée en logits et d'utiliser chaque logit comme entrée d'un réseau de neurones. Les sorties de chaque réseau sont alors sommées :

$$G(x) = \beta + \sum_{i=1}^{n} f_i(x_i) \tag{2.34}$$

où le vecteur d'entrée est $x = (x_1, x_2, ..., x_n)$ et f_i est le réseau appris avec la composante i des vecteurs d'entrée.

Une fonction d'activation est alors appliquée à G pour obtenir la sortie désirée apprise de manière supervisée. Par exemple, pour une classification binaire, une sigmoïde peut être utilisée pour obtenir la probabilité d'appartenance à la classe 1 et le modèle appris avec une fonction de coût de type entropie croisée. A l'inférence, on peut alors aisément obtenir la contribution de chaque caractéristique de l'entrée dans la décision, même si les réseaux f_i ne sont pas explicables.

Une version pour la classification d'images a été proposée dans [Alvarez Melis et Jaakkola, 2018]. Dans cette méthode, deux structures de type encodeur sont utilisées avec l'image comme entrée. Le premier encodeur h donne un vecteur dit de "concepts". Le deuxième encodeur θ donne un vecteur de "pertinences". La sortie du réseau F est alors la somme pondérée de la multiplication terme à terme de ces deux vecteurs ($F(x) = \sum_i g_i \theta_i(x) h_i(x)$ avec x l'entrée et g le vecteur des pondérations). L'ensemble est entraîné avec trois fonctions de coût :

$$L(x) = L_c(F(x), y) + L_h(x) + L_\theta(F) \tag{2.35}$$

L'encodeur h est ainsi entraîné avec une fonction de reconstruction L_h en ajoutant un décodeur. Les pondérations g sont apprises grâce à une fonction de coût de classification supervisée L_c. Finalement, l'encodeur θ est entraîné avec une fonction de coût de linéarisation de l'ensemble visant à maintenir les valeurs de pertinences constantes, indépendamment de x et donc de son encodage $h(x)$. Cela est traduit par la minimisation de :

$$L_\theta(F(x)) = \|\nabla_x F(x) - \theta(x)^T J_x^h(x)\| \tag{2.36}$$

où J_x^h est le Jacobien de h par rapport à x et $\nabla_x F(x)$ est le gradient de $F(x)$ par rapport à x.

Les explications sont obtenues en regardant le produit des pertinences et des concepts. Néanmoins, ces explications ne sont pas données dans l'espace des images et donc difficilement interprétables car on ne connaît pas la correspondance entre l'image d'entrée et les concepts. Pour retrouver cette dimension, il est possible d'utiliser des prototypes pour h. Le vecteur de concepts est alors remplacé par le prototype le plus proche qui peut être décodé. Dans ce cas, ce pose les problématiques liées aux prototypes évoquées précédemment.

IV.4.3 Réseaux monotones

Une fonction multivariable à valeurs réelles $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est dite monotone si $\forall i \in \{1, .., n\}, x_i^0 > x_i^1 \Rightarrow f(x_0, ...x_i^0, ...x_n) > f(x_0, ...x_i^1, ...x_n)$. Un réseau de neurones peut être vu comme une fonction multivariable à valeurs vectorielles $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$ où chaque logit de la sortie est une fonction $f_j : \mathbb{R}^n \rightarrow \mathbb{R}$ avec $j \in \{1, ...m\}$. Un réseau de neurones est monotone si toutes les sorties sont des fonctions monotones au sens décrit précédemment. La décision du réseau peut être expliquée en raison du lien monotone entre les caractéristiques d'entrée et la décision.

La solution la plus simple pour rendre un réseau monotone est d'utiliser des couches linéaires (convolution ou totalement connectée) à poids positifs. En effet, pour une couche linéaire $y = Wx + b$ de poids W et de biais b, la dérivée de la sortie par rapport à l'entrée est $\frac{\partial y}{\partial x} = W$. Or si chaque couche est monotone, et que les fonctions d'activation sont croissantes, le réseau sera monotone par composition. Ainsi [Sill, 1997, Daniels et Velikova, 2010] utilisent respectivement des réseaux à poids positifs composés d'une et trois couches, des réseaux très peu profonds.

Cependant, ces réseaux sont limités. Notamment, [You *et al.*, 2017] soulèvent le problème des réseaux à poids positifs qui ne pourraient pas apprendre des tâches complexes et soulignent la faible capacité des réseaux proposés par [Sill, 1997, Daniels et Velikova, 2010]. Ils proposent donc de contraindre la monotonie seulement par rapport à une partie de chaque donnée d'entrée et utilisent une architecture composée de trois couches distinctes : des couches de calibration (*calibrator layers*), des couches linéaires de plongement (*linear embedding layers*) et des couches d'ensemble de treillis (*ensemble of lattices layer*). Les couches de calibration permettent de projeter les données depuis $\mathbb{R}$ vers $[0, 1]$. Pour cela, une transformation est apprise de manière discrète et une interpolation linéaire est faite pour les valeurs différentes des points appris. Cette transformation est contrainte à être croissante. Les couches linéaires de plongement sont du type $y = Wx$ où W est la matrice des poids positifs pour la partie des données concernée et sans contrainte pour le reste. La monotonie est ainsi vérifiée pour la partie des données voulue. Finalement, les couches d'ensemble de treillis sont des interpolations linéaires de plusieurs tables de correspondance (*look-up tables*) apprises. Ces treillis sont imposés monotones s'ils prennent en entrée des données issues de la partie sélectionnée pour la monotonie. En imposant les différentes monotonies, on s'assure ainsi que pour les données sélectionnées, le réseau est bien monotone. Une architecture globale est donnée en Figure 2.13. Cette architecture est donc imposée et seule une partie des caractéristiques de l'entrée seront explicables par la monotonie.

De la même manière, [Liu *et al.*, 2020] affirme que contraindre les poids à être positifs contraint trop le réseau qui n'est alors plus capable de résoudre des tâches complexes. Cela est d'autant plus vrai si uniquement des activations de type ReLU sont utilisées. En effet, cela crée un réseau convexe (au même sens que la monotonie pour les fonctions multivariables à valeurs vectorielles) : $\frac{\partial ReLU(Wx+b)}{\partial x} = W \cdot ReLU'(Wx + b)$ est croissante et la convexité est préservée par composition. Ils proposent donc un réseau monotone avec moins de contrainte. Pour cela, ils imposent la monotonie des couches par deux, c'est-dire que $\forall k \in \{1, ...L\}, f_{2k} \circ f_{2k-1}$ est monotone, où L, pair, est le nombre de couches. Cela est moins restrictif que d'imposer que toutes les couches soient monotones tout en préservant la monotonie globale du réseau. Le réseau est entraîné avec la fonction de coût

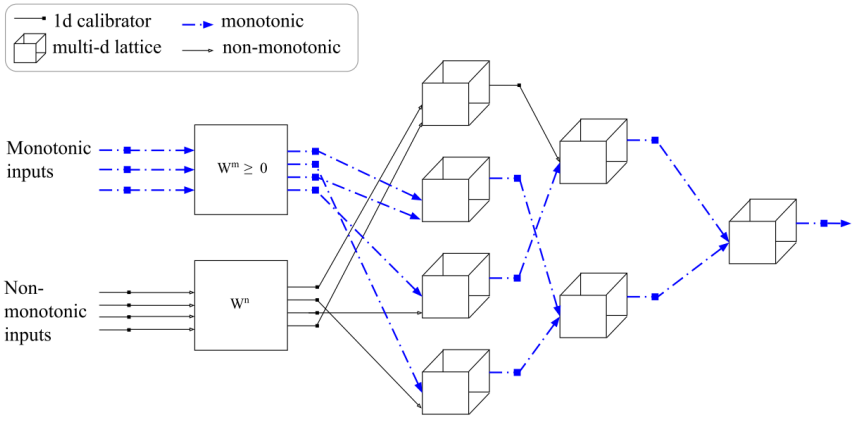


FIGURE 2.13 – *Exemple d'architecture à 9 couches de [You et al., 2017]*

correspondant à la tâche voulue ainsi qu'une fonction de coût de régularisation visant à rendre monotone chacun de ces couples de couches. Cette fonction somme les dérivées négatives (qui correspondent à des fonctions décroissantes) de la sortie par rapport à chaque entrée (du lot d'entrée pour cette itération). Le réseau est ainsi appris jusqu'à ce qu'il soit monotone en alternant les phases d'apprentissage avec les phases de vérification de la monotonie. Cette monotonie ne peut pas être vérifiée avec la fonction de coût proposée car il serait impossible de calculer les gradients pour tous les logits de toutes les entrées. Pour faire la vérification, [Liu *et al.*, 2020] utilisent donc plutôt la programmation linéaire mixte en nombres entiers (*Mixed-Integer Linear Programming* (MILP)) en transformant cette fonction en contrainte linéaire mixte d'entiers. Cette méthode impose donc une résolution d'un MILP à chaque itération et est donc fortement coûteuse en temps et ressources de calcul.

A l'inverse, [Sivaprasad *et al.*, 2021] affirment que les réseaux convexes monotones avec des poids positifs et des fonctions d'activation convexes ne sont pas trop restrictifs. En effet, ils montrent sur un MLP avec une sortie type vecteur d'encodage (un logit par classe) qu'il est possible d'apprendre une frontière de décision entre les classes non-convexes. Prenons l'exemple d'une classification à deux classes : g est la fonction convexe donnant la probabilité que l'entrée soit de la première classe $C1$ et f également convexe est celle de la deuxième classe. Alors pour une entrée x, si $g(x) - f(x) > 0$, la classe prédite est $C1$ et inversement. Cela est classiquement fait en prenant la classe à l'activation la plus forte (*argmax*). Or $g - f$ n'est pas convexe sauf si $g'' - f'' \geqslant 0$. La décision finale est donc non-convexe (Figure 2.14).

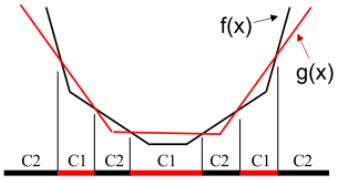


FIGURE 2.14 – *Exemple 1D de deux fonctions convexes g et f dont le argmax (classes $C1$ ou $C2$ en abcisse n'est pas convexe. Image tirée de [Sivaprasad et al., 2021]*

.

En utilisant cette formulation et des activations (toujours convexes) ELU, ils atteignent des performances de l'état de l'art pour la classification sur plusieurs bases de données comme CIFAR ou MNIST.

Dans [Runje et Shankaranarayana, 2022], la question de la convexité dans le cas d'un

MLP à poids positifs est résolue en utilisant des fonctions d'activation convexes sur une partie de la sortie d'une couche et des fonctions d'activation concaves sur la partie restante. En partant d'une fonction d'activation convexe a, ils construisent deux autres fonctions d'activation : $\hat{a} = -a(-x)$ qui est concave et $\tilde{a}$ qui est bornée et qui vaut $a(x+1) - a(1)$ si $x < 0$ et $a(x-1) - a(1)$ sinon. Ainsi à chaque couche, un tiers des sorties de la couche linéaire sont suivies de l'activation a, un tiers de l'activation $\hat{a}$ et le dernier tiers de $\tilde{a}$. Le réseau reste ainsi monotone tout en n'étant pas convexe et donc moins contraint. En outre, ils proposent de ne forcer les poids positifs qu'à partir de la deuxième couche. Cette proposition est également utilisée sur un MLP dans [Nguyen *et al.*, 2023] dans un contexte médical. Ainsi ce n'est plus l'entrée qui peut être interprétée grâce à la monotonie mais la dernière couche avant la partie monotone du réseau. Cette couche est ainsi appelée "caractéristiques interprétables" (*interpretable features*). Cela permet d'utiliser les réseaux monotones même s'il n'y a pas de relation monotone existante entre les données d'entrée et la sortie désirée. Les couches non monotones ne peuvent cependant pas être expliquées.

Conclusion

Dans ce chapitre, nous avons présenté les différentes architectures, métriques et fonctions de coût qui seront utilisées par la suite que ce soit pour des tâches de classification ou de segmentation par apprentissage profond. Nous nous intéresserons tout particulièrement aux méthodes de segmentation faiblement ou non supervisées qui nous serviront de point de comparaison par rapport aux méthodes développées. Nous avons également décrit les principales méthodes de l'état de l'art pour l'explicabilité des réseaux de neurones. Parmi celles-ci, trois d'entre elles seront exploitées dans nos travaux : les méthodes d'attributions basées sur le gradient qui permettent d'expliquer la décision d'un réseau de manière efficace, l'explication par génération d'exemples contrefactuels et les réseaux monotones qui présentent des qualités intrinsèques d'explicabilité mais qui ne sont pas encore totalement exploités aujourd'hui du fait que seuls des réseaux monotones peu profonds ont été proposés.

CHAPITRE

3

RÉSEAU ADVERSAIRE GÉNÉRATIF POUR LA SEGMENTATION FAIBLEMENT SUPERVISÉE DES LÉSIONS DE SCLÉROSE EN PLAQUES

Introduction

Nous avons vu dans le chapitre précédent que les méthodes de l'état de l'art en détection d'anomalies n'utilisent que des bases d'images issues de sujets sains. C'est le cas des auto-encodeurs [Baur *et al.*, 2019], des auto-encodeurs variationnels [Kingma et Welling, 2013, Zimmerer *et al.*, 2019] et de f-AnoGAN [Schlegl *et al.*, 2019] pour lesquels aucune information sur la pathologie considérée n'est fournie au réseau lors de l'apprentissage. Or des bases pathologiques sont souvent accessibles et ajouter cette information pourrait permettre au réseau de concentrer sa décision sur la pathologie. Il nous semble qu'ajouter des images pathologiques, sans annotation supplémentaire autre que le label "pathologique", est peu coûteux par rapport à l'apport d'information pertinente que cela peu introduire notamment pour les méthodes de segmentation. De telles bases de d'IRM de patients, sans autre forme d'annotation, existent largement car les examens radiologiques sont majoritairement réalisés sur des patients par définition "malades". Dans [Schlegl *et al.*, 2019], l'architecture f-AnoGAN dépasse les performances des autres méthodes de l'état de l'art (AE et VAE) pour la détection d'anomalies sur des images de rétines acquises par tomographie par cohérence optique dans le domaine spectral (SD-OCT). De manière générale, les réseaux du type adversaire génératif (GAN) ont démontré leurs performances dans de nombreux domaines [Gui *et al.*, 2021]. Dans ce chapitre, nous proposons donc d'utiliser un réseau adversaire génératif pour segmenter les lésions de sclérose en plaques de manière faiblement supervisée en utilisant à la fois une base de données avec des images de sujets sains et une base d'images de patients atteints de sclérose en plaques. Nous aurons donc accès à la classe de l'image mais à aucune segmentation manuelle.

Contributions

Les contributions présentées dans ce chapitre sont :

- L'utilisation d'un réseau adversaire génératif pour la segmentation faiblement supervisée des lésions de sclérose en plaques à partir d'IRM T1.
- La génération d'images simulant une atrophie et leur intégration dans le processus d'apprentissage du GAN pour diminuer l'influence de l'atrophie pour différencier les images saines des images pathologiques.
- Une analyse des différences entre la base saine et la base pathologique dont se sert le GAN.

I CycleGAN pour la segmentation des lésions de sclérose en plaques

I.1 CycleGAN

Nous proposons d'utiliser une architecture de type CycleGAN [Zhu *et al.*, 2017] permettant de passer d'une image dans un domaine à une image dans un autre domaine sans avoir besoin d'images appairées. Ici les deux domaines considérés sont celui des images RM saines et celui des images RM SEP. Cette architecture est composée :

- d'un générateur G_H permettant de passer d'une image SEP I_{MS} à une image $I_{MS \to H}$ dans le domaine "sain",
- d'un générateur G_{MS} permettant de passer d'une image saine I_H à une image $I_{H \to MS}$ dans le domaine SEP,
- d'un discriminateur D_H qui classifie les vraies images saines I_H issues de la base d'entraînement des images $I_{MS \to H}$ générées par le générateur G_H,

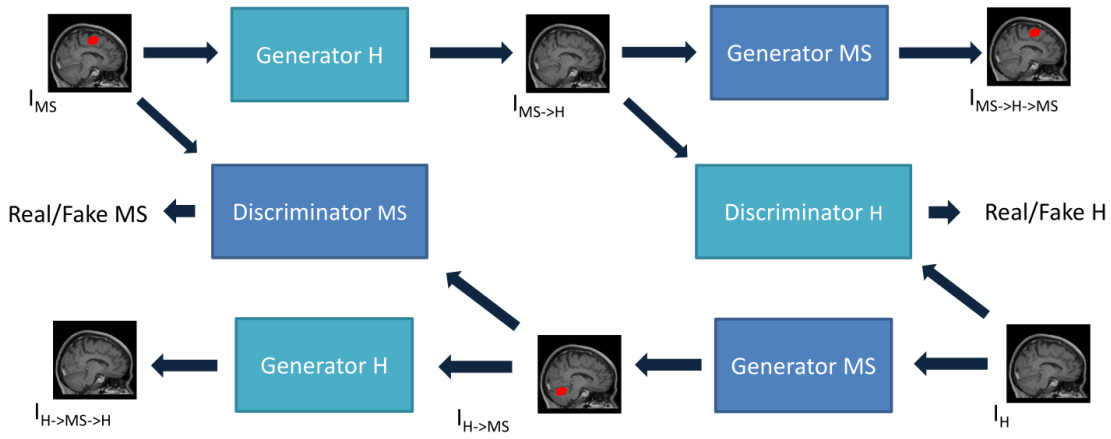


Figure 3.1 – *Schéma de la méthode proposée utilisant une architecture de CycleGAN. Elle est composée de deux générateurs : un générateur vers le domaine sain (Generator H) et un vers le domaine SEP (Generator MS). Elle est également composée de deux discriminateurs : un discriminateur différenciant les vraies des fausses images saines (Discriminator H) et un pour les images SEP (Discriminator MS). La zone rouge représente la pathologie.*

— d'un discriminateur D_{MS} qui classifie les vraies images SEP I_{MS} issues de la base d'entraînement des images $I_{H\to MS}$ générées par le générateur G_{MS}.

Dans le cas idéal, le générateur G_H devrait transformer une image SEP en image saine en retirant les lésions de sclérose en plaques. A l'inverse, le générateur G_{MS} devrait transformer une image saine en image SEP en lui ajoutant des lésions plausibles. Un schéma récapitulatif est donné Figure 3.1.

Comme tous les réseaux adversaires génératifs, les générateurs et discriminateurs sont entraînés en compétition : les discriminateurs doivent distinguer au mieux les images réelles des images générées alors que les générateurs doivent produire des images suffisamment réalistes pour tromper les discriminateurs. Cela se traduit par deux fonctions de coût utilisées pendant l'apprentissage pour chaque couple D_H/G_H et D_{MS}/G_{MS}, une de classification pour le discriminateur L_{class} et une dite adversaire pour le générateur L_{adv} :

$$L_{class}^{D_C} = BCE\left(D_C(I_C), 1\right) + BCE\left(D_c\left(G_C(I_{\overline{C}})\right), 0\right) \tag{3.1}$$

$$L_{adv}^{G_C} = BCE\left(D_C\left(G_C\left(I_{\overline{C}}\right)\right), 1\right) \tag{3.2}$$

où BCE est l'entropie croisée pour une classification binaire, $C \in \{H, MS\}$ est un domaine (c'est-à-dire une classe) et $\overline{C}$ l'autre domaine. La classe "réelle" du discriminateur correspond ici à la classe 1.

A cela s'ajoute des fonctions de coût spécifiques au CycleGAN. Tout d'abord, à la manière d'un auto-encodeur, passer d'une image à une image du même domaine ne devrait conduire à aucune modification de l'image. Ainsi, une fonction de coût dite d'identité est utilisée pour entraîner les deux générateurs. Elle se définit comme :

$$L_{id}^{G_C} = \|G_c(I_C) - I_C\|_1 \tag{3.3}$$

où $\|.\|_1$ mesure l'erreur de reconstruction, ici avec une norme $L1$.

Ensuite, pour conserver l'information du domaine d'origine lorsqu'on change de domaine, une fonction de coût dite cyclique est utilisée sur les générateurs et définie comme :

$$L_{cycl}^{G_C} = \|G_C\left(G_{\overline{C}}(I_C)\right) - I_C\|_1 \tag{3.4}$$

La fonction de coût utilisée pour entraîner les générateurs est une somme pondérée de ces trois fonctions :

$$L^{G_C} = L^{G_C}_{adv} + \alpha_{id} L^{G_C}_{id} + \alpha_{cycl} L^{G_C}_{cycl} \tag{3.5}$$

A l'inférence, le générateur G_H est utilisé sur les images SEP à segmenter. En passant dans le domaine sain, les lésions doivent être remplacées par du tissu sain. La différence entre l'image d'entrée I_{MS} et l'image de sortie de ce générateur $I_{MS \to H}$ permet donc de segmenter ces lésions.

I.2 Invariance à l'atrophie cérébrale

La méthode proposée repose sur une hypothèse : la différence entre l'image d'un sujet sain et l'image d'un patient atteint de sclérose en plaques doit être les lésions que nous voulons segmenter. Cette hypothèse est fragile, car en plus des biais potentiels des bases de données, d'autres critères radiologiques permettent de différencier une image SEP d'une image saine. Par exemple, l'atrophie cérébrale peut être importante chez les patients atteints de sclérose en plaques. Elle correspond à la mort des neurones et à la perte de volume cérébral. Puisque cette atrophie pourrait être prise en compte pour passer du domaine SEP au domaine sain, nous empêchant de segmenter uniquement les lésions à l'inférence, nous proposons de rendre le passage entre ces deux domaines invariant à l'atrophie.

I.2.1 Simulation d'une atrophie cérébrale dans les images

La première étape consiste à générer des images saines présentant de l'atrophie. L'atrophie correspond à une perte du volume cérébrale, c'est-à-dire du volume représenté par la substance blanche (WM) et la substance grise (GM). En conséquence, le liquide cérébrospinal (CSF) est plus représenté. Nous proposons donc de partir d'images saines et d'appliquer une transformation imposant une diminution des volumes de la substance blanche et de la substance grise tout en conservant la forme du cerveau.

Il est nécessaire d'obtenir la segmentation de l'image en ces différents types de tissus cérébraux : le liquide cérébropinal, la substance grise et la substance blanche. Cette segmentation peut être obtenue par exemple avec le logiciel FSL FAST [Zhang *et al.*, 2001]. Des détails sur le fonctionnement de ce logiciel sont donnés en Section I.1 du Chapitre 4.

Pour générer la transformation réduisant le volume des substances grise et blanche, nous nous sommes inspirés des méthodes de recalage déformable utilisant des B-splines [Sdika, 2008]. L'objectif est de trouver la transformation T tel que $I \circ T$ soit une version de l'image originale I du sujet après progression du processus d'atrophie. On cherche la transformation T sous la forme :

$$T(x) = x + \sum_i c_i \beta \left(\frac{x}{h} - i \right) \tag{3.6}$$

où x représente les coordonnées du voxel dans l'image d'origine, h est l'espacement entre les noeuds de B-splines, les $c_i \in \mathbb{R}^3$ (pour une image 3D) représentent les coefficients à optimiser, β est une B-spline cubique et $i \in \mathbb{Z}^3$. On impose que pour tout x, $det(T'(x)) > 0$ afin de pénaliser les transformations non inversibles.

Pour obtenir la déformation voulue, il est nécessaire d'imposer que cette transformation soit l'identité à l'extérieur du cerveau afin de préserver sa forme. Dans les substances grise et blanche, on impose un jacobien constant $det(T'(x)) = J_0 > 1$ afin que leur volume soit réduit. Aucune contrainte spécifique n'est imposée dans le CSF autre que $det(T'(x)) > 0$).

L'optimisation est faite avec l'algorithme L-BFGS [Liu et Nocedal, 1989]. Du fait d'une optimisation difficile, nous avons procédé par étape : une image est générée avec un J_0 proche de 1, puis à partir de cette image, une nouvelle image est générée avec le même J_0 proche de 1, etc. Cela correspondrait à une valeur de jacobien plus élevée depuis l'image d'origine pour obtenir l'image finale.

I.2.2 Utilisation d'une fonction de coût perceptuelle pour l'invariance

Ces images saines avec de l'atrophie peuvent être utilisées pour rendre invariant le cycle-GAN. La solution la plus simple est d'utiliser de l'augmentation de données en considérant ces images comme des images saines de la base d'entraînement.

Pour aller plus loin, nous proposons en plus de cela, d'utiliser une fonction de coût perceptuelle sur le discriminateur D_H. Cela consiste à imposer la même sortie, sur chaque couche du discriminateur, pour l'image originale et les images d'atrophie générées à partir de cette image. Concrètement, la fonction de coût peut s'exprimer comme :

$$L_{percep}^{D_H} = \sum_{k,\theta} \alpha_\theta \| D_H^\theta(I_H) - D_H^\theta(I_H^k) \|_1 \tag{3.7}$$

où $D_H^\theta(x)$ est la sortie de la couche numéro θ de D_H pour l'entrée x, I_H^k est l'image avec atrophie de niveau k générée à partir de l'image I_H et α_θ permet de pondérer l'invariance des différentes couches.

Avec cette fonction de coût, on impose que le discriminateur se comporte de manière indifférenciée pour tous les niveaux d'atrophie d'une même image et soit donc invariant à l'atrophie. La pénalisation sur les cartes de caractéristiques des différentes couches du réseau discriminateur permet de faciliter la convergence de cette contrainte.

II Protocole expérimental

II.1 Données

Nous avons utilisé cinq bases de données d'IRM T1 : les bases de données publiques de sujets sains IXI et HCP et les bases de patients atteints de sclérose en plaques OFSEP1, OFSEP2 et MSSEG. La répartition entre les jeux d'entraînement (N_{train}), de validation (N_{val}) et de test (N_{test}) est indiquée dans le tableau 3.1.

Tableau 3.1 – *Bases de données d'IRM T1. H fait référence à une base saine et MS à une base de patient atteints de sclérose en plaques. La dernière colonne précise si nous disposons de segmentations manuelles des lésions.*

Base	N_{train}	N_{val}	N_{test}	H/MS	Annotée
IXI	400	130	50	H	Non
HCP	800	200	113	H	Non
OFSEP1	383	97	30	MS	Non
OFSEP2	429	100	42	MS	Non
MSSEG	0	0	37	MS	Oui

Ces données ont été prétraitées selon le processus décrit dans la Section III.2 du Chapitre 1 à savoir un recalage sur l'atlas du MNI, une extraction du cerveau et une correction des inhomogénéités de champ. La taille des images est de $91 \times 109 \times 91$ avec des voxels de $2mm^3$.

Les deux bases de données de sujets sains ont été utilisées séparément. En effet, la base IXI est composée de sujets plutôt âgées (50 ± 17 ans) alors que la base HCP est composée de sujets jeunes (29 ± 4 ans). En vieillissant, il est normal que de l'atrophie et des lésions apparaissent dans le cerveau. On retrouve ces deux éléments dans la base IXI. Nous avons donc étudié l'influence du choix de la base saine pour l'entraînement du GAN. Dans le cas où IXI était utilisée comme base saine, seul OFSEP1 a été utilisée comme base pathologique afin d'avoir un équilibre du nombre d'images entre les deux classes. Dans le cas où HCP était utilisée, les deux bases OFSEP1 et OFSEP2 ont été utilisées comme bases pathologiques. Dans les deux cas, la base MSSEG a été utilisée pour l'évaluation des segmentations en

utilisant l'annotation manuelle des lésions FLAIR (seule disponible). A noter que toutes les lésions FLAIR ne sont pas visibles sur les IRM T1 et que les lésions visibles en T1 sont souvent plus petites. La base HCP a été utilisée pour générer cinq niveaux d'atrophie.

II.2 Implémentation

Les deux discriminateurs suivent une architecture PatchGAN 3D ($70 \times 70 \times 70$). Les générateurs sont des U-Net de profondeur 4. Chaque étage est composé de deux convolutions chacune suivie d'une activation ReLU. Le nombre de cartes de caractéristiques pour chaque étage est, dans l'ordre : 32, 64, 128, 256. Du *dropout* et une normalisation de type *Squeeze and Excitation* sont utilisés à chaque étage de l'encodeur. SGD a été utilisé comme optimiseur avec un *learning rate* de 10^{-4} pour les générateurs et 10^{-5} pour les discriminateurs. En outre, les paramètres des discriminateurs ne sont optimisés qu'une itération sur 10 pour éviter un déséquilibre entre les deux réseaux adversaires. Nous avons fixé les pondérations des différentes fonctions de coût tel que $\alpha_{id} = 10^{-1}$, $\alpha_{cycl} = 1$ et $\alpha_\theta \in \{0.1, 0.22, 0.47, 1\}$ respectivement pour chaque couche du discriminateur. Nous avons réalisé de l'augmentation de données lors de l'apprentissage : des déformations élastiques, des variations de luminosité et une inversion de l'image le long du plan sagittal.

Pour la génération d'atrophie, nous avons utilisé une grille de B-splines espacée de 2 voxels et avons fixé $J_0 = 1.07$.

II.3 Évaluation

Une comparaison visuelle des résultats a été faite entre le cycleGAN décrit en Section I.1 appris soit avec la base HCP (nommé cyGAN) soit avec la base IXI (nommée cyGAN-Old) comme base saine, le cycleGAN appris avec la base HCP où les images avec atrophie générées ont été utilisées comme base saine en plus des images originales (cyGAN+AD) et enfin cette même configuration où la fonction de coût perceptuelle a été ajoutée pour les images avec atrophie (cyGAN+Per). Le Dice, a été également été utilisé pour mesurer quantitativement la segmentation des lésions. Nous nous sommes comparés avec un auto-encodeur (AE) utilisant la même architecture que les générateurs (sans *skip connection*) et entraîné sur la base HCP.

Pour évaluer les modifications apportées par le générateur H lors du passage du domaine SEP au domaine sain, nous avons mesuré les volumes de la substance blanche et du CSF. Pour cela, la segmentation en trois types de tissus (CSF, substance grise et substance blanche) a été faite avec FSL FAST permettant ainsi le calcul des volumes du CSF et de la substance par rapport au volume global du cerveau.

III Résultats

III.1 Génération des images avec atrophie

Un exemple de l'atrophie générée sur une image est donnée en Figure 3.2. On constate que le volume du CSF a bien augmenté au niveau des ventricules. A noter que les derniers stades d'atrophie sont exagérés par rapport ce que l'on observe en SEP. On peut relever plusieurs problèmes de réalisme. Tout d'abord, on voit apparaître un motif dans les ventricules. Il s'agit d'une dilatation des plexus choroïdes qui sécrètent le liquide cérébrospinal. Or l'atrophie réelle n'induit pas d'augmentation de ces derniers. En outre, l'atrophie devrait également augmenter l'espace entre les sillons, ce qui n'est pas visible dans notre génération. Cela est dû à la segmentation initiale. En effet, les images utilisées ne présentent aucune atrophie et les sillons sont donc très fermés sans présence de CSF à la segmentation. Il n'y a donc pas de réduction de la substance blanche ou grise à ce niveau pour conserver la forme du cerveau.

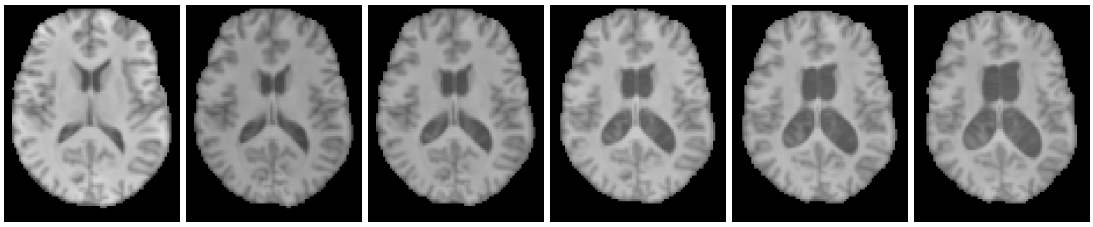


FIGURE 3.2 – *Exemple d'IRM T1 où une atrophie a été simulée. A gauche, l'image originale puis les cinq niveaux d'atrophie générés.*

III.2 Vérification du fonctionnement du cycleGAN

Les réseaux adversaires sont généralement difficiles à entraîner. Il peuvent facilement diverger ou entrer en mode "collapse" en ne générant que peu d'images différentes. La première étape est donc de vérifier que le comportement du cycleGAN est cohérent. Dans l'idéal, nous souhaiterions que le générateur H, qui va du domaine pathologique au domaine sain, efface les lésions alors que pour le générateur MS, qui va dans du domaine sain au domaine pathologique, nous souhaiterions qu'il crée des lésions. En outre, les deux fonctions de coût spécifiques au cycleGAN, L_{id} et L_{cycl}, imposent une reconstruction à l'identique avec l'image d'entrée pour le générateur vers le même domaine ($C \rightarrow C$) et pour le cycle ($C \rightarrow \overline{C} \rightarrow C$).

Un exemple de cycle pour une image pathologique et une image saine est donné en Figure 3.3. Ici le cycleGAN avec la fonction de coût perceptuelle, cyGAN+Per, est utilisé. On peut voir qu'il donne des résultats cohérents. En effet, en passant du domaine SEP au domaine sain (flèche bleu clair en haut), les lésions sont effacées (flèches vertes). Néanmoins, cela modifie également les ventricules. On note que les ventricules sont également modifiés pour les images saines ($H \rightarrow H$, ligne du bas). Le générateur vers le domaine pathologique semble, de plus, modifier légèrement le contraste des images. On note que, comme voulu, les lésions ne sont pas effacées avec ce dernier ($MS \rightarrow MS$). Pour l'image saine, le phénomène inverse à G_H se produit : les ventricules sont augmentés en passant dans le domaine pathologique alors qu'ils étaient diminués pour passer dans le domaine sain. On note également une légère modification des sillons. La fonction cyclique marche correctement sur le domaine sain ($H \rightarrow MS \rightarrow H$) : le générateur génère bien une lésion puis le générateur H l'efface (flèches vertes, ligne du bas).

Finalement, même si les lésions SEP semblent prises en compte pour passer d'une image SEP à une image saine, une différence au niveau des ventricules entre les deux bases semble également influencer fortement le comportement des générateurs.

III.3 Capacité à effacer les lésions du générateur H

Pour segmenter les lésions, il est nécessaire d'avoir un générateur H qui ne modifie que les lésions. Nous avons vu dans le paragraphe précédent que cela n'était pas le cas mais que les ventricules étaient également largement modifiés. Cela peut être dû à l'atrophie davantage présente chez les patients SEP. Nous allons ici évaluer l'impact de nos propositions pour rendre ce générateur invariant à l'atrophie. La Figure 3.4 montre plusieurs exemples de passage d'un sujet SEP au domaine sain par le biais du générateur H selon plusieurs configurations. Nous comparerons ainsi le cycleGAN de base (cyGAN), le cycleGAN où des images avec atrophie générée ont été utilisées comme base de données saine sans (cyGAN+AD) et avec (cyGAN+Per) la fonction de coût perceptuelle de l'Equation 3.7. Pour rappel, dans ces cas, une base de données de sujets jeunes et donc sans lésions ou atrophie est utilisée. Nous comparerons également le cycleGAN appris sur une base de données saine avec des sujets plus âgés et donc une plus grande atrophie et la présence de lésions

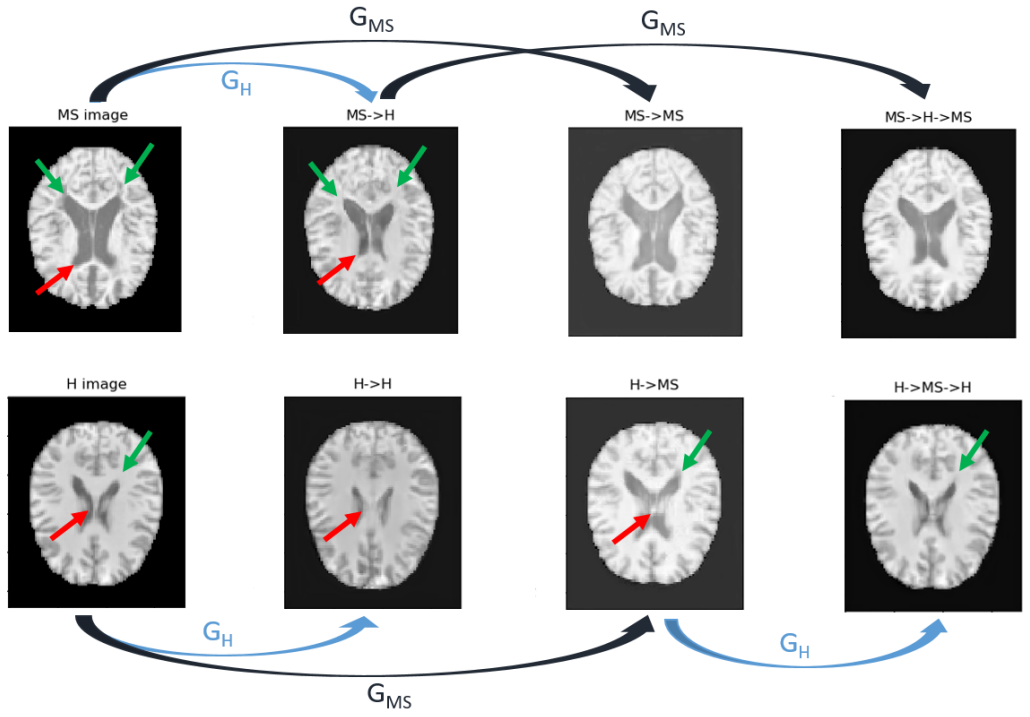


FIGURE 3.3 – *Exemple de cycle obtenu à partir d'une image saine et d'une image SEP. MS correspond au domaine SEP et H au domaine sain. Les flèches vertes correspondent à des modifications voulues de la part des générateurs (inpainting des lésions pour G_H et génération de lésions pour G_{MS}). Les flèches rouges correspondent à des modifications non souhaitées (modifications des ventricules).*

(cyGAN-Old).

Tout d'abord, toutes les configurations effacent partiellement les lésions en les remplaçant par du tissu sain de type substance blanche. Ainsi le volume de cette substance augmente avec toutes les configurations par rapport à l'image pathologique d'origine comme on peut le voir sur la Figure 3.5 (barres bleues).

On peut aussi remarquer que l'ajout des images où une atrophie a été simulée en tant qu'images saines (avec ou sans fonction de coût perceptuelle) diminue fortement la modification des ventricules par le générateur H. Les images en sortie de cyGAN (deuxième ligne de la Figure 3.4) ont des ventricules fortement réduits par rapport à l'image originale et aux autres configurations. Lorsque les images où une atrophie a été générée sont utilisées pour entraîner le GAN (ligne 3), les images en sorties du générateur H ont tendance à avoir des ventricules difformes et assez éloignés d'une atrophie naturelle. On ne retrouve pas ce problème lorsque la fonction de coût perceptuelle est utilisée (quatrième ligne) ou lorsque la base de sujets âgés est utilisée comme base saine (cinquième ligne). On peut aussi voir que l'utilisation de cette base permet de générer des images mieux reconstruites au niveau des sillons qui semblent moins modifiés par rapport à l'image originale qu'avec les autres configurations. La configuration avec la fonction de coût perceptuelle semble néanmoins effacer plus de lésions. Par exemple, pour le patient C de la troisième colonne, les lésions ne sont pas complètement remplacées par du tissu sain dans le cas où la base de sujets âgés sert pour l'entraînement alors que cyGAN parvient à faire l'*inpainting*. Cela pourrait être dû au fait que les sujets de cette base présentent eux mêmes des lésions, non liées à la SEP mais à la vieillesse. Le générateur (par le biais du discriminateur et de la fonction de coût adversaire) peut donc être moins sensible à certaines lésions pour passer du domaine pathologique au domaine sain. Dans le cas du patient D de la dernière colonne, seule la configuration avec la fonction de coût perceptuelle parvient à effacer complètement les lésions.

