



Université 20 Aout 1955- Skikda
Faculté des Sciences
Département informatique



Mémoire

En vue de l'obtention du diplôme de Master

Filière : Informatique

Spécialité : Intelligence Artificielle

Détection Assistée par Ordinateur pour l'Analyse de Mammographies Utilisant des Techniques d'Apprentissage Profond

Présenté par

Djeffal Nour El'houda

Superviseur : Dr. Adel Lahsasna

JUIN 2024

Remerciements

Je voudrais exprimer ma profonde gratitude à ALLAH pour m'avoir donné la force et la détermination nécessaires pour mener à bien ce mémoire.

Je tiens également à remercier mon superviseur, Dr. Adel Lahsasna, pour ses conseils inébranlables, son expertise et son engagement envers ce projet. Son mentorat et son approche disciplinée ont été essentiels dans l'élaboration de ce mémoire.

Je suis reconnaissante aux membres du jury qui ont accepté la responsabilité d'évaluer mon travail. Merci pour votre dévouement, vos conseils et votre considération tout au long de ce processus.

Enfin, je me remercie moi-même d'avoir entrepris ce projet. J'ai été mon plus grand soutien, me poussant à surmonter les obstacles et à persévérer durant cette période.

Encore une fois, j'exprime ma profonde gratitude à tous ceux qui ont contribué à mon parcours académique.

Nour El'houda



Dédicace

Je dédie ce mémoire à tous ceux qui ont cru en moi et m'ont soutenu tout au long de ce parcours.

À mes parents, dont les encouragements constants et la confiance en mes capacités ont été une source inépuisable de motivation.

À mes sœurs et leurs fils, ainsi qu'à mon frère, mon bras droit, pour leur soutien indéfectible. Un merci particulier à ma sœur Wafa et à son mari Chakib Berrahal pour leur soutien infailible.

À mes amis et à la PROMO IA 2024, en particulier à mes meilleures amies Dalel et Kawther, pour leur soutien et leur amitié indéfectibles.

À mes amis sur Instagram , qui ont été à mes côtés avec leurs paroles gentilles et leur soutien infini.

Nour El'houda

Abstract

Breast cancer screening poses significant challenges due to high rates of false positives and false negatives, limiting the effectiveness of traditional computer-aided diagnostic (CAD) systems. To enhance the accuracy and reliability of breast anomaly detection, this study aimed to develop an advanced CAD system based on deep learning.

By utilizing a combination of images from the DDSM (Digital Database for Screening Mammography) and CBIS-DDSM (Curated Breast Imaging Subset of DDSM) datasets, we trained and tested various models, including traditional machine learning algorithms such as Random Forest, Naive Bayes, Decision Tree, and XGBoost, as well as artificial neural networks (ANN), along with deep learning models like DenseNet121, Xception, InceptionResNetV2, and MobileNet. Model performances were assessed using standard metrics including accuracy, sensitivity, specificity, precision, and AUC.

The results indicated that deep learning models significantly outperformed traditional methods, with InceptionResNetV2 achieving an accuracy of 96.58%, sensitivity of 96.5%, specificity of 96.58%, and an AUC of 0.987.

These findings demonstrate a notable improvement in performance and suggest that integrating deep learning into CAD systems could reduce diagnostic errors and enhance breast cancer screening efficacy.

Keywords : Breast cancer screening, computer-aided diagnostic (CAD), deep learning, machine learning, ANN, DenseNet121, Xception, InceptionResNetV2, MobileNet, accuracy, sensitivity, specificity, precision, AUC, false positives, false negatives.

Résumé

Le dépistage du cancer du sein présente des défis importants en raison des taux élevés de faux positifs et de faux négatifs, limitant l'efficacité des systèmes de diagnostic assisté par ordinateur (CAD) traditionnels. Pour améliorer la précision et la fiabilité de la détection des anomalies mammaires, cette étude visait à développer un système CAD avancé basé sur l'apprentissage profond.

En utilisant une combinaison d'images provenant des ensembles de données DDSM (Digital Database for Screening Mammography) et CBIS-DDSM (Curated Breast Imaging Subset of DDSM), nous avons entraîné et testé divers modèles, y compris des algorithmes de machine learning traditionnels tels que Random Forest, Naive Bayes, Decision Tree et XGBoost, et MLP, ainsi que des modèles d'apprentissage profond comme DenseNet121, Xception, InceptionResNetV2, MobileNet. Les performances des modèles ont été évaluées à l'aide de métriques standard, notamment l'exactitude, la sensibilité, la spécificité, la précision et l'AUC.

Les résultats ont montré que les modèles d'apprentissage profond surpassent largement les méthodes traditionnelles, avec InceptionResNetV2 atteignant une exactitude de 96,58%, une sensibilité de 96,5 %, une spécificité de 96,58 % et une AUC de 0,987.

Ces résultats démontrent une amélioration significative des performances et suggèrent que l'intégration de l'apprentissage profond dans les systèmes CAD peut réduire les erreurs de diagnostic et augmenter l'efficacité du dépistage du cancer du sein.

Mots-clés : Dépistage du cancer du sein, diagnostic assisté par ordinateur (CAD), apprentissage profond, apprentissage automatique, MLP, DenseNet121, Xception, InceptionResNetV2, MobileNet, exactitude, sensibilité, spécificité, précision, AUC, faux positifs, faux négatifs.

ملخص

تواجه فحوصات سرطان الثدي تحديات كبيرة بسبب ارتفاع معدلات الإيجابيات الكاذبة والسلبيات الكاذبة، مما يحد من فعالية أنظمة التشخيص بمساعدة الكمبيوتر التقليدية. لتحسين دقة وموثوقية اكتشاف الشذوذات الثديية، هدفت هذه الدراسة إلى تطوير نظام CAD متقدم يعتمد على التعلم العميق.

باستخدام مجموعة من الصور من مجموعات البيانات DDSM و CBIS-DDSM ، قنا بتدريب واختبار نماذج مختلفة، بما في ذلك خوارزميات التعلم الآلي التقليدية مثل Forest Random ، Tree Decision ، bayes Naive ، XGBoost، و MLP، بالإضافة إلى نماذج التعلم العميق مثل InceptionResNetV2، Xception، DenseNet121، و MobileNet. تم تقييم أداء النماذج باستخدام مقاييس معيارية، بما في ذلك الدقة، الحساسية، النوعية، الدقة، ومنطقة تحت المنحنى . أظهرت النتائج أن نماذج التعلم العميق تتفوق بشكل كبير على الأساليب التقليدية، حيث حقق InceptionResNetV2 دقة بلغت 96.58%، حساسية بلغت 96.5%، نوعية بلغت 96.58% و AUC بلغت 0.987.

توضح هذه النتائج تحسناً كبيراً في الأداء وتقتصر أن دمج التعلم العميق في أنظمة CAD يمكن أن يقلل من أخطاء التشخيص ويزيد من فعالية فحص سرطان الثدي. الكلمات المفتاحية: فحص سرطان الثدي، التشخيص بمساعدة الكمبيوتر، التعلم العميق، التعلم الآلي، MobileNet، InceptionResNetV2، Xception، DenseNet121، MLP، النوعية، الدقة، AUC، الإيجابيات الكاذبة، السلبيات الكاذبة.

Table des matières

Abstract	IV
Résumé	V
ملخص	VI
Liste des figures	8
Liste des tableaux	11
Chapitre 1: Introduction	12
1.1 Contexte	12
1.1.1 Aperçu du cancer du sein et de sa prévalence	12
1.1.2 Importance du dépistage précoce du cancer du sein	13
1.2 La Mammographie comme Outil de Diagnostic	13
1.2.1 Description de la mammographie et de son rôle dans la détection des anomalies mammaires	13
1.2.2 Types de mammographie :	14
1.2.3 Les Défis de la Mammographie : Efficacité, Faux Positifs et Radiations	14
1.3 Limitations de l'Analyse Traditionnelle de la Mammographie	15
1.3.1 Erreur humaine, fatigue et interprétation subjective	15
1.4 Rôle de la Détection Assistée par Ordinateur (CAD)	15
1.4.1 Les principaux avantages des systèmes CAD	15
1.5 Des motivations pour l'utilisation de l'apprentissage profond dans les systèmes CAD en mammographie	16
1.5.1 Les méthodes CAD traditionnelles ont des limitations	16

1.5.2	L'apprentissage profond offre des avantages significatifs pour l'analyse automatique des mammographies	16
1.6	Objectifs du Mémoire	16
1.6.1	Développer un système CAD basé sur l'apprentissage profond pour l'analyse des mammographies	16
1.6.2	Évaluer l'efficacité et la précision du système proposé	17
1.7	Structure du Mémoire	17
Chapitre 2: Présentation des algorithmes d'apprentissage automatique		19
2.1	Introduction	19
2.2	L'intelligence artificielle	19
2.3	L'apprentissage automatique	21
2.3.1	L'apprentissage supervisé	21
2.3.1.1	Les machines à vecteur de support SVM	22
2.3.1.2	L'Arbre de décision	23
2.3.1.3	Forêt aléatoire	24
2.3.1.4	Régression logistique	24
2.3.1.5	La régression linéaire	24
2.3.1.6	L'algorithme des k plus proches voisins	26
2.3.2	L'apprentissage non supervisé	27
2.3.2.1	Le clustering K-moyenne	28
2.3.3	L'apprentissage semi-supervisé (SSL)	28
2.3.4	L'apprentissage par renforcement	29
2.4	Les réseaux de neurones	30
2.4.1	Introduction	30
2.4.1.1	Neurone biologique	31
2.4.1.2	Neurone Artificiel	32
2.4.1.3	Structure des réseaux de neurones artificiels	32
2.4.1.4	Types des réseaux de neurones artificiels	33
2.4.2	L'initialisation du poids	34

2.4.3	les fonctions d'activation	36
2.4.3.1	La fonction sigmoïde	37
2.4.3.2	La fonction tangente hyperbolique	37
2.4.3.3	La fonction ReLU	37
2.4.3.4	La fonction LReLU	38
2.4.3.5	La fonction PReLU	39
2.4.3.6	La fonction RReLU	39
2.4.3.7	La fonction ELU	39
2.4.3.8	La fonction SELU	40
2.4.3.9	La fonction Swish	40
2.4.4	Les fonctions de pertes (LOSS)	41
2.4.4.1	Erreur Quadratique Moyenne(MSE)	42
2.4.4.2	Erreur Absolue Moyenne	42
2.4.4.3	Entropie croisée binaire	42
2.4.4.4	Entropie croisée catégorielle	43
2.4.5	Les Optimiseurs	43
2.4.5.1	La descente de gradient (GD)	43
2.4.5.2	Momentum	47
2.4.5.3	Nesterov Momentum	47
2.4.5.4	Le gradient accéléré de Nesterov (NAG)	48
2.4.5.5	Adagrad	48
2.4.5.6	RMSprop	49
2.4.5.7	Adadelta	49
2.4.5.8	Adam	50
2.4.5.9	AdaMax	50
2.4.5.10	Nadam	50
2.5	Apprentissage profond (Deep learning ou DL)	51
2.5.1	Les réseaux de neurones convolutifs (CNN)	51
2.5.2	Les avantages CNN	52

2.5.3	Les inconvénients de CNN	52
2.5.4	L'opération de convolution	53
2.5.5	Mise en commun (Pooling)	54
2.5.6	Aplanissement (Flattening)	55
2.5.7	Quelques Architecture basées CNN	56
2.5.7.1	DenseNet	56
2.5.7.2	VGGNet	56
2.5.7.3	Xception	56
2.5.7.4	MobileNet	56
2.6	L'apprentissage par transfert	56
2.7	Domaines d'utilisation de l'apprentissage automatique	57
2.8	Conclusion	58
Chapitre 3: REVUE DE LA LITTERATURE		59
3.1	Introduction	59
3.2	Systèmes CAD Traditionnels en Mammographie	59
3.2.1	Développement historique et méthodologies clés	59
3.3	Apprentissage Profond en Imagerie Médicale	61
3.3.1	Aperçu de l'apprentissage profond	61
3.3.2	Applications en imagerie médicale	61
3.4	Bases de Données et Prétraitement	62
3.4.1	Base de données MG	63
3.4.2	Méthodes de Prétraitement d'Images :	64
3.5	Les performances des modèles d'apprentissage automatique	65
3.6	Les techniques d'apprentissage profond appliquées à l'analyse des mammogra- phies	66
3.6.1	classification multiclasse des anomalies du cancer du sein à l'aide du CNN	66
3.6.1.1	Transférer l'apprentissage à l'aide de ResNet50 pré-entraîné	67

3.6.2	Modèle CNN Amélioré avec Noyaux de Petite Dimension et Cadres d'Apprentissage Résiduel Profond	67
3.6.3	Classification BI-RADS basée sur l'Architecture Profonde EfficientNET	68
3.6.4	Modèle Hybride d'Apprentissage Profond pour la Classification Précise des Anomalies Mammaires	69
3.6.4.1	Réduction dimensionnelle à l'aide de LR-PCA	70
3.7	Les performances des modèles d'apprentissage profonds	71
3.8	Défis et Lacunes dans la Recherche Actuelle	72
3.9	Conclusion	74
Chapitre 4: La méthodologie		75
4.1	Introduction	75
4.2	Collecte des données	75
4.2.1	Description du jeu de données DDSM	75
4.2.2	Prétraitement des données	77
4.2.2.1	Chargement des données :	77
4.2.2.2	Mélange du Dataset	78
4.2.2.3	Mise en Cache du Dataset	79
4.2.2.4	Définition du Dictionnaire de Caractéristiques	80
4.2.2.5	Mapping de la Fonction de Parsing	80
4.2.2.6	Conversion Eager Execution de TensorFlow à NumPy	81
4.2.2.7	Décompression et décodage des images :	82
4.2.2.8	Conversion des images en images 299x299 pixels :	82
4.2.2.9	Découpage et redimensionnement des images négatives et positives :	83
4.2.2.10	Conversion en format RVB (Red, Green, Blue) :	84
4.2.2.11	Normalisation des données :	84
4.2.2.12	Ajout d'Éléments à une Liste (Append) :	85
4.3	Architecture des modèles	86
4.3.1	Modèles d'apprentissage automatique	86

4.3.1.1	XGBoost (Extreme Gradient Boosting)	86
4.3.1.2	Forêts aléatoires (Random Forest)	86
4.3.1.3	Arbres de décision (Decision Trees)	87
4.3.1.4	Le réseau neuronal multicouche perceptron (MLP)	87
4.3.1.5	Naive Bayes	88
4.3.2	Description des algorithmes Deep CNN	89
4.3.3	Modèles de deep learning testées	89
4.3.3.1	DenseNet121 :	89
4.3.3.2	MobileNet	91
4.3.3.3	InceptionResNetV2	92
4.3.3.4	Xception	93
4.4	Entraînement des Modèles	94
4.4.1	Modèles d'apprentissage automatique traditionnel :	94
4.4.2	Modèles de Réseaux de Neurones Convolutionnels (CNN) :	94
4.5	Évaluation des Modèles	95
4.5.1	Métriques d'Évaluation des Performances en Classification des Anomalies Mammaires	95
4.6	Conclusion	96
Chapitre 5: Mise en Œuvre du Système		97
5.1	Introduction	97
5.2	Vue d'Ensemble de l'Architecture du Système	97
5.3	Technologies et Outils	99
5.3.1	Langages de Programmation	99
5.3.2	Frameworks et Bibliothèques	99
5.3.3	Environnement de Développement	100
5.4	Interface Utilisateur	100
5.4.1	Considérations de Conception pour l'Interface Utilisateur	100
5.4.2	Captures d'Écran de l'Interface Utilisateur	101
5.4.3	Description des Fonctionnalités Fournies par l'Interface Utilisateur	103