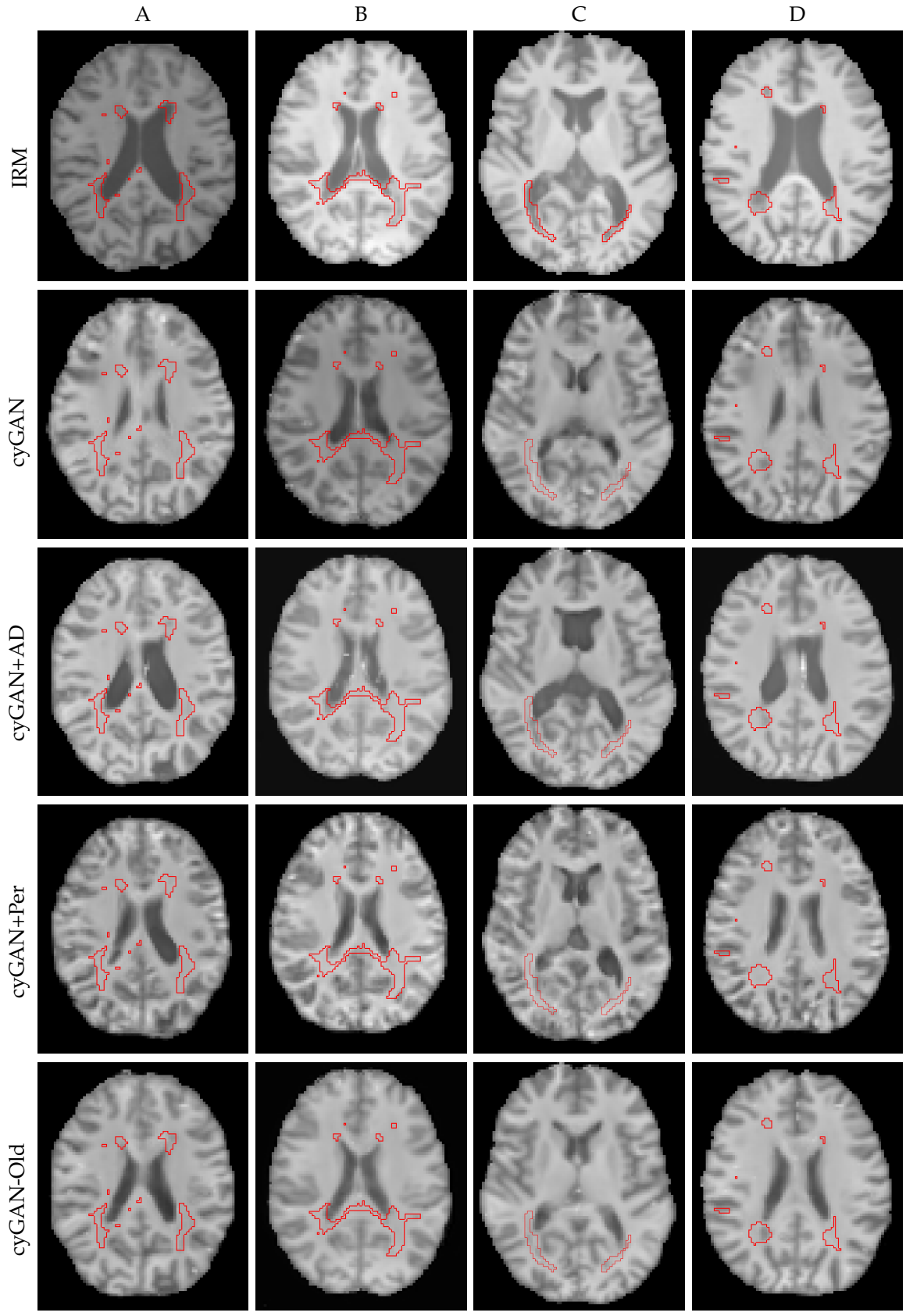


FIGURE 3.4 – *Sorties du générateur H. De haut en bas : image pathologique d'entrée puis sortie du générateur H pour quatre configurations : cycleGAN de base, avec ajout des images avec atrophie dans la base saine, avec ajout de la fonction de coût perceptuelle pour les images avec atrophie et cycleGAN entraîné avec une base saine constituée de sujets âgés.*

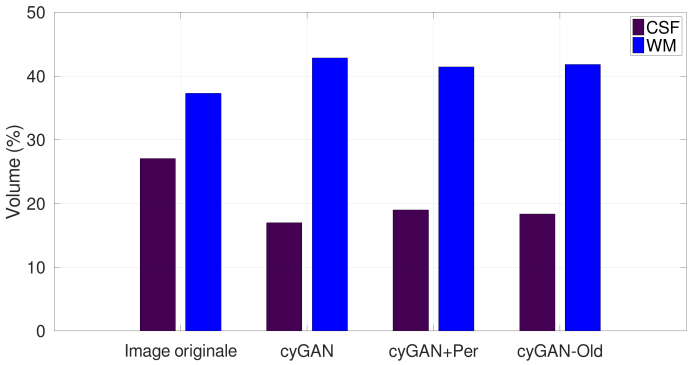


FIGURE 3.5 – *Volume du CSF et de la substance blanche. De gauche à droite : pour l'image pathologique originale, pour la sortie du générateur H du cycleGAN de base, du cycleGAN avec la fonction de coût perceptuelle pour les images avec atrophie et du cycleGAN entraîné avec la base de sujets âgés comme base saine.*

Ainsi, visuellement, cette configuration permet de mieux remplacer les lésions par du tissu sain tout en modifiant de manière similaire à cyGAN-Old, les ventricules grâce à la prise en compte de l'atrophie. Cela peut être vérifié numériquement en regardant le volume du CSF pour ces différentes configurations (Figure 3.5) puisque les ventricules font partie du CSF. Ce volume est moins important avec le cycleGAN de base qu'avec les deux autres configurations, à savoir cyGAN+Per et cyGAN-Old, pour lesquelles il est similaire. Le volume des ventricules est donc plus important dans les images générées par G_H avec ces deux configurations mais toujours plus faible que dans l'image originale. La reconstruction des sillons est meilleure lorsque la base saine est celle des sujets âgées. Aucune configuration ne permet donc de proprement segmenter les lésions avec un Dice moyen inférieur à 5% comme le montre le Tableau 3.2.

Méthode	cyGAN	cyGAN+AD	cyGAN+Per	cyGAN-Old	AE
Dice	0.02	0.02	**0.04**	0.03	0.006

TABLEAU 3.2 – *Dice moyen pour la segmentation des lésions SEP sur IRM T1.*

Conclusion

Dans ce chapitre, nous avons proposé d'utiliser un réseau adversaire génératif pour segmenter de manière faiblement supervisée les lésions SEP sur des IRM T1. Quantitativement, nous n'avons pas obtenu de résultats suffisants pour utiliser cette méthode pour segmenter les lésions. En effet, le passage d'une image dans le domaine SEP à une image dans le domaine sain ne modifie pas seulement les zones lésionnelles. Parmi les modifications de l'image lors de ce passage, les ventricules sont fortement impactés. Nous expliquons cela par le fait que les patients atteints de sclérose en plaques présentent une atrophie précoce et donc un élargissement des ventricules. Nous avons proposé une méthode pour rendre le discriminateur invariant à cette atrophie : à partir de l'image d'un sujet sain, nous simulons de l'atrophie avec une méthode basée sur du recalage et nous forçons toutes les caractéristiques du discriminateur à être les mêmes pour un sujet donné avec les différents niveaux d'atrophie. Cette approche permet de réduire la modification des ventricules par les générateurs mais n'est pas encore suffisante pour que les résultats de segmentation soient satisfaisants.

D'autres solutions ont été testées sans être présentées dans ce chapitre. Nous avons notamment essayé de restreindre les modifications possibles du générateur. Pour cela, plutôt que de laisser le générateur passer d'une image à une autre image en espérant qu'il réalise l'*inpainting* des lésions, nous l'avons contraint à générer une carte de segmentation des lésions. A partir de cette carte, nous procédions à un *inpainting* ad hoc, en remplaçant les zones segmentées par de la substance blanche saine obtenue par interpolation avec les zones de substance blanche voisines. Cette image était alors donnée en entrée du discriminateur. Cette proposition rendait cependant la convergence du modèle plus difficile. Pour effacer les différences dues au passage dans un générateur (flou, changement de contraste, etc), nous avons également utilisé les sorties des générateurs du même domaine ($H \rightarrow H$ plutôt que H et $MS \rightarrow MS$ plutôt que MS) comme images réelles pour les discriminateurs. Ces propositions ne permettaient cependant toujours pas une segmentation satisfaisante.

Puisque la décision des générateurs dépend du comportement des disciminateurs par le biais des fonctions de coût adversaires, il est nécessaire d'avoir un discriminateur s'appuyant au maximum sur les lésions pour discriminer les bases saines des bases pathologiques afin de pouvoir correctement segmenter les lésions. Par la suite, nous proposons donc de nous intéresser à ces classifieurs.

CHAPITRE

4

IDENTIFICATION DE BIAIS DANS LA CLASSIFICATION D'IRM ET AMÉLIORATION DE L'INTERPRÉTABILITÉ PAR L'UTILISATION DE CARTES DE PROBABILITÉ D'APPARTENANCE AUX TISSUS CÉRÉBRAUX

Introduction

Les résultats obtenus avec le GAN dans le chapitre précédent restent peu satisfaisants. Puisque la sortie du générateur et donc les performances de segmentation sont conduites par la décision du discriminateur à travers la fonction de coût adversaire, il est important que cette dernière soit focalisée sur la pathologie. En effet, si ce qui différencie les images de la base saine par rapport au images pathologiques est uniquement cette pathologie, pour passer du domaine pathologique au domaine sain, il faudrait remplacer la zone pathologique par du tissu sain. La différence de reconstruction nous permettrait alors de segmenter la pathologie comme voulu. Pour étudier sur quoi la décision du discriminateur est basée, nous pouvons utiliser les cartes d'attributions vu en Section IV.1.1 du Chapitre 2. Les résultats obtenus sur les deux discriminateurs du CycleGAN, à savoir celui pour les images SEP (MS) et celui pour les images saines (H), sont donnés en Figure 4.1. On peut voir que la décision des deux discriminateurs n'est absolument pas basée sur des éléments radiologiques caractéristiques de la pathologie (lésions), ce qui pourrait expliquer les performances décevantes de cette architecture.

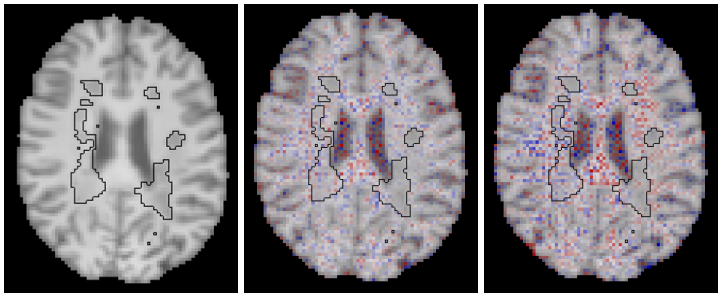


Figure 4.1 – *Exemple de cartes d'attributions Integrated Gradient obtenues avec les discriminateurs d'un GAN entraîné. L'IRM T1 d'un patient SEP est à gauche et les cartes d'attributions pour le discriminateurs MS et H sont au milieu et à droite. Les lésions sont entourées en noir. La pertinence pour la classe SEP est en rouge et celle pour la classe saine en bleu.*

Nous souhaitons donc améliorer le discriminateur pour que sa décision soit davantage basée sur la pathologie. En effet, le discriminateur peut différencier la base saine de la base pathologiques en exploitant des biais présents dans les bases d'entraînement et non les critères que des experts du domaine utiliseraient telle que la présence d'un signe radiologique comme les lésions. En outre, un des problèmes de l'IRM est que les intensités ne sont mesurées que de manière relative. Ainsi, deux acquisitions IRM réalisées sur le même patient en utilisant le même scanner avec des paramètres d'acquisition légèrement différents mais qui correspondent à une même séquence présenteront des variations d'intensité. Ce problème est d'autant plus présent lorsque les acquisitions proviennent de différents scanners et qu'il y a une différence de marques, d'intensités de champ, de protocoles d'acquisition, etc. En effet, pour générer une image dans une même modalité, les paramètres d'acquisition (temps de relaxation, temps d'écho, etc) peuvent varier. Or pour obtenir des bases de données suffisamment grandes pour entraîner des réseaux de neurones, il est souvent nécessaire d'agréger des bases de données issues de différents centres d'acquisition. Une signature d'acquisition présente dans les images peut être un des biais utilisé par le réseau pour prendre sa décision. Dans ce chapitre, nous nous intéressons à la classification d'images RM saines vs pathologiques. Nous nous intéresserons tout particulièrement aux patients atteints de sclérose en plaques. Nous souhaitons donc que la décision du réseau soit basée sur la présence de lésions. Dans un premier temps, nous proposerons une méthode de normalisation des IRM pour la classification par apprentissage profond qui permet au réseau de davantage basée sa décision sur les lésions. Puis, nous étudierons plus en détails

un biais possiblement utilisé par les réseaux pour prendre leur décision dans le cadre de la classification d'IRM cérébrales : la forme du cerveau.

Contributions

Les contributions présentées dans ce chapitre sont :

— L'utilisation de cartes de probabilité d'appartenance aux tissus qui permet une classification d'images sain vs SEP davantage basée sur les lésions SEP.

— L'identification d'un biais spécifique lors de la classification d'images saines vs pathologiques à savoir la forme du cerveau.

La première partie de ces travaux a été présentée à une conférence internationale et lors d'un congrès national :

— IEEE International Symposium on Biomedical Imaging (ISBI) 2021 [Wargnier-Dauchelle *et al.*, 2021b]

— Congrès de la Société Française de Résonance Magnétique en Biologie et Médecine (SFRMBM) 2021 [Wargnier-Dauchelle *et al.*, 2021a]

I Normaliser l'IRM par l'utilisation de cartes de probabilité d'appartenance aux tissus cérébraux

Les travaux présentés dans cette section ont été présentés à la conférence IEEE International Symposium on Biomedical Imaging (ISBI) 2021 [Wargnier-Dauchelle *et al.*, 2021b] ainsi qu'au congrès de la Société Française de Résonance Magnétique en Biologie et Médecine (SFRMBM) 2021 [Wargnier-Dauchelle *et al.*, 2021a]. Ces travaux portent sur l'utilisation de cartes de probabilité d'appartenance aux tissus cérébraux comme entrée d'un réseau de neurones pour une classification davantage basée sur les lésions de sclérose en plaques.

I.1 Utilisation de cartes de probabilité d'appartenance aux tissus

Pour lutter contre le problème des intensités non normalisées des images RM ou tout autre biais résultant d'une signature de l'acquisition, nous proposons un protocole permettant une forte normalisation des images d'entrée du réseau de classification. Pour cela, nous proposons d'apprendre un réseau de classification non pas avec les images RM en entrée mais avec des cartes de probabilités d'appartenance aux tissus cérébraux à savoir le liquide cérébrospinal (*cerebrospinal fluid* (CSF)), la substance grise (*grey matter* (GM)) et la substance blanche (*white matter* (WM)). Cette normalisation réduit la dépendance aux artefacts, bruits, signatures de scanner, etc.

Pour générer ces cartes de probabilité, nous avons utilisé FSL FAST [Zhang *et al.*, 2001]. On suppose que la distribution des intensités des images RM suit un modèle de mélange de gaussiennes. Dans le cadre de la segmentation en trois types de tissus (CSF, GM, WM), la distribution est modélisée par un mélange de trois gaussiennes comme illustré en Figure 4.2. Un *apriori* sur l'image nous donne la correspondance entre la gaussienne considérée et le type de tissu. Ainsi pour une image T1, le CSF correspondra aux intensités les plus basses, suivi de la substance grise puis de la substance blanche. L'objectif est donc de trouver les paramètres (moyenne et écart-type) de ces gaussiennes.

Néanmoins, en utilisant uniquement ce modèle simple, les multiples bruits et artefacts présents dans les images IRM conduirait à une segmentation peu fiable car très sensible à ces bruits. Pour améliorer la segmentation, le voisinage et les inhomogénéités de champ sont considérés dans l'estimation du modèle. Le voisinage est pris en considération grâce à un

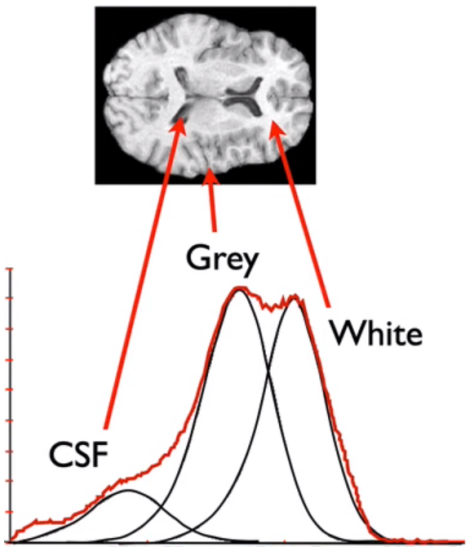


FIGURE 4.2 – *Modèle de mélange de 3 gaussiennes pour la segmentation en tissus cérébraux (liquide cérébro-spinal, substance grise et substance blanche) sur une IRM T1 avec FAST. Image issue de la chaîne YouTube FSLCourse.*

champ aléatoire de Markov, c'est-à-dire que $\forall x \in X, P(x_i|x_{S\setminus i}) = P(x_i|N_i)$ où X est l'ensemble des réalisations possibles d'un champ aléatoire, S est l'ensemble des index possibles et N_i est l'ensemble des index d'un voisinage de i. Les inhomogénéités de champ sont considérées comme un bruit multiplicatif gaussien et sont estimées itérativement en même temps que les paramètres des gausiennes. Finalement, le modèle peut être défini comme :

$$P(y_i|x_{N_i}, \theta) = \sum_{l \in L} g(y_i, \theta_l) P(l|y_i, x_{N_i}, \theta) \tag{4.1}$$

où y_i est l'intensité du voxel i, L est l'ensemble des labels des classes de tissus considérés, g est une gaussienne de moyenne et variance $\theta_l = (\mu_l, \sigma_l)$, x_{N_i} est le label de la classe de tissu des voxels voisins N_i et θ inclus à la fois θ_l et les paramètres des artefacts comme les inhomogénéités de champ. Le modèle est appris itérativement avec un algorithme d'espérance-maximisation (EM) en estimant tout d'abord du bruit lié aux inhomogénéités puis, en calculant la vraisemblance de la distribution des intensités et en maximisant cette vraisemblance pour obtenir les nouveaux paramètres des gaussiennes.

La carte de probabilité conditionnelle $P(l|y_i, x_{N_i}, \theta)$ est calculée pour trois classes (qui représenteront le CSF, la substance grise et la substance blanche) puis donnée en entrée du réseau de neurones comme une image avec trois canaux. Un exemple de cartes est donné Figure 4.3.

I.2 Évaluation de l'influence des lésions de sclérose en plaques

Pour évaluer l'influence des lésions de sclérose en plaques dans la décision, nous proposons de comparer la réponse du classifieur pour la même image avec et sans la présence de ces lésions. Pour cela, il est nécessaire de remplacer les zones lésionnelles par du tissu sain de type substance blanche.

Cet *inpainting* est réalisé avec la méthode de [Sdika et Pelletier, 2009] qui utilise les voxels de la substance blanche voisins pour remplir itérativement depuis le bord jusqu'au centre

FIGURE 4.3 – *De gauche à droite : IRM T1 (vue axiale) et les cartes de probabilités correspondantes pour le CSF, la substance grise et la substance blanche obtenue avec FSL FAST.*

la lésion. Ainsi, à chaque itération :

$$\forall x \in B(\Omega), I(x) = \frac{\sum\limits_{n \in V(x) \cap \overline{\Omega} \cap WM} w(x-n) I(n)}{\sum\limits_{n \in V(x) \cap \overline{\Omega} \cap WM} w(x-n)} \tag{4.2}$$

où $w(u) = \frac{1}{\sqrt{2\pi} e^{-\frac{1}{2}u^2}}$ est un noyau gaussien, I est l'image, Ω est la zone à inpainter, $B(\Omega)$ sont les bords de cette zone, $\overline{\Omega} = I \setminus \Omega$, $V(x)$ est le voisinage du voxel x et WM est la substance blanche comme elle pourrait être segmentée par FAST. Après chaque itération les voxels modifiés sont ajoutés à WM. Cette opération est répétée jusqu'à l'*inpainting* complet.

Une fois cette opération d'*inpainting* réalisée, l'image "inpaintée" est identique à l'image originale à l'exception des zones lésionnelles qui ont désormais des intensités similaires à la matière blanche environnante. Cette opération permet de comprendre l'impact de ces lésions sur la décision. Pour cela, nous proposons deux métriques qui seront utilisées à l'inférence.

Premièrement, nous proposons d'inspecter la probabilité pour la classe pathologique donnée en sortie du réseau pour l'image avant et après *inpainting*. En effet, en remplaçant la lésion par du tissu sain type substance blanche, l'image devrait être classée moins pathologiques que l'image originale. Pour éviter un effet de saturation, nous proposons de regarder la probabilité avant la sigmoïde et donc de regarder la log-probabilité. Nous définissons ainsi la métrique suivante :

$$plog_{\text{Diff}} = P_{log}(I) - P_{log}(I_{inpaint}) \tag{4.3}$$

où P_{log} est la log-probabilté, I l'image et $I_{inpaint}$ l'image ayant subi l'*inpainting*.

Ensuite, nous proposons d'utiliser les cartes d'attributions pour distinguer les voxels de l'image d'entrée importants pour la décision. En effet, après *inpainting* les attributions devraient indiquer davantage de voxels pertinents pour la décision saine et inversement. Cette inspection peut être faite visuellement. Pour une évaluation quantitative, nous définissons la métrique suivante :

$$\mu_{\text{Diff}} = \frac{\mu - \mu_{inpaint}}{max(|\mu_{inpaint}|, |\mu|)} \tag{4.4}$$

où μ et $\mu_{inpaint}$ sont respectivement la moyenne des attributions dans une zone donnée pour l'image d'origine et l'image après *inpainting* des lésions.

I.3 Protocole expérimental

Nous avons comparé notre méthode utilisant les cartes de probabilité d'appartenance aux tissus cérébraux (classifieur appelé par la suite C-PMAPS) à un classifieur appris classiquement sur les images IRM (C-MRI).

I.3.1 Données

Nous avons utilisé trois bases de données d'IRM T1 : la base de données publique de sujets sains IXI et les bases de patients atteints de sclérose en plaques OFSEP1 et MSSEG. La base de tumeurs cérébrales n'a été utilisée qu'à partir de la Section I.6. La répartition entre les jeux d'entraînement (N_{train}), de validation (N_{val}) et de test (N_{test}) est indiquée dans le tableau 4.1.

Tableau 4.1 – *Bases de données d'IRM T1. H fait référence à la base saine, MS à la base de patients atteints de sclérose en plaques et T à la base de tumeurs cérébrales.*

Base	N_{train}	N_{val}	N_{test}	H/MS/T	Annoté
IXI	400	130	50	H	Non
OFSEP1	383	97	30	MS	Non
MSSEG	0	0	52	MS	Oui
BraTS 2020	280	40	40	T	Oui

Ces données ont été prétraitées selon le processus décrit dans la Section III.2 du Chapitre 1 à savoir un recalage sur l'atlas du MNI, une extraction du cerveau et une correction des inhomogénéités de champ. La taille des images est de $91 \times 109 \times 91$ avec des voxels de 2mm^3.

I.3.2 Implémentation

Le classifieur implémenté est une architecture PatchGAN 3D ($70 \times 70 \times 70$). Pour l'entraînement, la fonction de coût utilisée est l'entropie croisée binaire et l'optimisation a été faite avec Adadelta [Zeiler, 2012] et un *learning rate* initial de 1 a été utilisé. Nous avons utilisé différentes méthodes d'augmentation de données lors de l'apprentissage : des déformations élastiques, des variations de luminosité et une inversion de l'image le long du plan sagittal. Un arrêt anticipé de l'entraînement a été fait en choisissant l'*epoch* avec les meilleures performances de classification sur le jeu de validation.

I.3.3 Métriques

Pour évaluer les capacités de classification, nous avons mesuré l'exactitude (*accuracy*) pondérée. L'évaluation de l'influence des lésions de sclérose en plaques a été quantifiée avec les métriques présentées dans la Section I.2. Le masque de vérité terrain permettant de vérifier l'impact des lésions dans la décision est celui annoté sur la modalité FLAIR recalée sur la modalité T1 du même patient pour la base MSSEG. En effet, il s'agit de l'unique segmentation manuelle des lésions SEP disponible dans nos bases de données. Comme en FLAIR, les lésions apparaissent plus grandes qu'en T1, l'utilisation de ce masque ne pose pas de problème pour les métriques calculées ou l'*inpainting* réalisé. En effet, si une partie du cerveau dans le masque ne présente en réalité aucune lésion visible en T1, le tissu sain sera remplacé par du tissu sain sans impact. Les attributions ont été calculées avec Integrated Gradient, une image de référence nulle et dans sa version globale (multiplication par l'entrée) pour notre protocole d'évaluation. L'homogénéité de ces cartes a été évaluée par le biais d'une variation totale définie comme :

$$TV(x) = \sum_{i \in N} \frac{1}{|V(i)|} \sum_{j \in V(i)} |x_i - x_j| \tag{4.5}$$

où x est l'image, N représente les indexes des pixels et $V(i)$ est le voisinage du pixel i.

I.4 Résultats

I.4.1 Performances de classification sain vs SEP

Tout d'abord, nous avons évalué les performances de classification de chaque classifieur : celui entraîné sur les IRM (C-MRI) et celui entraîné sur les cartes de probabilité (C-PMAPS). Les résultats sont présentés dans le Tableau 4.2. On peut remarquer que le classifieur C-PMAPS classe mieux les images avec une *accuracy* entre 7.5% et 10% plus haute que celle de C-MRI.

Tableau 4.2 – *Précision de classification (accuracy) pour C-MRI et C-PMAPS sur les différentes bases.*

	Bases d'évaluation	
Classifieur	IXI/OFSEP1	MSSEG
C-MRI	0.88	0.85
C-PMAPS	**0.95**	**0.94**

I.4.2 Contribution des lésions dans la décision du classifieur

Pour évaluer l'impact des lésions dans la décision, nous utilisons la métrique $plog_{\text{Diff}}$ proposée (Equation 4.3). Plus sa valeur est élevée, plus la présence de lésions a un impact pour classer une image comme pathologique. La valeur de cette métrique ainsi que le nombre de patients de la base de test classés moins pathologiques lorsque les lésions sont effacées sont reportés dans le Tableau 4.3.

On peut voir que la contribution des lésions dans la décision est plus importante pour C-PMAPS puisque la différence de log-probabilité est plus importante avec ce classifieur qu'avec C-MRI. De manière similaire, la probabilité d'être classé SEP diminue avec l'*inpainting* des lésions pour plus d'images quand C-PMAPS est utilisé.

Tableau 4.3 – *Moyenne ± écart-type de la différence des log-probabilités ($plog_{\text{Diff}}$) de chaque classifieur . La colonne "/52" fait référence au nombre de patients (sur un total de 52 patients) pour qui l'image après inpainting est classée comme moins pathologique.*

Classifieur	$plog_{\text{Diff}}$	/52
C-MRI	1.20 ± 7.43	44
C-PMAPS	$\mathbf{4.04 \pm 8.68}$	**49**

I.4.3 Des cartes d'attributions moins bruitées

Les cartes d'attributions ont été générées pour C-MRI et C-PMAPS sur les images pathologiques MSSEG (voir Figure 4.4). Visuellement, C-MRI semble générer des cartes d'attributions plus bruitées. Le calcul de la variation totale confirme cette impression puisqu'elle est 10 fois plus élevée pour C-MRI ($TV = 35339 \pm 4576$) que pour C-PMAPS ($TV = 3326 \pm 484$ en moyenne sur les 3 canaux). Les cartes de C-PMAPS sont donc plus homogènes.

I.4.4 Une décision davantage basée sur les lésions

Sur la Figure 4.4), les attributions pour C-MRI semblent également moins axées sur les lésions que les attributions de C-PMAPS. Pour C-PMAPS, le CSF semble moins porteur d'information que la substance blanche ou la substance grise. En effet, d'une part, dans la substance grise, les zones lésionnelles sont associées à la pertinence SEP (attributions

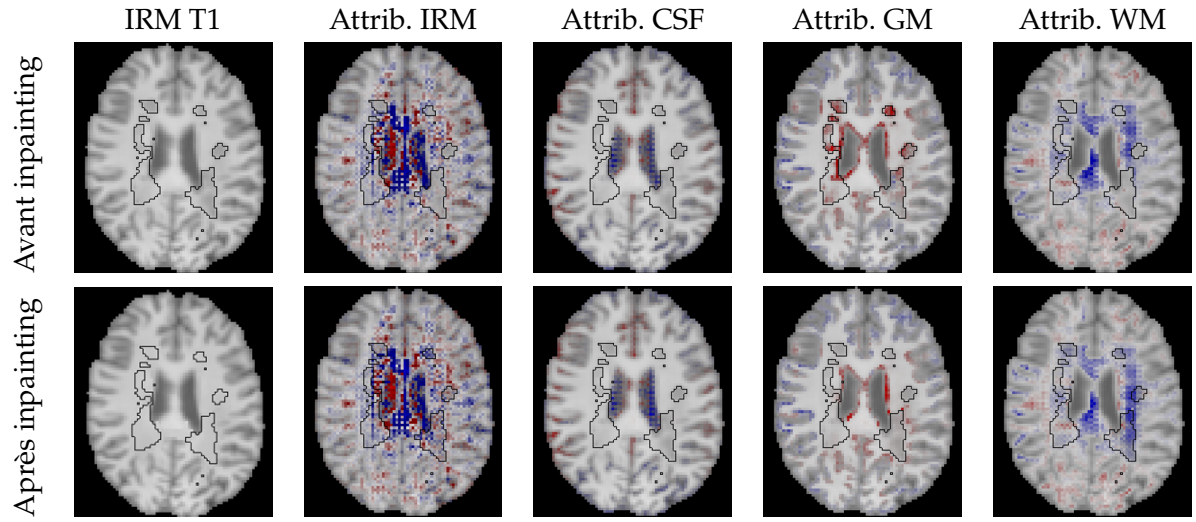


FIGURE 4.4 – *Vue axiale pour un patient SEP. L'annotation manuelle des lésions FLAIR lésions est en noir. Le bleu représente les voxels pertinents pour la décision "sain" et le rouge pour la décision "pathologique".*

TABLEAU 4.4 – *Moyenne ± écart-type de la différence relative μ_{Diff} des moyennes des attributions entre les images avant et après inpainting dans les lésions ou dans toute l'image. La colonne "/52" fait référence au nombre de patients (sur un total de 52 patients) avec plus de voxels pertinents pour la classe SEP avant inpainting.*

	Image complète		Lésions					
			Toute pertinence		Pertinence < 0		Pertinence > 0	
Classifieur	μ_{Diff}	/52	μ_{Diff}	/52	μ_{Diff}	/52	μ_{Diff}	/52
C-MRI	0.16 ± 0.31	37	0.40 ± 0.81	40	0.56 ± 0.16	**51**	−0.50 ± 0.14	1
C-PMAPS (tout)	0.34 ± 0.44	44	**1.07 ± 1.05**	**44**	0.32 ± 0.37	43	0.38 ± 0.33	45
C-PMAPS (CSF)	−0.23 ± 0.33	6	−0.35 ± 0.98	19	−0.72 ± 0.51	6	−0.74 ± 0.33	9
C-PMAPS (GM)	−0.34 ± 0.49	13	0.48 ± 0.67	41	−0.73 ± 0.28	2	**0.75 ± 0.26**	**51**
C-PMAPS (WM)	**0.36 ± 0.48**	**45**	0.53 ± 0.77	42	**0.77 ± 0.23**	**51**	−0.75 ± 0.21	1

positives) avant l'*inpainting*, alors qu'il n'y a plus cette pertinence après l'élimination des lésions. D'autre part, dans la substance blanche, les tissus sains apportent une pertinence saine contrairement aux lésions. Par conséquent, avec ce type d'entrée, le classifieur semble en accord avec les *aprioris* cliniques. On note également une activation positive autour des ventricules pour la carte de substance grise. Cela pourrait être lié à l'atrophie que l'on peut observer chez les patients SEP notamment sur la modalité T1.

Pour une évaluation quantitative, les résultats sur la métrique μ_{Diff} sont présentés dans Tableau 4.4. Elle a été calculée sur l'ensemble de l'image mais aussi uniquement dans le masque des lésions pour comprendre leur influence sur la décision. Les attributions négatives correspondant à la classe saine et les positives correspondant à la classe pathologique ont été également étudiées séparément. Pour C-PMAPS, l'évaluation a été faite sur la moyenne des trois canaux puis sur chaque canal séparément.

Les résultats montrent que C-PMAPS est plus interprétable car la différence entre les attributions d'images avant et après *inpainting* est plus importante, en particulier dans les lésions. En effet, les images pour lesquelles les lésions ont été remplacées par du tissu sain contiennent plus d'information pour la classe saine et les images pathologiques originales contiennent plus d'information pour la classe pathologique. Nous remarquons également que la carte de probabilité pour la substance blanche est le canal le plus important pour la

pertinence saine alors que celle pour substance grise est le plus important pour la pertinence de la SEP.

Ainsi la décision du réseau utilisant les cartes de probabilité d'appartenance aux tissus cérébraux semble davantage basée sur les lésions et donc plus interprétable. En outre, la distinction en canaux permet une analyse plus poussée de la décision en donnant l'importance de chaque tissu.

I.5 Un exemple de limite : les lésions trop proches des ventricules

Nous avons proposé d'utiliser des cartes de probabilité d'appartenance aux tissus cérébraux comme entrée d'un classifieur profond sain vs SEP à la place des images RM classiquement utilisées. Ces cartes de probabilité représentent une forte normalisation des images RM permettant de supprimer les bruits sur lesquels la décision du classifieur pourrait se baser plutôt que les informations pertinentes comme la présence de signes radiologiques de la pathologie dans l'image. Nous avons ainsi montré qu'avec ces cartes, la décision est davantage basée sur la pathologie tout en améliorant les performances de classification. Il existe cependant quelques limites à notre proposition. En effet, l'objectif est de conserver l'information pertinente tout en retirant les bruits et biais. Il semble y avoir moins de biais puisque la décision est davantage basée sur les lésions. Néanmoins, le risque est de perdre de l'information pertinente.

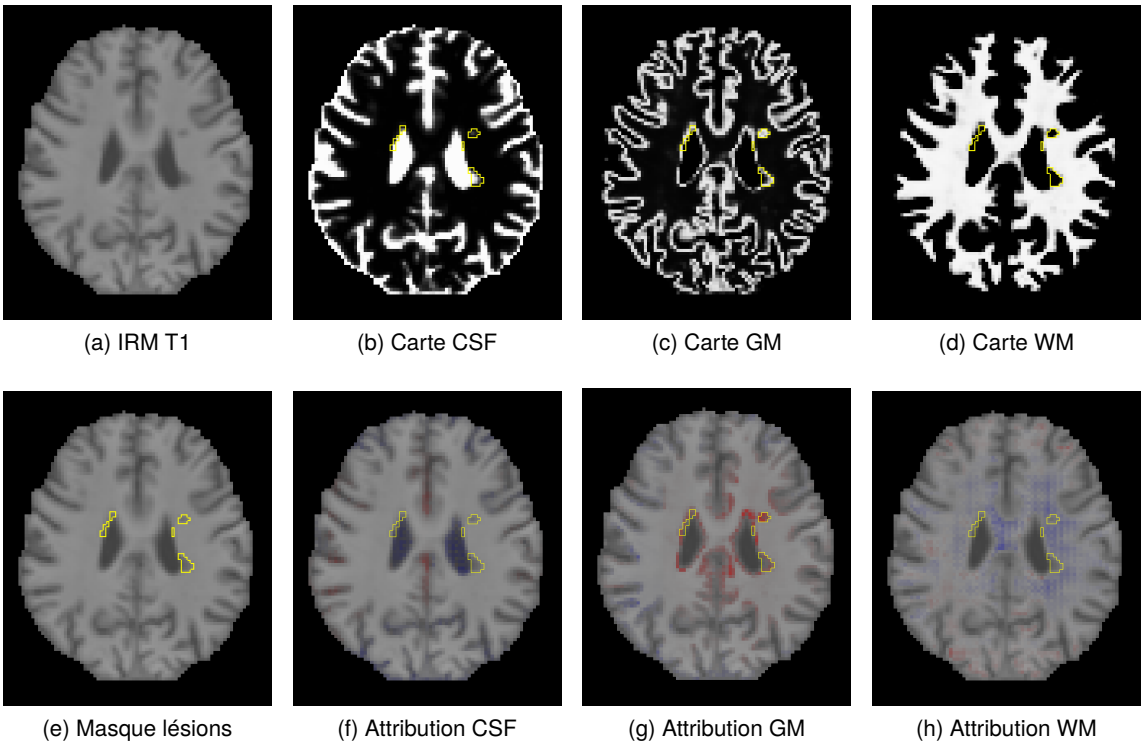


Figure 4.5 – *Exemple de cas SEP où la segmentation en tissu cérébraux inclus une lésion proche des ventricules dans ces derniers. La première colonne présente l'IRM et le masque des lésions (en jaune). Les cartes de probabilité d'appartenance aux tissus sont à en haut et les attributions correspondantes en bas.*

.

En effet, tout repose sur une bonne génération de ces cartes. Si trop d'information est éliminée avec cette forte normalisation, le réseau ne pourra plus utiliser cette information perdue pour la classification. Dans le cas d'une sous segmentation en tissus cérébraux, des zones voisines aux intensités trop proches seront regroupées dans la même classe alors que,

sur l'IRM, la distinction de ces deux zones était perceptible. Par exemple, lorsque la lésion est trop proche du ventricule comme pour le cas de la Figure 4.5, la lésion est incluse dans le CSF produisant un ventricule déformé et plus gros mais pas anatomiquement aberrant. Il est donc difficile pour le réseau de comprendre la présence d'une lésion à cet endroit et donc de baser sa décision dessus. On peut le voir sur les attributions où seul le bord de la lésion est pertinent pour la décision pathologique dans la substance grise, de la même manière que le reste du ventricule. Il n'y a donc pas de distinction entre le ventricule et la lésion.

I.6 Extension aux tumeurs cérébrales

La même méthode peut être appliquée au IRM T1 des patients atteints de tumeurs cérébrales. Néanmoins, il n'est pas possible d'évaluer l'impact des tumeurs avec de l'*inpainting* puisque les tumeurs sont beaucoup plus grosses et déforment les tissus sains environnants. Remplacer la zone tumorale par de la substance blanche créerait des artefacts.

Un exemple visuel de carte d'attributions issues d'une classification C-PMAPS de sujets sains vs avec tumeurs est donné en Figure 4.6. Les attributions semblent moins en adéquation avec la pathologie. En effet, même si des attributions positives (pour la classe pathologique, en rouge) se situent dans la tumeur, on en retrouve également en dehors. La décision n'est pas uniquement basée sur les tumeurs. Les attributions semblent néanmoins meilleures que lorsque C-MRI est utilisé. Le bord du cerveau semble également avoir une place prépondérante dans la décision puisque les attributions y sont fortes. Cela pourrait être dû à une discrimination à partir de la forme du cerveau. Nous allons analyser cette conjecture par la suite.

II Impact de la forme du cerveau dans la classification

En normalisant l'IRM, nous souhaitons supprimer les bruits et biais qui pourraient influencer la décision des réseaux de neurones plus que l'information pertinente d'un point de vue expert comme la présence d'une structure radiologique particulière pour la classification d'images saines vs pathologiques. Nous avons vu que la normalisation proposée en remplaçant les IRM par les cartes d'appartenance aux tissus, peut supprimer de l'information pertinente qui serait utile au réseau pour prendre sa décision. A l'inverse, on peut se questionner sur la suppression totale ou non des biais que nous ne souhaitons pas voir être utilisés par le réseau. Pour répondre en partie à cette question, nous pouvons pousser à l'extrême cette suppression d'information.

II.1 Normalisation extrême : utilisation d'un masque binaire

Nous proposons d'utiliser uniquement un masque binaire comme entrée du réseau de neurones de classification d'images saines vs pathologiques. Ce masque binaire indique pour chaque voxel, s'il se trouve dans le cerveau (valeur de 1) ou en dehors (valeur de 0). Il peut être classiquement obtenu avec des méthodes de l'état de l'art comme HD-BET décrit en Section III.2. Il ne contient donc que l'information de la forme du cerveau de chaque sujet. Un exemple est présenté Figure 4.7. Le réseau est alors entraîné pour classer les masques de la base de données saine par rapport à ceux des bases pathologiques. Cette expérience nous permettra de déterminer si un réseau de neurones est capable ou non de discriminer les deux bases par rapport à la forme des cerveaux uniquement.

II.2 Utilisation de masque du cerveau de la classe opposée

Nous pouvons également nous intéresser au rôle de cette information dans la classification à partir des images. Pour évaluer cet impact, nous proposons d'entraîner le réseau

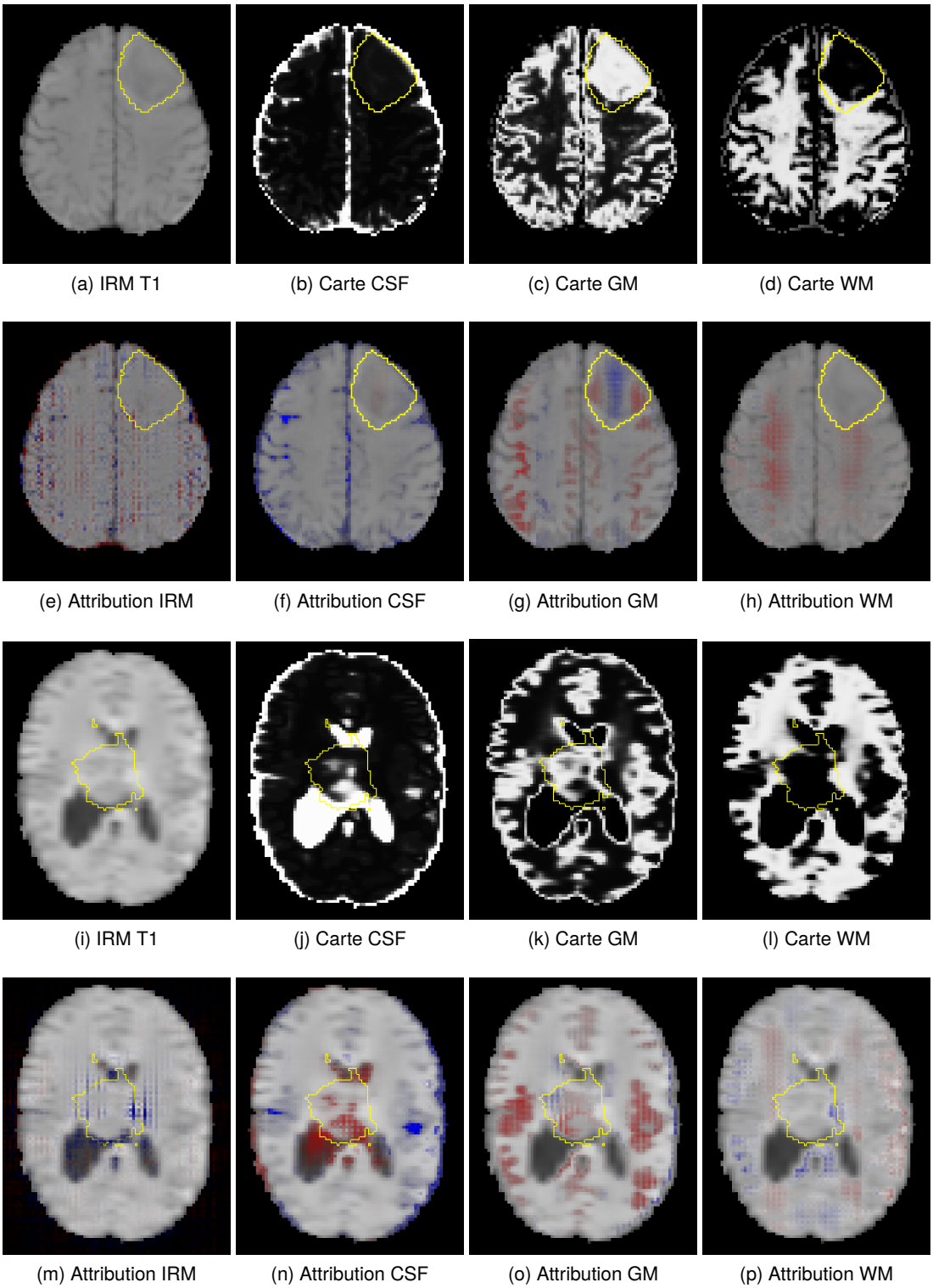
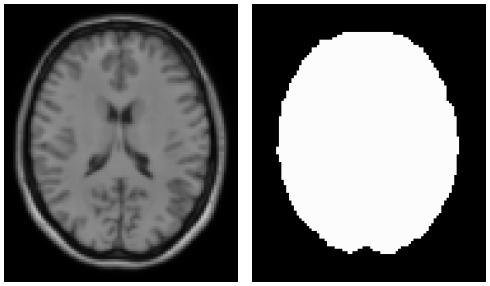


FIGURE 4.6 – *Deux exemples d'un patient avec une tumeur cérébrale. De gauche à droite, nous avons l'IRM T1 et les cartes de probabilité et en dessous les cartes d'attributions associées. La tumeur est entourée en jaune.*



(a) IRM (b) Masque du cerveau

FIGURE 4.7 – *IRM T1 et le masque du cerveau (extrait avec HD-BET) associé.*

sur les images mais d'évaluer les performances sur les mêmes images où la forme du cerveau serait plus proche de celle de la base opposée. Pour cela, un masque du cerveau de la base opposée est tiré aléatoirement et appliqué à l'image originale : les pixels en dehors du masque sont mis à la valeur du fond. Cette image modifiée est passée dans le réseau et les performances de classification entre l'image originale et l'image modifiée sont comparées. Des exemples d'images avant et après application du masque de la base opposée sont donnés en Figure 4.8.

II.3 Protocole expérimental

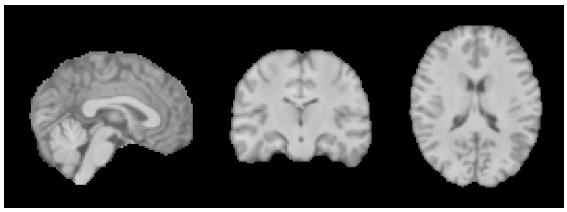
Les données utilisées sont les mêmes que dans la Section précédente. Pour la deuxième expérience consistant à inverser les formes de cerveau, les IRM et les cartes de probabilité d'appartenance aux tissus cérébraux ont été utilisées. Le protocole d'entraînement est également le même. Pour l'évaluation de la classification à partir des masques de cerveau, nous avons choisi de visualiser l'*accuracy* au cours de l'entraînement sur les jeux d'entraînement et de validation. Pour la classification en changeant la forme du cerveau, la différence d'*accuracy* entre l'image originale et l'image après avoir appliqué le masque de la classe opposée a été utilisée. Cette différence a été moyennée sur 50 epochs à partir de la convergence.

II.4 Résultats

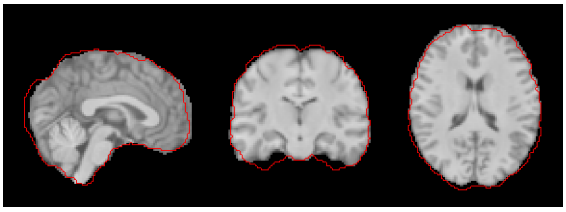
II.4.1 Classification à partir du masque

Les courbes d'*accuracy* lors de l'entraînement du classifieur en utilisant les masques du cerveau comme entrée sont données Figure 4.9. Lorsqu'aucune déformation élastique n'est réalisée sur les masques, la classification est parfaite sur les tumeurs. Le classifieur est donc capable de distinguer la base saine de la base tumorale à partir de la forme du cerveau. Pour la SEP, un *overfitting* a lieu mais le classifieur peut quand même bien classer 70% des masques sur la validation. Il semble donc plus facile de discriminer la base saine de la base tumorale sur la forme des cerveaux mais cela reste quand même possible pour la base SEP. En ajoutant des déformations élastiques comme augmentation de données, on peut s'attendre à distinguer plus difficilement les bases sur la forme du cerveau. Néanmoins, les performances de classification restent très bonnes avec environ 90% de masques bien classés pour les tumeurs et 80% pour la SEP. Dans ce dernier cas, l'*overfitting* est moins présent, l'augmentation de données apportant de la robustesse à l'apprentissage.

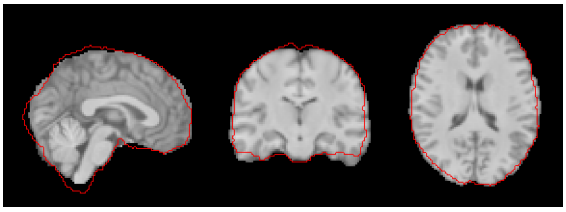
Il semble ainsi possible de distinguer les bases selon la forme des cerveaux surtout pour les tumeurs. Cela peut être lié au fait que la base BraTS 2020 est fournie déjà prétraitée avec un recalage sur un atlas différent et surtout une autre méthode pour segmenter le cerveau. Alors que pour la classification sain vs SEP, la même méthode a été utilisé sur les deux bases



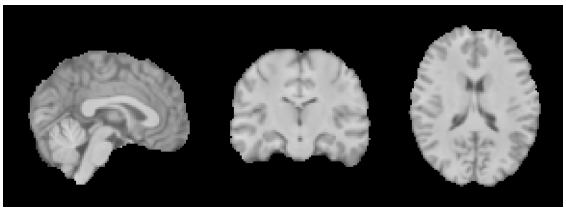
(a) Image H



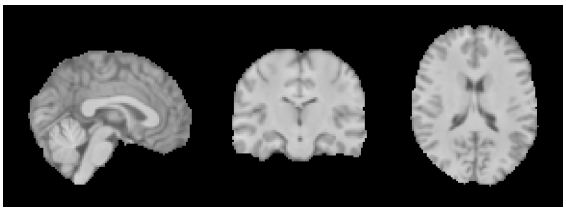
(b) Image H avec un masque MS



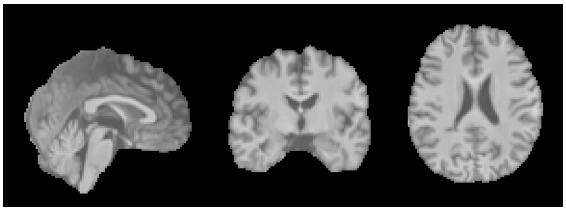
(c) Image H avec un masque T



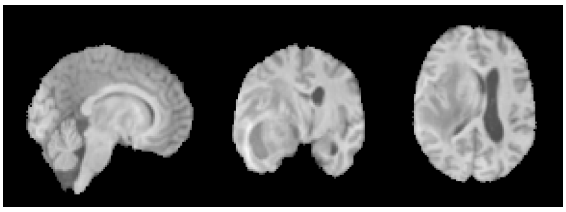
(d) Image H après application du masque MS



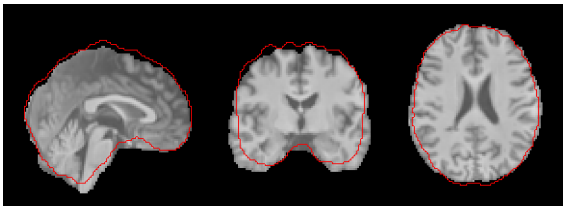
(e) Image H après application du masque T



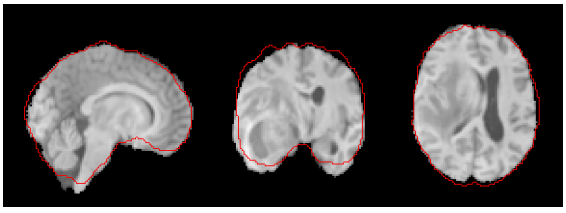
(f) Image MS



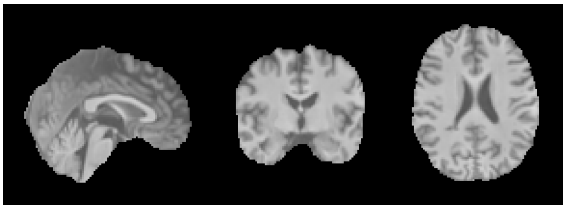
(g) Image T



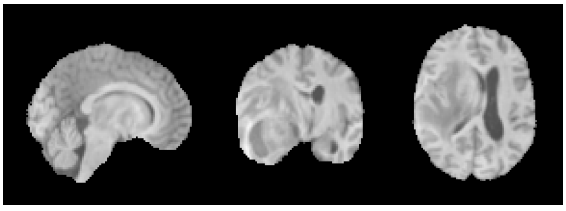
(h) Image MS avec un masque H



(i) Image T avec un masque H



(j) Image MS après application du masque H



(k) Image T un masque H

Figure 4.8 – *Modification de l'image en appliquant un masque du cerveau de la base opposée. T fait référence à la base de tumeurs cérébrales, MS à la base SEP et H à la base saine.*

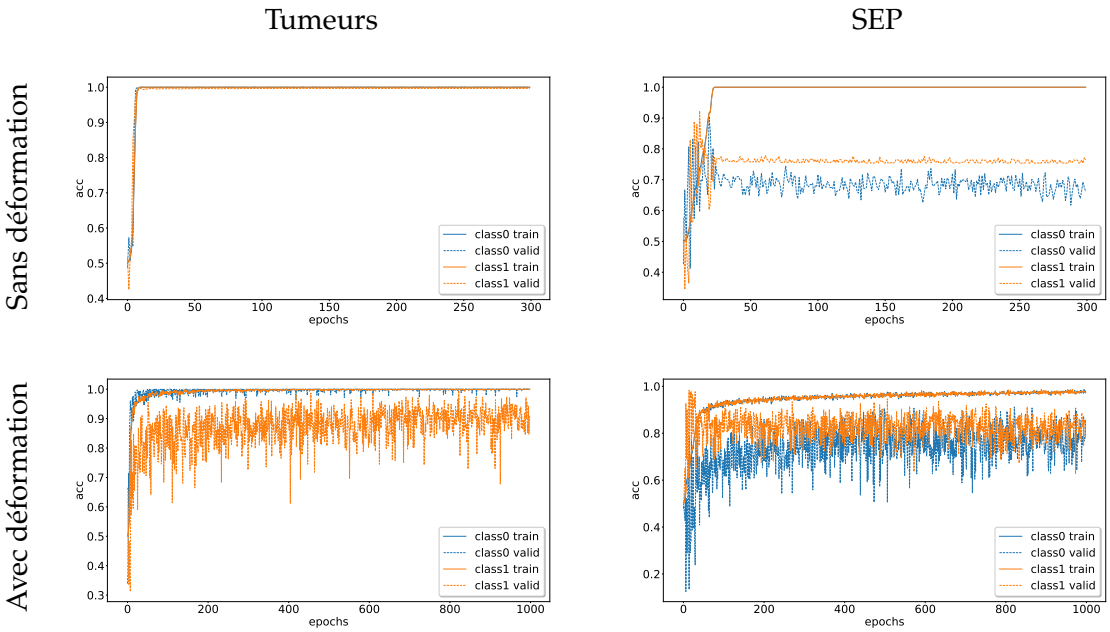


FIGURE 4.9 – *Accuracy au cours de l'entraînement pour un classifieur appris sur les masques du cerveau (de sujets sains vs atteints de tumeurs ou de SEP) avec ou sans déformations élastiques. La classe 0 correspond à la classe saine alors que la classe 1 correspond à la classe pathologique.*

pour extraire le cerveau, deux méthodes différentes ont été utilisées pour les bases de la classification sain vs tumeur. Le prétraitement n'est donc pas tout à fait identique.

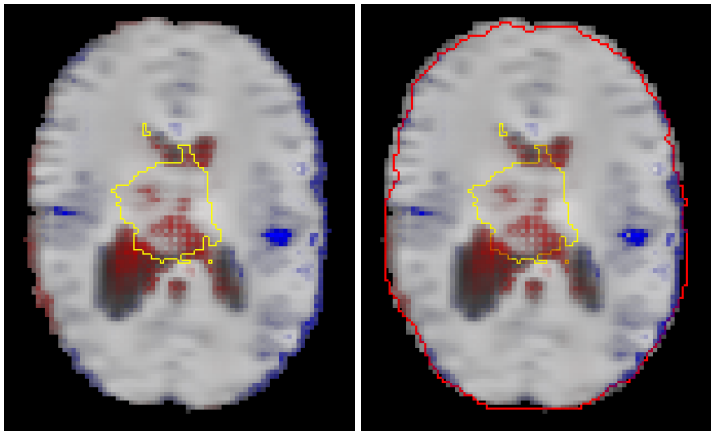
II.4.2 Influence de l'échange de forme

Pour vérifier, l'influence de la forme du cerveau sur la classification, nous avons changé la forme des cerveaux par rapport à ceux appris et évalué les conséquences sur les performances de classification. Pour cela, un masque de la classe opposée est tiré aléatoirement et appliqué sur l'image à classer. Les résultats sont reportés dans le Tableau 4.5.

	Jeu d'entraînement		Jeu de validation	
Expérience	Classe 0	Classe 1	Classe 0	Classe 1
C-MRI tumeurs	0.22	0.03	0.09	0.08
C-PMAPS tumeurs	0	0	0	0
C-MRI SEP	0.04	0.02	0.07	0.03
C-PMAPS SEP	0.13	0	0.15	0.02

TABLEAU 4.5 – *Différence absolue moyenne d'accuracy entre les images originales et les images sur lesquelles un masque du cerveau choisi aléatoirement dans la classe opposée a été appliqué. La classe 0 correspond à la classe de sujets sains et la classe 1 à celle de patients (tumeurs ou SEP).*

On peut voir que ce changement de forme impacte fortement les performances de classification de C-MRI pour la classification des patients atteints de tumeurs cérébrales notamment sur le jeu d'entraînement de la classe saine. Ainsi, appliquer un masque de la classe tumorale sur les images de la classe saine semble perturber la classification. La forme du cerveau semble donc être prise en compte dans la décision. Le changement est moins important pour l'autre classe mais reste présent. Il est important de noter que seul un masque est appliqué sur l'image originale. La forme du cerveau n'est donc pas complètement changée : les pixels



(a) Image originale (b) Image rognée

FIGURE 4.10 – *Différence d'attributions sur une image tumorale avec C-PMAMPS (CSF) avant et après application d'un masque aléatoire de la classe saine. Le masque est dessiné en rouge, la tumeur en jaune.*

en dehors du masque sont mis à la valeur du fond mais le cerveau n'est pas déformé lorsque le cerveau de l'image originale est plus petit (voir Figure 4.8). Les "petits" cerveaux ne seront donc pas ou moins impactés. Cela peut expliquer la différence entre les classes.

Lorsque C-PMAPS est utilisé, les performances sont identiques (et même parfaites avec une *accuracy* de presque 100%) pour les images originales et les images où le masque a été appliqué. Les attributions originales montraient pourtant une activation sur les contours notamment dans le CSF. Si on regarde les attributions sur l'image rognée par rapport à un masque aléatoire de la base saine (Figure 4.10), on voit qu'il y a toujours une activation sur le bord non rogné car dans le masque (en bas à droite de l'image) mais qu'il n'y en a plus sur le bord rogné (en haut à droite). Le bord inchangé reste donc impliqué dans la décision alors que celui modifié ne l'est pas. L'information fournie par les cartes de probabilité est suffisamment riche et propre pour que les performances de classification ne baissent pas dans le cas de la classification sain/tumeurs.

Pour la SEP, l'influence du changement de masque semble moins importante que pour les tumeurs. Cela peut-être lié au fait qu'il est plus difficile de discriminer les masques sains des masques pathologiques pour les bases utilisées dans le cas de la SEP comme nous l'avons vu en Section II.4.1. Néanmoins, les performances de C-PMAPS sont davantage dégradées avec cette modification par rapport à C-MRI. La forme du cerveau semble donc plus prépondérante pour la décision de C-PMAPS que pour C-MRI qui, au vu des attributions (Figure 4.4), base sa décision sur un autre biais. Utiliser les cartes de probabilité d'appartenance aux tissus ne permet pas de supprimer complètement le biais lié à la forme du cerveau pour la prise de décision même si les lésions contribuent davantage à la décision que pour C-MRI.

Ainsi, la forme du cerveau semble bien prise en considération dans la décision des réseaux de neurones dans les différentes configurations testées.

Conclusion

Dans ce chapitre, nous avons montré qu'il existait des biais sur lesquels un réseau de neurones de classification d'images saines vs pathologiques se base en partie pour prendre sa décision notamment lorsque plusieurs bases de données sont utilisées. Parmi ces biais, nous pouvons citer la forme du cerveau. En outre, il est impossible d'identifier tous les biais des bases de données qui peuvent être exploités par le réseau de neurones et donc

de les corriger un à un. Pour éliminer ces biais, nous avons proposé d'utiliser en entrée du réseau de classification des cartes de probabilité d'appartenance aux tissus plutôt que l'IRM T1. Nous avons montré que cette forte normalisation élimine certains biais et permet une décision davantage basée sur les signes radiologiques. Elle permet également de comprendre l'implication de chaque tissu dans la décision du réseau de neurones. Néanmoins, elle peut également enlever de l'information pertinente dont le réseau pourrait se servir pour prendre sa décision. Un protocole de normalisation sur les images IRM n'est donc peut-être pas la meilleure solution pour obtenir un réseau interprétable avec une prise de décision pertinente. Dans la suite, nous proposerons de contraindre le réseau lors de son apprentissage plutôt que de changer son entrée.

CHAPITRE

5

APPRENTISSAGE SOUS CONTRAINTE POUR UNE CLASSIFICATION INTERPRÉTABLE ET LA DÉTECTION D'ANOMALIES FAIBLEMENT SUPERVISÉE

Introduction

Dans le chapitre précédent, nous avons vu qu'il existait plusieurs biais dans les bases de données qui sont susceptibles d'être utilisés par un classifieur (notamment pour la classification d'images saines vs pathologiques) pour prendre sa décision plutôt que les signes radiologiques cliniquement pertinents. L'utilisation des cartes de probabilités d'appartenance aux tissus cérébraux comme entrée du réseau de classification améliore l'interprétabilité de ce dernier mais il subsiste des limites : cette forte normalisation peut supprimer de l'information pertinente et ne supprime malheureusement pas tous les biais. Pour aller plus loin, nous souhaitons donc, non plus modifier l'entrée pour éliminer les biais dans l'image, mais plutôt contraindre le réseau à s'affranchir de ces biais et à n'utiliser que l'information pertinente dans l'image à savoir les signes radiologiques de la pathologie. Nous avons vu dans les chapitres précédents que les attributions étaient un bon indicateur pour déterminer les zones de l'image d'entrée servant à la décision du réseau. Dans ce chapitre, nous proposons donc de les utiliser pendant l'entraînement pour contraindre le réseau et pas seulement à l'inférence pour expliquer la décision. Pour cela, nous contraignons, pendant l'entraînement, les attributions d'un réseau de classification d'IRM de sujets sains vs de patients atteints soit de tumeurs cérébrales soit de sclérose en plaques, de telle sorte que chaque voxel des images saines soit pertinent pour cette classe. L'ajout de cette contrainte permet de rendre la classification davantage basée sur la pathologie mais permet également la segmentation faiblement supervisée de la pathologie. Ces travaux ont été publiés dans le journal IEEE Transactions on Medical Imaging [Wargnier-Dauchelle *et al.*, 2023b] et présentés au Colloque Français d'Intelligence Artificielle en Imagerie Biomédicale (IABM) 2023 [Wargnier-Dauchelle *et al.*, 2023a] ainsi qu'à deux journées thématiques sur l'explicabilité des réseaux de neurones. Dans ce chapitre, nous commencerons par présenter les méthodes de l'état de l'art qui nous serviront de comparaisons. Puis nous détaillerons la méthode proposée pour contraindre le réseau. Enfin, nous présenterons les expériences et résultats obtenus lors de l'évaluation de notre méthode.

Contributions

Les contributions présentées dans ce chapitre sont :

- Une méthode non supervisée pour contraindre les attributions d'un classifieur à être négatives en dehors des zones pathologiques (et par conséquent positives à l'intérieur) par le biais d'une nouvelle fonction de coût.
- Avec seulement le label de l'image pour annotation, notre classifieur contraint atteint de bonnes performances de classification et de segmentation, surpassant les méthodes de détection d'anomalies et de classification interprétable de l'état de l'art.
- Une nouvelle intégration des contraintes sur les attributions dans l'apprentissage de sorte que le réseau résultant est invariant au choix de la méthode d'attributions (basée sur le gradient) utilisée lors de l'inférence.
- La preuve que contraindre avec le Gradient est, dans la plupart des cas, équivalent à une contrainte avec Expected Gradient, tout en permettant un apprentissage plus facile et plus rapide.

Ces travaux ont été publiés dans une revue international et présentés lors d'une conférence nationale et lors deux deux journées thématiques :

- IEEE Transactions on Medical Imaging [Wargnier-Dauchelle *et al.*, 2023b]
- Colloque Français d'Intelligence Artificielle en Imagerie Biomédicale (IABM)

2023 [Wargnier-Dauchelle *et al.*, 2023a]

— Journée GDR-ISIS sur l'IA et les réseaux de neurones profonds, ouvrir la boite noire : du modèle explicable à la synthèse et présentation d'explications en signal et image.

— Demi-journée de la Fédération d'Informatique de Lyon (FIL) sur l'explicabilité des modèles d'apprentissage automatique pour les signaux 2D/3D

I Travaux connexes

I.1 Classification interprétable

La plupart des méthodes d'attributions ont été proposées comme procédure pour visualiser, après entraînement, quels pixels contribuent positivement ou négativement à la décision du réseau. Ces informations sont particulièrement importantes lorsque l'on travaille sur des images médicales [Salahuddin *et al.*, 2021], une décision prises pour de mauvaises raisons pouvant alors avoir des conséquences particulièrement néfastes. Ces cartes d'attributions peuvent être seuillées pour réaliser une segmentation de la structure d'intérêt pour la classification. Ainsi dans [Selvaraju *et al.*, 2017], les fortes valeurs d'attributions pour la classe "chien" sont situées sur le chien, etc. On pourrait s'attendre à la même chose dans le domaine médical. Par exemple, classer des images saines vs pathologiques nous permettrait de segmenter la pathologie en question si la décision du réseau était basée sur cette dernière. Néanmoins, nous avons vu dans le chapitre précédent que ce n'est pas forcément le cas dans notre contexte. Il est donc nécessaire d'agir lors de l'entraînement du réseau pour le rendre interprétable et que sa décision soit basée sur des *aprioris* cliniques comme la présence d'une pathologie visible dans l'image.

Dans quelques travaux, les attributions ont également été utilisées comme régularisation pour l'entraînement d'un réseau de classification. Pour cela, le réseau est entraîné avec deux fonctions de coût : une pour la tâche de classification et une qui régularise les attributions calculées pendant l'entraînement.

Ainsi, dans [Erion *et al.*, 2021], on suppose que les pixels voisins de l'image d'entrée doivent avoir un impact similaire sur la décision. Pour modéliser cette contrainte, les auteurs ont choisi d'utiliser la variation totale anisotropique des attributions. Pour des images 2D, cette fonction de coût peut être écrite comme :

$$L_{TV} = \sum_{(i,j)\in N\times M} |EG_{i+1,j} - EG_{i,j}| + |EG_{i,j+1} - EG_{i,j}| \tag{5.1}$$

où $EG_{x,y}$ est la valeur carte d'attributions (dans [Erion *et al.*, 2021] Expected Gradient est utilisé) au pixel de position x, y et N et M sont l'ensemble des index des pixels pour les 2 dimensions spatiales.

Cette fonction régularise efficacement les cartes d'attributions qui, de ce fait, deviennent plus propres et plus lisses. Cependant, il n'y a aucun lien avec la tâche initiale du réseau comme la classification sain vs pathologique. Pour limiter le coût computationnel, ils ne calculent pas les intégrales sur l'image de référence et le paramètre d'intégral $alpha$ mais plutôt une somme discrète et une version stochastique d'Expected Gradient est proposée. Ainsi, à chaque itération, un tirage aléatoire uniforme est fait pour $\alpha \in [0, 1]$ et pour la référence $x' \in X$ et la carte d'attributions pour un réseau F est définie comme :

$$\nabla_x \mathrm{F}\Big(\alpha x + (1-\alpha)x'\Big) \tag{5.2}$$

Plusieurs valeurs de α et x' peuvent être utilisées par itération. Dans ce cas, l'erreur est calculée pour chacune de ces combinaisons de valeurs.

Dans [Ross *et al.*, 2017], l'idée est de rendre faibles les gradients par rapport à l'entrée dans les zones inintéressantes. En supposant qu'un masque de ces zones est disponible, la norme L_2 du gradient est pénalisée dans ce masque :

$$L_{L2} = \sum_{(i,j)\in N\times M} (A_{i,j}G_{i,j})^2 \tag{5.3}$$

où $G_{i,j}$ est le gradient de la sortie du réseau par rapport au pixel (i, j) de l'entrée et A est le masque donnant les zones sans intérêt.

Dans le contexte d'une classification sain/pathologique, la contrainte proposée par [Ross *et al.*, 2017] pourrait être adaptée pour forcer le réseau à être insensible à la région saine. Néanmoins, les voxels dans les régions saines ne devraient pas être neutres, ils devraient conduire la décision vers la classe "saine".

I.2 Détection d'anomalies

Les méthodes de l'état de l'art en détection d'anomalies, notamment médicales, reposent sur une tâche de reconstruction : le réseau est entraîné à reconstruire des images saines et les anomalies sont segmentées à l'inférence par seuillage de l'erreur de reconstruction pour les sujets pathologiques. Les AE [Baur *et al.*, 2018] et VAE [Zimmerer *et al.*, 2019] présentés enSection III.2 sont typiquement utilisés pour ce type d'approche. Néanmoins, l'image reconstruite est souvent floue. Cela produira de grandes erreurs de reconstruction dans les zones de changement de contraste. Ces zones n'étant pas forcément des anomalies, il peut être difficile de les différencier de ces fausses détections. La distribution des images saines peut également être apprise par des architectures de type GAN comme dans [Schlegl *et al.*, 2019]. Les images générées par ces architecture sont souvent de meilleure qualité que celles produites par les (V)AE. Cependant, il est bien connu que les GAN sont difficiles à entraîner. Il existe ainsi des problèmes de non-convergence ou de non-équilibre entre le générateur et le discriminateur qui peuvent conduire à un sur-apprentissage (*overfitting*) du générateur si le discriminateur est trop faible ou au contraire une evanescence du gradient dans le générateur si le discriminateur est trop fort. Les GAN sont aussi très sensibles aux choix des hyper-paramètres. Enfin, un problème récurrent est le mode "collapse" dans lequel le réseau ne génére qu'un nombre très limité d'images différentes.

En partant d'une méthode de reconstruction, un VAE, [Silva-Rodríguez *et al.*, 2021] propose d'ajouter une régularisation en utilisant GradCam [Selvaraju *et al.*, 2017]. L'attribution générée à l'inférence est également utilisée comme base de segmentation contrant les limites citées précédemment quant à la segmentation d'anomalies avec les VAE. Dans un premier temps, le VAE est appris classiquement avec une fonction de coût de reconstruction et une fonction de coût de régularisation sur la distribution de l'espace latent. Après un certain nombre d'*epochs*, s'ajoute une fonction de coût sur les attributions générées avec GradCam pendant l'entraînement. Ici GradCam est calculé en utilisant comme entrée la sortie du premier bloc convolutionnel (l'architecture de l'encodeur est basée sur les convolutions d'un ResNet18) et comme sortie l'espace latent correspondant à la moyenne. Cette carte est ensuite normalisée grâce à une sigmoïde. L'objectif est ici de maximiser les valeurs de l'attribution sur toute l'image (saine). Pour cela, la fonction de coût suivante est utilisée pour maximiser (à $1 - m$) la moyenne de l'attribution :

$$L_S = 1 - m - \frac{1}{|N|}\sum_{n\in N} GC_n \tag{5.4}$$

où N est l'ensemble des pixels de la carte d'attributions calculée avec GradCam GC et m permet de définir une marge de tolérance.

L'objectif ici est de converger vers $L_S \leq 0$. Pour une meilleure convergence, la fonction barrière logarithmique proposée par [Kervadec *et al.*, 2022] pour optimiser un problème sous contrainte d'inégalité, est utilisée :

$$\psi_t(L_S) = \begin{cases} \frac{1}{t} log(-L_S) \text{ si } L_S \leq -\frac{1}{t^2} \\ tL_S - \frac{1}{t} log(\frac{1}{t^2}) + \frac{1}{t} \text{ sinon} \end{cases} \tag{5.5}$$

où t contrôle cette barrière (fixé à 20 dans ce papier).

A l'inférence, les mêmes cartes GradCam des images pathologiques sont seuillées pour obtenir la segmentation. Puisque GradCam est utilisée sur le premier bloc convolutionnel plutôt que sur des convolutions en fin de réseau, les cartes générées ont peu de sémantique selon le papier original [Selvaraju *et al.*, 2017]. Récemment, une version étendue a été proposée [Silva-Rodríguez *et al.*, 2022]. Dans cette version, les attributions de GradCAM sont remplacées par la moyenne sur les canaux de la carte des caractéristiques à la sortie du premier bloc convolutionnel montrant que la pondération par le gradient est inutile. Le manque de sémantique est d'autant plus présent. La fonction de coût est également remplacée car la carte des caractéristiques n'est pas normalisée et on ne peut donc pas imposer une valeur comme précedemment. La carte est ici normalisée avec un softmax et l'objectif est de rendre cette carte homogène. Il faut donc que la carte soit similaire à une distribution uniforme. Cela peut être implémenté en utilisant une divergence de Kullback-Leibler, elle-même équivalente à une entropie. Les auteurs ont donc choisi d'implémenter la fonction de coût $L_H = -H(C)$ (où H est l'entropie de Shannon et C la carte normalisée des caractéristiques) comme régularisation des cartes d'activations. Dans ce chapitre, les comparaisons avec la méthode proposée seront faites avec la première version [Silva-Rodríguez *et al.*, 2021].

Par rapport aux méthodes de détection d'anomalies, qui ne sont entraînées que sur des images saines, les méthodes de classification interprétables décrites dans la Section I.1 ont besoin d'images pathologiques et sont faiblement supervisées : il est nécessaire d'avoir le label de l'image (sain/pathologique). Néanmoins, des bases de données pathologiques sont souvent disponibles et il serait intéressant de prendre en compte l'information concernant la pathologie considérée dans les caractéristiques du réseau. Plus précisément, les attributions des méthodes de classification sont basées sur la différenciation sain/pathologique alors que dans [Silva-Rodríguez *et al.*, 2021, Silva-Rodríguez *et al.*, 2022], les attributions sont basées sur une tâche de reconstruction, sans aucune information sur la pathologie. De plus, les méthodes de classification interprétables peuvent être utilisées pour deux tâches : la classification et la segmentation.

II Ajout d'une contrainte non supervisée à un classifieur

II.1 Apprentissage d'une classification sous contrainte sur les attributions

Nous proposons d'entraîner un réseau profond à classer des images de sujets sains vs pathologiques avec la contrainte supplémentaire que sa décision soit en accord avec les *aprioris* médicaux, c'est-à-dire que pour classer une image comme pathologique, la décision doit être prise par rapport à un signe radiologique de cette pathologie. La méthode est illustrée Figure 5.1. Nous supposons que la décision du réseau, pour une image d'entrée donnée, est reflétée par une carte d'attributions de la même taille que l'image d'entrée. Pendant l'entraînement, le réseau apprend à correctement classer les images mais aussi à satisfaire une contrainte sur les carte d'attributions produites par le réseau. Pour ce faire, il minimise la fonction de coût suivante :

$$L = L_C + \alpha_A L_A \tag{5.6}$$

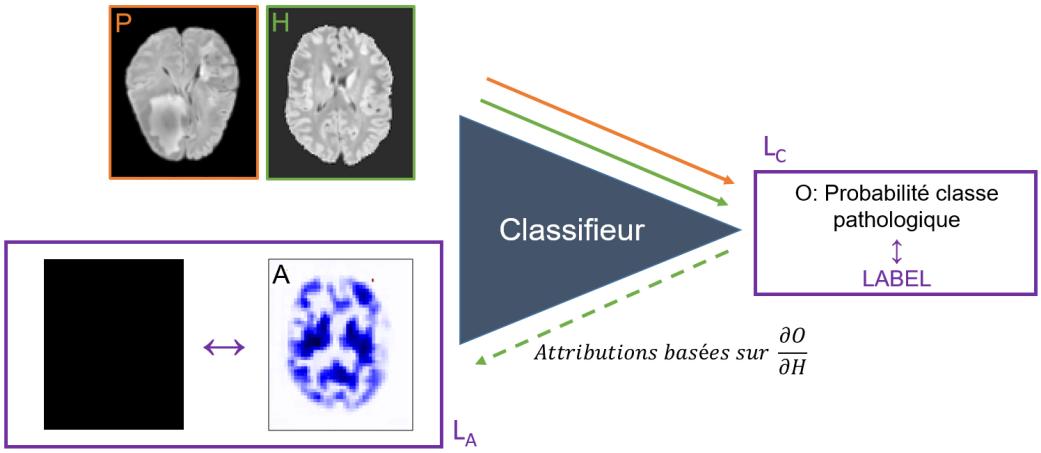


FIGURE 5.1 – *Schéma de la méthode en mode faiblement supervisé. Le réseau de classification est entraîné avec la fonction de coût de classification L_C sur des images pathologiques (P, en orange) et saines (H, en vert). La fonction de coût L_A, qui contraint les attributions basées sur le gradient L_A, n'est appliquée que sur les images saines. A l'inférence, la segmentation des structures pathologiques est obtenue par seuillage des attributions.*

où L_C est la fonction de coût de classification, L_A est la fonction de coût visant à pénaliser les contraintes non satisfaites sur les cartes d'attributions avec un coefficient de pénalisation α_A. La fonction de coût utilisée pour la classification est l'entropie croisée binaire. L_A est détaillée dans la Section II.2.

Une fois entraîné, ce réseau peut être utilisé comme un classifieur interprétable, c'est-à-dire dont la décision est davantage basée sur la pathologie ou pour segmenter la pathologie. Pour cela, la carte d'attributions de l'image passée en entrée du réseau est calculée puis seuillée pour obtenir un masque de segmentation.

II.2 Contrainte sur les attributions avec une entropie croisée

Les cartes d'attributions visent à révéler les régions de l'image d'entrée qui contribuent positivement ou négativement à la décision du réseau. Pour les images de sujets sains, aucune région ne doit conduire le réseau à une classification pathologique, mais chaque région doit conduire la décision vers la classe saine. Ainsi, les cartes d'attributions des sujets sains doivent être négatives sur l'ensemble de l'image. Nous avons choisi d'utiliser l'entropie croisée binaire comme moyen efficace d'imposer cette contrainte. Avec cette fonction de coût, les valeurs d'attributions sont considérées comme les logits d'un classificateur virtuel pixel à pixel. Pour ce classifieur virtuel, les voxels provenant d'images de sujets sains devraient tous être classés comme sains, c'est-à-dire avec un logit négatif alors que pour les images pathologiques, les voxels des parties saines devraient être négatifs et ceux des parties pathologiques devraient être positifs. Ici, nous souhaitons une contrainte non supervisée. Or pour les images saines, aucune donnée supplémentaire n'est nécessaire pour utiliser cette contrainte. La seule annotation nécessaire est l'étiquette au niveau de l'image (sain/pathologique) déjà requise pour la classification. Par conséquent, notre perte L_A peut être utilisée sur toutes les images de sujets sains de manière non supervisée. On peut imposer la négativité des logits de manière efficace avec une entropie croisée de la même manière que pour une classification. La fonction de coût utilisée sur les sujets sains est donc :

$$L_A(x) = BCE\big(\sigma(A(x)), 0_{Size(A(x))}\big) \tag{5.7}$$

où σ est la fonction sigmoïde, BCE est la fonction de coût d'entropie croisée binaire, x est l'image d'entrée, $A(x)$ est la carte d'attributions pour l'image x (même taille que x) et

$0_{Size(A(x))}$ est une image entièrement nulle de même taille que $A(x)$.