5.5 Conclusion	104
Chapitre 6: Résultats	105
6.1 Introduction	105
6.2 Présentation des Performances du Modèle sur des Ensembles de Test	105
6.3 Analyse Visuelle et Interprétation des Performances des Modèles	106
6.3.1 Matrices de Confusion	109
6.4 Analyse Comparative avec les Systèmes CAD Existants	111
6.5 Discussion des Résultats	113
6.6 Implications pour la Pratique Clinique	113
6.7 conclusion :	114
Chapitre 7: Conclusions et Travaux Futurs	115
7.1 Résumé du Problème de Recherche, de la Méthodologie, des Résultats et des Contributions	115
7.2 Limitations	116
7.3 Directions Futures	116
Références	117

Liste des figures

1.1	Une gynécologue consulte une mammographie.	12
1.2	Machine de mammographie	13
1.3	Processus de dépistage du cancer du sein avec utilisation de la mammographie et de l'intelligence artificielle	17
2.1	Les relations entre l'intelligence artificielle, l'apprentissage automatique et l'apprentissage profond.	20
2.2	Types d'apprentissage en apprentissage par machine	21
2.3	Schéma d'apprentissage supervisé "supervised learning"	22
2.4	Exemple simple représentant la classification par SVM	22
2.5	Un arbre de décision pour un apprentissage non linéaire.	23
2.6	Un exemple de la structure d'une forêt aléatoire prenant en compte plusieurs arbres de décision	24
2.7	la régression Linéaire.	25
2.8	Exemple de K-plus proche voisins (KPPV).	26
2.9	Schéma d'apprentissage non supervisé "unsupervised learning"	27
2.10	Schéma d'apprentissage semi supervisé "semi supervised learning"	29
2.11	: Schéma d'apprentissage par renforcement "reinforcement learning".	30
2.12	Architecture d'un MLP à deux couches cachées.	31
2.13	neurone biologique	31
2.14	neurone artificielle.	32
2.15	Structure des réseaux de neurones artificiels	33
2.16	Types des réseaux de neurones artificiels.	34

2.17	L'initialisation du poids.	35
2.18	fonction sigmoid et tanh	37
2.19	fonction RELU	38
2.20	fonction ELU	40
2.21	fonction Switch	41
2.22	Algorithme de descente de gradient	44
2.23	Descente de gradient stochastique	45
2.24	Minima globaux Vs. Minima locaux	46
2.25	La comparaison des algorithmes SGD avec et sans quantité de mouvement	47
2.26	Une architecture typique d'un CNN	52
2.27	Exemple d'une opération de convolution	53
2.28	Exemple d'opération de convolution avec padding, un tenseur d'entrée de taille 5×5, noyau de 3×3 et un stride de 1.	54
2.29	Types d'opération de pooling	55
2.30	schéma d'un réseau à convolution avec la couche Flattening	55
2.31	Apprentissage par transfert	57
3.1	Système CAD de segmentation et classification des tumeurs mammaires	60
3.2	(a) Image originale (b) Image après CLAHE.	65
3.3	Layers in CNN.	67
3.4	Organigramme du modèle de classification BI-RADS présenté.	69
3.5	Cadre CAD proposé pour la classification des lésions mammaires à partir des mammographies.	70
4.1	Exemple des mammographies bénignes et malignes DDSM	76
4.2	Images bénignes, malignes et normales de la base de données DDSM	76
4.3	architecture globale de xgboost	86
4.4	Un arbre de décision pour un apprentissage non linéaire.	87
4.5	Structure MLP	88
4.6	L'architecture DenseNet	90

4.7	L'architecture Inceptionresnet	92
5.1	Diagramme en Blocs Illustrant les Composants Principaux de la phase 1	98
5.2	Diagramme en Blocs Illustrant les Composants Principaux de la phase 2:Dé- ploiment	98
5.3	interface principale	101
5.4	Interface de connexion pour un système de classification du cancer du sein. . .	102
5.5	Interface d'inscription pour un système de classification du cancer du sein. . .	102
5.6	Interface d'enregistrement d'image et de saisie des données patient.	103
5.7	Interface de visualisation des résultats.	103
6.1	Comparaison de l'Accuracy des modèles de classification	106
6.2	Comparaison de la précision des modèles de classification	107
6.3	Comparaison de recall des modèles de classification	107
6.4	Comparaison de F1 Score des modèles de classification	108
6.5	Comparaison de l'AUC des modèles de classification	108
6.6	Comparaison de le des modèles de classification	109
6.7	Matrice de confusion-InceptionResNetV2	109
6.8	Matrice de confusion-Random Forest	110

Liste des tableaux

2.1	Comparaison entre les techniques d'initialisation du poids.	36
3.1	Méthodes CAD basées sur l'apprentissage automatique pour le diagnostic du cancer du sein.	66
3.2	Méthodes CAD basées sur l'apprentissage profond pour le diagnostic du cancer du sein.	72
6.1	Performances des Modèles Traditionnels de Machine Learning.	105
6.2	Performances des Modèles Basés sur l'Apprentissage Profond.	106
6.3	Performances des Systèmes CAD basés sur les Modèles Traditionnels et d'Apprentissage Profond.	112

Chapitre 1

Introduction

1.1 Contexte

1.1.1 Aperçu du cancer du sein et de sa prévalence

Le cancer du sein est défini comme une croissance incontrôlée de cellules anormales dans les tissus mammaires. Ces cellules peuvent envahir les tissus environnants et se propager à d'autres parties du corps via la circulation sanguine ou lymphatique, un processus connu sous le nom de métastase. Le cancer du sein représente une menace grave pour la vie humaine en raison de sa capacité à détruire les tissus corporels. En 2019, le cancer a été classé comme la principale cause de décès dans le monde, avec environ 10 millions de vies perdues et 19,3 millions de nouveaux cas enregistrés. Les prévisions du Centre International de Recherche sur le Cancer indiquent que ces chiffres pourraient atteindre 29,5 millions de nouveaux cas et 16,4 millions de décès d'ici 2040 [1].



Fig. 1.1 – Une gynécologue consulte une mammographie.

[2]

1.1.2 Importance du dépistage précoce du cancer du sein

Le dépistage précoce du cancer du sein est crucial pour améliorer les chances de traitement efficace et réduire la mortalité. La détection à un stade précoce permet de traiter les tumeurs avant qu'elles ne se propagent, augmentant ainsi les chances de guérison. Cependant, détecter les cellules malignes à un stade précoce reste un défi majeur en raison de leur petite taille et de leur difficulté à être suivies[1].

1.2 La Mammographie comme Outil de Diagnostic

1.2.1 Description de la mammographie et de son rôle dans la détection des anomalies mammaires

La mammographie est une technique d'imagerie médicale qui utilise des rayons X pour examiner les seins. Elle est principalement employée pour détecter des anomalies mammaires, y compris des tumeurs bénignes et malignes, avant même qu'elles ne soient détectables par un examen physique. La mammographie joue un rôle crucial dans le dépistage précoce du cancer du sein, permettant d'identifier des tumeurs à un stade où elles sont plus facilement traitables. Bien que passer une mammographie ne réduise pas directement le risque de développer un cancer du sein, elle facilite la détection précoce de tout signe de la maladie. Avant même qu'une grosseur ne soit palpable lors d'un examen physique, la mammographie peut révéler la présence de tumeurs, qu'elles soient bénignes ou malignes, ainsi que de kystes. Si la mammographie détecte quelque chose de suspect, comme une masse, un échantillon de tissu est prélevé de la zone concernée pour une biopsie, qui est ensuite examinée en détail[3].



Fig. 1.2 – Machine de mammographie
[4]

1.2.2 Types de mammographie :

1. **Mammographie de dépistage :** La mammographie de dépistage est effectuée sur des femmes qui ne présentent aucun symptôme, dans le but de détecter tout changement dans la forme ou la structure du sein. Cette procédure implique l'utilisation de faibles doses de rayons X pour obtenir des images des seins. Elle permet d'identifier les problèmes avant qu'ils ne deviennent apparents.
2. **Mammographie diagnostique :** Si la mammographie de dépistage révèle quelque chose de suspect, une mammographie diagnostique peut être nécessaire. Elle permet d'examiner de près les changements inhabituels dans les seins, comme les masses palpables ou les altérations de la forme des seins. La mammographie diagnostique est souvent recommandée aux femmes de plus de 30 ans ou à celles qui ont des antécédents familiaux de cancer du sein.

1.2.3 Les Défis de la Mammographie : Efficacité, Faux Positifs et Radiations

Certains des Défis de la Mammographie sont les suivants[3] :

1. **Variabilité de l'efficacité :** L'efficacité de la mammographie varie d'une femme à l'autre.
2. **Sensibilité limitée :** La sensibilité de la mammographie se situe entre 83 % et 95 %, ce qui signifie qu'elle peut manquer jusqu'à 17 % des cas de cancer du sein.
3. **Résultats erronés :** Les faux positifs et les faux négatifs sont fréquents, ce qui peut entraîner des diagnostics incorrects et la nécessité de tests supplémentaires.
4. **Exposition aux radiations :** La mammographie expose le corps à des radiations, ce qui peut poser des risques, notamment pendant la grossesse en raison du potentiel de malformations congénitales.
5. **Qualité des images :** Les mammographies produisent des images en noir et blanc, où les cancers et les microcalcifications apparaissent comme des zones plus blanches.
6. **Densité des tissus mammaires :** Les tissus mammaires denses peuvent compliquer la détection du cancer, augmentant ainsi le risque de faux négatifs.
7. **Besoin d'optimisation :** Il est nécessaire d'optimiser la mammographie pour améliorer sa précision en tant que méthode de dépistage du cancer du sein.

1.3 Limitations de l'Analyse Traditionnelle de la Mammographie

1.3.1 Erreur humaine, fatigue et interprétation subjective

Malgré l'efficacité de la mammographie en tant qu'outil de dépistage du cancer du sein, plusieurs limitations liées à l'analyse traditionnelle subsistent. L'interprétation des images mammographiques repose principalement sur l'expertise des radiologues. Cependant, ces professionnels peuvent commettre des erreurs en raison de la fatigue, du stress ou d'autres limitations cognitives. De plus, l'interprétation subjective des images peut entraîner des variations dans le diagnostic, augmentant ainsi les risques de faux positifs (diagnostic erroné de cancer) et de faux négatifs (non-détection d'un cancer existant). Ces limitations soulignent la nécessité de solutions avancées pour améliorer la précision et la fiabilité des diagnostics mammographiques[5].

1.4 Rôle de la Détection Assistée par Ordinateur (CAD)

Les systèmes CAD en mammographie sont des logiciels conçus pour assister les radiologues dans l'analyse des images mammographiques. Ils sont capables de détecter et de caractériser automatiquement les anomalies potentiellement cancéreuses sur les mammographies[6].

1.4.1 Les principaux avantages des systèmes CAD

1. **Amélioration de la précision diagnostique** : Les systèmes CAD peuvent aider les radiologues à identifier des lésions qui auraient pu être manquées lors de l'analyse manuelle, augmentant ainsi le taux de détection du cancer du sein.
2. **Réduction des erreurs d'interprétation** : Les systèmes CAD sont moins sujets à la fatigue et à l'erreur humaine que les radiologues, offrant une analyse plus objective et cohérente des images.
3. **Aide à la prise de décision** : Les informations fournies par les systèmes CAD, comme le niveau de suspicion des lésions détectées, peuvent guider les radiologues dans leur interprétation et leur prise de décision. Ainsi, l'utilisation des systèmes CAD en mammographie permet d'améliorer la précision du dépistage et de la détection précoce du cancer du sein, contribuant à réduire la mortalité liée à cette maladie[6].

1.5 Des motivations pour l'utilisation de l'apprentissage profond dans les systèmes CAD en mammographie

1.5.1 Les méthodes CAD traditionnelles ont des limitations

- Elles nécessitent une extraction manuelle de caractéristiques pertinentes, un processus long et fastidieux.
- Leur performance dépend fortement de la qualité de ces caractéristiques extraites.
- Elles ont une capacité de généralisation limitée à de nouvelles données[7].

1.5.2 L'apprentissage profond offre des avantages significatifs pour l'analyse automatique des mammographies

- Apprentissage automatique des caractéristiques pertinentes directement à partir des données, sans intervention humaine.
- Meilleure capacité de généralisation grâce à l'apprentissage de représentations de haut niveau.
- Amélioration des performances de détection et de classification des anomalies mammaires.
- Potentiel pour une analyse plus précise et objective des images mammographiques. Ainsi, l'utilisation de l'apprentissage profond dans les systèmes CAD en mammographie permet de surmonter les limitations des méthodes traditionnelles et d'offrir de meilleures performances pour la détection précoce du cancer du sein[7].

1.6 Objectifs du Mémoire

1.6.1 Développer un système CAD basé sur l'apprentissage profond pour l'analyse des mammographies

L'objectif principal de cette mémoire est de développer un système CAD innovant utilisant des techniques d'apprentissage profond pour l'analyse des mammographies. Ce système vise à améliorer la détection des anomalies mammaires en fournissant des analyses plus précises et fiables que les méthodes existantes.

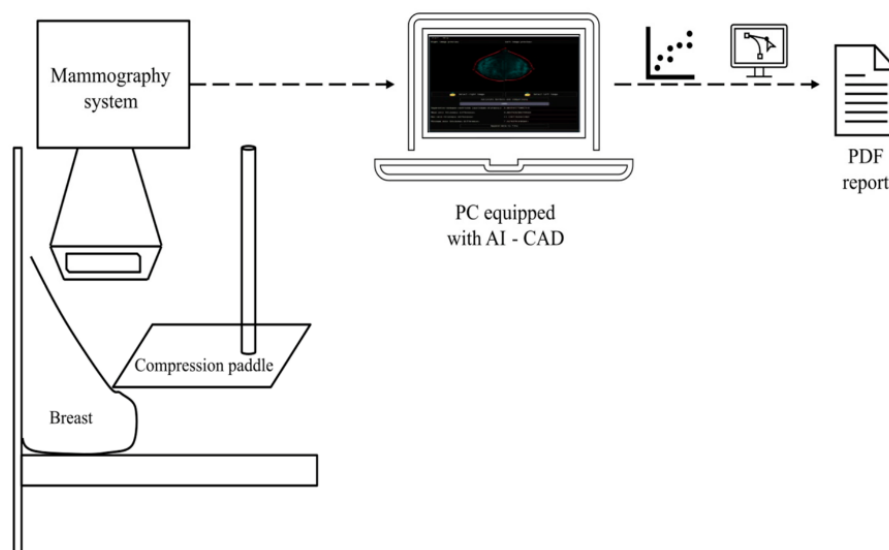


Fig. 1.3 – Processus de dépistage du cancer du sein avec utilisation de la mammographie et de l’intelligence artificielle

[8].