Bien que notre contrainte soit conçue pour être utilisée pour un apprentissage faiblement supervisé, elle peut facilement être étendue à l'apprentissage semi-supervisé. Dans ce cas, on suppose qu'une carte de pathologie $m(x)$, indiquant où les attributions positives devraient se trouver, est disponible pour certaines des images x de patients. Cette carte pathologique peut simplement être un masque de segmentation de la région d'intérêt pour la classification pathologique. Dans ce cas, la perte L_A devient :

$$L_A(x) = BCE(\sigma(A(x)), m(x)). \tag{5.8}$$

Apprendre ce classifieur avec la contrainte L_A est donc équivalent à apprendre un encodeur-décodeur très contraint dont la sortie serait une segmentation et l'espace latent serait la classification. Dans ce cas, l'encodeur serait le classifieur (flèches continues dans la Figure 5.1) alors que le décodeur serait paramétré avec les poids de l'encodeur et permettrait le calcul des attributions (flèche en pointillée).

Pour les attributions, nous avons décidé d'utiliser les méthodes basées sur le gradient : le gradient de la sortie par rapport à l'image (G), Integrated Gradient (IG) ou encore Expected Gradient. Ces méthodes présentent un intérêt pour l'interprétabilité en reflétant l'impact qu'une modification de l'entrée aura sur la sortie mais aussi un intérêt pour l'incorporation dans la procédure d'apprentissage. En effet, elles peuvent être facilement calculées avec une rétropropagation mais aussi dérivées pour le calcul du gradient de la fonction de coût. Nous avons choisi une approche locale en ne multipliant pas le gradient par l'entrée. Les attributions locales visent à expliquer comment l'entrée doit être modifiée afin d'obtenir une variation de la sortie. Elles correspondent parfaitement à notre problématique de passage d'un voxel sain à un voxel pathologique et vice-versa qui nous permettrait d'identifier les zones pathologiques sur les attributions.

Dans la suite, nous allons analyser le choix de la méthode d'attribution.

III Choix de la méthode d'attributions pour la contrainte

III.1 Equivalence entre une contrainte sur le gradient ou sur Expected Gradient

Même avec sa version stochastique, l'apprentissage avec Expected Gradient nécessite au moins un deuxième passage dans le réseau et une deuxième rétropropagtaion du gradient puisque le *minibatch* utilisé est différent de celui utilisé pour la tâche de classification. En effet, pour la classification, l'entrée $x \in X$, une image de la base d'apprentissage, est utilisée alors qu'Expected Gradient nécessite d'utiliser $x' + \alpha(x - x')$ où $x' \in X$ est également une image de la base d'apprentissage et α a été tiré aléatoirement dans $[0, 1]$. En comparaison, l'apprentissage avec le gradient ne nécessite qu'une double rétropropagation sur le même *minibatch* que celui utilisé par la fonction de coût de classification, réutilisant efficacement le calcul de la dérivée pour la descente du gradient. Par conséquent, le calcul du gradient est plus facile à implémenter et est moins coûteux en temps de calcul et en mémoire GPU qu'Expected Gradient. Il serait plus avantageux d'utiliser le gradient pour la contrainte sur les attributions si les résultats produits à l'inférence sont similaires. Nous défendons donc la conjecture suivante :

Conjecture 1. *Deux modèles entraînés soit avec un contrainte sur le gradient soit sur Expected Gradient produisent des cartes d'attributions de type Gradient ou Expected Gradient équivalentes à l'inférence.*

Pour étayer cette conjecture, définissons $LS(X)$:

Définition 2. *Si X est un sous-ensemble d'un espace vectoriel, l'ensemble des segments de droite de X peut s'exprimer comme suit :*

$$LS(X) = \left\{\alpha x + (1-\alpha)x' \mid (x, x', \alpha) \in X^2 \times [0,1]\right\}.$$

$LS(X)$ est donc l'ensemble des points appartenant aux segments dont les extrémités sont dans X. Autrement dit, il s'agit de l'ensemble des images obtenues par interpolation linéaire entre deux images de X.

L'enveloppe convexe de X peut être définie comme :

Définition 3. *Si X est un sous-ensemble d'un espace vectoriel, les éléments de l'enveloppe convexe $C(X)$ de X sont exactement les points qu'on peut écrire sous la forme $\sum_{i=0}^{p} \lambda_i x_i$ où p est un entier, $x_i \in X$ et $\sum_{i=0}^{p} \lambda_i = 1$ avec $\forall i, \lambda_i > 0$.*

On peut donc noter que $LS(X) \subset C(X)$ car $LS(X)$ est un sous-ensemble des combinaisons convexes de points de X ($\Lambda = \{\alpha, 1-\alpha\}$).

Nous pouvons montrer que :

Proposition 4. *L'apprentissage stochastique sous contrainte avec Expected Gradient sur X proposé dans [Erion et al., 2021] est équivalent à l'apprentissage stochastique sous contrainte avec le gradient sur $LS(X)$.*

Démonstration. Formellement, la contrainte sur les cartes d'attributions de type Expected Gradient consiste à ajouter à la fonction de coût de classification le terme suivant :

$$\sum_{x \in X} L\left(\sum_{x' \in X, \alpha \in [0,1]} \nabla_x \mathrm{F}\Big(\alpha x + (1-\alpha)x'\Big)\right) \tag{5.9}$$

où F est le réseau, x est l'image d'entrée, x' est l'image de référence, X est l'ensemble de données d'apprentissage et L est une fonction de coût utilisée pour contraindre la carte d'attribution. Dans la version stochastique proposée dans [Erion *et al.*, 2021], à chaque itération, les deux sommes sont supprimées et le terme

$$L\left(\nabla_x \mathrm{F}\Big(\alpha x + (1-\alpha)x'\Big)\right) \tag{5.10}$$

est ajouté à l'erreur de classification pour un *minibatch* de x, x' et α. La contrainte globale est donc :

$$\sum_{(x, x', \alpha) \in X^2 \times [0,1]} L\Big(\nabla_x \mathrm{F}\big(\alpha x + (1-\alpha)x'\big)\Big) \tag{5.11}$$

Ainsi, l'entraînement avec la contrainte stochastique EG sur X est équivalent à l'utilisation de la même contrainte à l'aide des cartes d'attributions G sur $LS(X)$. □

En utilisant la Proposition 4, la Conjecture 1 pourrait être prouvée si nous pouvions montrer que l'apprentissage contraint sur X ou $LS(X)$ (ou $C(X)$ car $LS(X) \subset C(X)$) a le même effet lorsque le modèle est appliqué aux données de test. Il est raisonnable de supposer que c'est le cas. En effet, pour des espaces de grande dimension d, chaque point d'un ensemble de N points sera en dehors de l'enveloppe convexe des autres points avec une probabilité proche de 1 (sauf si N croît exponentiellement avec d). Cela a été prouvé pour des points sur une hypersphère dans [Bárány et Füredi, 1988]. Cela a également été validé de manière heuristique pour des données réelles dans [Balestriero *et al.*, 2021, Yousefzadeh,

2021, Yousefzadeh, 2022] : pour des bases de données comme MNIST, CIFAR10 ou ImageNet, les images de l'ensemble de test sont en dehors de l'enveloppe convexe de l'ensemble d'apprentissage. Cela a conduit [Balestriero *et al.*, 2021] à la conclusion que le comportement d'un modèle dans l'enveloppe convexe d'un ensemble d'apprentissage a peu d'impact sur la performance de généralisation de ce modèle.

En ce qui concerne les contraintes d'attributions, notre intuition est que contraindre le gradient sur l'ensemble d'apprentissage X uniquement, et non comme EG sur $LS(X)$, est suffisant pour générer de bonnes cartes EG lors de l'inférence. Cela repose notamment sur le fait qu'une image dans l'ensemble de test a très peu de chances d'être le résultat d'une interpolation linéaire de deux images de l'ensemble d'apprentissage. Les résultats des expériences présentés en Section V.2 semblent confirmer cette conjecture sur des données réelles.

Le fait de n'utiliser que G pendant l'apprentissage permet une mise en œuvre plus simple, un chemin plus direct lors de la rétropropagation et un temps d'apprentissage plus court.

III.2 Une nouvelle contrainte robuste (IEG)

Dans la section précédente, nous avons soutenu que la contrainte sur le gradient devrait être suffisante pour une bonne inférence avec Expected Gradient. Bien qu'Integrated soit également basé sur le gradient, l'entraînement avec le gradient seulement pourrait ne pas être suffisant pour obtenir des cartes similaires à celle obtenues en contraignant les cartes d'Integrated Gradient. En effet, la référence utilisée (une image nulle) est toujours la même et en dehors de l'enveloppe convexe de l'ensemble d'apprentissage. En apprenant avec le gradient ou avec Expected Gradient, cette référence n'aura jamais été vue par le modèle lors de l'apprentissage alors que cela sera le cas si on entraîne avec Integrated Gradient. Si Integrated Gradient est utilisé à l'inférence, il y aura probablement une forte différence entre le modèle appris en ayant vu cette référence et un modèle ne l'ayant jamais vu. De la même manière, utiliser les cartes d'Integrated Gradient pour l'apprentissage pour une inférence avec Expected Gradient n'apporte rien car l'image nulle utilisée comme référence à l'apprentissage ne sera pas dans la distribution utilisée par Expected Gradient, à savoir celle des images de test. Si l'objectif n'est pas d'être plus rapide pendant l'apprentissage, mais de mettre en œuvre une contrainte robuste au choix de la méthode d'attributions utilisée à l'inférence, nous proposons l'usage d'EG avec une probabilité p_0 d'utiliser une référence nulle plutôt qu'une image de la base d'entraînement (c'est-à-dire d'utiliser IG) pendant l'apprentissage. Par la suite, cette proposition sera dénommée IEG. On peut noter que le début des chemins d'intégrations ($\alpha \approx 0$) pour IG et EG correspond à G, l'entraînement avec des contraintes sur IG et EG contraindra également les attributions de G.

Cette proposition devrait donc améliorer l'invariance du réseau par rapport au choix de la méthode d'attributions utilisée à l'inférence.

IV Protocole expérimental

IV.1 Comparaisons

Nous avons comparé plusieurs méthodes de l'état de l'art. En classification interprétable, une comparaison a été faite entre :

— **Unsup :** Un modèle appris avec notre méthode non supervisée basée sur l'Equation 5.7.

— **Sup :** Un modèle appris avec notre méthode supervisée pour laquelle $m(x)$ dans l'Equation 5.8 est disponible pour certaines images pathologiques pendant l'entraînement.

Cette méthode nous sert de référence pour ce qu'il est possible d'atteindre au mieux, avec une information plus forte.

— **UnsupTV :** Un modèle appris avec notre contrainte non supervisée et la contrainte sur la variation totale utilisée dans [Erion *et al.*, 2021].

— **Erion :** Un modèle appris avec la contrainte proposée par [Erion *et al.*, 2021].

— **Ross :** Un modèle appris avec la contrainte proposée par [Ross *et al.*, 2017] pour laquelle nous avons extrapolé que la zone sans intérêt citée dans le papier correspondait, dans notre cas, aux images saines.

— **NoConsG** : Un modèle appris sans contrainte sur les attributions et évalué avec le gradient.

— **NoConsGC** : Un modèle appris sans contrainte sur les attributions et évalué avec GradCam [Selvaraju *et al.*, 2017]. Nous n'avons pas appliqué l'activation ReLU dans la formulation de GradCam afin de visualiser les activations négatives et positives.

Lors de la comparaison à l'état de l'art, nous avons utilisé des contraintes sur le gradient sauf pour Erion qui utilise Expected Gradient. Le gradient a été utilisé à l'inférence car il est intégré dans chacune des méthodes d'attributions.

Pour étudier le choix de la méthode d'attributions, les modèles contraints cités précédemment ont été entraînés avec G, EG, IG ou IEG puis évalués en utilisant G, EG ou IG.

Pour la détection d'anomalies, nous avons comparé notre méthode avec :

— **Silva-Rodríguez :** La méthode proposée par [Silva-Rodríguez *et al.*, 2021].

— **AE :** La détection d'anomalies en utilisant un auto-encodeur [Baur *et al.*, 2018].

— **VAE :** La détection d'anomalies en utilisant un auto-encodeur variationnel [Zimmerer *et al.*, 2019].

— **f-AnoGAN :** La méthode proposée par [Schlegl *et al.*, 2019].

Pour ces dernières méthodes, nous avons utilisé l'implémentation de [Silva-Rodríguez *et al.*, 2021].

IV.2 Données

Pour nos expériences, nous avons utilisé des IRM cérébrales FLAIR de sujets sains et de patients atteints soit de tumeurs cérébrales soit de sclérose en plaques. Les bases de données utilisées ainsi que la répartition entre les jeux d'entraînement (N_{train}), de validation (N_{val}) et de test (N_{test}) sont indiquées dans Tableau 5.1.

Tableau 5.1 – *Bases de données utilisées. H correspond aux bases de sujets sains, T aux bases de patients avec tumeurs et MS aux bases de patients atteints de sclérose en plaques. La dernière colonne précise si nous disposons des masques de segmentation.*

Base	N_{train}	N_{val}	N_{test}	H/T/MS	Annotation
MPI	64	15	15	H	Non
kirby21	22	5	5	H	Non
IBC	8	2	2	H	Non
BraTS 2020	280	40	49	T	Oui
BraTS 2019 (2D)	2710	314	319	T	Oui
MSSEG	12	3	37	MS	Oui
OFSEP1	401	50	50	MS	Non

La base BraTS19 a été utilisée comme dans [Silva-Rodríguez *et al.*, 2021]. Ainsi, uniquement les 10 coupes centrales de chaque volume IRM ont été utilisées. Parmi ces coupes, celles sans tumeurs ont constitué la base de données de sujets sains alors que les coupes avec plus de 0.1% de volume tumoral ont constitué la base de patients avec tumeur cérébrale. Les images ont été ajustées pour obtenir une correspondance des histogrammes cumulés par rapport à une image de référence choisie aléatoirement parmi les coupes saines. Elles ont également été redimensionnées pour obtenir une image 224×224. Sur cette base, le réseau a été entraîné en 2D.

Les autres bases ont été utilisées en 3D en utilisant les bases de sujets sains pour la classe saine et les bases pathologiques pour la classe pathologique. La chaîne de prétraitements décrite en Section III.2 du Chapitre 1 a été utilisée en effectuant un recalage affine sur l'atlas du MNI, l'extraction du cerveau et la correction des inhomogénéités de champ. Pour les tumeurs cérébrales, deux versions des données ont été testées : avec ou sans cette correction des inhomogénéités pour simuler ou non un mauvais contraste dans les images. Deux résolutions d'images ont été testées : des voxels de 1mm^3 ou 2mm^3 donnant respectivement des images de taille $182 \times 218 \times 182$ ou $91 \times 109 \times 91$.

Pour le modèle supervisé sur la sclérose en plaques, les deux bases de données pour cette pathologie ont été utilisées mais seulement la base MSSEG a été utilisée pour la contrainte supervisée des attributions.

IV.3 Implémentation

Une architecture PatchGAN 3D ($70 \times 70 \times 70$) a été utilisée pour le classifieur. Notre méthode a été optimisée en utilisant l'optimiseur Yogi [Zaheer *et al.*, 2018] avec un *learning rate* initial de 1 et AMSGrad [Tan *et al.*, 2019] rendu robuste via l'utilisation d'une distribution de Student pour le *momentum* [Ilboudo *et al.*, 2022]. Pour le classifieur sans contrainte, nous avons utilisé Adam [Kingma et Ba, 2014] avec un *learning rate* initial de 10^{-3}. Ces choix ont été fait pour garantir la stabilité et la convergence des modèles. Les méthodes de l'état de l'art ont été utilisées avec les optimiseurs originaux.

Pour les volumes 3D, de l'augmentation de données a été ajoutée avec des déformations élastiques, des variations de luminosité et en retournant l'image autour du plan sagittal. Pour les images 2D, comme dans [Silva-Rodríguez *et al.*, 2021], les modèles ont été entraînés sans augmentation de données.

Nous avons utilisé une seule image de référence et un seul α pour le calcul de EG et une référence nulle pour IG. Pour IEG, la probabilité d'utiliser une référence nulle a été fixée à $p_0 = 0.25$. Le choix de ce paramètre n'a pas de grande influence.

Pour notre méthode, le coefficient de pondération de la contrainte sur les attributions a été fixé à $\alpha_A = 10^8$. Une analyse de l'influence de cet hyper-paramètre est réalisée Section V.1. Pour les méthodes de l'état de l'art, les coefficients originaux ont été utilisés.

Enfin, un arrêt prématuré de l'entraînement a été fait sur les méthodes de classification en choisissant le modèle avec les meilleures performances de classification sur le jeu de validation. Pour les méthodes de détection d'anomalies, les modèles ont été entraînés sur 300 *epochs*.

IV.4 Métriques

Pour évaluer les performances de classification au niveau de l'image, nous avons utilisé les taux de vrais négatifs/sains (TNR) et le taux de vrais positifs/pathologiques (TPR).

Pour évaluer les performances de segmentation, nous avons utilisé le Dice sur les attributions seuillées, l'aire sous la courbe ROC (AUROC), la même aire en se limitant à 10% de faux positifs (AUROC10) et l'aire sous la courbe de précision/rappel (AUPRC). Les seuils

ont été choisis de manière optimale en utilisant le point opérationnel de la courbe précision/rappel sur le jeu de validation des images de tumeurs avec des voxels de $2mm^3$ ou le jeu de validation de la base 2D.

Pour calculer la corrélation entre les cartes d'attributions, nous avons utilisé le coefficient de Pearson défini pour deux cartes X et Y comme :

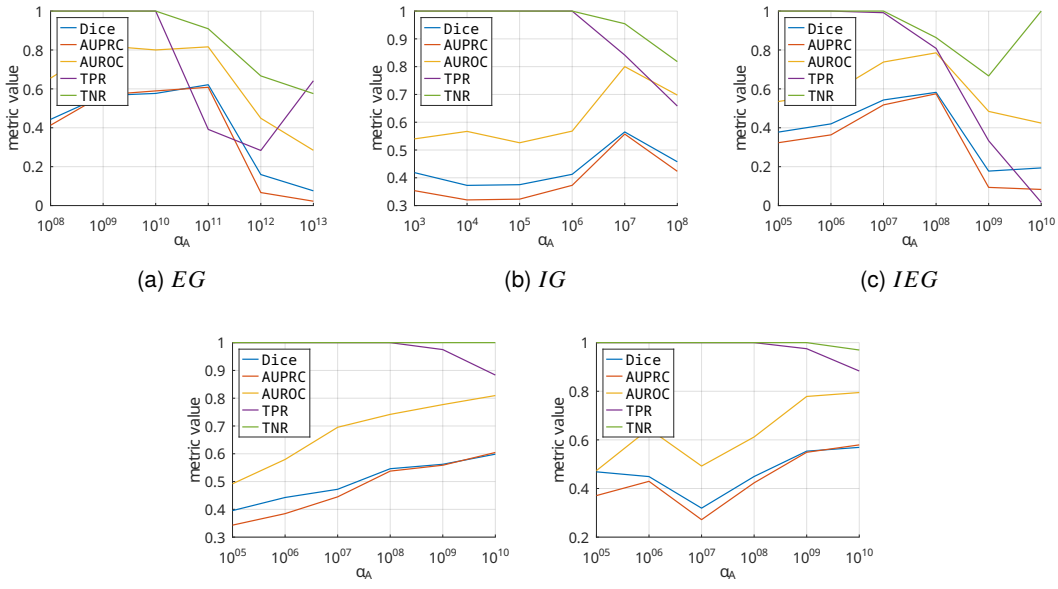
$$\text{corr}(X,Y) = \frac{\text{Cov}(X,Y)}{\sigma_X \sigma_Y} \tag{5.12}$$

où σ est l'écart type et Cov la covariance.

Les tests statistiques ont été effectués en comparant Unsup aux autres méthodes et en utilisant un test de permutations avec 10000 permutations et un seuil de confiance de 95%. Une correction de Bonferroni a été appliquée pour corriger le seuil de significativité des comparaisons multiples effectuées.

V Résultats

V.1 Coefficient de pondération de la perte



(a) EG (b) IG (c) IEG

(d) G 2mm (e) G 1mm

FIGURE 5.2 – *Influence du coefficient de pondération du coût α_A. Métriques (Dice, AUROC, AUPRC, TPR, TNR) en fonction de la valeur α_A pour les différentes méthodes d'attributions. Les expériences ont été conduites sur le jeu de validation de BraTS 2020 (sans correction N4) avec des voxels de $2mm^3$ (+$1mm^3$ pour le gradient).*

Le coefficient α_A, introduit dans l'Equation 5.6 et pondérant les fonctions de coût de classification et de contrainte sur les attributions l'une par rapport à l'autre, a été choisi de manière optimale sur le jeu de validation pour la base de données BraTS 2020 sans correction des inhomogénéités de champ. Pour cela, nous avons entraîné le modèle non supervisé proposé avec différentes valeurs pour ce coefficient puis nous avons évalué ses performances. Nous avons utilisé les métriques précédemment décrites pour la segmentation de la pathologie (Dice, AUPRC et AUROC) et la classification des images (TNR et TPR). Les résultats sont présentés en Figure 5.2.

On peut voir que les métriques de segmentation augmentent avec α_A et que les performances de classification sont maintenues jusqu'à une certaine valeur de α_A. Nous avons choisi le coefficient maximal qui conservait une classification presque parfaite (TPR et TNR proches de 1) pour chaque méthode d'attributions. Avec ce choix, nous voulons obtenir un bon modèle de classification et de segmentation. Néanmoins, les performances de segmentation peuvent être améliorées en choisissant un coefficient plus élevé au détriment des performances de classification. On peut remarquer que le volume du pixel (et donc la taille de l'image) n'a pas d'impact sur le coefficient optimal puisque les courbes TPR commencent à chuter pour le même α_A pour des images avec des voxels de $2mm^3$ et de $1mm^3$ (Figure 5.2d et 5.2e). Nous avons donc fixé $\alpha_A = 10^{10}$ pour EG, $\alpha_A = 10^6$ pour IG, $\alpha_A = 10^7$ pour IEG et $\alpha_A = 10^8$ pour G pour les modèles appris avec notre contrainte soit de manière supervisée soit de manière non supervisée. Nous avons dû réduire le coefficient du modèle supervisé (Sup) et le modèle où la variation totale a été ajoutée (UnsupTV) entraîné avec EG à 10^9. En effet, dans ces conditions et avec le coefficient optimal choisi sur le modèle non supervisé, le modèle ne converge pas. L'apprentissage avec EG semble donc moins stable et plus dépendant des hyperparamètres qu'avec les autres attributions. Les coefficients optimaux α_A décrits ici ont été utilisés pour les autres expériences.

V.2 Équivalence entre le Gradient et Expected Gradient

Tableau 5.2 – *Équivalence G/EG. Coefficient de Pearson entre les cartes d'attributions A1 et A2 données dans les deux premières colonnes pour différentes contraintes. La base de données tumorales en $2mm^3$ a été utilisée. Dans la notation X_Y, X est l'attribution utilisée pour la contrainte et Y celle utilisée pour l'inférence.*

A1	A2	Sup	Unsup	UnsupTV	Ross	Erion	Moyenne
EG_EG	NO_EG	0.09 ± 0.04	0.46 ± 0.07	0.45 ± 0.06	0.48 ± 0.08	0.30 ± 0.09	0.36 ± 0.07
	IG_EG	0.36 ± 0.09	0.59 ± 0.10	0.57 ± 0.07	0.83 ± 0.03	**0.54 ± 0.07**	0.58 ± 0.07
	G_EG	**0.82 ± 0.06**	**0.81 ± 0.06**	**0.65 ± 0.09**	**0.93 ± 0.02**	0.40 ± 0.06	**0.72 ± 0.06**
IG_IG	G_IG	0.64 ± 0.10	0.44 ± 0.14	0.64 ± 0.13	0.58 ± 0.17	0.50 ± 0.05	0.56 ± 0.12

Dans cette section, nous allons donner des éléments quantitatifs pour montrer qu'une contrainte sur le gradient pendant l'apprentissage est suffisante pour obtenir de bonnes cartes d'attributions de type Expected Gradient à l'inférence.

Dans le Tableau 5.2, nous indiquons la corrélation entre les cartes d'attributions à l'inférence lorsque différentes cartes d'attributions ont été utilisées pour la contrainte au cours de l'apprentissage et pour les différentes cartes d'attributions utilisées à l'inférence. L'expérience a été réalisée avec plusieurs contraintes. On constate tout d'abord que les cartes EG produites à l'inférence à la suite d'un entraînement avec une contrainte sur EG (EG_EG) et pour le modèle non contraint (NO_EG) ne sont pas corrélées. L'ajout d'une contrainte, que ce soit avec IG ou G, augmente considérablement la corrélation : entre 12% et 73%. On peut également constater que la corrélation est plus élevée pour les cartes où G (G_EG) a été utilisé pour la contrainte que pour celles où IG (IG_EG) a été utilisé dans la plupart des cas. En effet, la corrélation augmente en moyenne de 14%.

L'entraînement sous contrainte avec la carte G est également bénéfique en ce qui concerne l'adéquation avec la pathologie. Si l'on considère le Dice entre la vérité terrain et les attributions positives (présenté dans la Figure 5.3) et l'AUPRC (présenté dans la Figure 5.4), on constate que les résultats pour les deux métriques sont équivalents ou améliorés lorsque la contrainte est appliquée sur G au lieu de EG pendant l'apprentissage.

En ce qui concerne le temps de calcul, EG est environ 50% plus lent que G à chaque itération et la convergence avec G est généralement plus facile à obtenir.

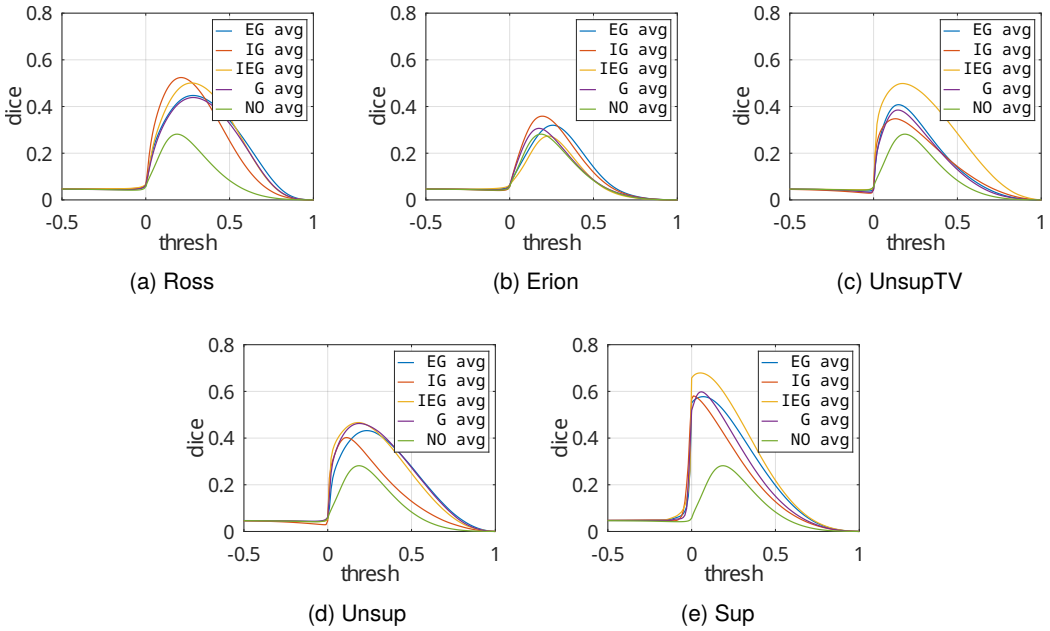


FIGURE 5.3 – *Equivalence G/EG. Courbes du Dice en fonction du seuil (thresh) pour différentes contraintes sur les images de tumeurs en $2mm^3$. Le Dice est moyenné sur toutes les méthodes d'attributions à l'inférence.*

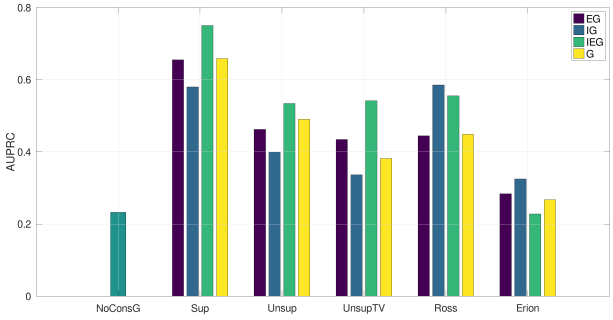


FIGURE 5.4 – *Equivalence G/EG. Influence de la méthode d'attributions utilisée pour la contrainte sur l'AUPRC pour les images de tumeurs en $2mm^3$. Les barres sont les AUPRC moyennées sur toutes les méthodes d'attributions à l'inférence.*

La dernière ligne du Tableau 5.2 montre que les cartes après contrainte sur G sont moins corrélées pour IG qu'elles ne le sont pour EG. Ainsi, pour la plupart des cas testés, la corrélation est plus faible entre IG_IG et G_IG qu'entre EG_EG et G_EG. La contrainte sur G n'est donc, a priori, pas suffisante pour assurer une robustesse à l'utilisation d'IG lors de l'inférence, mais elle est suffisante pour obtenir de bonnes performances à l'inférence avec EG tout en proposant un apprentissage plus facile et rapide.

V.3 Robustesse d'IEG

L'entraînement avec un mélange de EG et IG (nommé IEG) est meilleur que le Gradient en termes de Dice (courbe jaune dans la Figure 5.3) et d'AUPRC (barre verte dans la Figure 5.4). C'est la méthode la plus efficace en moyenne : avec cet entraînement, les performances (Dice et AUPRC) restent constantes selon le choix de la contrainte (Sup, Unsup, UnsupTV, Erion ou Ross) alors que EG ou IG atteignent de très bonnes performances pour

Tableau 5.3 – *Robustesse d'IEG. Coefficient de Pearson entre les cartes d'attributions A1 et A2 données dans les deux premières colonnes pour différentes contraintes. La base de données tumorales en 2mm^3 a été utilisée. Dans la notation X_Y, X est l'attribution utilisée pour la contrainte et Y celle utilisée pour l'inférence.*

A1	A2	Sup	Unsup	UnsupTV	Ross	Erion	Moyenne
EG_EG	IG_EG	0.36 ± 0.09	0.59 ± 0.19	0.57 ± 0.07	0.83 ± 0.03	**0.54 ± 0.07**	0.58 ± 0.09
	IEG_EG	**0.95 ± 0.03**	**0.86 ± 0.07**	**0.75 ± 0.07**	**0.87 ± 0.03**	0.27 ± 0.07	**0.74 ± 0.05**
IG_IG	EG_IG	0.36 ± 0.12	0.43 ± 0.09	0.52 ± 0.10	0.63 ± 0.23	**0.48 ± 0.10**	0.48 ± 0.13
	IEG_IG	**0.66 ± 0.09**	**0.67 ± 0.05**	**0.61 ± 0.09**	**0.79 ± 0.09**	0.25 ± 0.08	**0.59 ± 0.08**

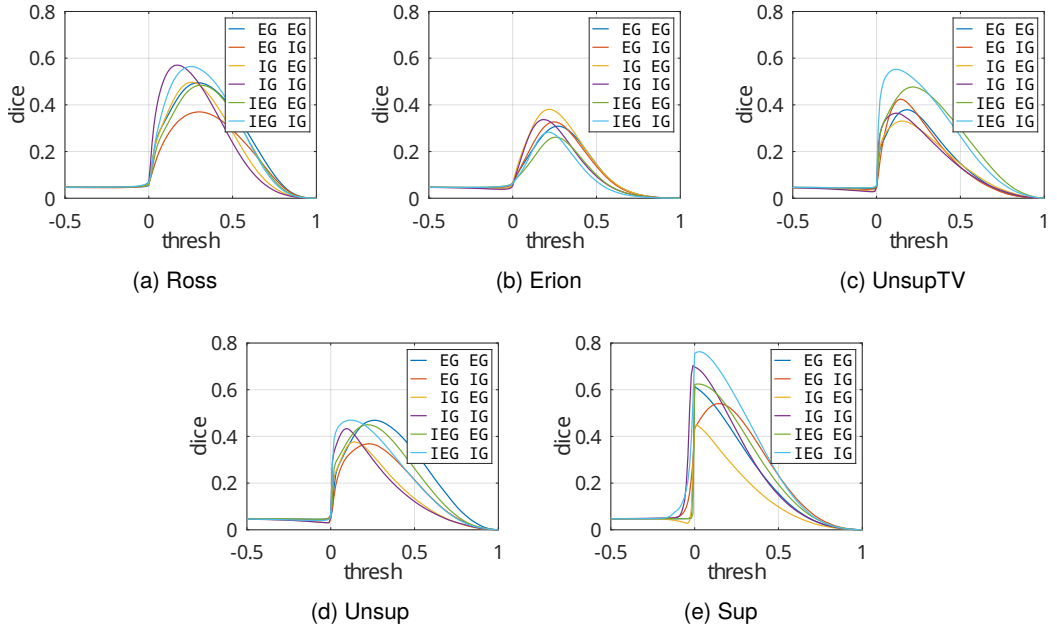


Figure 5.5 – *Robustesse d'IEG. Dice en fonction du seuil (thresh) pour différentes contraintes sur les images de tumeurs en 2mm^3. Dans la légende, la méthode de gauche est celle de l'entraînement et celle de droite celle utilisée à l'inférence.*

.

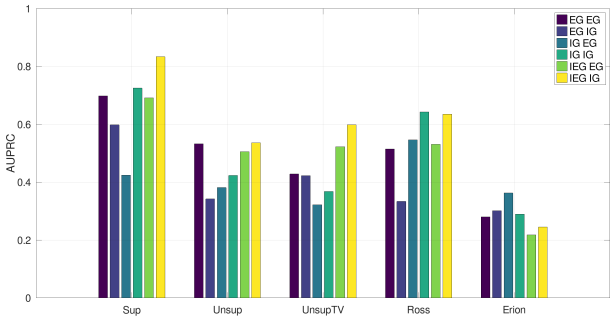


Figure 5.6 – *Robustesse d'IEG. Influence de la méthode d'attributions utilisée pour la contrainte sur l'AUPRC pour les images de tumeurs en 2mm^3. Dans la légende, la méthode de gauche est celle de l'entraînement et celle de droite celle utilisée à l'inférence.*

certaines et chutent drastiquement pour d'autres. Par exemple, l'entraînement avec IG n'est pas efficace pour UnsupTV et la carte EG_IG est la pire pour Ross en termes d'AUPRC.

Ce mélange pour la contrainte, IEG, fournit un modèle plus robuste au choix de la méthode d'attributions à l'inférence au prix d'un coût computationnel plus élevé par rapport à G mais sans coût supplémentaire par rapport aux contraintes avec EG ou IG. En effet, la contrainte IEG est meilleure ou équivalente à la contrainte EG évaluée sur IG et vice versa, en particulier pour le Dice (Figure 5.5 : courbes bleu clair vs orange et vert vs jaune) et l'AURPC (Figure 5.6 : barres bleu vs jaune et bleu pigeon vs vert clair).

En outre, avec cette contrainte, les cartes d'attributions sur IG sont 11% plus corrélées à la référence IG_IG que lorsque EG est utilisée pour l'entraînement et les cartes d'attributions sur EG sont 14% plus corrélées à la référence qu'avec la contrainte sur IG uniquement (Tableau 5.3).

V.4 Classification interprétable

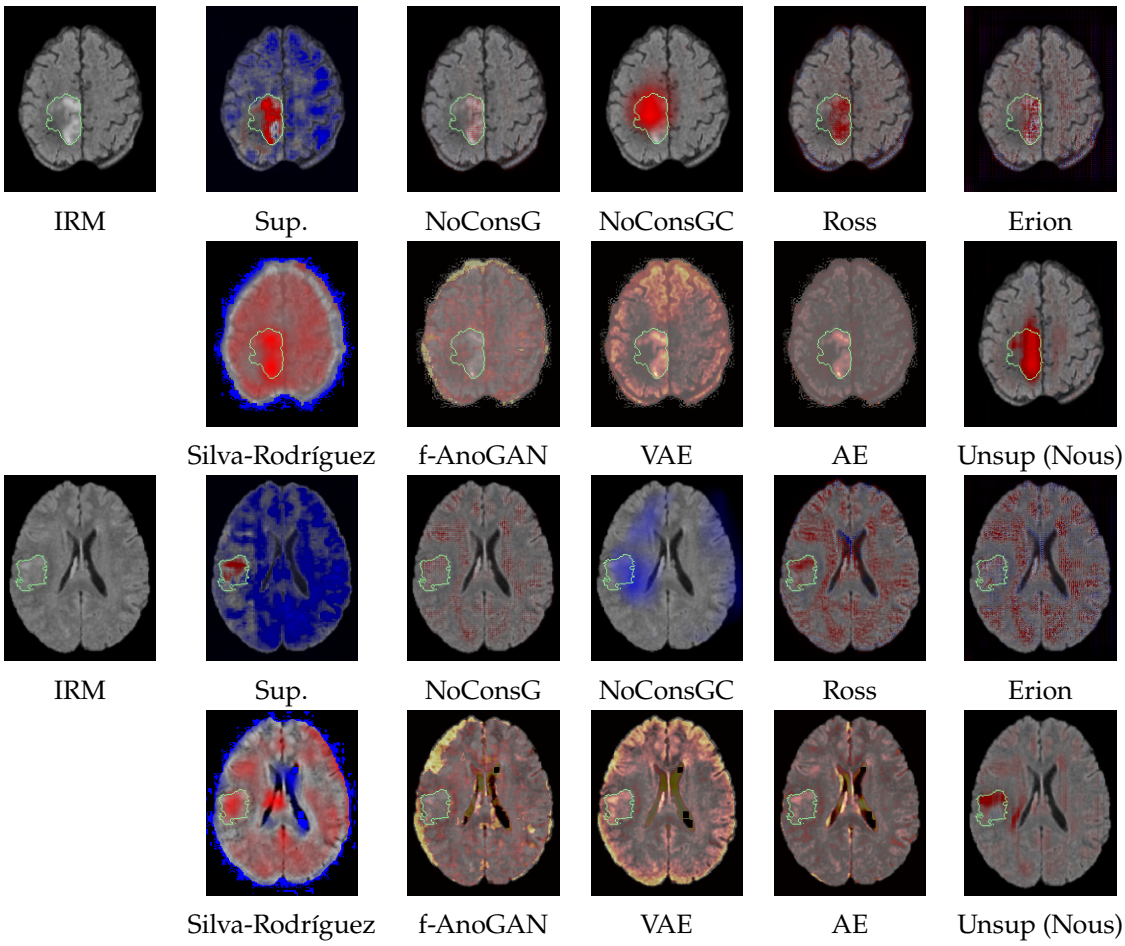


Figure 5.7 – *Carte de segmentation (attributions ou erreur de reconstruction) pour différentes méthodes sur les images de tumeurs en* $1mm^3$ *avec correction N4. Les contours de l'annotation manuelle sont en vert. Dans les attributions, le bleu représente la pertinence pour la classe saine et le rouge pour la classe pathologie. Les fortes valeurs d'attributions sont également en rouge pour Silva-Rodríguez. Pour les méthodes de reconstruction, l'échelle va du noir (faible erreur de reconstruction) au jaune (grande erreur). De gauche à droite, de haut en bas, nous avons : l'image RM, le modèle supervisé, les modèles sans contrainte évalués avec le gradient (NoConsG) et GradCam (NoConsGC), les méthodes de Ross, Erion, Silva-Rodríguez, f-AnoGAN, VAE et AE et finalement notre méthode non supervisée.*

Nous nous sommes comparés à l'état de l'art en utilisant le gradient comme méthode d'attributions à l'inférence puisqu'il est commun à toutes les méthodes (début du chemin

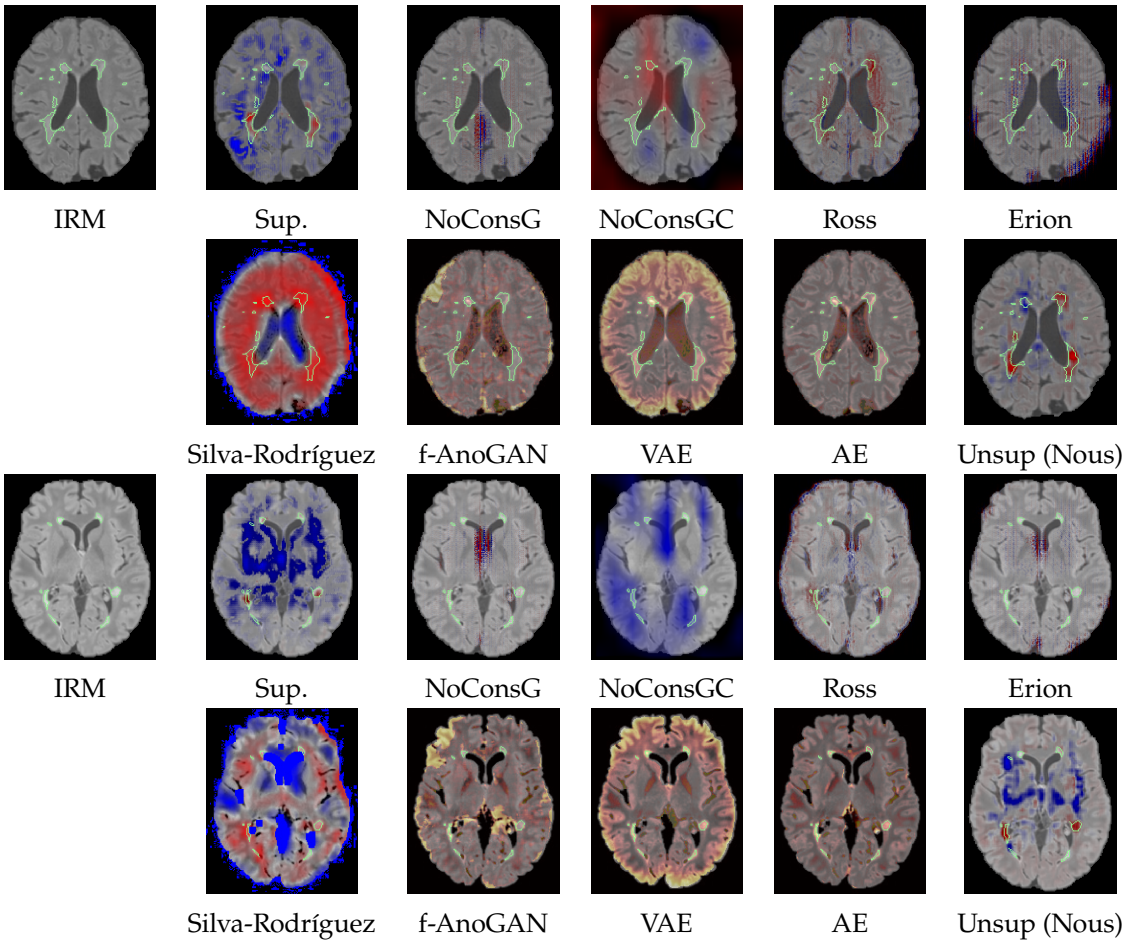


Figure 5.8 – *Carte de segmentation (attributions ou erreur de reconstruction) pour différentes méthodes sur les images de sclérose en plaques. Les contours de l'annotation manuelle sont en vert. Dans les attributions, le bleu représente la pertinence pour la classe saine et le rouge pour la classe pathologie. Les fortes valeurs d'attributions sont également en rouge pour Silva-Rodríguez. Pour les méthodes de reconstruction, l'échelle va du noir (faible erreur de reconstruction) au jaune (grande erreur). De gauche à droite, de haut en bas, nous avons : l'image RM, le modèle supervisé, les modèles sans contrainte évalués avec le gradient (NoConsG) et GradCam (NoConsGC), les méthodes de Ross, Erion, Silva-Rodríguez, f-AnoGAN, VAE et AE et finalement notre méthode non supervisée.*

d'intégration de EG et IG). Pour les méthodes de l'état l'art, nous avons utilisé les méthodes originellement utilisées pour la contrainte. Pour notre contrainte, nous avons utilisé le gradient pour sa rapidité et son entraînement facilité.