1.6.2 Évaluer l’efficacité et la précision du système proposé

Un autre objectif important est d’évaluer l’efficacité et la précision du système proposé. Cela impliquera des tests rigoureux sur des ensembles de données mammographiques réelles pour comparer les performances du système à celles des méthodes d’apprentissage automatique traditionnelles et des systèmes CAD existants. Les résultats de cette évaluation permettront de déterminer si le système basé sur l’apprentissage profond peut effectivement améliorer le dépistage et le diagnostic du cancer du sein.

1.7 Structure du Mémoire

Le mémoire est organisé en plusieurs chapitres, chacun abordant un aspect clé de la recherche sur la détection assistée par ordinateur pour l’analyse de mammographies utilisant des techniques d’apprentissage profond.

Le **Chapitre 1 : Introduction** fournit le contexte général du cancer du sein et son impact sur la santé publique, en soulignant l’importance de la mammographie pour la détection précoce des anomalies mammaires. Il discute des limitations de l’analyse traditionnelle dues à l’erreur humaine et à l’interprétation subjective, avant de présenter les avantages des systèmes de détection assistée par ordinateur (CAD). Ce chapitre justifie également l’utilisation de l’apprentissage profond pour surmonter les limitations des méthodes CAD traditionnelles, en énonçant les objectifs du mémoire et en présentant un plan général des chapitres suivants.

Le **Chapitre 2 : Présentation des algorithmes d’apprentissage automatique** présente un panorama des recherches en intelligence artificielle, avec un accent particulier sur les techniques d’apprentissage automatique, notamment les approches supervisées et non supervisées. Il ex-

plorent les avantages et les inconvénients de chaque méthode et introduit les concepts de base des réseaux de neurones artificiels et convolutifs. Ce chapitre vise à fournir une compréhension approfondie des dernières technologies utilisées dans le domaine de la classification automatique.

Le **Chapitre 3 : Revue de la Littérature** examine les systèmes CAD traditionnels en mammographie, retrace leur développement historique et explore les méthodologies clés. Il passe en revue les modèles existants d'apprentissage profond pour l'analyse des mammographies et identifie les défis non résolus et les lacunes dans la recherche actuelle, mettant en lumière les domaines nécessitant des améliorations.

Le **Chapitre 4 : Méthodologie** décrit la collecte et le prétraitement des données mammographiques, y compris la normalisation, l'augmentation et la segmentation des images. Il justifie le choix des architectures d'apprentissage profond spécifiques et détaille les procédures d'entraînement et de validation des modèles, ainsi que les métriques utilisées pour évaluer leur performance.

Le **Chapitre 5 : Mise en Œuvre du Système** présente une vue d'ensemble de l'architecture du système proposé, illustrée par un diagramme en blocs des composants principaux. Ce chapitre discute des technologies et outils utilisés, notamment les langages de programmation, les frameworks et bibliothèques, ainsi que les spécifications de l'environnement de développement. Il décrit également les considérations de conception de l'interface utilisateur, en fournissant des captures d'écran et en expliquant les fonctionnalités offertes.

Le **Chapitre 6 : Résultats** expose les performances du modèle sur des ensembles de test et effectue une analyse comparative avec les systèmes CAD existants. Il discute des résultats obtenus et de leurs implications pour la pratique clinique.

Enfin, le **Chapitre 7 : Conclusions et Travaux Futurs** résume le problème de recherche, la méthodologie, les résultats et les contributions du mémoire. Il identifie les limitations de l'étude actuelle et propose des directions futures pour surmonter ces limitations et explorer de nouvelles avenues de recherche.

La section **Références** fournit une liste complète des sources et références bibliographiques utilisées tout au long du mémoire.

Chapitre 2

Présentation des algorithmes d'apprentissage automatique

2.1 Introduction

L'apprentissage est un processus naturel du comportement humain et devient également une partie importante des machines. Aujourd'hui, l'intelligence artificielle (IA), l'apprentissage automatique et l'apprentissage profond (Deep Learning) sont trois termes populaires qui sont parfois utilisés de manière interchangeable pour décrire des systèmes ou des logiciels qui se comportent intelligemment. Ces technologies apparaissent dans d'innombrables travaux scientifiques. A travers ces techniques plusieurs programmes sont développés pour automatiser des tâches spécifiques et qui sont utilisée pour un large éventail d'activités. Grâce à l'apparition de ces technologies ont permet de simplifier les taches dans divers domaines y compris le contrôle des robots des voitures autonomes et d'assistants virtuels, la télédétection, le commerce électronique et le diagnostic médical... ect. Dans ce chapitre nous présentons les concepts essentiels pour comprendre les techniques de l'intelligence artificielle et de l'apprentissage automatique.

2.2 L'intelligence artificielle

L'histoire de l'intelligence artificielle remonte aux années 1950, lorsque les pionniers de l'inform-atique ont commencé à explorer la possibilité que les ordinateurs puissent imiter les processus de pensée humaine. Cette exploration, qui demeure centrale dans nos préoccupations actuelles, a ouvert la voie à un domaine d'étude fascinant. En termes simples, selon Chollet, l'intelligence artificielle vise à automatiser des tâches cognitives généralement accomplies par les humains. Au commencement, les premiers programmes d'échecs étaient fondés sur des règles établies par les programmeurs, sans recours à des techniques d'apprentissage automatique.

L'IA symbolique, aussi connue sous le nom d'IA classique, a émergé de l'idée dominante selon laquelle une intelligence artificielle comparable à celle humaine pourrait être obtenue en établissant un ensemble étendu de règles explicites. Les années 1980 ont été marquées par un

progrès significatif de l'IA avec l'apparition des systèmes experts. Ces systèmes ont démontré leur efficacité dans la résolution de problèmes logiques bien définis, comme ceux du jeu d'échecs. Cependant, ils ont montré leurs limites face à des problèmes complexes et ambigus tels que la classification d'images, la reconnaissance vocale et la traduction automatique, pour lesquels la définition de règles explicites était difficile. C'est alors que s'est manifesté un nouveau paradigme : l'apprentissage automatique. [9]

L'apprentissage automatique propose une approche différente, utilisant des outils de programmation, des langages et des algorithmes pour traiter un large éventail d'applications au-delà de la simple robotique. De la traduction automatique à la sécurité, en passant par la reconnaissance vocale, la reconnaissance d'écriture manuscrite, la médecine et bien d'autres domaines, l'IA s'est étendue à divers secteurs informatiques, souvent de manière discrète, mais avec un potentiel d'amélioration de quasiment tous les types de logiciels. Le domaine de l'IA évolue rapidement grâce aux progrès des capacités informatiques qui permettent le développement d'algorithmes auparavant inimaginables. Étant donné le rôle crucial de l'IA dans la construction de notre avenir, il est impératif que les développeurs et les informaticiens comprennent ses mécanismes sous-jacents et les appliquent efficacement [10].

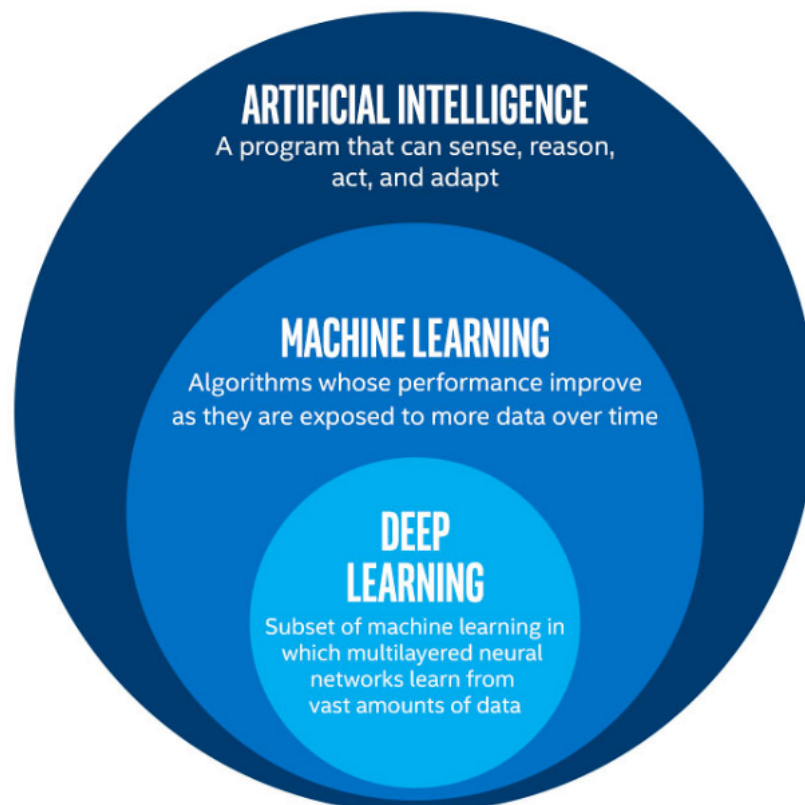


Fig. 2.1 – Les relations entre l'intelligence artificielle, l'apprentissage automatique et l'apprentissage profond.

2.3 L'apprentissage automatique

L'apprentissage automatique, une sous-catégorie de l'intelligence artificielle, se caractérise par son auto-apprentissage basé sur des algorithmes. Concrètement, cela signifie que le système apprend de ses expériences. Par exemple, en fournissant des données en entrée, le système apprend à partir de ces données pour générer des réponses. Ainsi, il devient de plus en plus performant au fil du temps sans nécessiter d'intervention humaine. Ce processus repose sur des algorithmes d'apprentissage statistique qui évoluent et s'améliorent de manière autonome, sans besoin d'intervention humaine [12].

Généralement, les approches d'apprentissage automatique peuvent être catégorisées en quatre types principaux : supervisé, non supervisé, semi-supervisé et par renforcement.

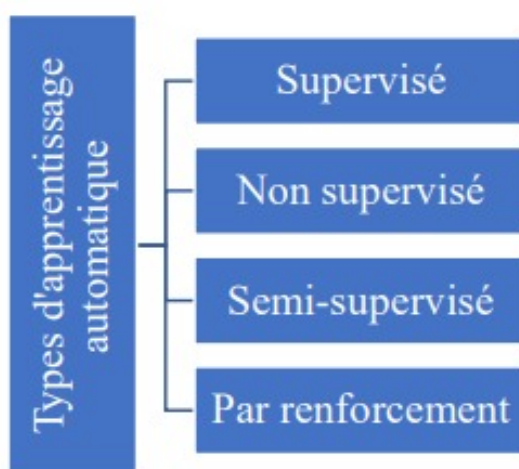


Fig. 2.2 – Types d'apprentissage en apprentissage par machine [13]

2.3.1 L'apprentissage supervisé

L'objectif de l'apprentissage supervisé est de s'appuyer sur des données étiquetées pour apprendre. Deux principaux types d'algorithmes supervisés sont généralement distingués : les algorithmes de régression, qui modèlent l'espace d'entrée en un domaine de valeurs réelles (par exemple, la régression bayésienne et la régression par vecteur de support), et les algorithmes de classification, qui segmentent l'espace d'entrée en classes prédéfinies (tels que les réseaux neuronaux artificiels, les arbres de décision, les forêts d'arbres et les machines à vecteurs de support SVM) [14].

2.3.1.2 L'Arbre de décision

Un arbre de décision représente graphiquement les choix et leurs résultats sous forme d'une structure en arbre. Les arêtes de cet arbre représentent les critères ou les règles de décision, tandis que les nœuds représentent des événements ou des options. Chaque nœud contient un ensemble de caractéristiques pour une classification, et chaque branche indique une valeur possible pour le nœud [17]. Lors de la catégorisation des instances à l'aide d'arbres de décision, l'algorithme commence au nœud racine et descend dans l'arbre en évaluant l'attribut défini par chaque nœud. Il suit ensuite la branche correspondant à la valeur de l'attribut de l'instance à catégoriser. Une fois qu'un nœud feuille est atteint, représentant la classification finale de l'instance, la procédure prend fin.

Deux mesures couramment utilisées pour diviser les nœuds d'un arbre de décision sont le "gini" pour l'impureté de Gini et l'entropie pour le gain d'information. Ces mesures aident à déterminer l'attribut optimal pour diviser les données à chaque nœud, en vue d'optimiser le gain d'information ou de réduire l'impureté.

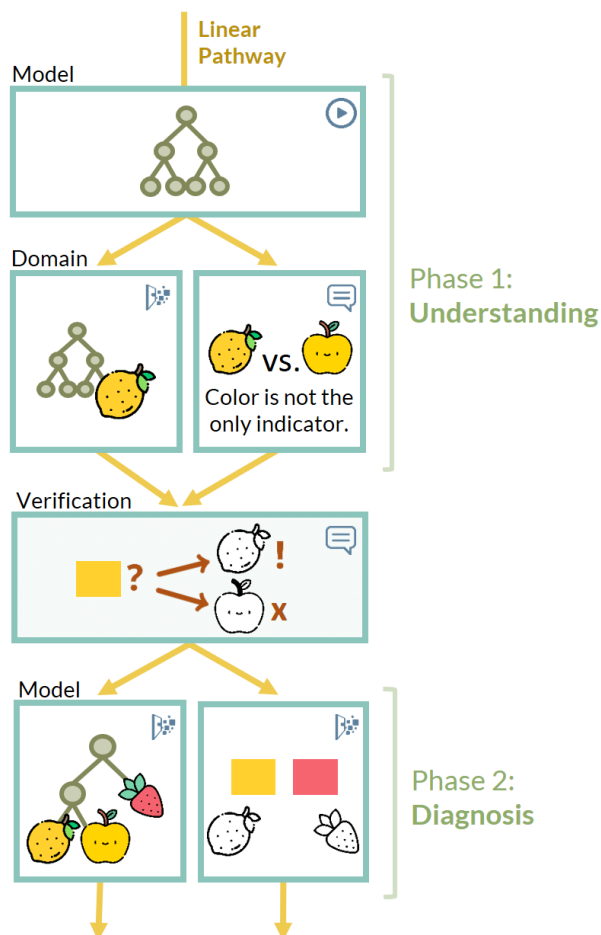


Fig. 2.5 – Un arbre de décision pour un apprentissage non linéaire. [18]

2.3.1.3 Forêt aléatoire

La forêt aléatoire est une technique d'apprentissage d'ensemble qui combine simultanément plusieurs arbres de décision et agrège leurs résultats par moyennage ou vote, dans le but de réduire le sur-apprentissage et d'améliorer la précision des prédictions. Cette méthode fonctionne en construisant un certain nombre d'arbres de décision avec une variance contrôlée, en utilisant des techniques telles que l'agrégation bootstrap (Bagging) et la sélection aléatoire des caractéristiques. Les variables catégorielles et continues peuvent être intégrées avec succès dans les forêts aléatoires, ce qui en fait une technique adaptée à la fois pour la classification et la régression.