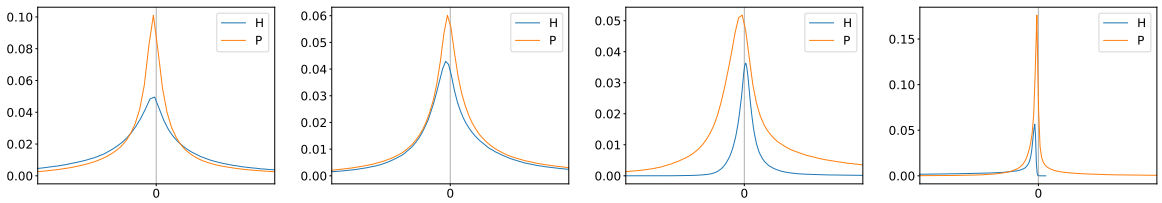
Comme le montre la Figure 5.7, notre contrainte non supervisée permet une décision visiblement basée sur la zone tumorale puisque les attributions se concentrent sur cette dernière. Ross est visuellement équivalent à la notre proposition pour le premier exemple mais moins spécifique dans le deuxième exemple pour lequel la tumeur est plus difficile à identifier. Les décisions prises avec les autres méthodes sont moins basées sur la région d'intérêt. Ceci est confirmé par les métriques présentées dans le Tableau 5.4. En effet, pour la base de données BraTS 2mm^3, le Dice est bien meilleur : il est plus de 20 points plus élevé que le modèle appris uniquement sur la tâche de classification et évalué avec le gradient (NoConsG) et environ 40 points plus élevé que lorsque GradCam (NoConsGC) est utilisé. Notre modèle Unsup surpasse les méthodes de l'état de l'art avec un gain d'au moins 3 points de Dice (statistiquement significatif). Notre modèle est plus précis avec une AUPRC de 6

Tableau 5.4 – *Comparaison avec les méthodes de classification interprétable de l'état de l'art pour les différentes bases de données. Les différences statistiques avec notre modèle Unsup sont indiquées avec une †.*

		Segmentation attributions				Class. images	
Dataset	Méthode	Dice	AUROC	AUROC10	AUPRC	TPR	TNR
BraTS 2020 2mm	Supervisé	0.71 ± 0.17†	0.85†	0.76†	0.73†	1.00	0.95
	NoConsG	0.29 ± 0.16†	0.61†	0.34†	0.16†	**1.00**	**1.00**
	NoConsGC	0.12 ± 0.16†	0.62†	0.15†	0.04†	**1.00**	**1.00**
	Ross	0.48 ± 0.20†	**0.80**	**0.63**	0.39	**1.00**	**1.00**
	Erion	0.29 ± 0.15†	0.70†	0.40†	0.19†	**1.00**	**1.00**
	Unsup	**0.51 ± 0.16**	0.73	0.62	**0.45**	**1.00**	0.95
BraTS 2020 1mm	Supervisé	0.70 ± 0.15†	0.78†	0.68†	0.66†	0.93	0.95
	NoConsG	0.27 ± 0.13†	0.70†	0.38†	0.18†	0.86	**1.00**
	NoConsGC	0.48 ± 0.20†	**0.90**†	0.65	0.40	0.86	**1.00**
	Ross	0.40 ± 0.19†	0.89†	**0.66**†	0.44	**1.00**	**1.00**
	Erion	0.29 ± 0.18†	0.78†	0.36†	0.19†	**1.00**	**1.00**
	Unsup	**0.52 ± 0.17**	0.69	0.56	**0.45**	**1.00**	**1.00**
BraTS 2020 1mm avec N4	Supervisé	0.55 ± 0.18†	0.70†	0.55	0.49†	0.98	1.00
	NoConsG	0.18 ± 0.07†	0.71†	0.33†	0.14†	**1.00**	**1.00**
	NoConsGC	0.20 ± 0.17†	0.79†	0.19†	0.08†	**1.00**	**1.00**
	Ross	0.19 ± 0.11†	**0.77**†	0.38†	0.17†	**1.00**	**1.00**
	Erion	0.16 ± 0.07†	0.66†	0.23†	0.08†	**1.00**	**1.00**
	Unsup	**0.38 ± 0.15**	0.73	**0.50**	**0.30**	0.94	**1.00**
MS 1mm	Supervisé	0.24 ± 0.18	0.53†	0.46	0.24†	0.77	0.77
	NoConsG	0.001 ± 0.002†	0.63†	0.41	0.02†	**1.00**	**1.00**
	NoConsGC	0.0003 ± 0.0007†	0.34†	0.01†	0.002†	**1.00**	**1.00**
	Ross	0.09 ± 0.09†	**0.70** †	0.45	0.04†	**1.00**	**1.00**
	Erion	0.01 ± 0.01†	0.60	0.35†	0.01†	**1.00**	**1.00**
	Unsup	**0.25 ± 0.16**	0.60	**0.51**	**0.20**	0.89	0.91

points supérieure à celle de la meilleure méthode de l'état de l'art, à savoir Ross. L'AUROC est un peu plus faible pour notre proposition mais moins adaptée que l'AUPRC car les voxels pathologiques sont sous-représentés par rapport aux voxels sains. Si l'on considère l'AUROC avec un taux de faux positifs inférieur à 10% (AUROC10), Ross et notre méthode non supervisée sont équivalentes. Cette différence entre l'AUROC et l'AUROC10 montre que Ross pourrait atteindre une plus grande sensibilité mais avec une faible spécificité. En d'autres termes, Ross permet de détecter plus de lésions, mais au prix d'un grand nombre de faux positifs. Lorsqu'on augmente la résolution de $2mm^3$ à $1mm^3$, les performances des méthodes de l'état de l'art augmentent car la tâche de segmentation est plus facile. En particulier, NoConsGC est la deuxième meilleure méthode en termes de Dice alors qu'elle est la pire pour les images ayant des voxels de $2mm^3$. Néanmoins, notre méthode reste meilleure en termes de Dice et d'AUPRC. Dans le cas où la base de données tumorales BraTS est utilisée avec une correction des inhomogénéités de champ, des images avec un contraste plus faible sont considérées et la tâche de segmentation est donc plus difficile. Dans ce cas, notre méthode est bien meilleure avec un écart de 20 points de Dice et une AUPRC deux fois meilleure que la deuxième meilleure méthode. Dans ces conditions, toutes les mesures sont statistiquement significatives. Notre contrainte permet donc d'obtenir un classifieur plus pertinent dans le sens où la décision du réseau est davantage basée sur des structures cliniquement pertinentes, les tumeurs cérébrales, sans dégrader les performances de classification avec un TPR et un TNR supérieurs à 95%.

Si l'on considère la sclérose en plaques, les lésions sont plus petites que les tumeurs cérébrales et même dans ce cas, la méthode non supervisée que nous proposons se distingue



(a) NoConsG (b) Erion (c) Ross (d) Unsup

FIGURE 5.9 – *Histogramme des attributions pour différentes méthodes sur les images saines (H) et avec tumeurs (P) en* $2mm^3$.

par ses performances par rapport aux méthodes de la littérature. Les exemples visuels de la Figure 5.8 montrent que notre proposition détecte plus de lésions (deuxième exemple) et qu'elle est plus spécifique : dans le premier exemple, avec notre contrainte, les zones d'attributions élevées sont concentrées sur les lésions SEP alors qu'avec Ross, les attributions élevées sont réparties autour des ventricules. Ainsi, le Dice avec notre méthode est 25 points plus élevé que NoConsG qui discrimine les images saines des images de SEP sans tenir compte des lésions. Par rapport à la deuxième meilleure méthode, notre proposition obtient un Dice trois fois supérieur et une AUPRC cinq fois plus élevée. Les performances de classification ne sont pas trop dégradées avec une précision d'environ 90%. Nous remarquons que les contraintes (supervisées et non supervisées) rendent la classification plus difficile : la convergence lors de l'entraînement est plus lente et les performances de classification sont un peu plus faibles. Il est probable qu'en l'absence de contrainte, le réseau utilise un raccourci basé sur la signature des bases de données au lieu des caractéristiques cliniques. Il convient de noter que la répartition des images entre les jeux d'entraînement, de validation et de test, traditionnellement utilisée pour détecter un sur-apprentissage (*overfitting*) ne permet pas de détecter ce type de problème : si l'ensemble de données présente un biais quelconque, ce biais sera présent dans chacun des sous-ensembles et n'affectera pas les performances de classification sur le test.

Par conséquent, l'utilisation de l'entropie croisée pour contraindre de manière non supervisée les attributions des images saines pendant l'apprentissage permet une classification interprétable avec une décision davantage basée sur la zone de la pathologie, surpassant la classification sans contrainte et les contraintes proposées dans l'état de l'art. Notre méthode est particulièrement efficace pour les tâches de segmentation difficiles (par exemple pour la sclérose en plaques ou les images de tumeurs à faible contraste) pour lesquelles la différence de performances avec l'état de l'art est importante et significative.

De plus, avec notre méthode, les voxels des images saines contribuent à la décision de classer l'image comme "saine" car les attributions des images saines sont négatives. En effet, pour notre méthode, l'histogramme des attributions saines (Figure 5.9 en bleu) disparaît dans la zone positive et la courbe pathologique (en orange) est à la fois négative, pour les régions saines, et positive, pour les régions tumorales comme le montre l'évaluation de la segmentation. En comparaison, les attributions saines des autres méthodes sont partiellement positives et les courbes des histogrammes des deux classes sont mélangées.

V.5 Détection d'anomalies

Le modèle contraint proposé peut être utilisé pour la détection d'anomalies faiblement supervisée en ayant seulement besoin du label de l'image. Dans le Tableau 5.5, les performances de notre méthode et de celle de l'état de l'art sur différentes bases de données sont rapportées. Notre méthode est plus performante que les autres méthodes pour le Dice,

Tableau 5.5 – *Comparaison avec les méthodes de détection d'anomalies de l'état de l'art pour les différentes bases de données. Les différences statistiques avec notre modèle Unsup sont indiquées avec une †.*

Dataset	Méthode	Dice	AUROC	AUROC10	AUPRC
BraTS 2020 2mm	Silva-Rodríguez	$0.37 \pm 0.17^{\dagger}$	$\mathbf{0.92}^{\dagger}$	0.56	$0.32^{\dagger}$
	AE	$0.26 \pm 0.11^{\dagger}$	$0.90^{\dagger}$	$0.36^{\dagger}$	$0.16^{\dagger}$
	VAE	$0.25 \pm 0.14^{\dagger}$	$0.91^{\dagger}$	$0.30^{\dagger}$	$0.15^{\dagger}$
	f-AnoGAN	$0.17 \pm 0.10^{\dagger}$	0.79	$0.06^{\dagger}$	$0.06^{\dagger}$
	Unsup	**0.51 ± 0.16**	0.73	**0.62**	**0.45**
BraTS 2020 1mm	Silva-Rodríguez	$0.33 \pm 0.20^{\dagger}$	$\mathbf{0.91}^{\dagger}$	0.52	$0.32^{\dagger}$
	AE	$0.28 \pm 0.11^{\dagger}$	$\mathbf{0.91}^{\dagger}$	0.43	$0.17^{\dagger}$
	VAE	$0.24 \pm 0.13^{\dagger}$	$\mathbf{0.91}^{\dagger}$	$0.34^{\dagger}$	$0.15^{\dagger}$
	f-AnoGAN	$0.16 \pm 0.10^{\dagger}$	$0.84^{\dagger}$	$0.16^{\dagger}$	$0.09^{\dagger}$
	Unsup	**0.52 ± 0.17**	0.69	**0.56**	**0.45**
BraTS 2020 1mm avec correction N4	Silva-Rodríguez	$0.18 \pm 0.09^{\dagger}$	$\mathbf{0.88}^{\dagger}$	$0.19^{\dagger}$	$0.14^{\dagger}$
	AE	$0.23 \pm 0.10^{\dagger}$	$\mathbf{0.88}^{\dagger}$	0.35	$0.15^{\dagger}$
	VAE	$0.18 \pm 0.10^{\dagger}$	$0.87^{\dagger}$	$0.26^{\dagger}$	$0.11^{\dagger}$
	f-AnoGAN	$0.16 \pm 0.09^{\dagger}$	$0.84^{\dagger}$	$0.18^{\dagger}$	$0.09^{\dagger}$
	Unsup	**0.38 ± 0.15**	0.73	**0.50**	**0.30**
BraTS 2019 1mm coupes centrales	Silva-Rodríguez	0.54 ± 0.22	**0.97**	**0.78**	$0.52^{\dagger}$
	Unsup (2D)	**0.62 ± 0.14**	0.96	**0.78**	**0.72**
MS 1mm	Silva-Rodríguez	$0.10 \pm 0.08^{\dagger}$	$0.93^{\dagger}$	0.50	$0.05^{\dagger}$
	AE	$0.10 \pm 0.09^{\dagger}$	$\mathbf{0.95}^{\dagger}$	0.67	$0.05^{\dagger}$
	VAE	$0.03 \pm 0.03^{\dagger}$	$0.92^{\dagger}$	0.36	$0.01^{\dagger}$
	f-AnoGAN	$0.02 \pm 0.02^{\dagger}$	$0.81^{\dagger}$	$0.10^{\dagger}$	$0.01^{\dagger}$
	Unsup	$\mathbf{0.25 \pm 0.16}^{\dagger}$	0.60	**0.51**	**0.18**

l'AUPRC et l'AUROC10 (statistiquement significatif pour les deux premières métriques). Pour les images à faible résolution, la méthode proposée surpasse l'état de l'art de près de 15 points de Dice et d'AUPRC. En augmentant la résolution à $1mm^3$, l'écart de Dice entre notre méthode et la meilleure méthode de l'état de l'art (Silva-Rodríguez) augmente de 5 points. Pour les tâches plus difficiles (BraTS avec correction des inhomogénéités de champ et sclérose en plaques), la meilleure méthode de la littérature est l'auto-encodeur mais notre méthode est toujours plus efficace avec 15 points de Dice et d'AUPRC en plus.

Visuellement, dans les Figures 5.7 et 5.8, notre méthode non supervisée semble plus spécifique que les autres, en particulier que le VAE et Silva-Rodríguez qui détectent les anomalies dans les tissus sains. Silva-Rodríguez ne semble compétitif qu'avec la configuration où des coupes centrales de BraTS 2019 sont utilisées à la fois pour la base saine et la base pathologique. Néanmoins, il atteint 8 points de Dice de moins que notre méthodes et reste moins précis avec une AUPRC de 20 points inférieure. Notons que dans la plupart des applications réelles, l'image entière est utilisée et l'algorithme de segmentation doit être capable de traiter les coupes de l'ensemble du cerveau et en particulier de produire une segmentation vide pour les coupes saines. En outre, en extrayant les coupes de la même base pour l'apprentissage, le réseau n'est pas confronté aux difficultés de biais entre les bases de données.

Conclusion

Dans ce chapitre, nous avons proposé une nouvelle contrainte dans le cas de l'apprentissage d'un classifieur d'images saines vs pathologiques sans ajouter de supervision supplémentaire et donc uniquement à partir du label de l'image. Nous avons contraint les attributions qui indiquent quels pixels/voxels de l'image d'entrée servent pour la prise de décision du réseau. Cette contrainte impose que tous les voxels des images saines servent pour une classification dans la classe saine. Le choix de la méthode d'attributions pour la contrainte a également été étudié avec deux conclusions. Tout d'abord, il est inutile d'utiliser Expected Gradient qui est coûteux en temps et en ressources tout en rendant les apprentissages moins stables. En effet, nous avons montré qu'en utilisant le gradient plutôt qu'Expected Gradient, on réduit le temps et les ressources nécessaires avec une meilleure convergence à l'apprentissage tout en ayant des cartes similaires à l'inférence. Ensuite, nous avons proposé une nouvelle contrainte mixant Expected Gradient et Integrated Gradient qui est robuste au choix de la méthode d'attributions à l'inférence tout en ayant le même coût qu'Expected ou Integrated Gradient. L'ajout de cette contrainte permet une classification plus interprétable avec une décision davantage basée sur la pathologie mais également une segmentation faiblement supervisée des anomalies. Les performances de notre proposition dépassent largement celle de l'état de l'art, notamment dans les cas les plus difficiles comme la sclérose en plaques ou des images à faible contraste.

Puisque notre proposition consiste à ajouter une simple fonction de coût pendant l'entraînement, elle peut être facilement utilisée pour tout type de modèles profonds sans modifier leur architecture. Ainsi, ce travail pourrait être utilisé dans plusieurs domaines. Les discriminateurs étant un élément constitutif des réseaux adversaires, nous pourrions utiliser les contraintes d'attributions pour augmenter les performances et la pertinence des méthodes de type GAN pour la détection d'anomalies par exemple. Des classifieurs ont également été utilisés dans d'autres modèles génératifs comme les modèles profonds de diffusion [Wolleb *et al.*, 2022] pour de la détection d'anomalies et notre contrainte pourrait être incorporée dans ce type d'approche également. Ce travail pourrait également être étendu à d'autres réseaux et pas seulement aux classifieurs comme les réseaux de régression ou de prédiction avec par exemple l'estimation du grade d'une maladie qui se concentre sur la signature radiologique de la pathologie.

CHAPITRE

6

CONSTRUCTION ET APPRENTISSAGE SOUS CONTRAINTE DE RÉSEAUX MONOTONES POUR AMÉLIORER L'EXPLICABILITÉ

Introduction

Dans le chapitre précédent, nous avons vu que contraindre les attributions d'un classifieur pendant l'entraînement permettait d'obtenir une décision plus interprétable et la détection d'anomalies faiblement supervisée. Pour améliorer l'explicabilité du classifieur, nous souhaiterions avoir des propriétés intrinsèques au réseau qui ne sont pas seulement apprises. En effet, en imposant certaines propriétés à notre modèle à travers l'apprentissage d'une contrainte donnée par une fonction de coût, on compte sur la généralisation de l'apprentissage pour que ces propriétés soit respectées sur des données de test. En outre, nous utilisions le gradient de la sortie par rapport à l'image. Ceci présente de nombreux avantages puisqu'il possède des informations sur le contenu sémantique de tout le réseau et nous permet d'obtenir une carte de segmentation à la même échelle que celle de l'image. Néanmoins, il peut exister de nombreux bruits dans l'image qui peuvent parasiter les cartes obtenues. En appliquant quelques couches de convolution, on peut s'attendre à éliminer une partie de ce bruit. Dans ce chapitre, nous proposons donc d'utiliser des réseaux qui sont par définition explicables : les réseaux monotones et plus précisément la solution la plus simple de les obtenir à savoir les réseaux à poids positifs. Nous avons vu, dans le Chapitre 2, que ces réseaux, malgré des qualités d'explicabilité importantes, sont peu utilisés et sont souvent des architectures peu profondes, car soi-disant trop contraints et donc difficilement entraînables. Nous levons donc, dans ce chapitre, un premier verrou qui est d'entraîner n'importe quelle architecture transformée en réseau à poids positifs avec les mêmes capacités que les réseaux classiques, notamment grâce à une initialisation des poids permettant de conserver une variance unitaire dans tout le réseau. Ensuite, pour rendre le réseau interprétable, nous proposons de le contraindre, à travers à la fois les cartes de caractéristiques et les gradients, de telle sorte que les tissus sains soient représentés de la même manière dans les images. Pour vérifier l'interprétabilité de notre réseau, nous générons des exemples contrefactuels sur les cartes de caractéristiques situées juste avant un réseau monotone. Ces cartes sont dépourvues du bruit présent dans l'image et directement liées à la sortie par une relation de monotonie permettant une lecture plus facile de l'exemple contrefactuel généré. Ce chapitre est organisé comme suit. Tout d'abord, nous présenterons notre méthode pour transformer un réseau en réseau monotone et l'initialisation des poids nécessaire à leur fonctionnement. Une discussion sur le préjugé de sur-contrainte des réseaux à poids positifs sera également discuté. Puis, nous détaillerons les différentes contraintes utilisées sur notre classifieur lors de l'entraînement. Enfin, nous finirons par présenter les expériences et les résultats obtenus.

Contributions

Les contributions présentées dans ce chapitre sont :

- Une méthodologie permettant de transformer n'importe quel réseau de neurones en réseau monotone.
- Une description paramétrique de l'ensemble des réseaux de neurones monotones à deux couches.
- Une analyse théorique permettant d'identifier les causes de dysfonctionnement des initialisations des poids des réseaux de neurones aléatoires de l'état-de-l'art pour certaines couches (couches résiduelles, couches de *maxpooling*, couches linéaires à poids positifs).
- Une initialisation efficace des poids des réseaux de neurones permettant de conserver une variance unitaire dans toutes les couches du réseau indépendamment de leur nature.
- Une méthode non-supervisée utilisant un réseau monotone contraint et la gé-

nération d'exemples contrefactuels permettant une classification plus interprétable et la détection d'anomalies.

Ces travaux sont en cours de soumission.

I Transformation d'un réseau en réseau monotone

Un réseau $\mathrm{M} : \mathbb{R}^n \to \mathbb{R}^m$ est monotone croissant si :

$$\forall x, \forall i, \forall j, \frac{\partial \mathrm{M}_j(x)}{\partial x_i} \geq 0 \tag{6.1}$$

où x est l'entrée, i l'indice dans l'entrée et j l'indice dans la sortie. Grâce à cette propriété, un lien peut être facilement établi entre l'entrée et la sortie, ce qui rend le réseau plus explicable : la sortie du réseau augmente (et respectivement diminue) avec l'entrée.

Étant donné une architecture de réseau neuronal quelconque, nous proposons de la convertir en un réseau monotone à l'aide de l'ensemble de règles suivant. Un schéma de la méthode est donné Figure 6.1.

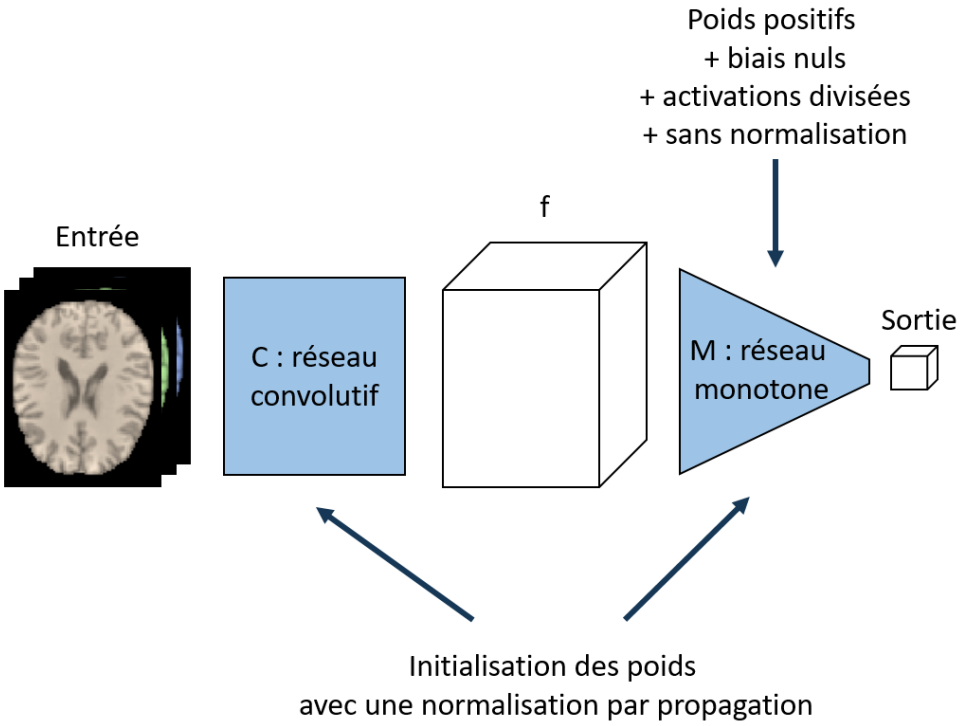


Figure 6.1 – *Schéma explicatif pour transformer un réseau en réseau monotone. Un réseau convolutif C est ajouté pour construire les caractéristiques interprétables f de mêmes dimensions spatiales que l'entrée (Section I.1). Ces dernières sont données en entrée du réseau monotone M pour lequel : 1/ on a paramétré les couches linéaires de telle sorte que les poids soient positifs et les biais nuls (Section I.2), 2/ les activations sont remplacées par des activations convexes croissantes sur la moitié des canaux et concaves croissantes sur le reste (Section I.4) et 3/ les couches de normalisation sont retirées (Section I.3). Les poids des réseaux sont initialisés avec la méthode de normalisation par propagation, décrite en Section I.5.*

I.1 Encodage des caractéristiques interprétables

Comme dans [Nguyen *et al.*, 2023, Sivaprasad *et al.*, 2021], notre réseau monotone est précédé d'un réseau non monotone standard qui vise à extraire les caractéristiques f de l'entrée. Comme les caractéristiques extraites constitueront l'entrée du réseau monotone, elles seront explicables dans le sens où elles seront liées de manière monotone croissante à la sortie : augmenter (respectivement diminuer) la valeur d'une caractéristique quelconque conduira nécessairement à augmenter (respectivement diminuer) la prédiction du réseau de neurones. Il n'est pas possible d'utiliser directement l'image comme entrée du réseau monotone à moins qu'une relation de monotonie entre les valeurs des voxels et la sortie voulue n'existe déjà. En précédant la partie monotone par ce réseau non-monotone, on peut réarranger les caractéristiques de l'image d'entrée de telle sorte qu'elles soient monotones par rapport à la sortie du réseau.

D'autres propriétés de ce réseau peuvent également améliorer l'explicabilité. Par exemple, un réseau composé de quelques couches convolutives, qui conservent la taille de l'entrée, permet de rendre également les caractéristiques localisables par rapport à l'image d'entrée et évite une interpolation comme dans les méthodes de type GradCam [Selvaraju *et al.*, 2017]. Un compromis doit ainsi être trouvé entre un bon encodage avec plus de couches et l'explicabilité du réseau avec moins de couches. Il peut être aussi intéressant de forcer les biais à être nuls pour cet encodeur comme nous le verrons en Section IV.1.1.

I.2 Couches linéaires

Pour obtenir des réseaux monotones à poids positifs comme dans [Sill, 1997, Daniels et Velikova, 2010], dans la partie monotone du réseau, les poids des couches linéaires, c'est-à-dire les couches entièrement connectées (FC) et les convolutions (Conv), sont paramétrés par une fonction positive. Nous proposons d'utiliser la fonction $\varphi : W \mapsto W^2/\sqrt{1+W^2}$ qui est différentiable et permet d'avoir des poids nuls. Les biais sont, quant à eux, fixés à zéro. Cette dernière contrainte, qui, à notre connaissance, n'a pas été proposée dans la littérature sur les réseaux monotones, implique la propriété suivante pour le réseau : si toutes les composantes des caractéristiques interprétables f sont positives (resp. négatives), alors la sortie de chaque couche (y compris la décision finale) sera la combinaison positive de nombres positifs (respectivement négatifs) et sera par conséquent également positive (respectivement négative). Le fait de fixer les biais à zéro dans la partie monotone du réseau implique par conséquent une propriété de calibrage intéressante qui permet une interprétation plus facile des caractéristiques interprétables f en fonction de leur signe.

I.3 Couches de normalisation

Les couches de normalisation sont souvent utilisées dans les architectures profondes pour stabiliser l'apprentissage. Elles consistent en l'opération suivante : $y = \frac{x-\mathrm{E}(x)}{\sigma(x)}$ où la moyenne et l'écart-type sont calculés sur certaines dimensions de l'entrée x en fonction du type de normalisation. Cette opération n'étant pas monotone, ces couches casseraient la propriété de monotonicité. Elles sont donc retirées de la partie monotone du réseau.

I.4 Fonctions d'activation

Dans [Runje et Shankaranarayana, 2022], des fonctions d'activations différentes sont utilisées sur les différents canaux d'une couche. Ainsi, certains canaux sont soumis à une activation convexe, d'autres à une activation concave et enfin d'autres à une activation bornée. Cela permet d'assouplir la propriété de convexité trop forte que l'utilisation de la même activation croissante convexe (comme par exemple une fonction ReLU ou LeakyReLU) imposerait au réseau. Nous proposons de n'utiliser que les deux premières formes qui

permettent déjà d'approximer un grand nombre de fonctions en sorties du réseau. Les fonctions d'activation sont donc fixées à n'importe quelle fonction d'activation croissante standard r sur la première moitié des canaux de la couche et à $-\mathrm{r}(-x)$ sur les canaux restants.

On retrouve ces fonctions d'activation en cherchant à caractériser les réseaux monotones à deux couches comme discuté en Section II.

I.5 Initialisation des poids

Les réseaux monotones à poids positifs proposés dans l'état de l'art [Sill, 1997, Daniels et Velikova, 2010] sont vraiment peu profonds. Nous supposons que l'initialisation des poids est cruciale pour l'entraînement de ces réseaux. En supprimant les couches de normalisation et en imposant des poids positifs, il est, en effet, impossible d'entraîner des réseaux monotones profonds avec les initialisations de poids classiques de l'état de l'art [He *et al.*, 2015, Glorot et Bengio, 2010]. Nous proposons d'utiliser l'initialisation de la Section III.2 pour entraîner nos réseaux à poids positifs. Cette initialisation assure une variance unitaire à travers le réseau et ceci, peu importe les couches qui le constituent.

II Une caractérisation des réseaux monotones à deux couches

A priori, la procédure décrite en Section I garantit l'aspect monotone du réseau en forçant chaque couche à être croissante. On peut se demander si cela ne sur-contraint pas trop le réseau. Avec la proposition ci-dessous, nous avons cherché à caractériser les réseaux monotones à deux couches afin de ne forcer l'aspect monotone que deux couches par deux couches.

Proposition 5. *Soit g le réseau à deux couches défini par $g(x) = A\sigma(Bx)$, avec σ une ReLU. Par la suite, on notera D_v une matrice diagonale avec le vecteur v sur sa diagonale.*

Assertion 1 : *S'il existe, $(a, b) \in \mathbb{R}^n \times \mathbb{R}^n$ avec $\operatorname{sign}(a) = \operatorname{sign}(b)$ et $\tilde{A}$ et $\tilde{B}$ positives tels que $A = \tilde{A}D_a$, $B = D_b\tilde{B}$, alors g est monotone.*

Assertion 2 : *Si B est surjective et g est monotone, alors il existe $(a, b) \in \mathbb{R}^n \times \mathbb{R}^n$ avec $\operatorname{sign}(a) = \operatorname{sign}(b)$ et $\tilde{A}$ et $\tilde{B}$ positives tels que $A = \tilde{A}D_a$ et $B = D_b\tilde{B}$.*

Démonstration. Notons tout d'abord que si pour un x donné, $w(x)$ est défini comme $w(x) = \sigma'(Bx)$, alors $w(x) \geq 0$ et $g'(x) = AD_{w(x)}B$.

Prouvons la première assertion. Si $A = \tilde{A}D_a$ et $B = D_b\tilde{B}$ alors $g'(x) = \tilde{A}D_aD_{w(x)}D_b\tilde{B}$, c'est-à-dire que :

$$\forall i, \forall j, \frac{\partial g_i}{\partial x_j} = \sum_k \tilde{a}_{i,k}a_k w_k b_k \tilde{b}_{k,j}$$

qui est toujours positive ou nulle car il s'agit d'une somme de termes positifs ou nuls. Ainsi g est monotone.

Prouvons maintenant la seconde assertion. Comme g est monotone, pour tout x :

$$\forall i, \forall j, \frac{\partial g_i}{\partial x_j} = \sum_k a_{i,k}w_k(x)b_{k,j} \geq 0.$$

Comme B est surjective, pour tout l, il existe x^l tel que Bx^l soit 1 pour la l^{e} composante et -1 pour toutes les autres composantes. On a donc $w(x^l)$ qui vaut 0 partout sauf pour la composante l qui vaut 1 et :

$$\forall l, \forall i, \forall j, \frac{\partial g_i}{\partial x_j}(x^l) = a_{i,l}b_{l,j} \geq 0. \tag{6.2}$$

Soit b le premier vecteur colonne de B. Alors pour tout i et l : $a_{i,l}b_l \geq 0$. Ceci implique que toutes les composantes du l^{e} vecteur colonne de la matrice A ont le même signe que b_l :

A peut être factorisée comme $A = \tilde{A}D_a$ avec $\tilde{A}$ non négative. En utilisant cette factorisation, l'inégalité 6.2 devient $\tilde{a}_{i,l}a_l b_{l,j} \geq 0$. Par conséquent, on peut factoriser B comme $B = D_b\tilde{B}$. □

Les réseaux monotones à deux couches peuvent donc s'écrire $f(x) = \tilde{A}D_a\sigma(D_b\tilde{B}x)$. Ce sont donc des réseaux dont les couches linéaires sont à poids positifs et dont les activations sont de la forme $D_a\sigma(D_b x)$. On retrouve les fonctions d'activations utilisées dans notre procédure en Section I.4 (ReLU(x) et −ReLU(−x)) en prenant $a = b$ où a est un vecteur valant 1 sur la moitié de ces composantes et −1 sur l'autre moitié.

Ainsi, si B est surjective, la configuration proposée avec les poids positifs et ces activations est quasiment la seule manière d'obtenir un réseau monotone deux couches par deux couches. En effet, la seule possibilité pour gagner en liberté serait de ne pas imposer le nombre et la place des composantes positives/négatives.

Dans [Liu *et al.*, 2020], les réseaux monotones deux couches par deux couches étaient obtenus par la résolution d'un MILP à chaque passage dans le réseau, ce qui est très coûteux. En outre, les approximations pour obtenir ce MILP ne permettent pas de certifier l'obtention d'un réseau monotone. A contrario, avec notre paramétrisation, il est certain que le réseau est monotone sans le moindre coût.

III Initialisation des poids avec conservation de la variance

III.1 Le besoin d'une nouvelle initialisation

Comme vu dans la Section III.2 du Chapitre 2, dans [Glorot et Bengio, 2010] et [He *et al.*, 2015], il est démontré que si les poids et les entrées d'une couche linéaire sont centrés, indépendants, identiquement distribués et mutuellement indépendants, la variance de l'entrée x, de la sortie y et des poids w satisfont $\mathrm{V}(y) = n\mathrm{V}(w)\mathrm{E}(x^2)$, où n est la taille du support de w. La variance des poids peut donc être choisie de telle sorte que si chaque caractéristique d'entrée est normalisée comme dans [Glorot et Bengio, 2010] ou normalisée suivie d'une activation ReLU comme dans [He *et al.*, 2015], la variance de la sortie sera égale à un.

Bien que ces initialisations aléatoires permettent de stabiliser l'apprentissage très efficacement, certaines limitations subsistent. Tout d'abord, la procédure est locale : elle ne tient pas compte de l'architecture complète du réseau et suppose que la stabilisation de chaque couche linéaire stabilisera le réseau complet par composition. Cependant, certaines couches différentes des couches FC/Conv et des couches d'activation peuvent briser cette hypothèse en ayant, en sortie, des caractéristiques non normalisées. En accumulant ces couches, la variance des caractéristiques peut fortement diverger de la variance unitaire du début du réseau notamment dans les réseaux très profonds très utilisés de nos jours.

Nous identifions ici trois types de couche qui engendrent ce problème : les blocs résiduels, les couches de *maxpooling* et les couches FC/Conv avec des coefficients positifs. Généralement, les couches de normalisation, comme les couches de *BatchNorm* [Ioffe et Szegedy, 2015] ou d'*InstanceNorm* [Ulyanov *et al.*, 2016], insérées dans le réseau permettent de renormaliser partiellement les caractéristiques entre les couches, rendant transparente la variation de la variance des caractéristiques entre les couches. Toutefois, ces couches de normalisation ne sont pas utilisables pour les réseaux monotones comme nous l'avons vu en Section I.3 .

III.1.1 Couches résiduelles

Les couches résiduelles [He *et al.*, 2016] codent l'opération $y = x + \mathrm{F}(x)$ où F représente généralement quelques couches de convolutions suivies d'activations et x l'entrée. Pour ces couches, la variance de la sortie y est donnée par :

$$\mathrm{V}(y) = \mathrm{V}(x) + \mathrm{V}(\mathrm{F}(x)) + 2\mathrm{Cov}(x, \mathrm{F}(x)) \tag{6.3}$$

Si x est normalisé (variance unitaire) et F préserve la variance à l'initialisation, on a :

$$\mathrm{V}(y) = 2 + 2\mathrm{Cov}(x, \mathrm{F}(x)) \tag{6.4}$$

La variance est donc susceptible d'au moins doubler à chaque bloc résiduel.