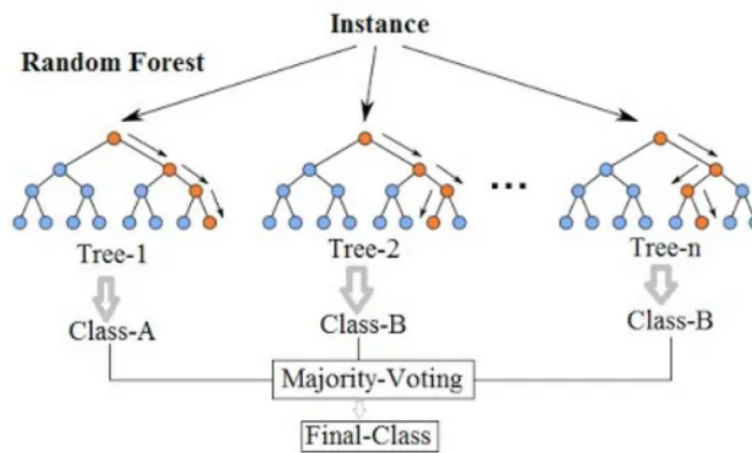


Fig. 2.6 – Un exemple de la structure d'une forêt aléatoire prenant en compte plusieurs arbres de décision

[19]

2.3.1.4 Régression logistique

La régression logistique est une approche statistique probabiliste très répandue pour traiter les problèmes de classification dans le cadre de l'apprentissage automatique. La fonction logistique, communément appelée fonction sigmoïde comme montré l'Équation,

$$g(z) = \frac{1}{1 + \exp(-z)}$$

est utilisée pour calculer la probabilité qu'une entrée entre dans une catégorie spécifique.

2.3.1.5 La régression linéaire

La régression est un concept largement employé en analytique et en science des données, servant à représenter la relation entre une fonction cible et des variables indépendantes. Elle est particulièrement utile lorsque l'objectif est de prédire une valeur continue. La Figure 1-9 illustre le fonctionnement de la régression linéaire [20].

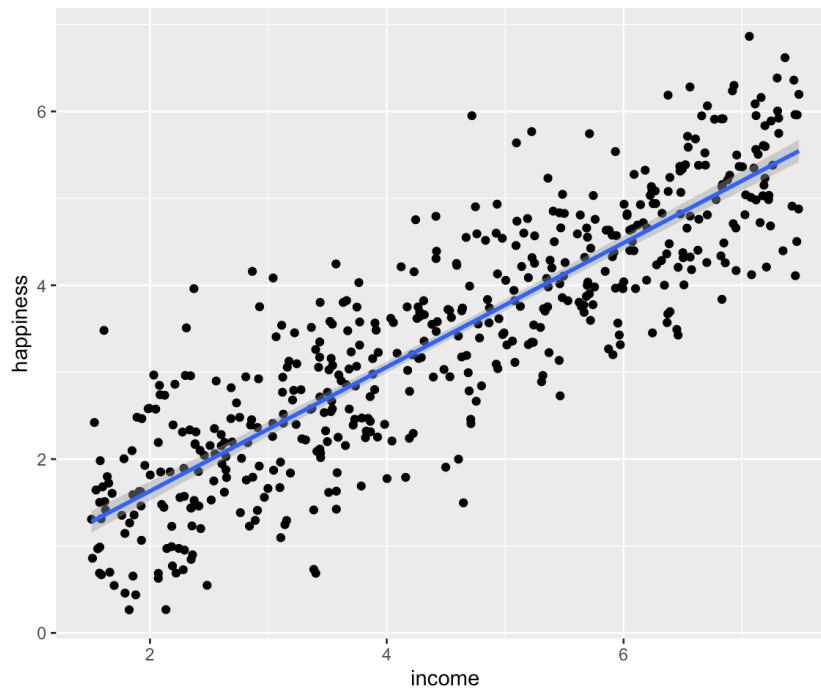


Fig. 2.7 – la régression Linéaire.
[21]

Parmi les hypothèses de la régression linéaire :

- Il doit toujours exister une relation linéaire entre les variables dépendantes et les variables indépendantes.
- Les données doivent suivre une distribution normale.
- Aucune colinéarité ne doit être présente entre les variables indépendantes.
- La multicollinéarité, caractérisée par une forte corrélation linéaire entre les variables indépendantes, doit être évitée, car ces variables corrélées peuvent être redondantes et doivent être traitées.
- L'homoscédasticité est nécessaire, ce qui signifie que la variance des erreurs doit être constante au fil du temps.
- La variance ne doit pas être hétéroscédastique, c'est-à-dire qu'elle ne doit pas être plus élevée pour les valeurs de sortie supérieures et plus faible pour les valeurs de sortie inférieures.
- Les termes d'erreur ne doivent pas être corrélés entre eux.

Une fois que le modèle est construit, il est important d'évaluer ses performances. Le coefficient de détermination R^2 est largement utilisé pour mesurer la précision de la régression linéaire. Il indique dans quelle mesure la variance des variables dépendantes est expliquée par les variables indépendantes, avec une plage de valeurs de 0 à 1. L'erreur quadratique moyenne (RMSE) est également utilisée pour mesurer la dispersion des valeurs prédites par rapport aux valeurs réelles.

Avantages :

- L'un des algorithmes les plus connus en statistiques et en apprentissage automatique.
- Technologie simple, rapide et un bon algorithme de départ.
- Facile à utiliser car il nécessite peu de réglages.

Inconvénients :

- Limité aux relations linéaires.
- Sensible aux valeurs aberrantes.
- Parfois incapable de modéliser des relations complexes en raison de sa simplicité.
- Nécessite des données indépendantes.
- Ne peut prédire que des sorties numériques.

2.3.1.6 L'algorithme des k plus proches voisins

L'algorithme du plus proche voisin (KNN) est une méthode d'apprentissage basée sur les instances, où il n'y a pas de phase d'apprentissage explicite. Au lieu de cela, les données d'entraînement sont simplement stockées en mémoire pour référence ultérieure. Lorsqu'un nouvel exemple doit être classé, il est comparé aux exemples de la base d'entraînement stockés, appelés prototypes, en utilisant une mesure de similarité.

Dans une approche de base de KNN, toutes les variables descriptives d'un exemple sont prises en compte de manière égale lors du calcul de la similarité avec un nouvel exemple. Cependant, dans les bases de données à haute dimension, certaines variables peuvent être moins pertinentes et affecter la précision de la classification. Pour remédier à cela, les variables peuvent être pondérées en fonction de leur pertinence, généralement déterminée par la validation croisée.

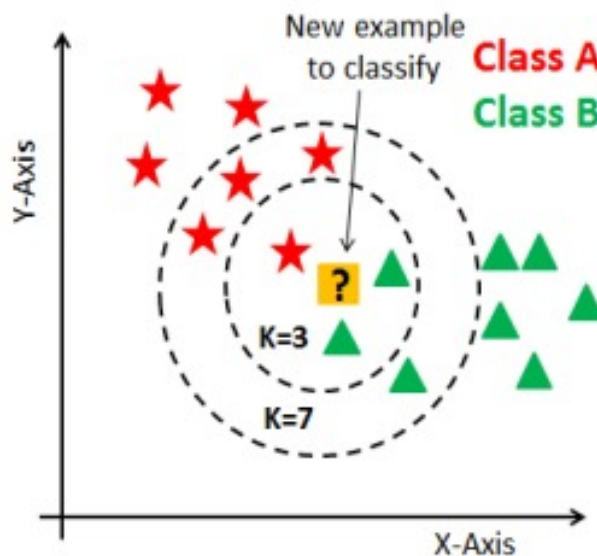


Fig. 2.8 – Exemple de K-plus proche voisins (KPPV).

L'efficacité de KNN dépend de plusieurs facteurs, tels que le nombre de voisins considérés (K), le choix de la métrique de distance et la méthode utilisée pour combiner les réponses des voisins. Un réglage précis de ces paramètres peut considérablement influencer les performances et la précision de l'algorithme.

Concernant la formule de la distance euclidienne entre deux vecteurs x_i et x_j , elle est donnée par :

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^n (x_{ik} - x_{jk})^2}$$

où n est le nombre de dimensions des vecteurs[23][24][25].

2.3.2 L'apprentissage non supervisé

L'apprentissage non supervisé est un type d'apprentissage automatique dans lequel un modèle est formé à travers une base de données non étiquetée et peut reconnaître des modèles sans l'aide d'un superviseur. Les algorithmes d'apprentissage non supervisé peuvent être classés en deux types de problèmes [26] :

Clustering : tels que K-means clustering, K-plus proches voisins et clustering hiérarchique, le rôle du clustering est de diviser automatiquement les données non étiquetées en différents groupes en fonction de la similarité.

Les algorithmes d'association : tels que l'analyse en composantes principales et la décomposition en valeurs singulières sont utilisés pour identifier les corrélations cachées entre différentes variables dans les grandes bases de données.

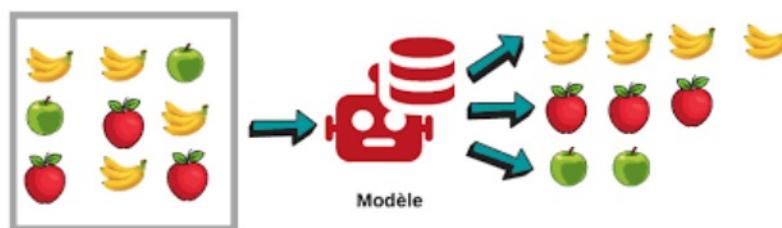


Fig. 2.9 – Schéma d'apprentissage non supervisé "unsupervised learning" [27]

Un médecin pourrait par exemple dans un ensemble assez large de patients de cancers du foie tenter de faire émerger des hypothèses explicatives, l'ordinateur pourrait différencier différents groupes, qu'on pourrait ensuite associer par exemple à leur provenance géographique, génétique, à l'alcoolisme ou à l'exposition à un métal lourd ou à une toxine telle que l'aflatoxine. Dans ce type de problèmes, un algorithme va apprendre les caractéristiques des données d'entraînement (Training Set) pour produire un modèle prédictif. On retrouve deux classes de prédictions : la régression, et la classification. Un algorithme de régression pourra, par exemple,

apprendre à prédire le prix d'une maison en fonction de ses caractéristiques. Par ailleurs, un algorithme de classification nous détectera si une transaction bancaire est frauduleuse ou non. La régression tout comme la classification fait partie de l'apprentissage supervisé (Supervised Learning). La régression permettra de prédire des valeurs continues (prix d'une maison, poids d'une personne...etc.). Quant à la classification, elle permet de déduire une valeur discrète (une classe) comme Spam/non Spam, tumeur maligne/bénigne etc..[28].

2.3.2.1 Le clustering K-moyenne

L'approche des K-moyennes (k-means en anglais), développée par MacQueen en 1967, est un algorithme bien connu d'apprentissage automatique non supervisé utilisé pour le regroupement de données. Son objectif est de diviser une base de données donnée en k groupes ou clusters, où la valeur de k est généralement prédéterminée par l'analyste.

L'idée fondamentale derrière l'algorithme des k-moyennes est de regrouper les éléments en clusters de sorte qu'ils soient très similaires les uns aux autres (présentant une forte similarité intra-classe) et différents des éléments d'autres clusters (présentant une faible similarité inter-classe).

Dans cet algorithme, chaque groupe est représenté par son centroïde, qui est calculé comme la moyenne de tous les points de données assignés à ce groupe. Le centroïde agit comme un point représentatif pour le cluster, résumant ainsi les caractéristiques de ce dernier.

2.3.3 L'apprentissage semi-supervisé (SSL)

L'apprentissage semi-supervisé (SSL) se positionne entre l'apprentissage non supervisé et l'apprentissage supervisé, en combinant une petite quantité de données étiquetées avec une plus grande quantité de données non étiquetées pendant l'entraînement [29]. Son objectif principal est d'utiliser les données non étiquetées (Unlabeled Data) pour améliorer la précision de la fonction de prédiction $f(\theta)$ avec des paramètres entraînaibles (θ), par rapport à ce qui pourrait être obtenu en se basant uniquement sur les données étiquetées (Labeled Data).

En effet, les données non étiquetées peuvent fournir des informations supplémentaires sur la structure de la distribution des données ($p(x)$), ce qui permet de mieux estimer la frontière de décision entre les différentes classes et ainsi améliorer la capacité du modèle de classifier correctement [30][31].



Fig. 2.10 – Schéma d'apprentissage semi supervisé "semi supervised learning" [32]

2.3.4 L'apprentissage par renforcement

L'apprentissage par renforcement représente un autre paradigme fondamental de l'apprentissage automatique. Dans ce type d'apprentissage, un agent interactif observe son environnement, effectue des actions spécifiques, puis reçoit des récompenses ou des pénalités en retour. Contrairement à l'apprentissage supervisé où l'agent dispose de données étiquetées pour guider son apprentissage, dans l'apprentissage par renforcement, l'agent doit apprendre par lui-même à travers des interactions itératives avec son environnement, ce qui est souvent appelé une politique [15].

Dans ce contexte, l'agent doit apprendre à naviguer dans un environnement dynamique en prenant des décisions basées sur ses observations et en recevant des rétroactions sous forme de récompenses ou de pénalités. À chaque étape de cette interaction, l'agent perçoit l'état courant de l'environnement (représenté par p) et génère une action (a) en réponse, qui influence ensuite l'état de l'environnement. La qualité de cette interaction est évaluée par un signal de renforcement ou de récompense (r), qui informe l'agent de la qualité de son action dans cet état particulier.

Le défi principal de l'apprentissage par renforcement est de développer une politique efficace qui permet à l'agent de maximiser la somme des récompenses sur le long terme. Cette politique associe les états possibles de l'environnement aux actions à entreprendre dans chacun de ces états. Formellement, le modèle est représenté par un triplet (S, A, R) , où S est un ensemble fini d'états, A est un ensemble fini d'actions, et R est une fonction de renforcement.

Il est important de noter que l'environnement dans lequel évolue l'agent peut être non déterministe, ce qui signifie que pour une action donnée dans un état donné, l'état résultant ou la récompense associée peuvent varier d'une interaction à l'autre [33].

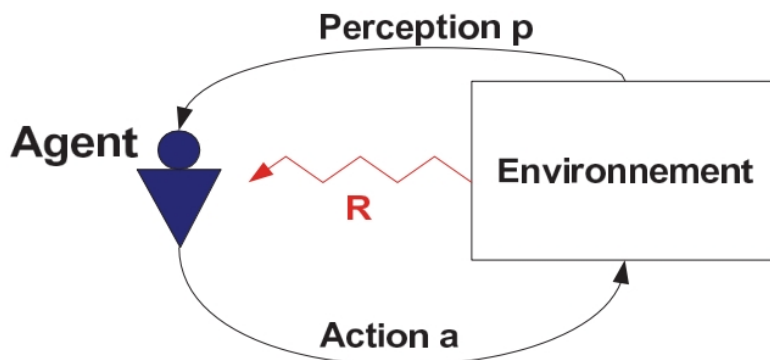


Fig. 2.11 – : Schéma d'apprentissage par renforcement "reinforcement learning".
[33]

2.4 Les réseaux de neurones

2.4.1 Introduction

Les réseaux neuronaux artificiels (RNA) sont une technique d'apprentissage automatique développée depuis les années 1940. Il existe deux types principaux de réseaux neuronaux : les réseaux de neurones artificiels (ANN) et les réseaux de neurones convolutifs (CNN) [34].

Les RNA sont des programmes informatiques capables de détecter des corrélations, qu'elles soient linéaires ou non linéaires, entre les variables d'entrée et de sortie. Ils sont particulièrement efficaces pour résoudre des problèmes complexes qui ne peuvent pas être facilement formulés mathématiquement. Parmi les réseaux de neurones artificiels les plus couramment utilisés, on trouve le perceptron multicouche (MLP). Son objectif principal est de créer un modèle capable d'associer avec précision les données d'entrée aux données de sortie, en utilisant les données précédentes, permettant ainsi de générer des prédictions même lorsque les données de sortie requises sont inconnues.

L'algorithme de rétropropagation est souvent utilisé avec le MLP pour apprendre à partir des données d'entrée. Cette méthode permet au réseau de s'ajuster progressivement afin de mieux correspondre aux données observées. Dans un MLP, l'information est transmise de manière unidirectionnelle, de la couche d'entrée à la couche de sortie, en passant par une ou plusieurs couches cachées. L'efficacité d'un modèle MLP dépend de divers facteurs, notamment le nombre de couches cachées et de nœuds, la taille de l'ensemble de données d'apprentissage, ainsi que des paramètres d'apprentissage tels que le taux d'apprentissage, l'élan et le nombre d'itérations[35].

Les réseaux de neurones artificiels ont été initialement conçus pour modéliser les neurones biologiques et imiter leurs capacités par analogie. Cependant, ces premiers modèles n'étaient pas capables d'apprendre, et il a fallu attendre l'apparition du perceptron pour cela. De plus, plusieurs innovations, telles que la rétropropagation du gradient et les modèles multicouches, couplées au développement des capacités des outils informatiques, ont permis les premières

utilisations pratiques des réseaux de neurones artificiels.

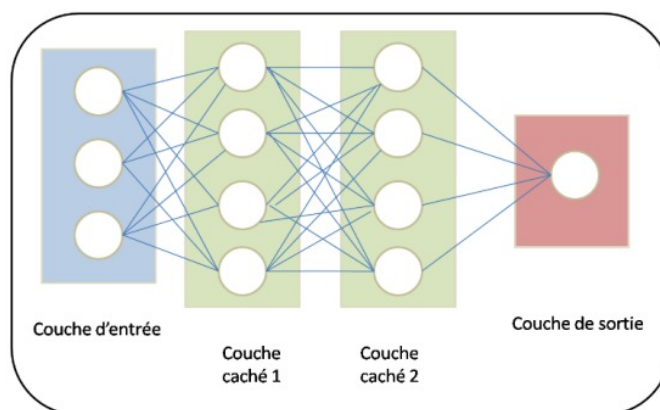


Fig. 2.12 – Architecture d'un MLP à deux couches cachées. [36]

2.4.1.1 Neurone biologique

Un réseau de neurone artificiel est un ensemble interconnecté d'éléments de traitement simple dont la fonctionnalité s'inspire du neurone biologique. Le cerveau humain est constitué d'environ 100 milliards de cellules nerveuse ou neurones (Figure 2-13). Les neurones communiquent par des signaux électriques et les connections entre elles sont assurées par des jonctions électrochimiques appelées synapse, située sur des branches de la cellule appelée dendrites. Chaque neurone reçoit généralement plusieurs milliers de connexion avec d'autres neurones et reçoit ainsi en permanence une multitude de signaux entrants qui finissent par atteindre le corps cellulaire. Si le signal résultant dépasse un certain seuil, le neurone se déclenche ou génère une impulsion de tension en réponse. Celle-ci est ensuite transmise à d'autres neurones par une fibre ramifiée appelée axone [37].

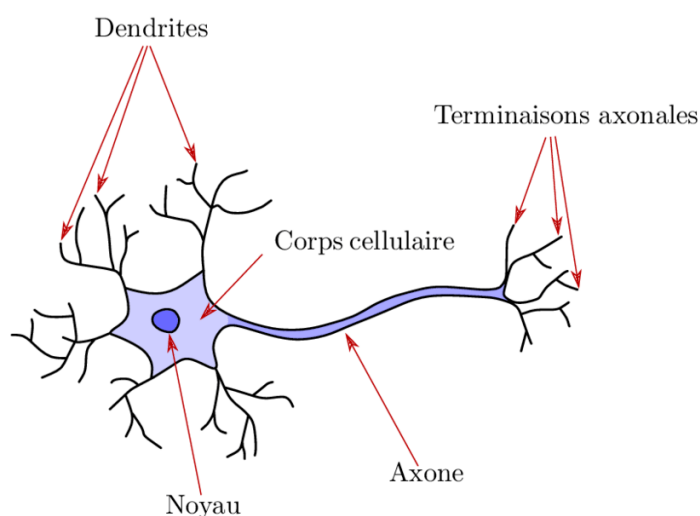


Fig. 2.13 – neurone biologique [38]

2.4.1.2 Neurone Artificiel

Un neurone artificiel est un système ayant plusieurs entrées et une seule sortie. Chaque neurone possède deux modes (le mode d'entraînement et le mode de fonctionnement). Dans le mode d'entraînement, le neurone apprend à être déclenché contre des signaux d'entrée spécifiques ou à se déclencher à un terme. En mode de fonctionnement, une sortie correspondante est fournie lorsque les informations fournis à l'entrée sont reconnus. Si l'entrée ne fait pas partie des entrées pré-identifiées, les règles fixés après la phase d'apprentissage décident de la déclencher ou non. En omettant certaines des propriétés critiques des neurones et leurs interconnexions [39]. Un modèle élémentaire du neurone artificiel typique peut être simulé par un ordinateur.

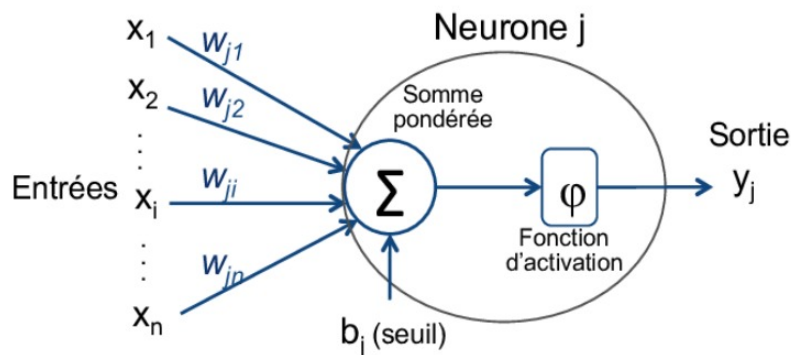


Fig. 2.14 – neurone artificielle.
[40]

Un neurone est composé d'une couche d'entrée acceptant des variables d'entrées $x_i \in \mathbb{R}$, d'un biais b_i , de poids w_i , d'une fonction d'activation σ , et d'une ou plusieurs valeurs de sortie y_i de sortie. On a ainsi la formule suivante :

$$y = \sigma \left(w_0 + \sum_{j=0}^p w_j x_j \right)$$

2.4.1.3 Structure des réseaux de neurones artificiels

Comme le montre la Figure 2-15, un réseau neurone artificiel est constitué des composants de couches et de poids de connexion. Le comportement du réseau dépend également de la communication entre les membres. Il existe généralement trois types de couches neuronales :

1. **La couche d'entrée** : cette couche reçoit les données brutes introduites dans le réseau
2. **Les Couches cachées** : La performance de ces couches est déterminée par les entrées et les poids de la connexion entre elles et les couches cachées. Les poids entre les neurones d'entrée et cachées détermine quand un neurone de cette couche doit être activée.
3. **La couche de sortie** : la sortie de l'unité de sortie dépend de l'activité de la couche cachée et des poids de la connexion entre l'unité cachée et la sortie.

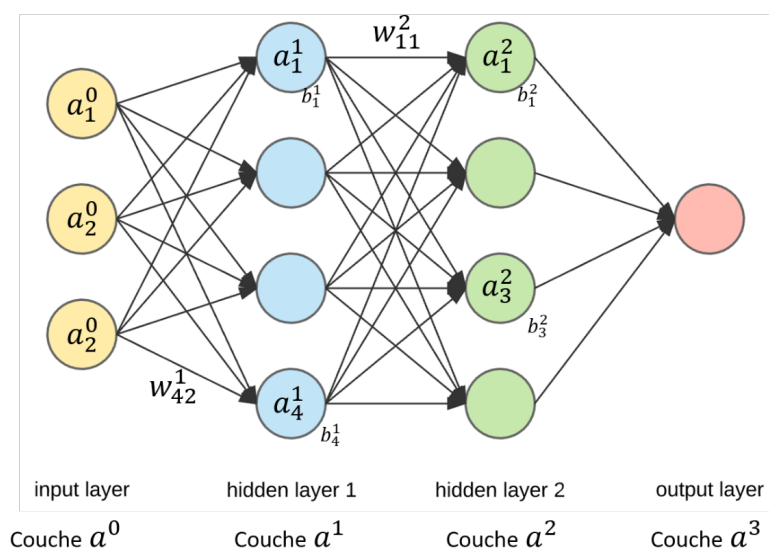


Fig. 2.15 – Structure des réseaux de neurones artificiels [41]

Il existe également des réseaux monocouches et multicouches ; dans l'organisation monocouche tous les neurones sont connectés à une couche, ce type de réseaux neuronaux est le plus utilisé et possède plus de potentiel de calcul que les organisations multicouches. Dans les réseaux multicouches, les neurones sont numérotés par couches (au lieu de suivre la numérotation globale). Les deux couches d'un réseau sont reliées par des poids et en fait des connexions [42]. Dans les réseaux de neurones, il existe plusieurs types de connectivité ou gain de poids :

- **Anticipation** : la plupart des connexions sont des types dans lesquelles les signaux se déplacent dans une seule direction. De l'entrée à la sortie, dans ce cas il n'existe pas de retour (boucle). La sortie de chaque couche n'affecte pas la même couche.
- **En arrière** : les données sont renvoyées des nœuds de la couche supérieure vers la couche inférieure nœuds de couche.
- **Côté** : les sorties des nœuds de chaque couche sont utilisées comme entrées pour les nœuds de la même couche

2.4.1.4 Types des réseaux de neurones artificiels

Voici les types de réseaux de neurones les plus connus (Figure 2-16) :

1. Réseau de neurones Perceptron multicouche (MLPNN).
2. Réseaux de neurones à fonction de base radiale (RBFNN).
3. Réseau de neurones Hopfield (HN).
4. Réseau de neurones de Hamming (HNN).
5. Réseau neuronal cartographique auto-organisé de Kohonen (KSOMNN).
6. Réseau de neurones à retardement (TDNN).
7. Réseaux de neurones à alimentation profonde (DFNN).

8. Réseaux de neurones récurrents (RNN).
9. Réseaux de neurones à mémoire à long et à court terme (NNLSTM).
10. Réseau neuronal des encodeurs automatiques (AENN).
11. Réseaux de chaînes de Markov (MCN).

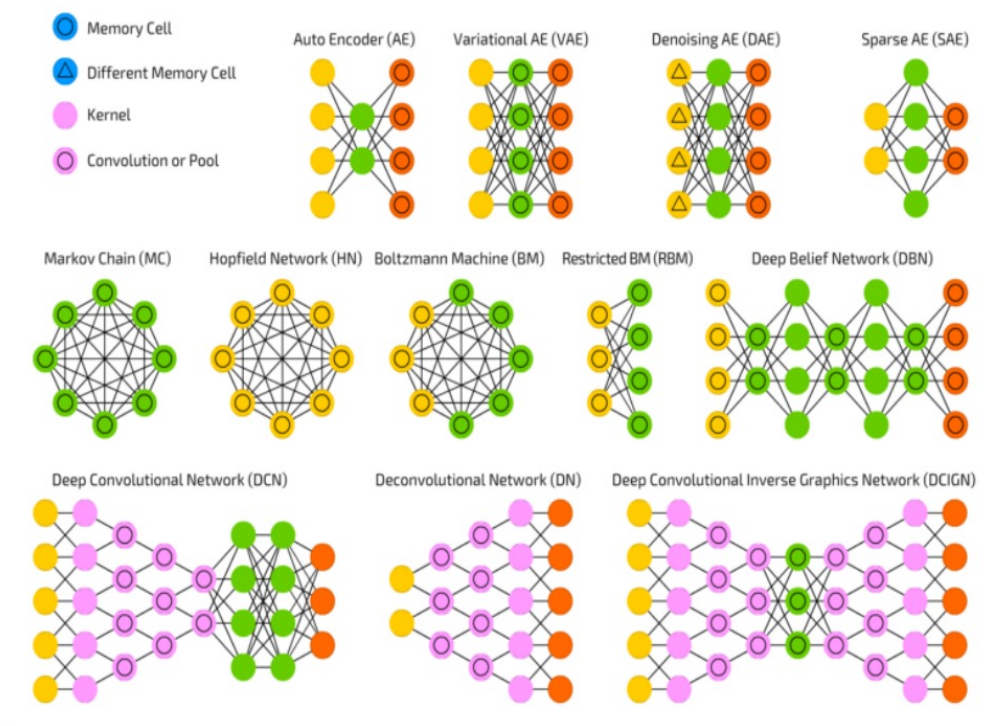


Fig. 2.16 – Types des réseaux de neurones artificiels.

2.4.2 L'initialisation du poids

L'objectif de l'initialisation des poids dans les réseaux de neurones artificiels est d'empêcher les sorties d'activation de couche d'exploser ou disparaître pendant la phase d'apprentissage. L'apprentissage du réseau sans une initialisation de poids peut conduire à une convergence très lente ou à une incapacité à converger [43].