III.1.2 Couches de *maxpooling*

Les couches de *maxpooling* peuvent être écrites sous la forme $y_i = max\,(x_{is+k})_{k\in N}$, avec i l'index sur les dimensions spatiales, s le *stride* et n la taille du voisinage N. Ces couches modifient la distribution des données. Par exemple, le maximum de n distributions uniformes $\mathcal{U}(0,1)$ suit une distribution bêta $B(n,1)$ [Gentle, 2010]. Cette distribution a pour espérance $\frac{n}{n+1}$ et pour variance $\frac{n}{(n+2)(n+1)^2}$. De la même manière, dans [Nadarajah et Kotz, 2008], il est montré que la distribution du maximum de deux distributions gaussiennes $\mathcal{N}(0,1)$ indépendantes a une espérance égale à $\frac{1}{\sqrt{\pi}}$ et une variance de $1 - \frac{1}{\pi}$. Dans le cas de n gaussiennes $\mathcal{N}(0,1)$ indépendantes, l'espérance est supérieure à celle avec deux gaussiennes puisque $max(X_1, X_2, ..., X_n) = max(max(X_1, X_2), X_3, ...X_n) \geq max(X_1, X_2)$. Pour la variance, l'inégalité de Poincaré permet d'établir que $V(\max_{i\in N}(X_i)) \leq \max_{i\in N} V(X_i) = 1$. On peut également démontrer que cette variance est sous la forme $\frac{C}{log(n)}$ (où C est une constante) et décroît avec n [Chatterjee, 2014, Tanguy, 2019]. Les caractéristiques après une couche de *maxpooling* ne seront donc plus centrées et leur variance sera diminuée par rapport à l'entrée.

III.1.3 Couches linéaires (FC/Conv) à poids positifs

Dans la proposition *originale* ci-dessous, nous donnons un aperçu des conséquences de ces initialisations aléatoires sur la corrélation des caractéristiques en général. Entre autre, l'analyse suivante explique pourquoi les méthodes d'initialisation de l'état de l'art ne peuvent pas fonctionner pour les réseaux à poids positifs.

Proposition 6. *Soit x l'entrée d'une couche linéaire (FC/Conv), notons $\mu = \mathrm{E}(x)$, $C = \mathrm{Cov}(x)$ et ρ défini comme suit*

$$\rho = \frac{\sum_i C_{i,i} + \mu_i^2}{\sum_{i,j} C_{i,j}}.$$

Soit également $y = \sum_{i=1}^{n} a_i x_i + t_a$ et $z = \sum_{i=1}^{n} b_i x_i + t_b$ deux composantes de la sortie de cette couche linéaire où n est le nombre de coefficients non nuls, a_i et b_i sont les poids et t_a et t_b sont les biais. Si,

1. *$\forall i$, a_i (respectivement b_i) sont indépendants et identiquement distribués avec une espérance μ_w et une variance σ_w^2,*
2. *$\forall i, \forall j, \forall k$, x_i, a_j et b_k sont indépendants,*

alors la corrélation entre y et z est donnée par :

$$\mathrm{corr}(y, z) = \frac{\mu_w^2}{\mu_w^2 + \sigma_w^2 \rho}. \tag{6.5}$$

Démonstration. Comme

$$\mathrm{E}(y) = E\left[\sum_i a_i x_i + t_a\right] = \sum_i \mathrm{E}(a_i x_i + t_a) = \sum_i \mathrm{E}(a_i)\mathrm{E}(x_i) + t_a = \mu_w \sum_i \mu_i + t_a \tag{6.6}$$

et de manière similaire $\mathrm{E}(z) = \mu_w \sum_i \mu_i + t_b$, la covariance entre y et z est :

$$\begin{aligned}
\mathrm{Cov}(y,z) &= E\left[(y-\mathrm{E}(y))(z-\mathrm{E}(z))\right] \\
&= E\left[\left(\sum_i a_i x_i + t_a - \mu_w \sum_i \mu_i - t_a\right)\left(\sum_j b_j x_j + t_b - \mu_w \sum_j \mu_j - t_b\right)\right] \\
&= E\left[\left(\sum_i a_i x_i - \mu_w \mu_i\right)\left(\sum_j b_j x_j - \mu_w \mu_j\right)\right] \qquad (6.7) \\
&= \sum_{i,j} E\left[(a_i x_i - \mu_w \mu_i)(b_j x_j - \mu_w \mu_j)\right] \\
&= \sum_{i,j} \mathrm{E}(a_i b_j x_i x_j) - \mu_w \mu_j \mathrm{E}(a_i x_i) - \mu_w \mu_i \mathrm{E}(b_j x_j) + \mu_w^2 \mu_i \mu_j] \qquad (6.8) \\
&= \sum_{i,j} \mu_w^2 \mathrm{E}(x_i x_j) - \mu_w^2 \mu_i \mu_j \\
&= \mu_w^2 \sum_{i,j} C_{i,j} \qquad (6.9)
\end{aligned}$$

En remarquant que

$$\mathrm{E}(a_i a_j) = \begin{cases} \mu_w^2 + \sigma_w^2 & \text{si } i = j \\ \mu_w^2 & \text{sinon,} \end{cases}$$

la variance de y (ou de z de manière similaire) est donnée par :

$$\begin{aligned}
\mathrm{V}(y) &= E\left[(y-\mathrm{E}(y))^2\right] \\
&= \sum_{i,j} E\left[(a_i x_i - \mu_w \mu_i)((a_j x_j - \mu_w \mu_j)\right] \\
&= \sum_{i,j} \mathrm{E}(a_i a_j)\mathrm{E}(x_i x_j) - \mu_w^2 \mu_i \mu_j \\
&= \sum_{i,j} \mathrm{E}(a_i a_j)(C_{i,j} + \mu_i \mu_j) - \mu_w^2 \mu_i \mu_j \\
&= \sum_{i \neq j} \mu_w^2 (C_{i,j} + \mu_i \mu_j) - \mu_w^2 \mu_i \mu_j + \sum_i \mu_w^2 (C_{i,i} + \mu_i^2) \quad - \mu_w^2 \mu_i^2 \quad + \sigma_w^2 (C_{i,i} + \mu_i^2) \\
&= \sigma_w^2 \sum_i (C_{i,i} + \mu_i^2) + \mu_w^2 \sum_{i,j} C_{i,j} \qquad (6.10)
\end{aligned}$$

On a donc bien :

$$\mathrm{corr}(x,y) = \frac{\mathrm{Cov}(y,z)}{\sqrt{\mathrm{V}(y)\mathrm{V}(z)}} = \frac{\mu_w^2}{\mu_w^2 + \sigma_w^2 \rho}.$$

□

D'après l'Equation 6.5, on peut voir que les caractéristiques ne sont pas corrélées si $\mu_w = 0$ comme supposé dans [Glorot et Bengio, 2010, He *et al.*, 2015]. Dans le cas d'un réseau à poids positifs, cela n'est pas possible.

Pour estimer la corrélation pour un réseau à poids positifs, prenons l'exemple d'un réseau à deux couches de convolution. On suppose que l'entrée x suit une loi $\mathcal{N}(0,1)$ et que ses canaux sont indépendants. On posera également les biais nuls. Pour deux caractéristiques y et z en sortie de la première convolution on a :

— $corr_1 = \mathrm{corr}(y,z) = \frac{\mu_{w_1}^2}{\mu_{w_1}^2 + \sigma_{w_1}^2}$, d'après l'Equation 6.5

— $\mathrm{E}(y) = t_a = 0$ et $\mathrm{E}(z) = 0$, d'après l'Equation 6.6

— $\mathrm{V}(y) = \mathrm{V}(z) = \sigma_{w_1}^2 n_1 + \mu_{w_1}^2 n_1$, d'après l'Equation 6.10

— $\mathrm{Cov}(y, z) = \mu_{w_1}^2 n_1$, d'après l'Equation 6.9

où n_1 est le nombre de coefficients non nuls de la première convolution (c'est-à-dire le nombre de canaux en entrée multiplié par la dimension du filtre) et w_1 correspond aux poids de cette première convolution.

La corrélation entre deux canaux y' et z' après la deuxième convolution vaut donc $corr_2 = \mathrm{corr}(y', z') = \frac{\mu_{w_2}^2}{\mu_{w_2}^2 + \sigma_{w_2}^2 \rho_1}$ avec :

$$\begin{aligned}
\rho_1 &= \frac{\sum_{i=0}^{n_2} \sigma_{w_1}^2 n_1 + \mu_{w_1}^2 n_1}{\sum_{i=0}^{n_2} \sigma_{w_1}^2 n_1 + \mu_{w_1}^2 n_1 + \sum_{i \neq j} \mu_{w_1}^2 n_1} \\
&= \frac{n_1 n_2 (\sigma_{w_1}^2 + \mu_{w_1}^2)}{n_1 n_2 (\sigma_{w_1}^2 + \mu_{w_1}^2) + n_1 (n_2^2 - n_2) \mu_{w_1}^2 n_1} \\
&= \frac{\sigma_{w_1}^2 + \mu_{w_1}^2}{\sigma_{w_1}^2 + \mu_{w_1}^2 + (n_2 - 1)\mu_{w_1}^2}
\end{aligned}$$

Dans la Figure 6.2, nous avons tracé la corrélation de ces caractéristiques ($corr_1$ et $corr_2$) en fonction de $\frac{\sigma_w^2}{\mu_w^2}$ pour plusieurs valeurs de n_2. Cette figure montre que les caractéristiques sont fortement corrélées dans la deuxième couche, même pour un grand rapport $\frac{\sigma_w^2}{\mu_w^2}$. Cette corrélation des caractéristiques en sortie d'une couche linéaire à poids positifs invalide la condition d'indépendance pour l'utilisation de l'initialisation aléatoire de Kaiming ou Xavier et explique pourquoi ces méthodes ne peuvent pas être utilisées dans ce contexte.

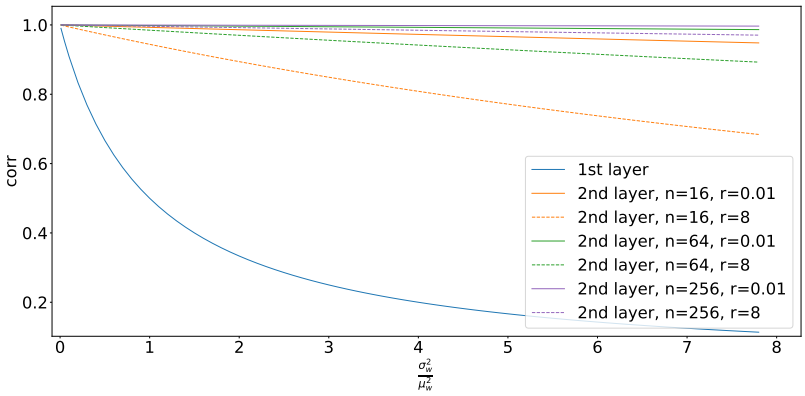


FIGURE 6.2 – *Corrélation entre les caractéristiques pour les deux premières couches en fonction de $\frac{\sigma_w^2}{\mu_w^2}$ pour différentes tailles de support des poids n de la couche concernée. Pour les deuxièmes couches, deux valeurs de $r = \frac{\sigma_{w_1}^2}{\mu_{w_1}^2}$ ont été tracées.*

III.2 Procédure pour l'initialisation

Nous proposons d'adopter une méthode programmatique simple pour initialiser les poids d'un réseau profond. Le but de cette initialisation est de préserver la variance des

caractéristiques sur l'ensemble du réseau comme dans [Glorot et Bengio, 2010, He *et al.*, 2015] mais sans hypothèse sur la distribution d'entrée de chaque couche. En d'autres termes, la procédure proposée permet de trouver, pour chaque couche linéaire (FC/Conv), le facteur par lequel les poids doivent être divisés pour que la variance de la sortie soit égale à un sans hypothèse sur la distribution des caractéristiques d'entrée.

Pour ce faire, les poids sont d'abord initialisés à l'aide d'une initialisation aléatoire standard comme [Glorot et Bengio, 2010, He *et al.*, 2015]. Nous générons également un *minibatch* x_{train} de données de bruit suivant une distribution gaussienne $\mathcal{N}(0,1)$ comme entrée du réseau. Il est possible de choisir n'importe quelle distribution, mais il est courant de centrer-réduire les données d'entrée des réseaux de neurones. Ensuite, à partir de x_{train}, nous suivons le graphe de calcul du réseau et divisons par un facteur, dans l'ordre, les poids de chaque couche linéaire de manière à ce que sa sortie ait une variance unitaire. Ce facteur de normalisation est donné par l'écart-type de la sortie (puisque $\mathrm{V}(Wx/\sqrt{\mathrm{V}(Wx)}) = 1$). Enfin, la sortie est calculée une seconde fois avec les nouveaux poids et utilisée comme entrée de la couche suivante. Cette opération est répétée sur toutes les couches restantes du graphe de calcul.

Notre méthode a l'avantage de prendre en compte l'architecture globale du réseau puisque l'entrée de chaque couche suit le graphe de calcul. Il n'est pas supposé que l'entrée de la couche linéaire soit normalisée (ou normalisée suivie d'une activation non linéaire). Ainsi, si un bloc résiduel ou une couche de *maxpooling* précède la couche linéaire, les poids seront naturellement ajustés. Cette méthode ne présume pas non plus de l'indépendance des caractéristiques d'entrée et peut donc fonctionner pour un réseau avec des poids positifs.

Du point de vue de la mise en œuvre, cette initialisation peut être effectuée en un seul passage dans le réseau. Chaque couche linéaire $y = Wx$ est remplacée par une fonction *wrapper* pour l'initialisation uniquement, selon l'Algorithme 1.

Algorithme 1 *Wrapper* d'initialisation

function LinearWrapper(x)
  $y \leftarrow Wx$
  $\sigma \leftarrow \sqrt{\mathrm{V}(y)}$
  $W \leftarrow W/\sigma$
  return Wx
end function

Lorsque les poids sont paramétrés par une fonction positive ($W = \varphi(\tilde{W})$) comme c'est le cas pour les réseaux à poids positifs, le *wrapper* d'initialisation présenté dans l'Algorithme 1 doit être adapté comme donné dans l'Algorithme 2. On peut noter que nous n'avons pas

Algorithme 2 *Wrapper* d'initialisation pour des poids paramétrés

function PositiveLinearWrapper(x)
  $y = \varphi(\tilde{W})x$
  $\sigma = \sqrt{\mathrm{V}(y)}$
  $\tilde{W} = \varphi^{-1}(\varphi(\tilde{W})/\sigma)$
  return $\varphi(\tilde{W})x$
end function

vraiment besoin de la fonction inverse réelle φ^{-1} de la fonction φ mais seulement de son inverse à droite, comme l'exige le module de paramétrage des poids de Pytorch.

Une fois ces *wrappers* mis en place, le modèle est exécuté avec l'entrée aléatoire x_{train}. Enfin, les couches linéaires (normalisées) sont remises à leur place initiale dans le réseau. En pratique, notre méthode d'initialisation peut être utilisée sur n'importe quel modèle,

indépendamment de l'architecture et de la distribution des caractéristiques d'entrée.

Une procédure d'initialisation des poids basée sur une normalisation des poids à l'aide de la variance en sortie de couche a déjà été proposée dans [Mishkin et Matas, 2015]. S'il y a de nombreuses similarités, on peut néanmoins énumérer plusieurs différences significatives. Tout d'abord, à chaque normalisation d'une couche, le réseau est parcouru dans son entièreté pour obtenir la sortie de la couche concernée alors qu'avec notre implémentation, lorsqu'une couche est normalisée, on ne repasse que dans cette couche. L'implémentation de [Mishkin et Matas, 2015] demande donc pour n couches, le passage à travers n^2 couches alors que la notre ne demande le passage que par $2n$ couches. Notre implémention réduit considérablement le temps nécessaire pour l'initialisation. Ensuite, alors que nous utilisons un *minibatch* de bruit aléatoire x_{train}, [Mishkin et Matas, 2015] utilise un *minibatch* issue des données d'entraînement. Le risque est ici que ce lot ne soit pas représentatif des données et qu'un *overfitting* soit fait sur ce dernier. Pour remédier à ce problème, le processus le normalisation (division des poids par l'écart-type de la sortie de la couche) est répété avec plusieurs lots jusqu'à ce que la variance obtenue en sortie de couche soit quasi-unitaire (une marge est tolérée). Cet algorithme n'est, en fait, pas un algorithme itératif comme on l'entend habituellement : puisque la variance sera bien unitaire pour le lot servant à la normalisation, la condition d'arrêt revient à ce que la sortie de la couche pour le lot suivant ait une variance quasi-unitaire après la normalisation avec le lot précédent. Néanmoins, cette condition ne garantit en rien que ces deux lots consécutifs sont représentatifs de l'ensemble de données et les itérations avant ces lots sont réalisées inutilement, avec un passage dans l'ensemble du réseau supplémentaire pour chaque *minibatch* tiré. Si la volonté était ici d'approximer la distribution des données d'entrée (où nous, nous supposons qu'elles sont centrées-réduites) en supposant que si deux *minibatches* sont de distributions similaires alors ils sont représentatifs, il serait plus efficace de calculer la moyenne et l'écart-type des données au préalable, puis d'utiliser notre méthode avec x_{train} suivant une loi normale avec la moyenne et l'écart-type calculés. Notre approche est donc beaucoup plus rapide. En effet, pour un ResNet152 en 2D et un *batch* de taille 50, le temps d'initialisation est de 12s pour notre méthode plutôt que 22min pour celle de [Mishkin et Matas, 2015]. A noter que dans l'implémentation Pytorch utilisée [1] pour reproduire les résultats de [Mishkin et Matas, 2015], un unique lot est utilisé. Il n'y a donc qu'une seule itération dans la boucle sur les différents lots puisque la variance pour le lot sera bien unitaire après normalisation avec ce même lot. Le temps de calcul de cette implémentation Pytorch est déjà réduit par rapport au papier original.

IV Explicabilité des réseaux monotones

Nous nous plaçons ici dans le cadre d'une classification binaire entre des images saines (classe négative) et des images de patients (classe positive). Dans ce cas, un réseau dit interprétable baserait sa décision sur les signes radiologiques de la pathologie.

IV.1 Contraintes pour un réseau interprétable

Pour rendre notre réseau interprétable, nous proposons d'utiliser différentes contraintes, basées sur les propriétés intéressantes des réseaux monotones et sous la forme de fonctions de coût ajoutées à la fonction de coût de classification lors de l'entraînement, de manière similaire au Chapitre 5. Notre réseau est ainsi entraîné avec la fonction de coût :

$$L = L_C + L_{nF} + L_{dF} + \alpha_G L_G \tag{6.11}$$

où L_C est la fonction de coût de classification (ici une entropie croisée binaire), L_{nF} et L_{dF} imposent des contraintes sur les caractéristiques données en entrée de la partie monotone

1. https://github.com/ducha-aiki/LSUV-pytorch

du réseau et enfin L_G est une régularisation des gradients de cette même partie monotone, pondérée par α_G.

Ces contraintes ne demandent aucune annotation supplémentaire : seul le label des images, nécessaire pour la tâche de classification, est requis. A l'inférence, ce réseau permet une classification plus interprétable dont la décision est davantage basée sur les signes radiologiques de la pathologie mais aussi une segmentation faiblement supervisée de ces dernières.

IV.1.1 Contrainte de négativité sur les caractéristiques des sujets sains

En imposant les biais nuls dans la partie monotone, nous avons vu que, pour une entrée complètement négative, la sortie du réseau monotone serait négative. Nous proposons donc de contraindre les cartes de caractéristiques données en entrée de la partie monotone à être négatives pour les images saines pour lesquelles on souhaite une sortie négative pour la classification. Pour cela, nous utilisons la fonction de coût suivante sur les images saines x_0 :

$$L_{nF}(x_0) = \|ReLU(C(x_0))\|_1 \tag{6.12}$$

où $C(x_0)$ correspond aux caractéristiques interprétables obtenues par la partie du réseau non-monotone et servant d'entrée à la partie monotone.

A travers cette fonction, nous souhaitons supprimer les éléments positifs de ces cartes pour les images saines. L'entropie croisée utilisée dans [Wargnier-Dauchelle *et al.*, 2023b] (Chapitre 5) serait moins adaptée puisqu'elle augmenterait considérablement l'échelle des caractéristiques (en tirant ces dernières vers $-\infty$) alors qu'ici l'échelle naturelle du réseau est conservée. Pour cette contrainte, il est important de contraindre l'encodeur C à avoir des biais nuls afin d'éviter qu'elle ne se traduise pas uniquement par l'apprentissage de biais très négatifs.

IV.1.2 Contrainte de distributions similaires des caractéristiques des deux classes

Les images saines et pathologiques devraient partager des caractéristiques communes puisqu'en dehors de la pathologie, les tissus sont sains. Nous souhaitons que dans les zones en dehors de la pathologie, les images soient encodées de la même manière au niveau des caractéristiques interprétables. Avec la contrainte précédente, les tissus sains devraient être encodés par des caractéristiques négatives. Nous voulons donc que les distributions des caractéristiques négatives soient similaires pour les deux classes. Pour cela, nous proposons d'utiliser une fonction de coût de type Kullback-Leibler sur les distributions des caractéristiques qui sont négatives :

$$L_{dF}(x_0, x_1) = KL(P_0, P_1)) \tag{6.13}$$

où KL est la divergence de Kullback-Leibler et P_0 et P_1 correspondent respectivement aux distributions des caractéristiques interprétables (celles juste avant le réseau monotone) qui sont négatives lorsque les images saines et pathologiques sont passées en entrée du réseau.

Cette contrainte permet d'éviter que le réseau ne classe les données sur une éventuelle différence sur la "partie saine" des images qui serait due à des différences sur l'acquisition ou les populations. Elle peut être vue comme une forme d'adaptation de domaine incluse dans l'apprentissage.

IV.1.3 Régularisation des gradients

Le gradient de la sortie par rapport à l'entrée de la partie monotone du réseau est positif par construction. Une adaptation de [Wargnier-Dauchelle *et al.*, 2023b] serait donc de contraindre les gradients de la partie monotone à être faibles pour les images saines plutôt

que négatifs. Cette contrainte doit être vue comme une régularisation de la même manière que dans [Varga *et al.*, 2017]. Le coefficient α_G pondérant la fonction de coût correspondante doit donc être relativement faible par rapport aux autres fonctions de coût. Nous avons ici utilisé une norme L1 pour cette régularisation :

$$L_G(x_0) = \left\| \frac{\partial F(x_0)}{\partial C(x_0)} \right\|_1 \tag{6.14}$$

où, pour rappel, F est le réseau complet et C la partie non-monotone. On pourra noter que, puisque c'est le gradient de la partie monotone du réseau qui apparaît dans L_G, on retrouve ici la somme des dérivées de la sortie par rapport à chaque logit de l'entrée sans la valeur absolue.

IV.2 Lecture des caractéristiques interprétables

Pour étudier l'interprétabilité des réseaux, nous proposons de générer un exemple contrefactuel sur les caractéristiques interprétables données en entrée du réseau monotone. L'objectif est de trouver les modifications minimales nécessaires à effectuer sur ces caractéristiques pour passer d'une classification "pathologique" à une classification "sain". Nous recherchons cette modification sous la forme d'un tenseur α tel que :

$$M(f - \alpha) < m \tag{6.15}$$

où $f = C(x)$ correspond aux caractéristiques interprétables, c'est-à-dire à la sortie du réseau convolutif non-monotone placé devant le réseau monotone pour une image x, M est le réseau monotone, α est de la même dimension que f et représente les modifications à apporter pour changer de classe et $m \leq 0$ est une marge permettant de s'éloigner de la frontière de décision.

On peut noter que pour un réseau monotone, il existe un α complètement positif qui permet de passer de la classe positive à la classe négative. Cela permet une lecture plus aisée et donc un réseau plus facilement explicable. Pour la recherche de cet exemple, nous nous inspirons de la méthode basique proposée par [Wachter *et al.*, 2017] en cherchant α, par descente de gradient, minimisant l'expression suivante :

$$\min_{\alpha} M(f - \alpha) + \lambda \|\alpha\|_1 \tag{6.16}$$

où λ pondère les deux parties de la fonction de coût. Cette fonction est optimisée jusqu'à obtenir la condition voulue $M(f - \alpha) < m$.

V Protocole expérimental

L'évaluation de notre méthode est faite en deux étapes. Tout d'abord, l'influence de l'initialisation des poids pour les réseaux monotones et non-monotones a été étudiée. Pour cela, nous nous sommes comparés à une initialisation des poids avec la méthode de Kaiming [He *et al.*, 2015] qui reste la méthode la plus utilisée. Ensuite, nous avons évalué l'interprétabilité de notre méthode et ses capacités de segmentation sur une tâche de classification d'images saines vs tumorales, en nous comparant aux meilleures méthodes du Chapitre 5 à savoir : l'auto-encodeur (AE) [Baur *et al.*, 2018], Silva-Rodríguez [Silva-Rodríguez *et al.*, 2021], Ross [Ross *et al.*, 2017] et notre méthode présentée dans le Chapitre 5, Wargnier-Dauchelle [Wargnier-Dauchelle *et al.*, 2023b]. Nous avons également établi deux références : un réseau de classification suivant la même architecture que notre proposition mais non-monotone et non-contraint (Baseline) et un réseau non-monotone mais contraint avec les fonctions de coût L_{dF} et L_{nF} de la Section IV.1. La fonction de coût L_G n'a pas été ajoutée car

elle empêchait la convergence sur la tâche de classification et n'avait pas de sens puisque, pour un réseau non-monotone, les gradients ne sont pas positifs. Pour notre méthode et les deux références, la segmentation a été faite à partir de la différence nécessaire (α dans l'Equation 6.15) pour générer un exemple contrefactuel sur les caractéristiques interprétables.

V.1 Données

Nous avons utilisé plusieurs bases de données médicales mélangeant différentes modalités d'imagerie, tâches de classification et dimensionnalités des données. Le Tableau 6.1 résume les bases et leur répartition selon les jeux d'entraînement, de validation et de test.

MedNIST Tout d'abord, nous avons utilisé la base de données "jouet" publique d'images médicales MedNIST [1]. Elle est composée d'images 2D (64×64) réparties en 6 classes : CT de l'abdomen, IRM du sein, CT du thorax, radiographie du thorax, radiographie de la main et CT du cerveau.

Brain tumors Nous avons également utilisé une base de données IRM présentant des tumeurs cérébrales [Bhuvaji *et al.*, 2020]. Elle est composée d'images 2D (512×512) d'IRM T1 (avec et sans rehaussement de contraste), T2 et FLAIR (vues axiale, coronale et sagitale). Il y a 4 classes : une sans tumeur dans l'image et 3 types différents de tumeurs cérébrales : les gliomes, méningiomes et tumeurs pituitaires.

Single-Cell La base de données protéomiques unicellulaires publiques [Levine *et al.*, 2015] propose 13 valeurs de marqueurs de surface cellulaire pour 20 types de cellules saines différentes.

Classification sain/tumeur Pour la classification des images saines vs des images de patients avec tumeurs cérébrales, comme dans le Chapitre 5, nous avons utilisé des IRM cérébrales FLAIR de sujets sains (MPI, kirby21 et IBC) et de patients atteints de tumeurs cérébrales (BraTS2020). Les volumes 3D ont été prétraités selon la chaîne de prétraitements décrite en Section III.2 du Chapitre 1 sans la correction des inhomogénéités de champ. Une résolution voxélique de 2mm^3 été utilisée, conduisant à des images de taille $91 \times 109 \times 91$.

Tableau 6.1 – *Bases de données utilisées pour chaque expérience avec la répartition entre les différents jeux, le nombre de classes, la dimensionnalité et les modalités d'imagerie utilisées. Pour l'expérience de classification sain/tumeur, H désigne les bases saines et T la base de tumeurs cérébrales.*

Base/expérience	N_{train}	N_{val}	N_{test}	Nb classes	Dim	Modalité
MedNIST	47163	5895	5995	6	2D	IRM, CT, Rayon X
H : MPI, kirby21, IBC	64, 22, 8	15, 5, 2	15, 5, 2			
T : BraTS20	280	40	49	2	3D	IRM
Brain Tumors	2870	99	295	4	2D	IRM
Single-Cell	58374	6486	16215	20	1D	Marqueurs cellulaires

1. La base de données MedNIST a été constituée à partir de plusieurs bases issues du TCIA, du RSNA Bone Age Challenge et de de la base Chest X-ray du NIH. Cette base est généreusement mis à dispositon par le Dr Bradley J. Erickson M.D., Ph.D. (Département de radiologie, Mayo Clinic) sous licence Creative Commons CC BY-SA 4.0.

V.2 Implémentation

Les réseaux ont été implémentés sous Pytorch. Plusieurs architectures de classification ont été utilisées : un PatchGan 3D 70x70 [Isola *et al.*, 2017] et des ResNets 2D [He *et al.*, 2016] (de 18 à 152 couches) qui ont été convertis ou non en un réseau monotone (comme décrit dans la Section I) et un perceptron multicouche monotone (MLP) [Nguyen *et al.*, 2023]. Pour les réseaux monotones, l'encodeur non-monotone est une convolution simple suivie d'une normalisation par instance et d'une activation LeakyReLU pour les expériences sur l'initilisation et ce sont deux couches convolutives à 8 canaux qui sont utilisées (avec respectivement une taille de noyau de 7 et 3) avec la même activation et normalisation pour les autres expériences. PatchGAN a été utilisé pour les classifications pathologiques vs saines avec l'optimiseur Adadelta [Zeiler, 2012] et un *learning rate* initial fixé à 1. Pour la tâche sur MedNIST, toutes les tailles de ResNet ont été utilisées et un ResNet152 entraîné avec Adam [Kingma et Ba, 2014] et un *learning rate* fixé à 10^{-5} a été utilisé pour l'évaluation de l'*accuracy* sur la base de test. Un ResNet34 a été utilisé pour la classification des types de tumeurs (Brain tumors) avec l'optimiseur Adam et un *learning rate* de 10^{-3}. Enfin, le MLP monotone a été entraîné sur Single-Cell comme dans [Nguyen *et al.*, 2023].

Pour obtenir les distributions des caractéristiques interprétables utilisées dans la Section IV.1.2, nous avons calculé leur histogramme (avec 50 *bins*) différentiable en utilisant une fonction triangle pour la répartition entre deux *bins* consécutifs (fenêtre de Parzen).

Pour la génération des exemples contrefactuels, l'optimiseur Adam a été utilisé avec un *learning rate* de 10^{-5}. Nous avons choisi une marge $m = -15$ et une pondération $\lambda = 10^{-4}$.

V.3 Métriques

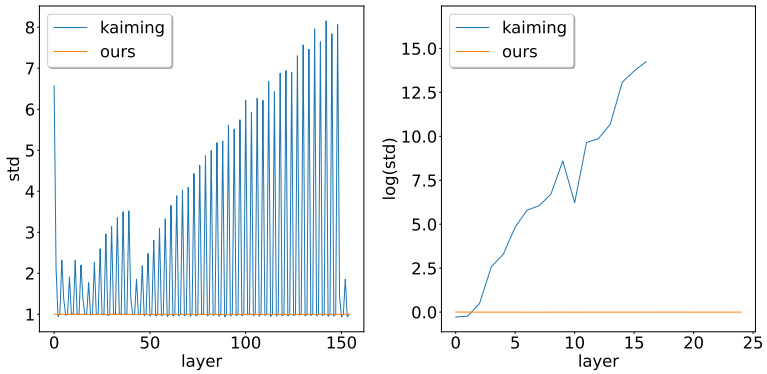
Les performances de classification ont été mesurées par l'*accuracy* moyenne (acc) et l'aire sous la courbe ROC (auc). Dans la Section VI.2.2, nous détaillons l'*accuracy* pour la classe saine (TNR) et la classe pathologique (TPR). Pour la détection des anomalies, la qualité des cartes de segmentation a été mesurée à l'aide du Dice [Sorensen, 1948] entre les cartes seuillées et le masque de vérité terrain, l'aire sous la courbe de précision-rappel (AUPRC) et la courbe réceptive-opératoire (AUROC) en considérant une classification au niveau du voxel. Les seuils ont été choisis comme point de fonctionnement de la PRC sur l'ensemble des données de validation pour chaque canal. Dans la Section VI.2.2, le meilleur canal sur la validation a été choisi pour notre proposition et les références (Baseline et BaselineC). La corrélation a été mesurée à l'aide du coefficient de Pearson.

VI Résultats

VI.1 Influence de l'initialisation des poids

VI.1.1 Influence sur la variance des cartes de caractéristiques

Une bonne initialisation des poids d'un réseau devrait maintenir la variance des caractéristiques stable dans chaque couche. En utilisant les méthodes de l'état de l'art comme Kaiming [He *et al.*, 2015], la variance n'est pas conservée dans certaines couches comme nous l'avons montré en Section III.1. Au fur et à mesure que nous accumulons ces couches dans une architecture profonde, l'écart avec la variance originale augmente. Dans la Figure 6.3, nous représentons l'écart-type des caractéristiques internes pour chaque couche dans un ResNet152 après l'initialisation de Kaiming et après notre procédure. Les caractéristiques sont calculées à l'aide d'une entrée aléatoire suivant une loi $\mathcal{N}(0, 1)$, différente de l'entrée x_{train} utilisée pour la procédure d'initialisation. On peut voir que la variance augmente considérablement avec la profondeur des couches lorsque l'initialisation de Kaiming est



(a) Non-monotone (b) Monotone

FIGURE 6.3 – *Ecart-type des caractéristiques en fonction de la profondeur dans le réseau de la couche étudiée pour un ResNet152 et une entrée aléatoire tirée dans $\mathcal{N}(0,1)$ (différente de celle utilisée pour notre initialisation), soit avec l'initialition de Kaiming soit avec celle proposée.*

utilisée mais reste constante pour notre méthode : elle est 8 fois plus élevée pour Kaiming sur les réseaux classiques (Figure 6.3a).

Ceci est encore plus vrai pour les réseaux monotones (Figure 6.3b) pour lesquels la variance tend vers l'infini à partir d'une certaine couche. Comme le montre la Section III.1.3, imposer des poids positifs à un réseau augmente la corrélation entre les canaux de ses caractéristiques internes. La Figure 6.4 montre la corrélation de ces canaux dans le discriminateur PatchGAN sous sa forme monotone et non-monotone. On constate que les canaux de la forme non-monotone ne sont pas corrélés alors que pour le réseau monotone, la corrélation passe de 50% pour la première couche à plus de 80% pour la dernière. Ainsi, la condition nécessaire pour pouvoir utiliser Kaiming [He *et al.*, 2015] ou Xavier [Glorot et Bengio, 2010] sur ce point est fausse pour les réseaux monotones. Cela pourrait donc expliquer pourquoi elle n'est pas appropriée pour ce type de réseaux et pourquoi la variance diverge.

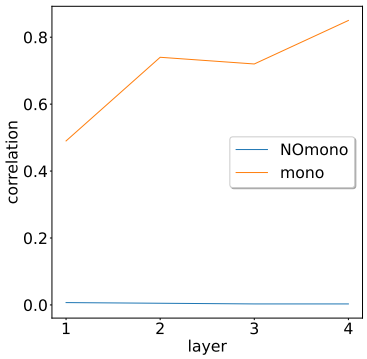
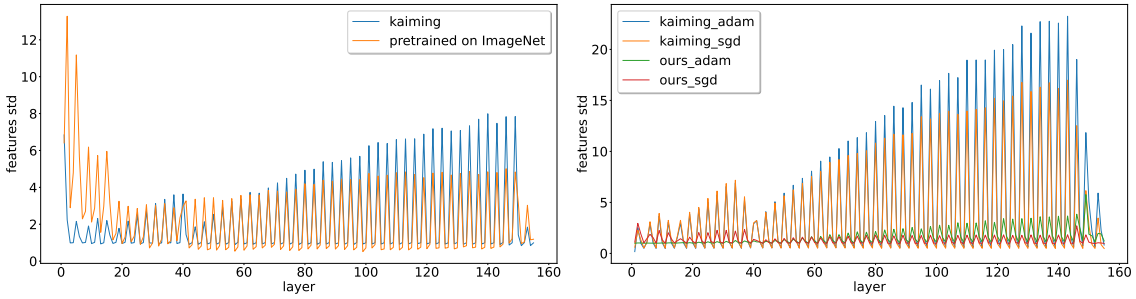


FIGURE 6.4 – *Corrélation entre les cartes de caractéristiques pour un PatchGAN monotone (mono) et non-monotone (NOmono).*

La variance initiale est très importante car les poids conserveront une variance similaire au cours de l'apprentissage. Dans la Figure 6.5, nous représentons l'écart-type des caractéristiques de modèles entraînés. Elle montre qu'avec un ResNet152, très utilisé dans la communauté, pré-entraîné sur ImageNet (Figure 6.5a), la variance est plus stable qu'avec les poids originaux de l'initialisation de Kaiming. Néanmoins, la variance des caractéristiques reste proche de l'initialisation originale. En effet, si l'on compare l'écart-type des caractéris-

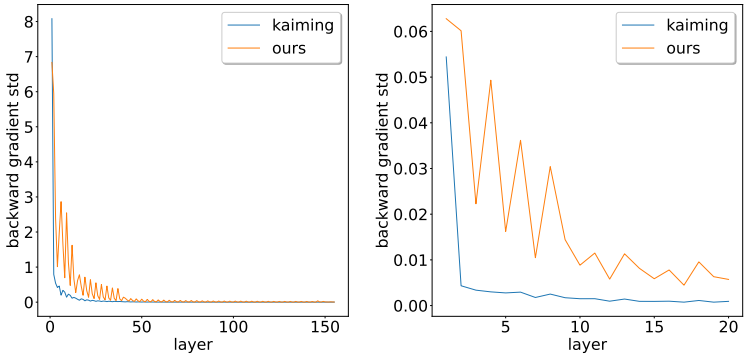
tiques pour un modèle entraîné après l'initialisation de Kaiming ou après notre proposition (Figure 6.5b), on voit qu'il est plus faible avec notre initialisation comme c'était le cas avant l'entraînement. La distribution des caractéristiques est similaire à celle de l'initialisation. Une bonne méthode d'initialisation est donc importante même dans un contexte de *fine-tuning* pour le pré-entraînement. Nous pouvons également remarquer que la variation de l'écart-type entre l'initialisation et la fin de l'entraînement, est plus importante avec l'optimiseur Adam [Kingma et Ba, 2014] qu'avec SGD.