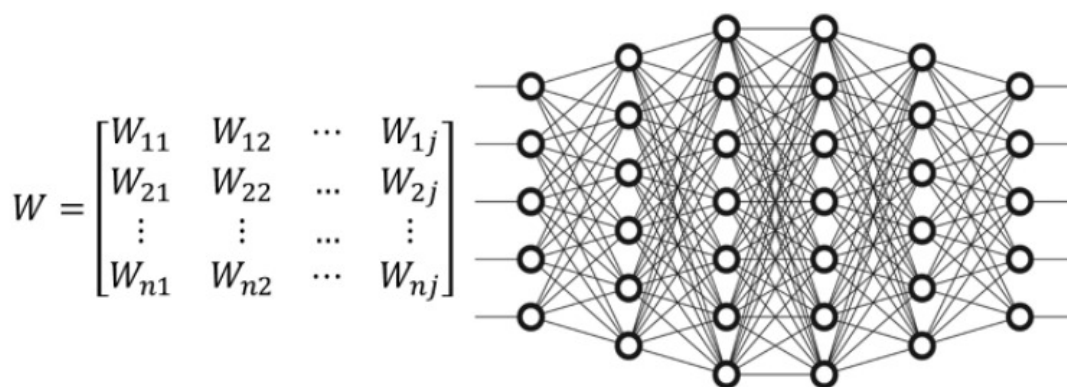


Fig. 2.17 – L'initialisation du poids.
[44]

Les techniques d'initialisation du poids [45] les plus utilisées sont les suivantes :

1. **Initialisation tous zéros et initialisation constante** Cette méthode définit tous les poids sur zéros (respectivement à constante). De plus, toutes les activations dans tous les neurones sont les mêmes, et donc tous les calculs sont les mêmes, ce qui fait ce type d'initialisation concerne les modèles linéaires.
2. **Initialisation aléatoire** Cette technique améliore le processus de brisure de symétrie et offre une plus grande précision. Les poids sont initialisés très près de zéro et aléatoirement. Cette méthode empêche l'apprentissage de la même fonctionnalité pour l'entrée paramètres.
3. **Initialisation LeCun (normaliser la variance)** L'initialisation LeCun vise à empêcher la disparition ou l'explosion des gradients lors de la rétro-propagation en résolvant la variance croissante avec le nombre d'entrées et en fixant une variance constante.
4. **Initialisation de Xavier (initialisation de Glorot)** Xavier a proposé une méthode plus simple, où les poids tels que la variance des activations sont le même sur chaque couche. Cela empêchera le dégradé d'exploser ou disparition.
5. **Initialisation HE (initialisation de Kaiming)** Cette initialisation préserve non linéarité des fonctions d'activation telles que les activations ReLU. L'initialisation des poids avec la méthode HE peut réduire ou amplifier les amplitudes des entrées de manière exponentielle.

Dans le Tableau 2-1, une comparaison entre les techniques d'initialisation de poids est présentée pour décrire les principales techniques d'initialisation de poids en fonction de leurs avantages et inconvénients.

Méthode d'initialisation	Avantages	Inconvénients
Tout-zéros/constante	— Simplicité	— Problème de symétrie amenant les neurones à apprendre la même chose
Aléatoire (Random)	— Améliore le processus de rupture de symétrie	— Une saturation peut se produire conduisant à un gradient de fuite. — La pente ou le gradient est faible, ce qui peut ralentir la descente du gradient
LeCun	— Résout la variance croissante et les problèmes de gradient	— Pas utile dans les réseaux à largeur constante. — Tient compte de la propagation vers l'avant du signal d'entrée. — Cette méthode n'est pas utile lorsque la fonction d'activation est non différentiable.
Xavier	— Diminue la probabilité du problème de disparition/explosion du gradient.	— Cette méthode n'est pas utile lorsque la fonction d'activation est non différentiable. — Problème de neurone mourant pendant l'entraînement.
HE	— Résout les problèmes de neurones mourants.	— Cette méthode est spécialement utile pour les couches avec les fonctions d'activations différentiables telles que ReLU ou LeakyReLU.

Tab. 2.1 – Comparaison entre les techniques d'initialisation du poids.

2.4.3 les fonctions d'activation

La fonction d'activation est introduite dans les CNN pour introduire de la non-linéarité. En pratique, les données ne sont généralement pas classifiables de manière linéaire. Pour les classificateurs basés sur les CNN, les performances diminueraient considérablement si aucune unité d'activation non linéaire n'était introduite dans les classificateurs. La fonction sigmoïde et la fonction tangente hyperbolique sont les deux fonctions d'activation les plus populaires dans les premiers réseaux neuronaux[46].

2.4.3.1 La fonction sigmoïde

peut être exprimée comme suit :

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}}$$

où e représente le nombre naturel. La fonction sigmoïde est introduite en tant que fonction d'activation car elle est lisse et différentiable. Cependant, l'un des principaux inconvénients qu'elle entraîne est la disparition du gradient à mesure que les réseaux deviennent plus profonds. De plus, la sortie de la fonction sigmoïde n'est plus centrée sur zéro et modifie donc la distribution de l'entrée[46].

2.4.3.2 La fonction tangente hyperbolique

, qui atténue dans une certaine mesure la disparition du gradient et recentre la sortie autour de zéro, peut être écrite comme suit :

$$\text{tanh}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

Ces deux fonctions peuvent être observées dans la Figure 2.6. Cependant, les limitations de la fonction tangente hyperbolique telles que le coût de calcul élevé et la disparition du gradient irréversible restreignent l'utilisation de la fonction tangente hyperbolique[46].

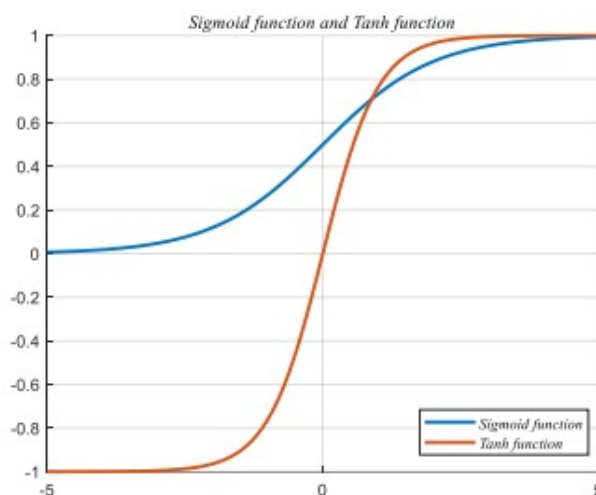


Fig. 2.18 – fonction sigmoïde et tanh
[46]

2.4.3.3 La fonction ReLU

La fonction d'activation Rectified Linear Unit (ReLU) devient la plus couramment utilisée car elle converge rapidement sans présenter de problème de disparition de gradient. Sa définition

peut être donnée de la manière suivante :

$$ReLU(x) = \begin{cases} x, & \text{si } x \geq 0 \\ 0, & \text{si } x < 0 \end{cases}$$

L'utilisation de ReLU comme fonction d'activation élimine le problème de disparition du gradient, permettant ainsi d'entraîner efficacement des réseaux plus profonds. Toutefois, certaines caractéristiques potentiellement utiles peuvent être perdues lorsque la sortie de ReLU est fixée à zéro pour les activations négatives. Pour atténuer cette perte de caractéristiques, de nombreuses variantes de ReLU ont été introduites dans le domaine[46].

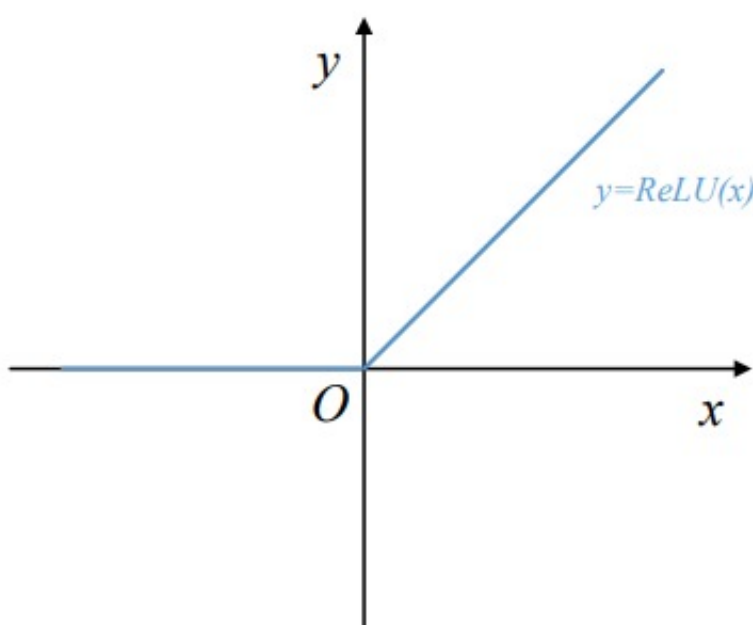


Fig. 2.19 – fonction RELU
[46]

2.4.3.4 La fonction LReLU

La fonction d'activation Leaky ReLU (LReLU) maintient les activations négatives en multipliant les valeurs négatives par une petite constante positive de 0,01, tandis que les valeurs des activations positives restent les mêmes que celles de ReLU[46]. Mathématiquement, LReLU peut être exprimé comme suit :

$$LReLU(x) = \begin{cases} x, & \text{si } x \geq 0 \\ 0.01x, & \text{si } x < 0 \end{cases}$$

2.4.3.5 La fonction PReLU

Pour permettre la flexibilité des pentes pour les activations négatives, PReLU a été proposé. PReLU peut être écrit sous la forme :

$$PReLU(x) = \begin{cases} x, & \text{si } x \geq 0 \\ \alpha x, & \text{si } x < 0 \end{cases}$$

où α' est une petite variable positive déterminée selon les réseaux neuronaux[46].

2.4.3.6 La fonction RReLU

Randomized ReLU (RReLU), la valeur de la pente pour les activations négatives est choisie de manière aléatoire dans une plage prédéfinie. Au lieu de déterminer la pente avant l'entraînement dans PReLU, la pente dans RReLU est sélectionnée dans une plage donnée à chaque époque d'entraînement. La fonction RReLU peut être étendue comme suit :

$$RReLU(x) = \begin{cases} x, & \text{si } x \geq 0 \\ \alpha' x, & \text{si } x < 0 \end{cases}$$

où α' suit une distribution uniforme de c à d , où c, d appartiennent à $[0, 1)$ [46].

2.4.3.7 La fonction ELU

L'unité linéaire exponentielle (ELU), une fonction typique qui permet aux activations négatives d'être proches de zéro, est également largement utilisée comme fonction d'activation. ELU peut être écrit comme suit :

$$ELU(x) = \begin{cases} x, & \text{si } x \geq 0 \\ \beta(e^x - 1), & \text{si } x < 0 \end{cases}$$

où β détermine l'échelle de proximité entre la sortie des activations et zéro. Plus β est grand, plus la distance est grande. Deux fonctions ELU avec différentes valeurs de β sont illustrées dans la Figure 2.20[46].

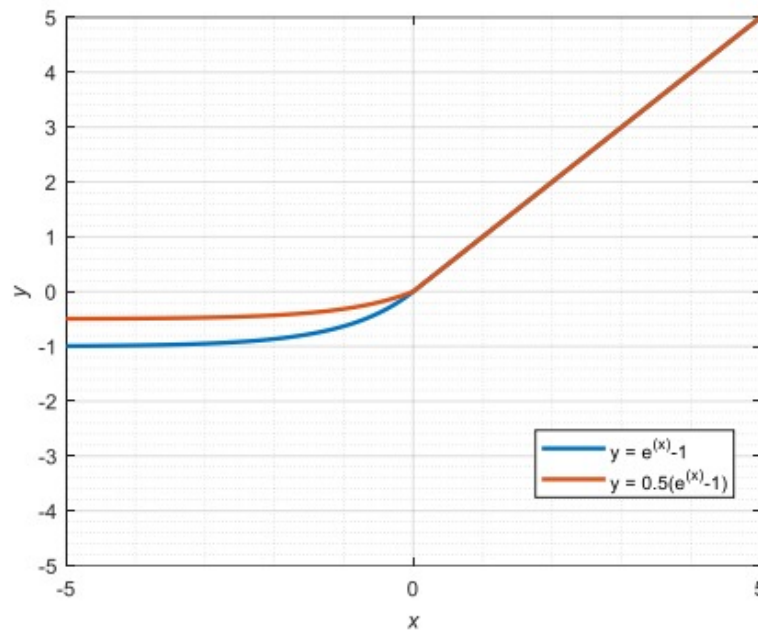


Fig. 2.20 – fonction ELU
[46]

2.4.3.8 La fonction SELU

La fonction SELU est une extension de la fonction ELU avec un paramètre supplémentaire de mise à l'échelle[46], noté γ . Elle peut être définie comme suit :

$$SELU(x) = \begin{cases} \gamma x, & \text{si } x \geq 0 \\ \gamma\beta(e^x - 1), & \text{si } x < 0 \end{cases}$$

2.4.3.9 La fonction Swish

est une fonction d'activation relativement nouvelle qui a été découverte par des chercheurs chez Google. La caractéristique distinctive de la fonction Swish est qu'elle n'est pas monotone, ce qui signifie que la valeur de la fonction peut diminuer même si les valeurs des entrées augmentent. Dans certains cas, la fonction Swish surpasse même la fonction ReLU[12].

Mathématiquement, la fonction Swish est exprimée comme suit :

$$f(x) = x \cdot \text{sigmoid}(x)$$

ou

$$f(x) = \frac{x}{1 + e^{-x}}$$

Cette fonction prend l'entrée x et la multiplie par la fonction sigmoïde de x , ou la divise par $1 + e^{-x}$. Elle est souvent utilisée comme fonction d'activation dans les couches cachées des réseaux de neurones profonds.

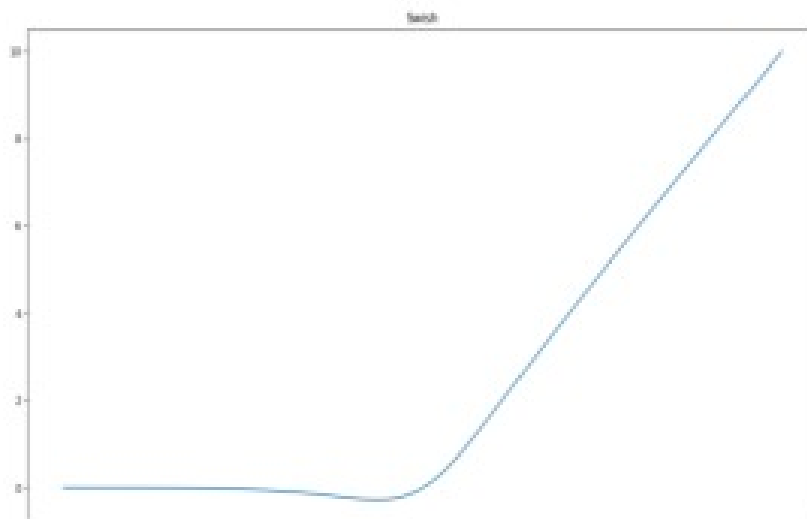


Fig. 2.21 – fonction Switch
[12]

2.4.4 Les fonctions de pertes (LOSS)

La fonction de perte joue un rôle crucial dans l'apprentissage d'un réseau neuronal, en lui fournissant un retour sur la qualité de ses prédictions à chaque étape de l'entraînement. Pour mieux comprendre ce concept, imaginez que vous êtes un étudiant passant plusieurs tests sur le même sujet. À chaque test, vous recevez un score qui vous indique à quel point vous avez bien performé. Si vos scores augmentent d'un test à l'autre, cela indique une amélioration de vos compétences dans le domaine.

De la même manière, un réseau neuronal utilise la fonction de perte pour évaluer ses performances à chaque itération de l'apprentissage. Cette fonction mesure essentiellement la différence entre les prédictions du réseau et les valeurs attendues (ou cibles). Si cette différence est grande, la perte sera élevée, ce qui signifie que le réseau a du mal à produire des résultats précis. En revanche, si la perte diminue au fil de l'entraînement, cela indique que le réseau s'améliore dans sa capacité à effectuer les prédictions souhaitées.

Pour les tâches de régression, où le réseau doit prédire des valeurs continues comme les notes d'un étudiant ou le nombre de produits vendus par un magasin, il existe plusieurs fonctions de perte couramment utilisées. Ces fonctions, telles que l'erreur quadratique moyenne (MSE) ou l'erreur absolue moyenne (MAE), mesurent la différence entre les prédictions du réseau et les valeurs réelles, fournissant ainsi un moyen de quantifier la performance du modèle. En ajustant les paramètres du réseau pour minimiser cette perte, on peut améliorer sa capacité à produire des prédictions précises[47].

2.4.4.1 Erreur Quadratique Moyenne(MSE)

Cette erreur est définie par la différence quadratique moyenne entre la valeur réelle et prédite. La différence au carré permet de pénaliser plus facilement le modèle pour une différence plus élevée. Ainsi, une différence de 3 entraînerait une perte de 9, mais une différence de 9 entraînerait une perte de 81. L'équivalent mathématique est défini par la fonction suivante[47] :

$$\text{MSE} = \frac{1}{K} \sum_{i=1}^K (y_i - \hat{y}_i)^2$$

Où : - K est le nombre total d'exemples dans l'ensemble de données. - y_i est la valeur réelle de l'exemple i . - \hat{y}_i est la prédiction du modèle pour l'exemple i .

2.4.4.2 Erreur Absolue Moyenne

L'erreur absolue moyenne entre le réel et le prévu. L'équivalent mathématique serait :

$$\text{MAE} = \frac{1}{K} \sum_{i=1}^K |y_i - \hat{y}_i|$$

Où : - K est le nombre total d'exemples dans l'ensemble de données.

- y_i est la valeur réelle de l'exemple i .

- \hat{y}_i est la prédiction du modèle pour l'exemple i .

De même, peu d'autres variantes sont :

— MAPE : Erreur absolue moyenne en pourcentage (Mean Absolute Percentage Error).

— MSLE : Erreur logarithmique quadratique moyenne (Mean Square Logarithmic Error)

Pour les résultats catégoriels, la prédiction serait pour une classe, comme si un étudiant réussira (1) ou échouera (0), si le client réussira un achat ou non, que le client soit en défaut de paiement ou non, etc.

Certains cas d'utilisation peuvent avoir plusieurs classes comme résultat, comme classer les types de maladies (type A, B ou C), classer les images en tant que chats, chiens, voitures, chevaux, paysages, etc. Dans de tels cas, les pertes définies ci-dessus ne peuvent être utilisées en raison à des raisons évidentes. Nous aurions besoin de quantifier le résultat de la classe comme probabilité et définir les pertes sur la base des estimations de probabilité comme prédictions[47].

2.4.4.3 Entropie croisée binaire

Définit la perte lorsque résultats catégoriels est une variable binaire, c'est-à-dire avec deux classes possibles : (Réussite / Échec) ou (Oui / Non), La forme mathématique de la fonction de l'entropie croisée binaire est la suivante [47] :

$$\text{Entropie croisée binaire} = -\frac{1}{N} \sum_{i=1}^N (y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i))$$

Où : - N est le nombre total d'exemples dans l'ensemble de données.

- y_i est la véritable étiquette de classe de l'exemple i , qui est soit 0 (pour la classe négative) soit 1 (pour la classe positive).

- \hat{y}_i est la probabilité prédite par le modèle que l'exemple i appartienne à la classe positive.

2.4.4.4 Entropie croisée catégorielle

Définit la perte lorsque les résultats catégoriques sont non binaires, c'est-à-dire le nombre de classes possibles est supérieur à 2, exemple : classes (Oui/Non/Peut-être) ou (Type 1/ Type 2/... Type n), la forme mathématique de l'entropie croisée catégorielle serait [47] :

$$\text{Entropie croisée catégorielle} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{i,c} \log(\hat{y}_{i,c})$$

Où : - N est le nombre total d'exemples dans l'ensemble de données.

- C est le nombre total de classes.

- $y_{i,c}$ est une variable indicatrice qui vaut 1 si l'exemple i appartient à la classe c , et 0 sinon.

- $\hat{y}_{i,c}$ est la probabilité prédite par le modèle pour l'exemple i appartenant à la classe c .

2.4.5 Les Optimiseurs

L'optimisation est vraiment importante pour l'apprentissage automatique et profond. Pour la plupart des modèles d'apprentissage profond, on utilise une technique appelée rétropropagation. C'est un peu comme corriger ses erreurs : on détecte où on s'est trompé dans nos prédictions et on ajuste nos réglages en conséquence. Les algorithmes d'optimisation sont comme des outils qui nous aident à faire ces ajustements pour que notre modèle puisse devenir meilleur à chaque itération. L'objectif est de minimiser la différence entre ce que notre modèle prédit et ce qui est réellement observé, ce qu'on appelle la fonction de perte. En gros, ces algorithmes nous aident à rendre notre modèle plus précis et efficace au fil du temps.[48]

2.4.5.1 La descente de gradient (GD)

Les algorithmes d'optimisation s'appuient sur un concept appelé descente de gradient, qui est crucial dans le processus d'apprentissage. Ces méthodes visent à ajuster de manière itérative les paramètres du réseau pour minimiser les erreurs. Essentiellement, ils calculent la direction dans laquelle les paramètres doivent être modifiés pour réduire l'erreur en fonction de la pente de la fonction objectif par rapport à ces paramètres. Plus précisément, après chaque époque d'entraînement, l'algorithme de descente de gradient calcule le gradient de la fonction objectif par rapport aux paramètres du réseau. Ce gradient indique la direction dans laquelle les paramètres doivent être ajustés pour minimiser l'erreur. Ensuite, les paramètres sont mis à jour dans la direction opposée du gradient, se rapprochant progressivement de la valeur optimale.[49] L'équation générale de la descente de gradient pour la mise à jour des paramètres d'un modèle lors de l'entraînement est la suivante :

$$\theta_{t+1} = \theta_t - \eta \nabla J(\theta_t)$$

Où : - θ_t représente les paramètres du modèle à l'itération t .

- η est le taux d'apprentissage, qui contrôle la taille des pas que nous faisons dans la direction opposée du gradient.

- $J(\theta_t)$ est la fonction de perte (ou la fonction objectif) du modèle par rapport aux paramètres θ_t .

- $\nabla J(\theta_t)$ est le gradient de la fonction de perte par rapport aux paramètres θ_t , indiquant la direction dans laquelle les paramètres doivent être ajustés pour minimiser la perte.

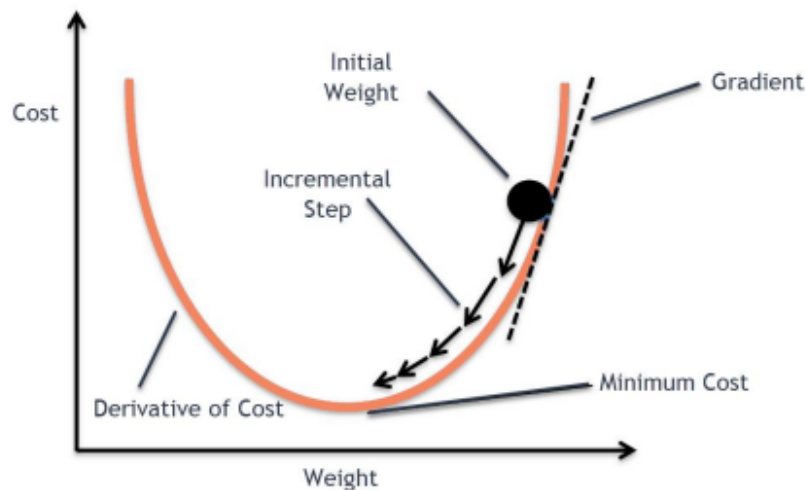


Fig. 2.22 – Algorithme de descente de gradient
[50]

Il existe trois variantes de descente de gradient sont disponibles et couramment utilisées, notamment les suivantes :

1. **La descente du gradient par batch** : Pour la descente de gradient par batch, les mises à jour des paramètres sont effectuées sur un ensemble de données (batch) à la fois. Voici l'équation :

$$\theta_{t+1} = \theta_t - \eta \cdot \frac{1}{B} \sum_{i=1}^B \nabla J(\theta_t; x_i, y_i)$$

Où : - θ_t représente les paramètres du modèle à l'itération t .

- η est le taux d'apprentissage.

- B est la taille du batch, c'est-à-dire le nombre d'exemples dans le batch.

- $\nabla J(\theta_t; x_i, y_i)$ est le gradient de la fonction de perte par rapport aux paramètres θ_t , calculé sur l'exemple (x_i, y_i) du batch.

- La somme est effectuée sur tous les exemples du batch. L'inconvénient de cet algorithme est qu'il prend beaucoup de temps pour les grands bases d'apprentissage car il calcule le

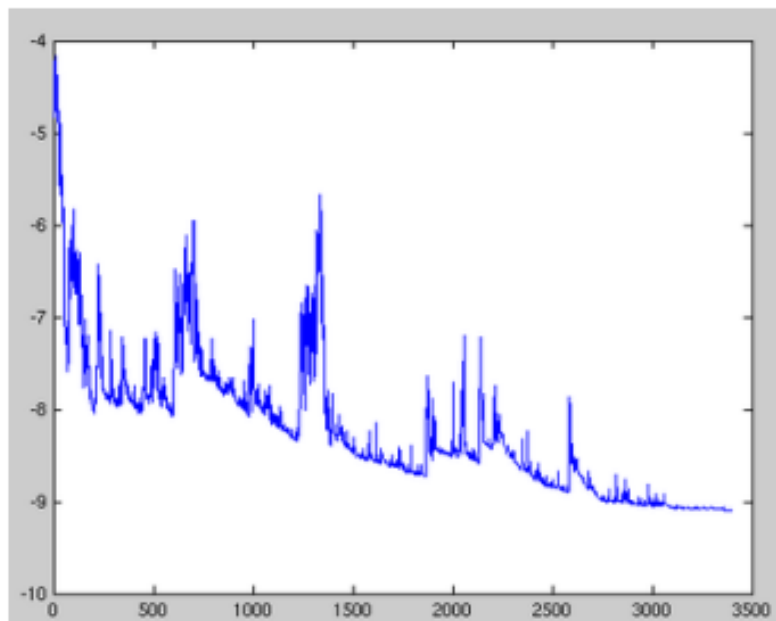


Fig. 2.23 – Descente de gradient stochastique
[51]

gradient de la fonction de perte par rapport aux paramètres pour met à jour les paramètres du modèle.

2. **La descente du gradient stochastique (SGD) :** La descente du gradient stochastique (SGD) est une variante de la descente de gradient où les mises à jour des paramètres sont effectuées sur un seul exemple d'entraînement à la fois, de manière aléatoire.

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla J(\theta_t; x_i, y_i)$$

Où : - θ_t représente les paramètres du modèle à l'itération t .

- η est le taux d'apprentissage.

- $\nabla J(\theta_t; x_i, y_i)$ est le gradient de la fonction de perte par rapport aux paramètres θ_t , calculé sur l'exemple (x_i, y_i) choisi aléatoirement parmi l'ensemble d'entraînement à cette itération.

Contrairement à l'algorithme de descente de gradient par batch, SGD met à jour les paramètres du modèle en calculant le gradient de la fonction de perte pour chaque instance et n'utilise pas l'ensemble d'apprentissage complet.

-SGD est beaucoup plus rapide que la descente de gradient par batch.

-La SGD peut presque certainement converger vers un optimum. et le processus demise à jour peut introduire de grandes fluctuations dans la fonction de perte.

3. **La descente du gradient avec mini-batch :**

La descente du gradient avec mini-batch est une technique qui trouve un équilibre entre la descente du gradient par batch et la descente du gradient stochastique. Au lieu de mettre

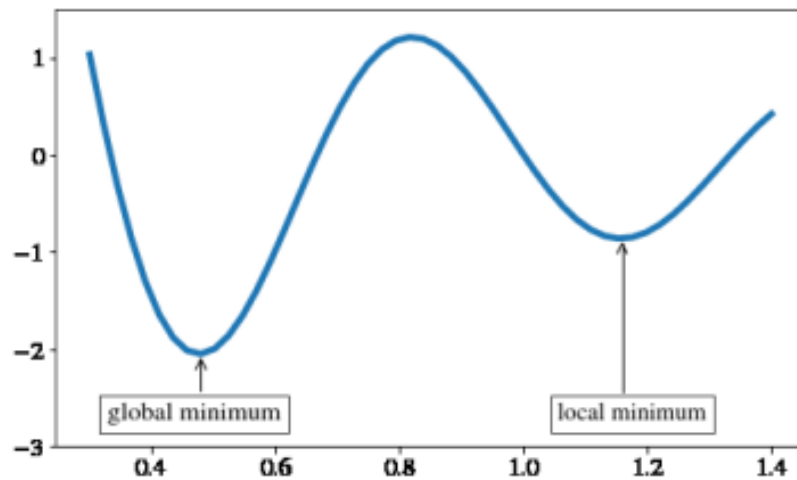


Fig. 2.24 – Minima globaux Vs. Minima locaux
[50]

à jour les paramètres du modèle sur un seul exemple ou sur l'ensemble complet des données d'entraînement, cette méthode utilise des mini-batch, c'est-à-dire de petits sous-ensembles d'exemples d'entraînement.

L'équation de mise à jour des paramètres du modèle avec un mini-batch est donnée par :

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla L(\theta_t; B)$$

Où : - θ_t représente les paramètres du modèle à l'itération t .

- η est le taux d'apprentissage.
- $\nabla L(\theta_t; B)$ est le gradient de la fonction de perte par rapport aux paramètres θ_t , calculé sur le mini-batch B sélectionné à partir de l'ensemble d'entraînement.
- La descente du gradient avec mini-batch est plus efficace que la descente par batch pour les ensembles de données volumineux.
- Elle offre une convergence plus stable que la descente stochastique.
- Cependant, choisir le bon taux d'apprentissage est crucial : un taux trop élevé peut causer une divergence, tandis qu'un taux trop bas peut ralentir la convergence.
- Il est souvent nécessaire d'ajuster le taux d'apprentissage au fur et à mesure de l'apprentissage pour obtenir de bons résultats.
- Les différents paramètres du modèle peuvent nécessiter des taux d'apprentissage différents en raison de leurs valeurs initiales ou de leur impact sur la fonction de perte.
- Il est important d'éviter les points de selle, où le gradient est nul dans toutes les dimensions, pour empêcher un arrêt prématuré de l'apprentissage.