(a) ResNet152 pré-entraîné sur ImageNet (b) ResNet152 entraîné

FIGURE 6.5 – *Ecart-type des caractéristiques en fonction de la profondeur dans un ResNet152 de la couche étudiée pour une entrée aléatoire tirée dans $\mathcal{N}(0,1)$. A gauche : comparaison entre l'initialisation de Kaiming et le modèle pré-entraîné sur ImageNet. A droite : comparaison entre les modèles entraînés sur MedNIST après une initialisation avec Kaiming ou celle proposée, avec deux optimiseurs (SGD et Adam).*

VI.1.2 Influence sur le gradient



(a) ResNet152 (b) ResNet18

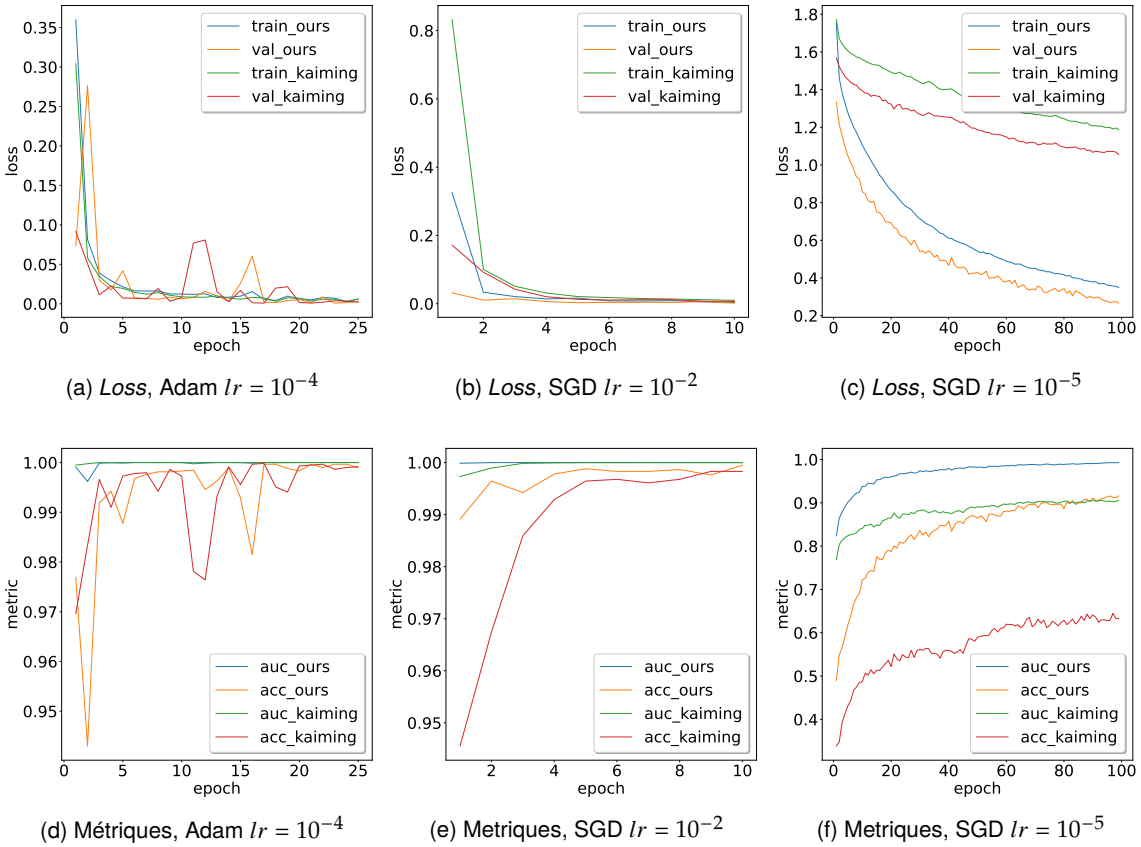
FIGURE 6.6 – *Comparaison entre l'initialisation de Kaiming et la notre sur l'écart-type du premier gradient de rétropropagation en fonction de la profondeur dans le réseau de la couche étudiée pour un ResNet152 (à gauche) et un ResNet18 (à droite).*

L'objectif final de la préservation de la variance et donc de l'initialisation des poids est d'éviter l'évanescence ou l'explosion du gradient. Dans la Figure 6.6, nous étudions l'écart-type (la moyenne est nulle) du gradient de la rétropropagation à travers les couches juste après une initialisation avec Kaiming ou notre initialisation. Les résultats montrent qu'avec Kaiming, le gradient croît de manière exponentielle vers les premières couches alors que la croissance est plus douce avec notre initialisation, en particulier avec l'architecture très

profonde ResNet152. Ainsi, l'initialisation proposée est moins susceptible de voir exploser le gradient.

VI.1.3 Influence sur la convergence en fonction de l'optimiseur et du *learning rate*

Pour évaluer l'influence de l'initialisation sur la convergence des modèles, nous avons choisi de les entraîner avec différents optimiseurs et *learning rate* sur la tâche de classification MedNIST en comparant l'initialisation proposée à celle de Kaiming. Les courbes d'apprentissage sont présentées dans la Figure 6.7 pour trois expériences utilisant soit l'optimiseur Adam avec son *learning rate* optimal (10^{-4}), soit SGD avec un *learning rate* élevé et un faible (10^{-2} et 10^{-5} respectivement).



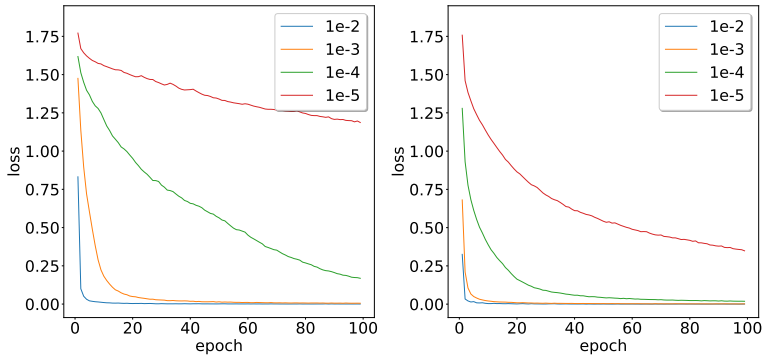
(a) *Loss*, Adam $lr = 10^{-4}$ (b) *Loss*, SGD $lr = 10^{-2}$ (c) *Loss*, SGD $lr = 10^{-5}$

(d) Métriques, Adam $lr = 10^{-4}$ (e) Metriques, SGD $lr = 10^{-2}$ (f) Metriques, SGD $lr = 10^{-5}$

FIGURE 6.7 – *Évolution de la fonction de coût (sur le jeu d'entraînement et de validation) en haut et de l'AUC et de l'accuracy en bas pour un ResNet152 entraîné sur MedNIST après une initialisation avec Kaiming ou notre proposition. Plusieurs optimiseurs sont testés : Adam avec un learning rate de* 10^{-4} *(optimal) et SGD avec un learning rate de* 10^{-2} *et* 10^{-5}*.*

Les deux initialisations sont équivalentes avec l'optimiseur Adam en termes de convergence. Avec SGD et un *learning rate* élevé, notre proposition semble un peu meilleure avec une convergence plus rapide de la fonction de coût et des métriques (en particulier l'*accuracy*). Cette différence entre les deux initialisations est plus importante pour un *learning rate* faible. En effet, avec notre initialisation, la convergence du modèle est meilleure puisque la fonction de coût diminue plus rapidement et les performances sur la validation (auc et acc) sont entre 10% et 30% plus élevées à la fin de l'apprentissage. Ainsi, Adam est capable d'effacer rapidement (avant la fin de la première *epoch*) l'avantage de notre initialisation visible avec SGD. Or, SGD est connu pour être moins enclin à l'*overfitting* qu'Adam et est souvent préférable.

Notre proposition semble également moins dépendante du *learning rate* que l'initialisation de Kaiming. La Figure 6.8 montre la convergence de la fonction de coût pour les deux initialisations avec différents *learning rate* et SGD. La convergence est toujours meilleure avec notre proposition. En effet, la convergence finale y est atteinte en moins de 100 epochs avec la plupart des *learning rate*.

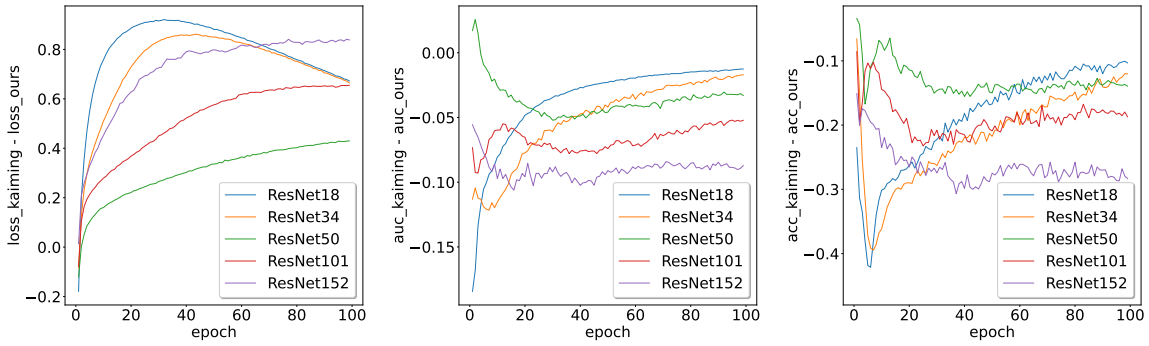


(a) Kaiming (b) Notre proposition

FIGURE 6.8 – *Évolution de la fonction de coût d'entraînement pour un ResNet152 entraîné sur MedNIST après une initialisation avec Kaiming (gauche) ou la notre (droite) pour différents learning rate et SGD.*

VI.1.4 Influence sur la convergence en fonction de la profondeur de l'architecture

Dans cette section, nous évaluons l'influence de notre initialisation en fonction de la profondeur de l'architecture. Pour cela, nous avons entraîné plusieurs ResNets sur la tâche de classification MedNIST. La Figure 6.9 montre la différence entre Kaiming et notre initialisation.



(a) Loss (b) AUC (c) Accuracy

FIGURE 6.9 – *Évolution de la différence de fonction de coût, d'AUC et d'accuracy entre un modèle initialisé avec Kaiming ou notre méthode pour différentes profondeurs de ResNet.*

La convergence est toujours meilleure avec notre proposition puisque la différence entre les fonctions de coût est positive et les différences de métriques sont négatives. La différence est plus importante avec les architectures plus profondes comme un ResNet152. Nous constatons une différence entre les architectures à *bottleneck* (ResNet50, 121 et 152) et les architectures classiques. En effet, pour les architectures sans *bottleneck*, la différence est la plus

importante au début de l'apprentissage, puis les modèles initialisés avec Kaiming semblent rattraper les modèles initialisés avec notre méthode.

VI.1.5 Influence sur la convergence en fonction de couches de normalisation

L'initialisation de Kaiming ne préserve pas la variance dans chaque couche. Les effets de ce problème sont réduits grâce aux couches de normalisation. Pour évaluer l'impact de ces couches, nous les retirons d'un ResNet152 entraîné sur MedNIST. Dans ce cas, le modèle diverge lorsqu'il est initialisé avec Kaiming alors qu'il atteint une meilleure convergence que le modèle avec les couches de normalisation lorsque notre initialisation est utilisée, comme le montre la Figure 6.10. Ainsi, les couches de normalisation semblent moins essentielles avec notre initialisation.

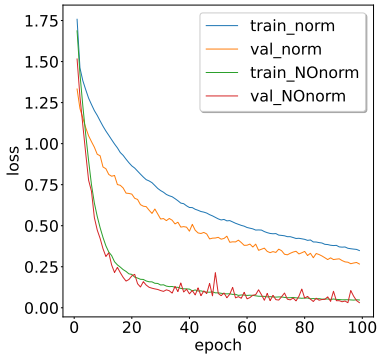


Figure 6.10 – *Évolution de la fonction de coût avec (norm) et sans (NOnorm) couche de normalisation dans un ResNet152 avec notre initialisation.*

VI.1.6 Influence sur les performances

Une comparaison de l'*accuracy* de classification sur le jeu de test pour plusieurs architectures (monotones ou non) et tâches de classification est présentée dans le Tableau 6.2.

Tableau 6.2 – *Accuracy sur le jeu de test pour les réseaux monotones ou non sur les différentes tâches. Une comparaison est faite entre une initialisation avec Kaiming ou avec notre méthode. NaN signifie que le modèle a divergé pendant l'entraînement.*

Tâche	Standard		Monotone	
	Kaiming	Nous	Kaiming	Nous
H vs T (PatchGAN)	**1.00**	**1.00**	0.98	**1.00**
Single Cell (MLP)	/	/	0.83	**0.92**
MedNIST (ResNet152)	**1.00**	**1.00**	NaN	**0.98**
Brain tumors (ResNet34)	0.72	**0.73**	NaN	**0.71**

Si nous examinons l'*accuracy* des réseaux non-monotones, nous constatons que les performances sont similaires pour les deux initialisations. Néanmoins, comme nous l'avons vu précédemment, la convergence est plus rapide.

Pour les réseaux monotones, comme la variance dans les couches du réseau diverge rapidement (Figure 6.3), l'initialisation de Kaiming est encore moins adaptée. Ainsi, pour les architectures profondes comme un ResNet34 ou un ResNet152, le modèle diverge avec l'initialisation de Kaiming.

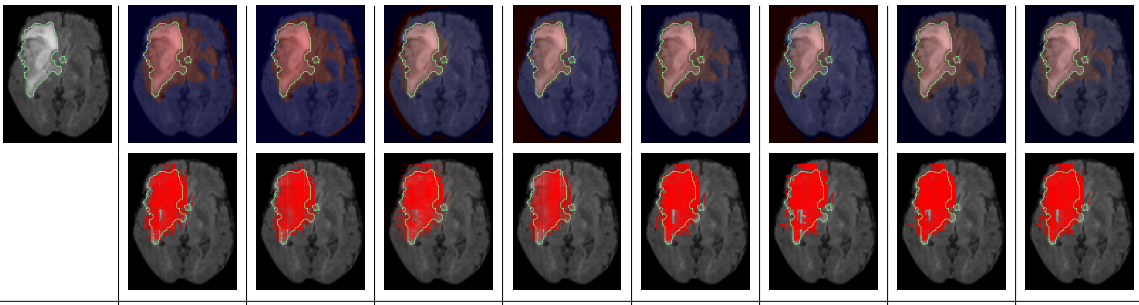
Pour les architectures plus petites, le modèle converge pour les deux initialisations mais l'*accuracy* du MLP est inférieure de 10 points avec l'initialisation de Kaiming. La convergence

est également plus rapide avec notre proposition. Par exemple, même si les *accuracies* pour la classification des images saines vs tumorales sont similaires pour les deux initialisations, la meilleure *accuracy* sur le jeu de validation est obtenue après un nombre similaire d'epochs (24 pour Kaiming et 28 pour notre initialisation) pour le réseau original, mais pour sa forme monotone, la meilleure précision est obtenue après 480 epochs pour Kaiming alors qu'elle est obtenue après seulement 182 avec notre proposition. Ainsi, toutes les architectures peuvent être converties en réseaux monotones en utilisant la méthode proposée avec une convergence du modèle et de bonnes performances.

VI.2 Interprétabilité des réseaux monotones contraints

VI.2.1 Exemples contrefactuels pour les différents canaux des caractéristiques interprétables

Nous proposons d'utiliser la différence α nécessaire pour générer l'exemple contrefactuel sur les caractéristiques interprétables afin de visualiser les zones de l'image importantes pour la décision mais aussi de l'utiliser pour segmenter de manière faiblement supervisée les tumeurs cérébrales. Cette différence α est calculée selon la méthode décrite dans la Section IV.2. Dans la Figure 6.11, on peut voir les différentes cartes de caractéristiques interprétables f, les différences contrefactuelles α correspondantes ainsi que les métriques de segmentation obtenues à partir de ces dernières lorsque notre proposition est utilisée (réseau monotone contraint avec les différentes fonctions de coût comme décrit dans la Section IV.1). On remarque que tous les canaux doivent être modifiés dans la région de la tumeur pour obtenir une image classée saine et cela, même si dans les caractéristiques, il n'y a pas que la tumeur qui est mise en avant. La décision de notre réseau semble donc pertinente. Au niveau quantitatif, les métriques valident cette hypothèse avec des Dice supérieurs à 50% pour tous les canaux.



Dice	0.55 ± 0.19	0.55 ± 0.19	0.55 ± 0.20	**0.56 ± 0.19**	0.54 ± 0.20	0.50 ± 0.19	0.56 ± 0.20	0.54 ± 0.20
AUROC	0.94	0.93	0.96	0.91	0.96	**0.97**	0.96	0.96
AUPRC	0.56	**0.58**	0.56	**0.58**	0.54	0.49	0.56	0.55

FIGURE 6.11 – *Exemple de caractéristiques interprétables (en haut), de différences contrefactuelles (α, en bas) et métriques (calculées sur toute la base de test) pour les différents canaux des caractéristiques interprétables avec notre proposition. La tumeur est délimitée en vert. L'IRM est en haut à gauche.*

VI.2.2 Détection d'anomalies faiblement supervisée

Dans la Figure 6.12, nous nous comparons visuellement aux méthodes de l'état de l'art en détection d'anomalies sur deux exemples. Pour le premier exemple, notre méthode est comparable à Ross et la méthode proposée dans le Chapitre 5 (Wargnier-Dauchelle). Néanmoins pour le deuxième exemple, notre méthode semble la plus focalisée sur la pathologie. On peut aussi noter que lorsque la génération d'exemples contrefactuels est faite sur les mêmes

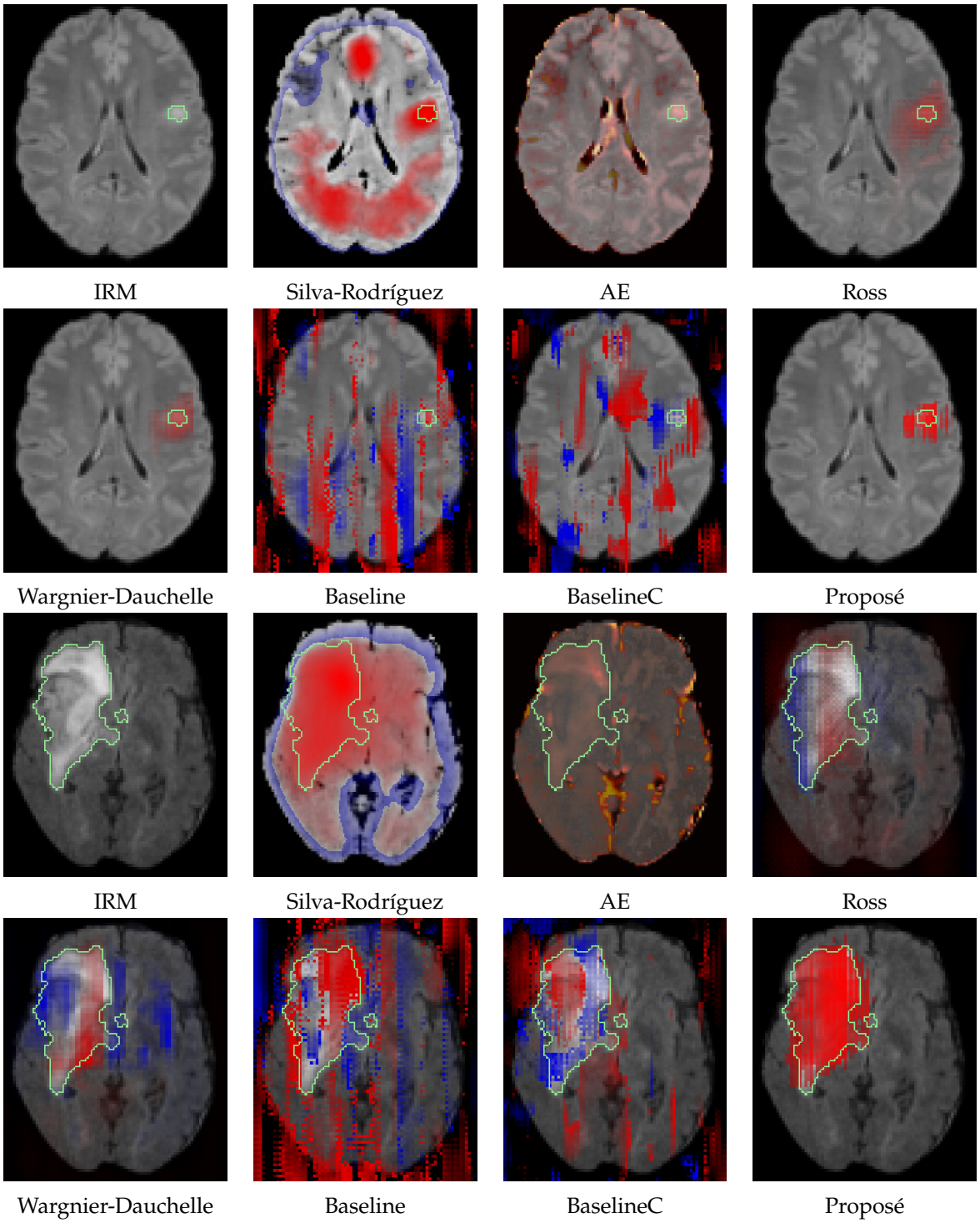


FIGURE 6.12 – *Carte de segmentation pour différentes méthodes sur les images de tumeurs en $2mm^3$ avec correction N4. Les contours de l'annotation manuelle sont en vert. Dans les attributions, le bleu représente la pertinence pour la classe saine et le rouge pour la classe pathologie. Les fortes valeurs d'attributions sont également en rouge pour Silva-Rodríguez. Pour les exemples contrefactuels, le bleu correspond aux valeurs de α négatives et le rouge à celles positives. Pour les méthodes de reconstruction, l'échelle va du noir (faible erreur de reconstruction) au jaune (grande erreur).*

caractéristiques pour un réseau non-monotone et non-contraint (Baseline) ou contraint (BaselineC), les résultats sont nettement moins bons. En effet, premièrement, la différence pour

générer l'exemple contrefactuel est à la fois positive et négative, contrairement à celle pour les réseaux monotones. Ceci nuit fortement à la lecture et à l'interprétation des cartes. Ensuite, le réseau ne semble pas utiliser la présence de la tumeur pour prendre sa décision puisque les fortes valeurs de cette carte (négatives et positives) ne se trouvent pas particulièrement dans la zone tumorale. La génération d'exemples contrefactuels, de la même manière qu'une attaque adversaire, produit, en effet, souvent des résultats difficilement interprétables. L'utilisation des réseaux positives est donc vraiment utile.

Tableau 6.3 – *Comparaison avec l'état de l'art pour la classification et la segmentation des tumeurs.*

	Segmentation			Class. images	
Méthode	Dice	AUROC	AUPRC	TPR	TNR
AE	0.26 ± 0.11	0.90	0.16	/	/
Silva-Rodríguez	0.37 ± 0.17	**0.92**	0.32	/	/
Ross	0.48 ± 0.20	0.80	0.39	**1.00**	**1.00**
Wargnier-Dauchelle	0.51 ± 0.16	0.73	0.45	**1.00**	0.95
Baseline	0.04 ± 0.03	0.60	0.03	**1.00**	**1.00**
BaselineC	0.04 ± 0.03	0.66	0.04	**1.00**	**1.00**
Proposé	**0.56 ± 0.19**	0.91	**0.58**	**1.00**	**1.00**

Les résultats quantitatifs en termes de segmentation sont donnés dans le Tableau 6.3. Notre proposition dépasse toutes les méthodes de l'état sur le Dice et l'AUPRC tout en conservant des performances de classification parfaites. Ainsi, le Dice est 5 points supérieur à la méthode présentée dans le Chapitre 5 et l'AUPRC est 13 points supérieure. L'AUROC est similaire à la meilleure valeur parmi les méthodes de l'état de l'art.

Conclusion

Dans ce chapitre, nous avons proposé d'utiliser des réseaux monotones contraints pour améliorer l'interprétabilité sur une tâche de classification. Pour cela, la première limitation des méthodes de l'état de l'art reposait sur les réseaux monotones eux-mêmes. En effet, les architectures de la littératures sont très limitées. Nous avons donc proposé une méthode générique pour transformer n'importe quelle architecture en réseau monotone. Nous avons notamment montré que les réseaux à poids positifs ne sont pas des réseaux trop contraints et qu'ils peuvent être entraînés sans difficulté tout en obtenant les mêmes performances que les réseaux non-monotones, s'ils sont construits en suivant notre méthode. Les étapes clés de cette méthode sont notamment l'initialisation des poids et le choix des activations. Les réseaux à poids positifs ainsi construits améliorent l'explicabilité en imposant un lien de monotonie entre la représentation intermédiaire qui est apprise et la sortie du réseau. Cela est d'autant plus intéressant pour la génération d'exemples contrefactuels puisqu'on est certain qu'il faut diminuer l'entrée pour diminuer la sortie, à l'inverse des réseaux non-monotones pour lesquels la différence contrefactuelle est souvent difficile à interpréter. Pour améliorer l'interprétabilité de ces réseaux intrinsèquement explicables, nous avons proposé une nouvelle contrainte pour l'apprentissage d'un classifieur d'images saines vs pathologiques qui ne demande aucune annotation supplémentaire par rapport à la tâche de classification. A travers des fonctions de coût, nous contraignons ainsi les cartes de caractéristiques et les gradients pour une représentation similaire des tissus sains présents dans les images. Avec ces réseaux à poids positifs contraints, nous obtenons une classification plus interprétable et notre méthode de segmentation faiblement supervisée des tumeurs cérébrales dépasse les performances de l'état de l'art.

Des expériences supplémentaires pourraient être menées. Tout d'abord, notre méthode pourrait être évaluée pour la segmentation des lésions de sclérose en plaques. Ensuite, dans certaines configurations (en enlevant certaines contraintes par exemple), notre méthode souffre de problème de reproductibilité. Une analyse plus poussée des hyperparamètres pourrait donc être réalisée. Nous avons également fait une analyse théorique de la corrélation des caractéristiques dans un réseau de neurones et avons notamment montré que les caractéristiques sont beaucoup plus corrélées dans les réseaux à poids positifs. Un axe de recherche future pourrait ainsi être de proposer des solutions pour diminuer la corrélation des caractéristiques dans le réseau monotone. Enfin, différentes architectures du réseau pré-monotone pourraient être considérées. En effet, nous avons choisi une architecture relativement petite avec deux couches de convolution mais une architecture permettant d'extraire davantage de sémantique (comme un U-Net) pourrait être examinée.

Enfin, de la même manière que notre contribution [Wargnier-Dauchelle *et al.*, 2023b], la méthode proposée dans ce chapitre pourrait être utilisée dans d'autres architectures comme les GAN ou les modèles de diffusion mais aussi pour d'autres tâches comme de la régression ou de la prédiction.

CONCLUSION

I Bilan

I.1 Bilan scientifique

Deux problématiques importantes de l'analyse d'images médicales par apprentissage profond ont été abordées dans ce manuscrit. Tout d'abord, le développement de méthodes faiblement supervisées est nécessaire en imagerie médicale où les données annotées sont peu nombreuses car très coûteuses alors que les méthodes d'apprentissage profond, très performantes, nécessitent de grandes bases d'entraînement. Ensuite, l'explicabilité et l'interprétabilité des réseaux de neurones sont cruciales dans un domaine critique comme celui de la santé. En effet, les cliniciens ont besoin d'avoir confiance dans la prédiction des méthodes automatiques alors que les méthodes d'apprentissage profond sont reconnues comme des "boîtes noires", dont la décision est difficilement explicable.

Pour aborder ces thématiques, nous avons commencé, dans le Chapitre 3, par proposer une méthode de segmentation des lésions de sclérose en plaques par un réseau génératif adversaire. Les résultats, peu concluants, nous ont amené à nous questionner sur la prise de décision des discriminateurs utilisés dans ces architectures et dirigeant le modèle global et donc ses performances.

Dans le Chapitre 4, nous avons étudié plus en profondeur les classifieurs d'images saines vs pathologiques, tels que ceux utilisés dans les GAN. Nous avons identifié des biais dans la classification mais également proposé une nouvelle entrée pour le réseau de classification : des cartes de probabilité d'appartenance aux tissus cérébraux. L'utilisation de ces cartes, et donc d'une forte normalisation par rapport aux données IRM brutes, permet d'obtenir un classifieur dont la décision est davantage basée sur les signes radiologiques de la pathologie. Néanmoins, comme toute normalisation, de l'information peut être perdue et rien ne certifie que les biais sont complètement retirés.

Ainsi, pour aller plus loin et limiter ces problèmes, nous avons proposé, dans le Chapitre 5, de contraindre un classifieur. Nous contraignons ainsi les attributions basées sur le gradient pour que chaque voxel des images saines contribue à la décision de classer l'image dans la classe correspondante. Cette contrainte ne nécessite aucune annotation supplémentaire. En entraînant le classifieur avec cette contrainte, nous obtenons une décision davantage basée sur la pathologie permettant la segmentation faiblement supervisée des lésions cérébrales. En outre, nous avons étudié l'influence du choix de la méthode d'attributions utilisée pour la contrainte et avons montré que le gradient était équivalent à Expected Gradient tout en étant plus rapide et stable. Nous avons également proposé une contrainte plus robuste au choix de la méthode à l'inférence.

Enfin, dans le Chapitre 6, pour renforcer la garantie d'une décision interprétable, nous avons utilisé des réseaux monotones qui sont intrinsèquement explicables. En proposantn après analyse des possibles causes de dysfonctionnement des méthodes de l'état de l'art,

une solution pour transformer un réseau classique en réseau à poids positifs entraînable et en les contraignant pendant l'apprentissage pour obtenir une représentation similaire (à travers les cartes de caractéristiques et les gradients) des tissus sains dans les images, et ceci sans annotation supplémentaire, nous avons obtenu un réseau de classification sain vs pathologique plus interprétable. Les performances de notre réseau pour de la segmentation faiblement supervisée dépassent celles des méthodes de l'état de l'art sur les images de tumeurs cérébrales.

I.2 Autres travaux

En plus de mes propres travaux, j'ai participé pendant ma thèse à plusieurs projets d'intérêt au sein de CREATIS sur du recalage par apprentissage profond, une étude de reproductibilité ainsi que de la détection d'anomalies par modèle de diffusion.

I.2.1 Recalage par apprentissage profond

Dans le Chapitre 1, nous avons vu que le prétraitement des données IRM est essentiel en vue d'une utilisation par une méthode automatique. Parmi ces prétraitements, le recalage permet de replacer toutes les images dans le même espace. Il est très utilisé en imagerie médicale que ce soit pour des études longitudinales, l'utilisation de plusieurs modalités, la visualisation d'organe en mouvement (cœur ou poumons par exemple) ou encore comme normalisation spatiale. Dans notre contexte de classification d'IRM cérébrales, il est important que les images soient dans le même espace afin que la classification ne soit pas basée sur des différences d'orientation de cerveaux entre les bases de données.

Les méthodes classiques de recalage [Sotiras *et al.*, 2013, Mattes *et al.*, 2001] cherchent à résoudre un problème d'optimisation pour minimiser la dissimilarité entre l'image à recaler et l'image fixe. Néanmoins, elles ne sont pas robustes à de fortes transformations ou aux artefacts. Dans le cadre de ma thèse, nous avons utilisé le logiciel FLIRT sur nos bases de données et dans certains cas, le recalage ne fonctionnait pas correctement. Tout d'abord, il est nécessaire de s'assurer que les images se trouvent bien dans le bon système de coordonnées (Right Anterior Inferior) sinon une étape supplémentaire était nécessaire. En outre, le recalage peut conduire à des images retournée à cause de la symétrie et la forme ovale du cerveau. Il est donc nécessaire d'avoir à disposition des méthodes de recalage vraiment robustes pour normaliser spatialement nos bases de données. De plus, le temps de calcul de la transformation est long. Des méthodes de recalage par apprentissage profond existent également avec un temps de calcul à l'inférence très court. Parmi elles, les méthodes supervisées [Liao *et al.*, 2017, Rohé *et al.*, 2017, Miao *et al.*, 2016] sont robustes mais nécessitent de connaître la transformation à appliquer pour le recalage pour l'entraînement. Les méthodes non supervisées [Li et Fan, 2018, Balakrishnan *et al.*, 2019, de Vos *et al.*, 2019] sont quant à elles souvent basées sur une fonction de coût de dissimilarité comme les méthodes classiques et présentent donc les limitations de robustesse de ces dernières.

Nos travaux proposent une méthode de recalage par apprentissage profond, non supervisée et n'utilisant pas de fonction de coût de dissimilarité. Pour cela, la sortie du réseau de neurones est la transformation affine à appliquer pour recaler l'image d'entrée sur l'image de référence. Trois fonctions de coût sont utilisées. La première force le réseau entraîné à être équivariant pour les transformations affines : pour toute transformation affine B, $B \circ R(I \circ B) = R(I)$ où I est l'image et R le réseau. La seconde fonction de coût régularise les transformations en sortie du réseau afin d'éviter les transformations irréalistes avec de trop fortes variations de taille ou une trop forte anisotropie. Enfin, sur l'image de référence, des transformations affines aléatoires sont générées et le modèle apprend à retrouver cette transformation de manière supervisée. De plus, un entraînement en deux temps est proposé pour rendre le réseau robuste à de fortes transformations. Le réseau est d'abord entraîné

avec de faibles transformations de manière non supervisée. Puis, à partir de ce modèle appris, les images d'entraînement peuvent être recalées et utilisées pour un entraînement avec des transformations plus grandes avec la même fonction de coût que celle utilisée sur l'image de référence.

Cette proposition permet d'obtenir un recalage robuste aux fortes transformations affines mais également si l'image est rognée ou si des occlusions sont ajoutées pour effacer une partie de l'image. Ces travaux ont été acceptés au workshop *Medical Image Learning with Limited and Noisy Data* de la conférence MICCAI 2023 [Hachicha *et al.*, 2023].

I.2.2 Reproductibilité des modèles de segmentation par apprentissage profond

Nous avons vu, dans le Chapitre 1, que le prétraitement des données est une étape importante pour l'analyse de ces dernières, permettant d'uniformiser les données, d'éliminer certains artefacts et biais potentiels qui pourraient perturber l'apprentissage d'un réseau de neurones. Or, de nombreuses méthodes de prétraitement sont proposées en accès libre. Tout le monde peut accéder à ces pipelines et les utiliser sur sa machine mais on sait peu de choses sur la variabilité et les conséquences du choix de la version du pipeline ou de l'environnement d'exécution sur l'analyse des images par un réseau de neurones. Cependant, il est important de comprendre l'impact de ces changements lors de partage de code comme la chaîne de prétraitements que nous avons établie et qui a été portée sur VIP.

Dans ces travaux, une analyse de l'influence de la version d'une même chaîne de prétraitements et du changement de système d'exploitation sur les performances de segmentation des tumeurs cérébrales d'un réseau de neurones à l'inférence a été proposée. Nous avons également analysé quels étaient les éléments de la chaîne de prétraitements qui avaient la plus forte influence sur la reproductibilité des résultats. Pour évaluer les variabilités apportées par le changement de version, deux versions de la chaîne proposée par le challenge BraTS [Bakas *et al.*, 2017, Bakas *et al.*, 2018, Menze *et al.*, 2014] ont été appliquées sur des images de tumeurs cérébrales non-prétraitées. Pour mimer un changement d'environnement, des perturbations numériques ont été introduites avec "Fuzzy libmath" [Salari *et al.*, 2021].

Les résultats montrent que les performances de segmentation varient considérablement d'une version à l'autre du pipeline BRATS. Même si, en moyenne, les coefficients de Dice sont élevés, les valeurs peuvent descendre très bas pour certains patients en fonction de la version utilisée. Les étapes du prétraitement les plus critiques pour cette variation semblent être la correction des inhomogénéités de champ ainsi que le recalage. En outre, la variabilité inter-environnement est du même ordre de grandeur que la variabilité inter-version, ce qui suggère que les causes sous-jacentes des variations observées peuvent être liées à la stabilité numérique. Ces travaux ont été présentés lors de la conférence ISBI 2023 [Des Ligneris *et al.*, 2023].

I.2.3 Segmentation faiblement supervisée par modèle de diffusion profond guidé par classification

Nous avons vu, dans le Chapitre 3, que les modèles génératifs adversaires sont difficiles à entraîner. Récemment, d'autres modèles génératifs plus stables ont été proposés à savoir les modèles de diffusion profonds. Dans ces modèles [Ho *et al.*, 2020], les images d'entrée sont, pas à pas, détruites pour obtenir un bruit gaussien. Pour cela, une chaîne de Markov est utilisée pour ajouter progressivement du bruit de manière connue en ne dépendant que de l'échantillon précédent (niveau de bruit $t-1$ à t). Un réseau est alors appris pour reconstruire l'image d'origine, pas à pas, en estimant le bruit à supprimer pour passer d'une image bruitée à un niveau t à une image bruitée de niveau $t-1$. Il est possible de guider le processus de reconstruction vers une classe donnée [Dhariwal et Nichol, 2021]. En effet, en

entraînant un classifieur à discriminer les images des différentes classes pour les différents niveaux de bruits, on peut utiliser le gradient de la sortie de ce classifieur $p(y|x)$ (où x est l'image d'entrée et y la classe choisie) par rapport à l'entrée pour tirer la reconstruction vers la classe voulue. Dans [Wolleb *et al.*, 2022], un classifieur d'images saines vs pathologiques est appris et utilisé pour guider un modèle de diffusion sur des images pathologiques en tirant la reconstruction vers la classe saine. Ainsi à partir d'une image d'un patient, une image de type "saine" est reconstruite. A la manière des méthodes de détection d'anomalies (AE, VAE, GAN, etc), la différence de reconstruction permet de segmenter la pathologie. Les expériences sont notamment faites sur la segmentation de tumeurs cérébrales à partir de la base de données BraTS.

Or, nous avons vu dans ce manuscrit que les classifieurs d'images saines vs pathologiques ne basent pas forcément leur décision sur les signes radiologiques de la pathologie. Nous avons également montré, dans le Chapitre 5 qu'en ajoutant la contrainte proposée, le discriminateur basait d'avantage sa décision sur la présence de la pathologie et que le gradient de la sortie par rapport à l'entrée se concentrait sur la pathologie. Puisque c'est ce gradient qui est utilisé pour guider le modèle de diffusion, on peut penser qu'en contraignant comme proposé le discriminateur utilisé pour guider le modèle de diffusion dans [Wolleb *et al.*, 2022], nous obtiendrions de meilleures segmentations.

De plus, un des problèmes des modèles de diffusion est le temps nécessaire pour reconstruire l'image. En effet, dans les modèles classiques, il faut de nombreux pas pour bruiter et débruiter l'image. Des méthodes récentes proposent des solutions pour espacer l'échantillonnage et donc réduire le temps de calcul [Lu *et al.*, 2022, Zhao *et al.*, 2023]. Utiliser ces modèles pour de la détection d'anomalies par diffusion guidée serait donc très intéressant.

J'ai co-encadré un stagiaire de Master 2 (avril-septembre 2023) pour travailler sur ces thématiques. Les expériences doivent être poursuivies.

I.3 Enseignements

Au cours de ma thèse, j'ai réalisé plus de 330h d'enseignement (équivalent TD) au sein du département FIMI (Formation Initiale aux Métiers d'Ingénieur) de l'INSA Lyon, à travers des vacations et un contrat d'ATER. Le département FIMI accueille les bacheliers sur deux ans en proposant un tronc commun avant d'intégrer les départements de spécialisation. J'ai ainsi pu enseigner à des élèves de première et deuxième année après le baccalauréat. Dans ce département, il existe différentes filières. J'ai dispensé des cours en filière classique mais également en filière EURINSA, qui mélange des élèves venant de France et du reste de l'Europe, et en filière Arts-Études.

Je suis intervenue dans plusieurs matières. J'ai principalement encadré des TD en algorithmique et programmation, dont l'objectif est de former les élèves au développement d'algorithmes efficaces et à leur programmation en Java puis en Python suite à un changement de programme du département. En première année, les bases du langage, les listes, les structures conditionnelles, les boucles et les fonctions sont traitées. Dans le nouveau programme, une introduction aux réseaux informatiques (architectures, protocoles, etc) est également réalisée. En deuxième année, la lecture de fichiers, les dictionnaires, le parcours d'arbres, la programmation orientée objets et la création d'interfaces graphiques utilisateur avec la programmation événementielle sont abordés. Les bases de données relationnelles et la recherche d'information dans des bases avec SQL y sont également étudiées. J'ai aussi donné 12h de CM à des élèves de deuxième année dans cette discipline. J'ai également été chargée de TD en systèmes et outils logiciels dont le but est de former les élèves en tableur, traitement de texte, markdown et bash mais également de les faire réfléchir sur l'hygiène numérique. Ensuite, j'ai encadré 16h de TP dans le cadre des parcours pluridisciplinaires d'initiation à l'ingénierie et plus précisément dans le parcours imagerie industrielle et médicale. Ces TP correspondaient à de l'acquisition et du traitement du signal d'ondes

acoustiques. Enfin, je suis intervenue dans des projets pédagogiques en tant que deuxième encadrant en outils mathématiques pour les sciences de l'ingénieur.

J'ai ainsi dispensé des TP, TD et CM mais aussi encadré des projets et des heures de soutien. J'étais également en charge de surveillances d'examens et de corrections des copies. J'ai aussi participé à la relecture de sujets d'examens.

Enfin, j'ai encadré 2h de TP en ligne pour une école d'été internationale ayant pour thématique l'apprentissage profond pour l'imagerie médicale [1]. Les TP encadrés portaient sur de la classification et de la segmentation.

II Perspectives

Pour poursuivre les travaux présentés dans cette thèse, de nombreuses pistes existent. Tout d'abord, les classifieurs sont des modèles utilisés au sein de nombreuses architectures plus importantes. En terme de segmentation faiblement supervisée, on retrouve par exemple, les GAN et les modèles de diffusion guidés. Des travaux ont été initiés pour introduire le classifieur contraint proposé dans le Chapitre 5 dans un modèle de diffusion (Section I.2.3) et des expériences sont encore nécessaires pour obtenir une méthode fonctionnelle. On pourrait également utiliser le classifieur proposé dans le Chapitre 6 pour guider le modèle de diffusion. Ces deux classifieurs pourraient aussi être utilisés dans des GAN pour améliorer les performances en détection d'anomalies avec des discriminateurs dont la décision serait basée sur la pathologie plutôt que sur divers biais.

Concernant ces classifieurs, nous n'avons traité que des cas de classifications binaires. Il pourrait être intéressant de considérer des cas de classification multi-classe. Par exemple, on pourrait classer les différents grades d'une maladie à partir d'images (comme dans le cas de la dégénérescence cérébrale) en incluant une classe de sujets sains. Dans ce cas, les mêmes contraintes sur les sujets sains que celles proposées dans cette thèse pourraient être utilisées sur cette classe. De la même manière, des tâches de régression ou de prédiction pourraient être abordées. Par exemple, si l'objectif est de prédire un score, encore une fois, des sujets sains au score nul pourraient être introduits pour l'apprentissage avec les contraintes proposées.

Enfin, il est possible d'améliorer les réseaux monotones du Chapitre 6, notamment pour diminuer la corrélation entre les cartes de caractéristiques. Pour cela, des initialisations des poids limitant cette corrélation peuvent être envisagées. Des travaux préliminaires utilisant des bases de permutations ont été initiés et les premiers résultats montrent une diminution de la corrélation à l'initialisation. En outre, la possibilité, grâce à notre méthode, d'entraîner n'importe quel réseau dans une version monotone, ouvre des possibilités qui permettront, peut-être, une généralisation de leur utilisation.

1. https://event.fourwaves.com/dlmi2022/pages

BIBLIOGRAPHIE

[Agarwal *et al.*, 2021] Agarwal, R., Melnick, L., Frosst, N., Zhang, X., Lengerich, B., Caruana, R. et Hinton, G. E. (2021). Neural additive models : Interpretable machine learning with neural nets. *Advances in Neural Information Processing Systems*, 34 : 4699–4711.

[Akhondi-Asl *et al.*, 2014] Akhondi-Asl, A., Hoyte, L., Lockhart, M. E. et Warfield, S. K. (2014). A logarithmic opinion pool based staple algorithm for the fusion of segmentations with associated reliability weights. *IEEE transactions on medical imaging*, 33(10) : 1997–2009.

[Alvarez Melis et Jaakkola, 2018] Alvarez Melis, D. et Jaakkola, T. (2018). Towards robust interpretability with self-explaining neural networks. *Advances in neural information processing systems*, 31.

[Ancona *et al.*, 2017] Ancona, M., Ceolini, E., Öztireli, C. et Gross, M. (2017). Towards better understanding of gradient-based attribution methods for deep neural networks. *arXiv preprint arXiv :1711.06104*.

[Ba *et al.*, 2016] Ba, J. L., Kiros, J. R. et Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv :1607.06450*.

[Babayan *et al.*, 2019] Babayan, A., Erbey, M., Kumral, D., Reinelt, J. D., Reiter, A. M., Röbbig, J., Schaare, H. L., Uhlig, M., Anwander, A., Bazin, P.-L. *et al.* (2019). A mind-brain-body dataset of mri, eeg, cognition, emotion, and peripheral physiology in young and old adults. *Scientific data*, 6(1) : 1–21.

[Bach *et al.*, 2015] Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R. et Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7) : e0130140.

[Bakas *et al.*, 2017] Bakas, S., Akbari, H., Sotiras, A., Bilello, M., Rozycki, M., Kirby, J. S., Freymann, J. B., Farahani, K. et Davatzikos, C. (2017). Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features. *Scientific data*, 4(1) : 1–13.

[Bakas *et al.*, 2018] Bakas, S., Reyes, M., Jakab, A., Bauer, S., Rempfler, M., Crimi, A., Shinohara, R. T., Berger, C., Ha, S. M., Rozycki, M. *et al.* (2018). Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge. *arXiv preprint arXiv :1811.02629*.

[Balakrishnan *et al.*, 2019] Balakrishnan, G., Zhao, A., Sabuncu, M. R., Guttag, J. et Dalca, A. V. (2019). Voxelmorph : a learning framework for deformable medical image registration. *IEEE transactions on medical imaging*, 38(8) : 1788–1800.

[Balestriero *et al.*, 2021] Balestriero, R., Pesenti, J. et LeCun, Y. (2021). Learning in high dimension always amounts to extrapolation.

[Bárány et Füredi, 1988] Bárány, I. et Füredi, Z. (1988). On the shape of the convex hull of random points. *Probability theory and related fields*, 77(2) : 231–240.

[Baur *et al.*, 2018] Baur, C., Wiestler, B., Albarqouni, S. et Navab, N. (2018). Deep autoencoding models for unsupervised anomaly segmentation in brain mr images. *International MICCAI brainlesion workshop*, p. 161–169. Springer.

[Baur *et al.*, 2019] Baur, C., Wiestler, B., Albarqouni, S. et Navab, N. (2019). Deep autoencoding models for unsupervised anomaly segmentation in brain mr images. *Brainlesion : Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries : 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part I 4*, p. 161–169. Springer.

[Bergmann *et al.*, 2021] Bergmann, P., Batzner, K., Fauser, M., Sattlegger, D. et Steger, C. (2021). The mvtec anomaly detection dataset : a comprehensive real-world dataset for unsupervised anomaly detection. *International Journal of Computer Vision*, 129(4) : 1038–1059.

[Bhuvaji *et al.*, 2020] Bhuvaji, S., Kadam, A., Bhumkar, P., Dedge, S. et Kanchan, S. (2020). Brain tumor classification (mri).

[Bloch, 1946] Bloch, F. (1946). Nuclear induction. *Physical review*, 70(7-8) : 460.

[Brisset *et al.*, 2020] Brisset, J.-C., Kremer, S., Hannoun, S., Bonneville, F., Durand-Dubief, F., Tourdias, T., Barillot, C., Guttmann, C., Vukusic, S., Dousset, V. *et al.* (2020). New ofsep recommendations for mri assessment of multiple sclerosis patients : special consideration for gadolinium deposition and frequent acquisitions. *Journal of Neuroradiology*, 47(4) : 250–258.

[Castro *et al.*, 2009] Castro, J., Gómez, D. et Tejada, J. (2009). Polynomial calculation of the shapley value based on sampling. *Computers & Operations Research*, 36(5) : 1726–1730.

[Charachon *et al.*, 2022] Charachon, M., Cournède, P.-H., Hudelot, C. et Ardon, R. (2022). Leveraging conditional generative models in a general explanation framework of classifier decisions. *Future Generation Computer Systems*, 132 : 223–238.

[Chatterjee, 2014] Chatterjee, S. (2014). *Superconcentration and related topics*, volume 15. Springer.

[Chen *et al.*, 2020] Chen, Z., Guo, B., Li, C. et Liu, H. (2020). Review on superpixel generation algorithms based on clustering. *2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education (ICISCAE)*, p. 532–537. IEEE.

[Commowick *et al.*, 2021] Commowick, O., Kain, M., Casey, R., Ameli, R., Ferré, J.-C., Kerbrat, A., Tourdias, T., Cervenansky, F., Camarasu-Pop, S., Glatard, T. *et al.* (2021). Multiple sclerosis lesions segmentation from multiple experts : The miccai 2016 challenge dataset. *Neuroimage*, 244 : 118589.

[Confavreux *et al.*, 1992] Confavreux, C., Compston, D., Hommes, O., McDonald, W. et Thompson, A. (1992). Edmus, a european database for multiple sclerosis. *Journal of Neurology, Neurosurgery & Psychiatry*, 55(8) : 671–676.

[Daniels et Velikova, 2010] Daniels, H. et Velikova, M. (2010). Monotone and partially monotone neural networks. *IEEE Transactions on Neural Networks*, 21(6) : 906–917.

[de Vos *et al.*, 2019] de Vos, B. D., Berendsen, F. F., Viergever, M. A., Sokooti, H., Staring, M. et Išgum, I. (2019). A deep learning framework for unsupervised affine and deformable image registration. *Medical Image Analysis*, 52 : 128–143.

[Deng, 2012] Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6) : 141–142.

[Des Ligneris *et al.*, 2023] Des Ligneris, M., Bonnet, A., Chatelain, Y., Glatard, T., Sdika, M., Vila, G., Wargnier-Dauchelle, V., Pop, S. et Frindel, C. (2023). Reproducibility of tumor segmentation outcomes with a deep learning model. *International Symposium on Biomedical Imaging (ISBI)*.

[Dhariwal et Nichol, 2021] Dhariwal, P. et Nichol, A. (2021). Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34 : 8780–8794.

[Du *et al.*, 2020] Du, G., Cao, X., Liang, J., Chen, X. et Zhan, Y. (2020). Medical image segmentation based on u-net : A review. *Journal of Imaging Science and Technology*.

[Erion *et al.*, 2021] Erion, G., Janizek, J. D., Sturmfels, P., Lundberg, S. M. et Lee, S.-I. (2021). Improving performance of deep learning models with axiomatic attribution priors and expected gradients. *Nature Machine Intelligence*, p. 1–12.

[Fisher *et al.*, 2019] Fisher, A., Rudin, C. et Dominici, F. (2019). All models are wrong, but many are useful : Learning a variable's importance by studying an entire class of prediction models simultaneously. *J. Mach. Learn. Res.*, 20(177) : 1–81.

[Fong et Vedaldi, 2017] Fong, R. C. et Vedaldi, A. (2017). Interpretable explanations of black boxes by meaningful perturbation. *Proceedings of the IEEE international conference on computer vision*, p. 3429–3437.

[Gautam *et al.*, 2022] Gautam, S., Boubekki, A., Hansen, S., Salahuddin, S., Jenssen, R., Höhne, M. et Kampffmeyer, M. (2022). Protovae : A trustworthy self-explainable prototypical variational model. *Advances in Neural Information Processing Systems*, 35 : 17940–17952.

[Gentle, 2010] Gentle, J. E. (2010). *Computational statistics*. Springer.

[Glorot et Bengio, 2010] Glorot, X. et Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, volume 9 de *Proceedings of Machine Learning Research*, p. 249–256. PMLR.

[Goodfellow *et al.*, 2020] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. et Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11) : 139–144.

[Gourraud *et al.*, 2013] Gourraud, P.-A., Sdika, M., Khankhanian, P., Henry, R. G., Beheshtian, A., Matthews, P. M., Hauser, S. L., Oksenberg, J. R., Pelletier, D. et Baranzini, S. E. (2013). A genome-wide association study of brain lesion distribution in multiple sclerosis. *Brain*, 136(4) : 1012–1024.

[Grath *et al.*, 2018] Grath, R. M., Costabello, L., Van, C. L., Sweeney, P., Kamiab, F., Shen, Z. et Lecue, F. (2018). Interpretable credit application predictions with counterfactual explanations. *arXiv preprint arXiv :1811.05245*.

[Gui *et al.*, 2021] Gui, J., Sun, Z., Wen, Y., Tao, D. et Ye, J. (2021). A review on generative adversarial networks : Algorithms, theory, and applications. *IEEE transactions on knowledge and data engineering*, 35(4) : 3313–3332.

[Guidotti, 2022] Guidotti, R. (2022). Counterfactual explanations and how to find them : literature review and benchmarking. *Data Mining and Knowledge Discovery*, p. 1–55.

[Hachicha *et al.*, 2023] Hachicha, S., Le, C., Wargnier-Dauchelle, V. et Sdika, M. (2023). Robust Unsupervised Image to Template Registration Without Image Similarity Loss. *Medical Image Learning with Limited and Noisy Data, Second International Workshop, MILLanD 2023, Held in Conjunction with MICCAI 2023, Vancouver, Proceedings*, Vancouver, Canada.

[Hahnloser *et al.*, 2000] Hahnloser, R. H., Sarpeshkar, R., Mahowald, M. A., Douglas, R. J. et Seung, H. S. (2000). Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *nature*, 405(6789) : 947–951.

[Hajnal *et al.*, 1992] Hajnal, J. V., Bryant, D. J., Kasuboski, L., Pattany, P. M., De Coene, B., Lewis, P. D., Pennock, J. M., Oatridge, A., Young, I. R. et Bydder, G. M. (1992). Use of fluid attenuated inversion recovery (flair) pulse sequences in mri of the brain. *Journal of computer assisted tomography*, 16(6) : 841–844.

[He *et al.*, 2015] He, K., Zhang, X., Ren, S. et Sun, J. (2015). Delving deep into rectifiers : Surpassing human-level performance on imagenet classification. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[He *et al.*, 2016] He, K., Zhang, X., Ren, S. et Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, p. 770–778.

[Hinton *et al.*, 2012] Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I. et Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv :1207.0580*.

[Ho *et al.*, 2020] Ho, J., Jain, A. et Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33 : 6840–6851.

[Hu *et al.*, 2018] Hu, J., Shen, L. et Sun, G. (2018). Squeeze-and-excitation networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, p. 7132–7141.

[Huang *et al.*, 2017] Huang, G., Liu, Z., Van Der Maaten, L. et Weinberger, K. Q. (2017). Densely connected convolutional networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, p. 4700–4708.

[Huttenlocher *et al.*, 1993] Huttenlocher, D. P., Klanderman, G. A. et Rucklidge, W. J. (1993). Comparing images using the hausdorff distance. *IEEE Transactions on pattern analysis and machine intelligence*, 15(9) : 850–863.

[Idbaih *et al.*, 2019] Idbaih, A., Canney, M., Belin, L., Desseaux, C., Vignot, A., Bouchoux, G., Asquier, N., Law-Ye, B., Leclercq, D., Bissery, A. *et al.* (2019). Safety and feasibility of repeated and transient blood–brain barrier disruption by pulsed ultrasound in patients with recurrent glioblastoma. *Clinical Cancer Research*, 25(13) : 3793–3801.

[Ilboudo *et al.*, 2022] Ilboudo, W. E. L., Kobayashi, T. et Sugimoto, K. (2022). Robust stochastic gradient descent with student-t distribution based first-order momentum. *IEEE Transactions on Neural Networks and Learning Systems*, 33(3) : 1324–1337.

[Ioffe et Szegedy, 2015] Ioffe, S. et Szegedy, C. (2015). Batch normalization : Accelerating deep network training by reducing internal covariate shift. *International conference on machine learning*, p. 448–456. pmlr.

[Isensee *et al.*, 2020] Isensee, F., Jäger, P., Wasserthal, J., Zimmerer, D., Petersen, J., Kohl, S., Schock, J., Klein, A., Roß, T., Wirkert, S., Neher, P., Dinkelacker, S., Köhler, G. et Maier-Hein, K. (2020). batchgenerators - a python framework for data augmentation.

[Isensee *et al.*, 2019] Isensee, F., Schell, M., Pflueger, I., Brugnara, G., Bonekamp, D., Neuberger, U., Wick, A., Schlemmer, H.-P., Heiland, S., Wick, W. *et al.* (2019). Automated brain extraction of multisequence mri using artificial neural networks. *Human brain mapping*, 40(17) : 4952–4964.

[Isola *et al.*, 2017] Isola, P., Zhu, J.-Y., Zhou, T. et Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, p. 1125–1134.

[Jeanneret *et al.*, 2022] Jeanneret, G., Simon, L. et Jurie, F. (2022). Diffusion models for counterfactual explanations. *Proceedings of the Asian Conference on Computer Vision*, p. 858–876.

[Jenkinson *et al.*, 2002] Jenkinson, M., Bannister, P., Brady, M. et Smith, S. (2002). Improved optimization for the robust and accurate linear registration and motion correction of brain images. *Neuroimage*, 17(2) : 825–841.

[Jenkinson et Smith, 2001] Jenkinson, M. et Smith, S. (2001). A global optimisation method for robust affine registration of brain images. *Medical image analysis*, 5(2) : 143–156.

[Joshi *et al.*, 2019] Joshi, S., Koyejo, O., Vijitbenjaronk, W., Kim, B. et Ghosh, J. (2019). Towards realistic individual recourse and actionable explanations in black-box decision making systems. *arXiv preprint arXiv :1907.09615*.

[Kalchbrenner et Blunsom, 2013] Kalchbrenner, N. et Blunsom, P. (2013). Recurrent continuous translation models. *Proceedings of the 2013 conference on empirical methods in natural language processing*, p. 1700–1709.

[Karimi et Salcudean, 2019] Karimi, D. et Salcudean, S. E. (2019). Reducing the hausdorff distance in medical image segmentation with convolutional neural networks. *IEEE Transactions on medical imaging*, 39(2) : 499–513.

[Kastler et Vetter, 2018] Kastler, B. et Vetter, D. (2018). *Comprendre l'IRM : manuel d'autoapprentissage*. Elsevier Health Sciences.

[Kervadec *et al.*, 2019] Kervadec, H., Bouchtiba, J., Desrosiers, C., Granger, E., Dolz, J. et Ayed, I. B. (2019). Boundary loss for highly unbalanced segmentation. *International conference on medical imaging with deep learning*, p. 285–296. PMLR.

[Kervadec *et al.*, 2022] Kervadec, H., Dolz, J., Yuan, J., Desrosiers, C., Granger, E. et Ayed, I. B. (2022). Constrained deep networks : Lagrangian optimization via log-barrier extensions. *2022 30th European Signal Processing Conference (EUSIPCO)*, p. 962–966. IEEE.

[Kingma et Ba, 2014] Kingma, D. P. et Ba, J. (2014). Adam : A method for stochastic optimization. *arXiv preprint arXiv :1412.6980*.

[Kingma et Welling, 2013] Kingma, D. P. et Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv :1312.6114*.

[Klein *et al.*, 2009] Klein, S., Staring, M., Murphy, K., Viergever, M. A. et Pluim, J. P. (2009). Elastix : a toolbox for intensity-based medical image registration. *IEEE transactions on medical imaging*, 29(1) : 196–205.

[Landman *et al.*, 2011] Landman, B. A., Huang, A. J., Gifford, A., Vikram, D. S., Lim, I. A. L., Farrell, J. A., Bogovic, J. A., Hua, J., Chen, M., Jarso, S. *et al.* (2011). Multi-parametric neuroimaging reproducibility : a 3-t resource study. *Neuroimage*, 54(4) : 2854–2866.

[LeCun, 1985] LeCun, Y. (1985). Une procedure d'apprentissage pour reseau a seuil asymetrique. proceedings of cognitiva 85.

[LeCun *et al.*, 1998] LeCun, Y., Bottou, L., Bengio, Y. et Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) : 2278–2324.

[Levine *et al.*, 2015] Levine, J. H., Simonds, E. F., Bendall, S. C., Davis, K. L., El-ad, D. A., Tadmor, M. D., Litvin, O., Fienberg, H. G., Jager, A., Zunder, E. R. *et al.* (2015). Data-driven phenotypic dissection of aml reveals progenitor-like cells that correlate with prognosis. *Cell*, 162(1) : 184–197.

[Li et Fan, 2018] Li, H. et Fan, Y. (2018). Non-rigid image registration using self-supervised fully convolutional networks without training data. *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, p. 1075–1078. IEEE.

[Li *et al.*, 2018] Li, O., Liu, H., Chen, C. et Rudin, C. (2018). Deep learning for case-based reasoning through prototypes : A neural network that explains its predictions. *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

[Liao *et al.*, 2017] Liao, R., Miao, S., de Tournemire, P., Grbic, S., Kamen, A., Mansi, T. et Comaniciu, D. (2017). An artificial agent for robust image registration. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1).

[Liu et Nocedal, 1989] Liu, D. C. et Nocedal, J. (1989). On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3) : 503–528.

[Liu *et al.*, 2020] Liu, X., Han, X., Zhang, N. et Liu, Q. (2020). Certified monotonic neural networks. *Advances in Neural Information Processing Systems*, 33 : 15427–15438.

[Liu *et al.*, 2021] Liu, X., Song, L., Liu, S. et Zhang, Y. (2021). A review of deep-learning-based medical image segmentation methods. *Sustainability*, 13(3) : 1224.

[Lu *et al.*, 2022] Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C. et Zhu, J. (2022). Dpm-solver++ : Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv :2211.01095*.

[Lundberg et Lee, 2017] Lundberg, S. M. et Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

[Mader *et al.*, 2009] Mader, S., Dubois, N., Devito, M., Pelletier, N., Fontaine, J. et Morin, L. (2009). *Biologie humaine*. Chenelière éducation.

[Maier-Hein *et al.*, 2022] Maier-Hein, L., Menze, B. *et al.* (2022). Metrics reloaded : Pitfalls and recommendations for image analysis validation. *arXiv. org*, (2206.01653).

[Mattes *et al.*, 2001] Mattes, D., Haynor, D. R., Vesselle, H., Lewellyn, T. K. et Eubank, W. (2001). Nonrigid multimodality image registration. *Medical imaging 2001 : image processing*, volume 4322, p. 1609–1620. Spie.

[McKinley *et al.*, 2016] McKinley, R., Wepfer, R., Gundersen, T., Wagner, F., Chan, A., Wiest, R. et Reyes, M. (2016). Nabla-net : A deep dag-like convolutional architecture for biomedical image segmentation. *Brainlesion : Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries : Second International Workshop, BrainLes 2016, with the Challenges on BRATS, ISLES and mTOP 2016, Held in Conjunction with MICCAI 2016, Athens, Greece, October 17, 2016, Revised Selected Papers 2*, p. 119–128. Springer.

[Menze *et al.*, 2014] Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R. *et al.* (2014). The multimodal brain tumor image segmentation benchmark (brats). *IEEE transactions on medical imaging*, 34(10) : 1993–2024.

[Miao *et al.*, 2016] Miao, S., Wang, Z. J. et Liao, R. (2016). Real-time 2d/3d registration via cnn regression.

[Minsky et Papert, 1969] Minsky, M. et Papert, S. (1969). An introduction to computational geometry. *Cambridge tiass., HIT*, 479 : 480.

[Mishkin et Matas, 2015] Mishkin, D. et Matas, J. (2015). All you need is a good init. *arXiv preprint arXiv :1511.06422*.

[Nadarajah et Kotz, 2008] Nadarajah, S. et Kotz, S. (2008). Exact distribution of the max/-min of two gaussian random variables. *IEEE Transactions on very large scale integration (VLSI) systems*, 16(2) : 210–212.

[Nguyen *et al.*, 2023] Nguyen, A.-P., Moreno, D. L., Le-Bel, N. et Rodríguez Martínez, M. (2023). Mononet : Enhancing interpretability in neural networks via monotonic features. *Bioinformatics Advances*, p. vbad016.

[Parker, 1985] Parker, D. B. (1985). Learning-logic. *Tech. Rep.*, 47.

[Pinho *et al.*, 2018] Pinho, A. L., Amadon, A., Ruest, T., Fabre, M., Dohmatob, E., Denghien, I., Ginisty, C., Becuwe-Desmidt, S., Roger, S., Laurier, L. *et al.* (2018). Individual brain charting, a high-resolution fmri dataset for cognitive mapping. *Scientific data*, 5(1) : 1–15.

[Ras *et al.*, 2022] Ras, G., Xie, N., Van Gerven, M. et Doran, D. (2022). Explainable deep learning : A field guide for the uninitiated. *Journal of Artificial Intelligence Research*, 73 : 329–397.

[Ribeiro *et al.*, 2016] Ribeiro, M. T., Singh, S. et Guestrin, C. (2016). " why should i trust you ?" explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, p. 1135–1144.

[Rocca *et al.*, 2017] Rocca, M. A., Battaglini, M., Benedict, R. H., De Stefano, N., Geurts, J. J., Henry, R. G., Horsfield, M. A., Jenkinson, M., Pagani, E. et Filippi, M. (2017). Brain mri atrophy quantification in ms : from methods to clinical application. *Neurology*, 88(4) : 403–413.

[Rohé *et al.*, 2017] Rohé, M.-M., Datar, M., Heimann, T., Sermesant, M. et Pennec, X. (2017). Svf-net : learning deformable image registration using shape matching. *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017 : 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part I 20*, p. 266–274. Springer.

[Rohlfing *et al.*, 2010] Rohlfing, T., Zahr, N. M., Sullivan, E. V. et Pfefferbaum, A. (2010). The sri24 multichannel atlas of normal adult human brain structure. *Human brain mapping*, 31(5) : 798–819.

[Ronneberger *et al.*, 2015] Ronneberger, O., Fischer, P. et Brox, T. (2015). U-net : Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015 : 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, p. 234–241. Springer.

[Rosenblatt, 1958] Rosenblatt, F. (1958). The perceptron : a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6) : 386.

[Ross *et al.*, 2017] Ross, A. S., Hughes, M. C. et Doshi-Velez, F. (2017). Right for the right reasons : training differentiable models by constraining their explanations. *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, p. 2662–2670.

[Rumelhart *et al.*, 1986] Rumelhart, D. E., Hinton, G. E. et Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088) : 533–536.

[Runje et Shankaranarayana, 2022] Runje, D. et Shankaranarayana, S. M. (2022). Constrained monotonic neural networks. *arXiv preprint arXiv :2205.11775*.

[Rymarczyk *et al.*, 2022] Rymarczyk, D., Struski, Ł., Górszczak, M., Lewandowska, K., Tabor, J. et Zieliński, B. (2022). Interpretable image classification with differentiable prototypes assignment. *Computer Vision–ECCV 2022 : 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XII*, p. 351–368. Springer.

[Salahuddin *et al.*, 2021] Salahuddin, Z., Woodruff, H. C., Chatterjee, A. et Lambin, P. (2021). Transparency of deep neural networks for medical image analysis : A review of interpretability methods. *arXiv preprint arXiv :2111.02398*.

[Salari *et al.*, 2021] Salari, A., Chatelain, Y., Kiar, G. et Glatard, T. (2021). Accurate simulation of operating system updates in neuroimaging using monte-carlo arithmetic.

[Salimans *et al.*, 2016] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A. et Chen, X. (2016). Improved techniques for training gans. *Advances in neural information processing systems*, 29.

[Schlegl *et al.*, 2019] Schlegl, T., Seeböck, P., Waldstein, S. M., Langs, G. et Schmidt-Erfurth, U. (2019). f-anogan : Fast unsupervised anomaly detection with generative adversarial networks. *Medical image analysis*, 54 : 30–44.

[Sdika, 2008] Sdika, M. (2008). A fast nonrigid image registration with constraints on the jacobian using large scale constrained optimization. *IEEE transactions on medical imaging*, 27(2) : 271–281.

[Sdika, 2013] Sdika, M. (2013). A sharp sufficient condition for b-spline vector field invertibility. application to diffeomorphic registration and interslice interpolation. *SIAM Journal on Imaging Sciences*, 6(4) : 2236–2257.

[Sdika et Pelletier, 2009] Sdika, M. et Pelletier, D. (2009). Nonrigid registration of multiple sclerosis brain images using lesion inpainting for morphometry or lesion mapping. *Human brain mapping*, 30(4) : 1060–1067.

[Selvaraju *et al.*, 2017] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D. et Batra, D. (2017). Grad-cam : Visual explanations from deep networks via gradient-based localization. *Proceedings of the IEEE international conference on computer vision*, p. 618–626.

[Shamonin *et al.*, 2014] Shamonin, D. P., Bron, E. E., Lelieveldt, B. P., Smits, M., Klein, S., Staring, M. et Initiative, A. D. N. (2014). Fast parallel image registration on cpu and gpu for diagnostic classification of alzheimer's disease. *Frontiers in neuroinformatics*, 7 : 50.

[Sill, 1997] Sill, J. (1997). Monotonic networks. *Advances in neural information processing systems*, 10.

[Silva-Rodríguez *et al.*, 2021] Silva-Rodríguez, J., Naranjo, V. et Dolz, J. (2021). Looking at the whole picture : constrained unsupervised anomaly segmentation. *BMVC*.

[Silva-Rodríguez *et al.*, 2022] Silva-Rodríguez, J., Naranjo, V. et Dolz, J. (2022). Constrained unsupervised anomaly segmentation. *Medical Image Analysis*, 80 : 102526.

[Simonyan *et al.*, 2013] Simonyan, K., Vedaldi, A. et Zisserman, A. (2013). Deep inside convolutional networks : Visualising image classification models and saliency maps. *arXiv preprint arXiv :1312.6034*.

[Sivaprasad *et al.*, 2021] Sivaprasad, S., Singh, A., Manwani, N. et Gandhi, V. (2021). The curious case of convex neural networks. *Machine Learning and Knowledge Discovery in Databases. Research Track : European Conference, ECML PKDD 2021, Bilbao, Spain, September 13–17, 2021, Proceedings, Part I 21*, p. 738–754. Springer.

[Smith, 2017] Smith, L. N. (2017). Cyclical learning rates for training neural networks. *2017 IEEE winter conference on applications of computer vision (WACV)*, p. 464–472. IEEE.

[Sorensen, 1948] Sorensen, T. A. (1948). A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons. *Biol. Skar.*, 5 : 1–34.

[Sotiras *et al.*, 2013] Sotiras, A., Davatzikos, C. et Paragios, N. (2013). Deformable medical image registration : A survey. *IEEE Transactions on Medical Imaging*, 32(7) : 1153–1190.

[Sturmfels *et al.*, 2020] Sturmfels, P., Lundberg, S. et Lee, S.-I. (2020). Visualizing the impact of feature attribution baselines. *Distill*, 5(1) : e22.

[Sudre *et al.*, 2017] Sudre, C. H., Li, W., Vercauteren, T., Ourselin, S. et Jorge Cardoso, M. (2017). Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support : Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*, p. 240–248. Springer.

[Sundararajan *et al.*, 2017] Sundararajan, M., Taly, A. et Yan, Q. (2017). Axiomatic attribution for deep networks. *International Conference on Machine Learning*, p. 3319–3328. PMLR.

[Szegedy *et al.*, 2015] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. et Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*, p. 1–9.

[Tan *et al.*, 2019] Tan, T., Yin, S., Liu, K. et Wan, M. (2019). On the convergence speed of amsgrad and beyond. *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, p. 464–470. IEEE.

[Tanguy, 2019] Tanguy, K. (2019). Non asymptotic variance bounds and deviation inequalities by optimal transport.

[Thakur *et al.*, 2020] Thakur, S., Doshi, J., Pati, S., Rathore, S., Sako, C., Bilello, M., Ha, S. M., Shukla, G., Flanders, A., Kotrotsou, A. *et al.* (2020). Brain extraction on mri scans in

presence of diffuse glioma : Multi-institutional performance evaluation of deep learning methods and robust modality-agnostic training. *Neuroimage*, 220 : 117081.

[Thompson *et al.*, 2018] Thompson, A. J., Banwell, B. L., Barkhof, F., Carroll, W. M., Coetzee, T., Comi, G., Correale, J., Fazekas, F., Filippi, M., Freedman, M. S. *et al.* (2018). Diagnosis of multiple sclerosis : 2017 revisions of the mcdonald criteria. *The Lancet Neurology*, 17(2) : 162–173.

[Tustison *et al.*, 2010] Tustison, N. J., Avants, B. B., Cook, P. A., Zheng, Y., Egan, A., Yushkevich, P. A. et Gee, J. C. (2010). N4itk : improved n3 bias correction. *IEEE transactions on medical imaging*, 29(6) : 1310–1320.

[Ulyanov *et al.*, 2016] Ulyanov, D., Vedaldi, A. et Lempitsky, V. (2016). Instance normalization : The missing ingredient for fast stylization. *arXiv preprint arXiv :1607.08022*.

[Varga *et al.*, 2017] Varga, D., Csiszárik, A. et Zombori, Z. (2017). Gradient regularization improves accuracy of discriminative models. *arXiv preprint arXiv :1712.09936*.

[Vukusic *et al.*, 2020] Vukusic, S., Casey, R., Rollot, F., Brochet, B., Pelletier, J., Laplaud, D.-A., De Sèze, J., Cotton, F., Moreau, T., Stankoff, B. *et al.* (2020). Observatoire français de la sclérose en plaques (ofsep) : A unique multimodal nationwide ms registry in france. *Multiple Sclerosis Journal*, 26(1) : 118–122.

[Wachter *et al.*, 2017] Wachter, S., Mittelstadt, B. et Russell, C. (2017). Counterfactual explanations without opening the black box : Automated decisions and the gdpr. *Harv. JL & Tech.*, 31 : 841.

[Wargnier-Dauchelle *et al.*, 2021a] Wargnier-Dauchelle, V., Grenier, T., Durand-Dubief, F., Cotton, F. et Sdika, M. (2021a). Un classifieur plus interprétable pour la sep. *Congrès Société Française de Résonance Magnétique en Biologie et Médecine (SFRMBM)*.

[Wargnier-Dauchelle *et al.*, 2021b] Wargnier-Dauchelle, V., Grenier, T., Durand-Dubief, F., Cotton, F. et Sdika, M. (2021b). A more interpretable classifier for multiple sclerosis. *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, p. 1062–1066. IEEE.

[Wargnier-Dauchelle *et al.*, 2023a] Wargnier-Dauchelle, V., Grenier, T., Durand-Dubief, F., Cotton, F. et Sdika, M. (2023a). Une contrainte faiblement supervisée sur les attributions basées sur le gradient pour une classification interprétable et la détection d'anomalies. *Colloque Français d'Intelligence Artificielle en Imagerie Biomédicale (IABM 2023).*

[Wargnier-Dauchelle *et al.*, 2023b] Wargnier-Dauchelle, V., Grenier, T., Durand-Dubief, F., Cotton, F. et Sdika, M. (2023b). A weakly supervised gradient attribution constraint for interpretable classification and anomaly detection. *IEEE Transactions on Medical Imaging*.

[Werbos, 1974] Werbos, P. (1974). Beyond regression : New tools for prediction and analysis in the behavioral sciences. *PhD thesis, Committee on Applied Mathematics, Harvard University, Cambridge, MA*.

[Wolleb *et al.*, 2022] Wolleb, J., Bieder, F., Sandkühler, R. et Cattin, P. C. (2022). Diffusion models for medical anomaly detection. *International Conference on Medical image computing and computer-assisted intervention*, p. 35–45. Springer.

[Wu et He, 2018] Wu, Y. et He, K. (2018). Group normalization. *Proceedings of the European conference on computer vision (ECCV)*, p. 3–19.

[You *et al.*, 2017] You, S., Ding, D., Canini, K., Pfeifer, J. et Gupta, M. (2017). Deep lattice networks and partial monotonic functions. *Advances in neural information processing systems*, 30.

[Yousefzadeh, 2021] Yousefzadeh, R. (2021). Deep learning generalization and the convex hull of training sets. *CoRR*, abs/2101.09849.

[Yousefzadeh, 2022] Yousefzadeh, R. (2022). Decision boundaries and convex hulls in the feature space that deep learning functions learn from images. *ArXiv*, abs/2202.04052.

[Zaheer *et al.*, 2018] Zaheer, M., Reddi, S., Sachan, D., Kale, S. et Kumar, S. (2018). Adaptive methods for nonconvex optimization. *Advances in neural information processing systems*, 31.

[Zeiler, 2012] Zeiler, M. D. (2012). Adadelta : an adaptive learning rate method. *arXiv preprint arXiv :1212.5701*.

[Zhang *et al.*, 2001] Zhang, Y., Brady, M. et Smith, S. (2001). Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE transactions on medical imaging*, 20(1) : 45–57.

[Zhao *et al.*, 2023] Zhao, W., Bai, L., Rao, Y., Zhou, J. et Lu, J. (2023). Unipc : A unified predictor-corrector framework for fast sampling of diffusion models. *arXiv preprint arXiv :2302.04867*.

[Zhao, 2020] Zhao, Y. (2020). Fast real-time counterfactual explanations. *arXiv preprint arXiv :2007.05684*.

[Zhu *et al.*, 2017] Zhu, J.-Y., Park, T., Isola, P. et Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. *Proceedings of the IEEE international conference on computer vision*, p. 2223–2232.

[Zimmerer *et al.*, 2019] Zimmerer, D., Isensee, F., Petersen, J., Kohl, S. et Maier-Hein, K. (2019). Unsupervised anomaly localization using variational auto-encoders. *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019 : 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part IV 22*, p. 289–297. Springer.

FOLIO ADMINISTRATIF

THESE DE L'INSA LYON, MEMBRE DE L'UNIVERSITE DE LYON

NOM : WARGNIER DAUCHELLE née WARGNIER
(avec précision du nom de jeune fille, le cas échéant)

DATE de SOUTENANCE : 08/12/2023

Prénoms : Valentine, Julia, Jessica, Justine, Charlotte

TITRE : Interprétabilité des réseaux de neurones profonds et segmentation faiblement supervisée des lésions cérébrales sur IRM

NATURE : Doctorat

Numéro d'ordre : 2023ISAL0110

Ecole doctorale : Interdisciplinaire Sciences Santé

Spécialité : Ingénierie biomédicale

RESUME : L'imagerie médicale est un outil fondamental pour diagnostiquer les maladies, suivre leur évolution mais aussi comprendre leur fonctionnement afin de mieux les soigner. L'imagerie par résonance magnétique est une méthode de choix pour visualiser le cortex cérébral et ses pathologies comme la sclérose en plaques, une maladie auto-immune inflammatoire et démyélinisante qui est la première cause de handicap non traumatique chez les jeunes adultes, ou encore les gliomes, qui sont les tumeurs primitives cérébrales les plus courantes.
Pour analyser ces images de manière automatique, les méthodes basées sur l'apprentissage profond présentent de très bonnes performances pour différents types de tâches comme la classification ou la segmentation. Ces méthodes automatiques apportent aux cliniciens une pré-analyse très utile dans leurs études ou diagnostics. Cependant, elles nécessitent beaucoup de données pour leur entraînement. Dans le cas des méthodes de segmentation supervisées, les annotations manuelles nécessaires pour chaque image sont très coûteuses. Le développement de méthodes faiblement ou non-supervisées performantes, ne nécessitant pas ou peu d'annotations manuelles, est donc nécessaire. En outre, dans un domaine critique comme celui de la médecine, il est important que les décisions des réseaux soient explicables et s'appuient sur les signes radiologiques de la pathologie présents dans l'image et utilisés par les cliniciens. Or, les réseaux de neurones profonds sont, de par leur grand nombre de paramètres et les interconnexions non linéaires dont ils sont composés, difficiles à expliquer. Proposer des réseaux explicables et interprétables est donc une problématique forte pour l'analyse d'images médicales par apprentissage profond.
Dans cette thèse, nous avons abordé ces deux thématiques. En nous focalisant sur une tâche de classification entre des images de sujets sains et des images de patients (notamment atteints de sclérose en plaques ou de gliomes), nous avons montré que la décision des classifieurs de l'état de l'art n'est pas forcément pertinente et en accord avec les aprioris médicaux. Cela peut avoir de lourdes conséquences : pour du diagnostic, l'utilisation de tels classifieurs biaisés n'est pas raisonnable et lorsqu'ils sont utilisés au sein d'autres modèles, comme les modèles génératifs, cela peut faire chuter les performances. Nous avons donc proposé des classifieurs plus interprétables avec une décision davantage basée sur les signes radiologiques de la pathologie considérée. Trois solutions ont été proposées. Tout d'abord, nous avons normalisé l'entrée des réseaux de neurones afin d'éliminer les biais présents dans l'image et qui peuvent être utilisés par les réseaux classiques pour prendre leur décision. Ensuite, nous avons contraint les classifieurs au cours de leur entraînement en utilisant les cartes d'attributions, des méthodes de l'état de l'art permettant d'identifier les zones de l'image d'entrée utilisées par le réseau pour prendre sa décision. Enfin, nous avons utilisé des réseaux intrinsèquement explicables : les réseaux monotones. Nous avons notamment proposé une méthode pour transformer n'importe quelle architecture en réseau monotone alors que les réseaux monotones de l'état de l'art étaient limités à des architectures de très faible profondeur. Avec ces réseaux de classification interprétables ne disposant que du label de l'image à l'entraînement, nous pouvons réaliser une segmentation faiblement supervisée des lésions cérébrales, la décision étant basée sur ces dernières.

MOTS-CLÉS : Apprentissage profond, Apprentissage faiblement supervisé, Interprétabilité des réseaux de neurones, Segmentation, Lésions cérébrales, IRM

Laboratoire(s) de recherche : Centre de Recherche en Acquisition et Traitement de l'Image pour la Santé (CREATIS)

Directeur de thèse : François Cotton

Président de jury : Christophe Garcia

Composition du jury :

Dolz, Jose	Professeur	ETS Montréal	Rapporteur
Hudelot, Céline	Professeure	CentraleSupélec	Rapporteure
Mateus, Diana	Professeure	Ecole Centrale Nantes	Rapporteure
Garcia, Christophe	Professeur	INSA Lyon	Examinateur
Petitjean, Caroline	Professeure	Université de Rouen	Examinatrice
Cotton, François	Professeur praticien hospitalier	Université Lyon 1	Directeur de thèse
Sdika, Michaël	Ingénieur de recherche HDR	CNRS	Co-directeur de thèse
Grenier, Thomas	Maître de conférences HDR	INSA Lyon	Encadrant