Pour résoudre les problèmes précédents, plusieurs algorithmes sont proposés :

2.4.5.2 Momentum

Le moment (ou momentum) est une technique utilisée pour accélérer la convergence lors de la mise à jour des paramètres d'un modèle. Il met à jour les paramètres selon l'équation suivante[52] :

$$\theta_{\tau} = \theta_{\tau-1} - \eta \cdot \Delta v_{\tau}$$

Où : - θ_{τ} représente les paramètres du modèle à l'itération τ .

- η est le taux d'apprentissage.

- Δv_{τ} est le terme de moment, qui enregistre les gradients historiques jusqu'à l'itération τ .

Le terme de moment est calculé comme suit :

$$\Delta v_{\tau} = \rho \cdot \Delta v_{\tau-1} - (1 - \rho) \cdot \nabla_{\theta} L(\theta_{\tau})$$

Où ρ est un paramètre qui contrôle le poids du terme de moment et $L(\theta_{\tau})$ est la fonction de perte à l'itération τ .

En utilisant le moment, les mises à jour des paramètres tiennent compte des gradients historiques, ce qui peut aider à accélérer la convergence et à éviter les oscillations indésirables.

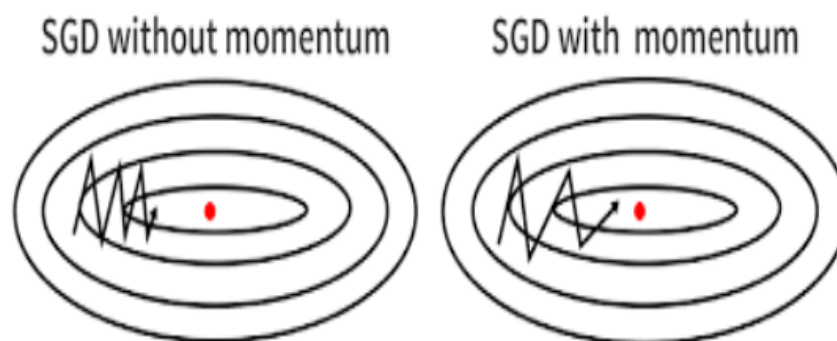


Fig. 2.25 – La comparaison des algorithmes SGD avec et sans quantité de mouvement [50]

2.4.5.3 Nesterov Momentum

Nesterov Momentum améliore la méthode de l'élan en ajustant les positions potentielles plutôt que les paramètres actuels. Contrairement à l'élan, il ne prédit pas directement l'emplacement futur du paramètre. Inspiré par la méthode du gradient accéléré de Nesterov, cette approche calcule le gradient en fonction des positions potentielles estimées. La différence fondamentale avec la méthode de l'élan est que Nesterov ajoute déjà la vitesse aux paramètres lors du calcul du gradient. Cela peut être interprété comme une tentative de correction de la forme traditionnelle de la quantité de mouvement.[53]

2.4.5.4 Le gradient accéléré de Nesterov (NAG)

Le gradient accéléré de Nesterov (NAG) vise à améliorer les algorithmes de descente de gradient en prenant en compte les gradients futurs lors de la mise à jour des paramètres. Contrairement aux approches traditionnelles qui se basent uniquement sur les gradients du point courant ou du passé, le NAG est plus futuriste, anticipant les points futurs approximatifs. Cela permet d'ajuster les paramètres de manière plus précise et efficace.

L'équation de mise à jour des paramètres avec Nesterov est la suivante :

$$\theta_\tau = \theta_{\tau-1} - \eta \cdot \Delta v_\tau$$

Où : - θ_τ représente les paramètres du modèle à l'itération τ .

- η est le taux d'apprentissage.

- Δv_τ est le terme de moment utilisé pour anticiper le gradient futur.

Ce terme de moment est calculé comme suit :

$$\Delta v_\tau = \rho \cdot \Delta v_{\tau-1} - \eta \cdot \nabla_{\theta} L(\theta_\tau - \rho \cdot \Delta v_{\tau-1})$$

Où ρ est un paramètre qui contrôle le poids du terme de moment et $L(\theta_\tau)$ est la fonction de perte à l'itération τ .

L'algorithme NAG est capable de mettre à jour les paramètres du modèle de manière plus efficace en prenant en compte les informations futures, ce qui améliore souvent la convergence et la stabilité du processus d'apprentissage.

2.4.5.5 Adagrad

Adagrad est un algorithme d'optimisation adaptatif qui ajuste automatiquement le taux d'apprentissage pour chaque paramètre du modèle en fonction de l'historique des gradients calculés. Contrairement aux méthodes traditionnelles avec un taux d'apprentissage fixe pour tous les paramètres, Adagrad utilise une approche adaptative qui peut accélérer la convergence pour les paramètres qui sont loin de l'optimum et ralentir pour ceux qui s'en approchent.

L'équation de mise à jour des paramètres avec Adagrad est la suivante :

$$\theta_\tau = \theta_{\tau-1} - \frac{\eta}{\sqrt{G_\tau + \varepsilon}} \odot g_\tau$$

Où : - θ_τ représente les paramètres du modèle à l'itération τ .

- η est le taux d'apprentissage initial.

- g_τ est le gradient de la fonction de perte par rapport aux paramètres θ_τ à l'itération τ .

- G_τ est une matrice diagonale qui contient les éléments diagonaux de la somme des carrés des gradients historiques jusqu'à l'itération τ .

- ε est un terme de régularisation pour éviter la division par zéro.

Dans Adagrad, le taux d'apprentissage s'ajuste automatiquement pour chaque paramètre en fonction de l'historique des gradients. Cela permet un apprentissage efficace avec un taux

d'apprentissage adaptatif, sans nécessiter de réglage manuel. Cependant, à mesure que l'entraînement progresse, les taux d'apprentissage diminuent, ce qui peut finalement entraîner un arrêt prématuré de la mise à jour des paramètres avec les données d'apprentissage[54].

2.4.5.6 RMSprop

RMSprop a été développé pour résoudre un problème rencontré par Adagrad, celui du taux d'apprentissage décroissant. RMSprop ajuste correctement le poids des gradients accumulés historiquement dans la matrice G introduite dans Adagrad, tout en permettant également d'adapter le taux d'apprentissage pendant la mise à jour des paramètres.

Voici les équations de mise à jour des paramètres dans RMSprop :

$$\theta_{\tau} = \theta_{\tau-1} - \frac{\eta}{\sqrt{G_{\tau} + \varepsilon}} \odot g_{\tau}$$

Où θ_{τ} représente les paramètres du modèle à l'itération τ , η est le taux d'apprentissage initial, g_{τ} est le gradient de la fonction de perte par rapport aux paramètres à l'itération τ , et G_{τ} est une matrice diagonale contenant les éléments diagonaux de la somme des carrés des gradients historiques jusqu'à l'itération τ . ε est un terme de régularisation pour éviter la division par zéro.

Le paramètre ρ désigne le poids des gradients historiquement accumulés, généralement fixé à 0,9. Dans cette formulation, le dénominateur est calculé par $\text{RMS}(\sqrt{G})$. Ainsi, l'équation de mise à jour des paramètres peut être généralement réécrite comme suit :

$$\theta_{\tau} = \theta_{\tau-1} - \frac{\eta}{\text{RMS}(\sqrt{G})} \odot g_{\tau}$$

Cette formulation permet à RMSprop d'ajuster le taux d'apprentissage de manière adaptative tout en prenant en compte l'historique des gradients, ce qui peut améliorer l'efficacité de l'optimisation des modèles d'apprentissage automatique[55].

2.4.5.7 Adadelta

Adadelta est une méthode d'optimisation qui se rapproche du dénominateur de RMSprop en tant que RMS du gradient précédent. Elle approxime également le numérateur dans la formule et remplace $\Delta\theta$ par $\text{RMS}(\Delta\theta)$.

Contrairement à Adagrad, Adadelta n'utilise pas de taux d'apprentissage dans son équation de mise à jour, ce qui permet de corriger efficacement deux faiblesses précédemment introduites[56]. Voici l'équation de mise à jour des paramètres dans Adadelta :

$$\theta_{\tau} = \theta_{\tau-1} - \frac{\text{RMS}(\Delta\theta_{\tau-1})}{\text{RMS}(g_{\tau})} \odot g_{\tau}$$

Où $\Delta\theta$ conserve les enregistrements des mises à jour précédentes jusqu'à l'itération précédente $\tau - 1$.

2.4.5.8 Adam

Adam est un algorithme d'optimisation qui combine les concepts de RMSprop et de Momentum. Il maintient une moyenne mobile du carré des gradients du premier ordre, ce qui décroît exponentiellement au fil de l'apprentissage[57].

Les équations de mise à jour des paramètres dans Adam sont les suivantes :

$$m_\tau = \beta_1 m_{\tau-1} + (1 - \beta_1) g_\tau$$

$$v_\tau = \beta_2 v_{\tau-1} + (1 - \beta_2) g_\tau^2$$

Où m_τ et v_τ sont des vecteurs stockant respectivement les moyennes mobiles des gradients de premier ordre et des gradients de premier ordre au carré.

Pour résoudre le problème de biais lorsque β_1 et β_2 sont proches de 1, Adam propose de redimensionner les termes comme suit :

$$\hat{m}_\tau = \frac{m_\tau}{1 - \beta_1^\tau}$$

$$\hat{v}_\tau = \frac{v_\tau}{1 - \beta_2^\tau}$$

En utilisant ces vecteurs redimensionnés et la matrice de redimensionnement, Adam met à jour les paramètres du modèle avec l'équation suivante :

$$\theta_\tau = \theta_{\tau-1} - \frac{\eta}{\sqrt{\hat{v}_\tau} + \varepsilon} \odot \hat{m}_\tau$$

Cette méthode permet une convergence plus rapide et un taux d'apprentissage adaptatif.

2.4.5.9 AdaMax

AdaMax est une variante de la descente de gradient stochastique (SGD) dotée d'un taux d'apprentissage adaptatif, qui est une version de Adam basée sur la norme infinie. Son avantage majeur réside dans sa faible sensibilité aux choix des hyperparamètres par rapport au SGD. L'équation d'AdaMax utilise la valeur maximale de la deuxième composante de la quantité de mouvement estimée par la méthode Adam, ce qui garantit une solution plus stable[58].

2.4.5.10 Nadam

Nadam est un algorithme d'optimisation qui combine les aspects de l'algorithme Adam et du gradient accéléré de Nesterov. Cette intégration permet une mise à jour adaptative des paramètres du modèle[59].

L'équation de mise à jour des paramètres dans Nadam est donnée par :

$$\theta_\tau = \theta_{\tau-1} - \frac{\eta}{\sqrt{v_\tau} + \varepsilon} \left(\beta m_\tau + \frac{(1 - \beta) g_\tau}{1 - \beta^\tau} \right)$$

Où : - θ_τ représente les paramètres du modèle à l'itération τ . - η est le taux d'apprentissage initial. - g_τ est le gradient de la fonction de perte par rapport aux paramètres θ_τ à l'itération τ . - m_τ et v_τ sont des vecteurs qui stockent respectivement les moyennes mobiles des gradients de premier ordre et des gradients de premier ordre au carré. - β est le coefficient de décroissance de la moyenne mobile. - ϵ est un terme de régularisation pour éviter la division par zéro.

Cette équation combine les aspects adaptatifs de l'algorithme Adam avec la correction de l'impulsion introduite par le gradient accéléré de Nesterov, permettant ainsi une convergence plus rapide et plus stable.

2.5 Apprentissage profond (Deep learning ou DL)

L'apprentissage en profondeur (DL) est une technique d'apprentissage très prometteuse utilisée dans divers domaines. Les architectures basées sur le DL se distinguent par leur grande profondeur, caractérisée par de nombreuses couches de neurones cachées. Cependant, le principal défi de ces architectures est le temps considérable nécessaire à l'apprentissage. L'objectif ambitieux de la communauté de recherche en DL est d'atteindre une précision exceptionnelle tout en conservant des temps d'apprentissage raisonnables.

2.5.1 Les réseaux de neurones convolutifs (CNN)

Les réseaux de neurones convolutifs (CNN) sont une technique d'apprentissage profond composée de plusieurs couches, chacune représentant une carte de caractéristiques. En imagerie, ces cartes sont des matrices 3D d'intensités de pixels pour différents canaux de couleur. Chaque couche interne représente une image induite à plusieurs canaux, où chaque "pixel" est une caractéristique spécifique. Les neurones sont connectés à une petite partie des neurones adjacents de la couche précédente, et des transformations telles que la convolution et la mise en commun peuvent être effectuées sur les cartes.

Dans la phase de convolution, une opération convolue une matrice de filtre (poids appris) avec les valeurs d'un champ récepteur de neurones, suivie d'une fonction non linéaire comme la sigmoïde ou ReLU pour obtenir les réponses. Ensuite, l'opération de mise en commun résume les réponses d'un champ récepteur en une seule valeur, produisant des descriptions de fonctionnalités plus robustes.[60]

Une hiérarchie initiale des caractéristiques est construite, pouvant être affinée de manière supervisée en ajoutant une couche de neurones entièrement connectés pour s'adapter à différentes tâches. Selon la tâche, une couche finale avec différentes fonctions d'activation est ajoutée pour obtenir une probabilité spécifique pour chaque neurone de sortie. Le réseau est ensuite optimisé sur une fonction objective via la descente de gradient stochastique.

Un exemple typique de CNN est le VGG16, composé de 13 couches convolutionnelles, 3 couches entièrement connectées, 3 couches de max-pooling et une couche finale de classification softmax. Les filtres de convolution sont produits par des fenêtres de filtre 3x3, et les

résolutions des cartes sont réduites avec 2 couches de max-pooling. Pour tester le modèle, une image de la même taille que les échantillons d'apprentissage peut être traitée, avec éventuellement une opération de remise à l'échelle ou de recadrage si différentes tailles sont fournies[61].

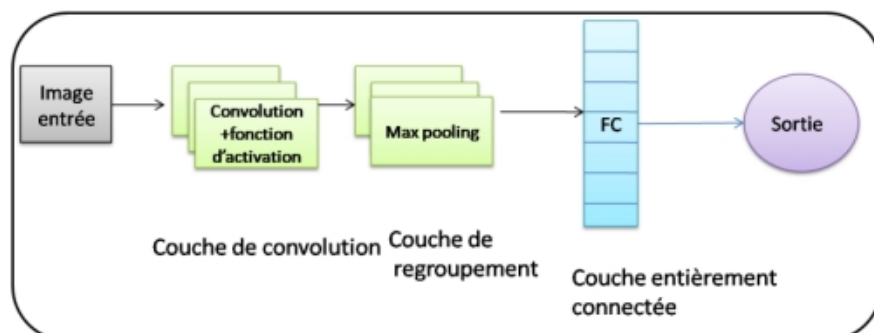


Fig. 2.26 – Une architecture typique d'un CNN
[49]

2.5.2 Les avantages CNN

Les avantages des réseaux de neurones convolutifs (CNN) par rapport aux méthodes de classification traditionnelles sont les suivants :

1. **Représentation hiérarchique des caractéristiques :** Les CNN apprennent une représentation hiérarchique des caractéristiques, allant des pixels aux caractéristiques sémantiques de haut niveau, à travers une structure hiérarchique à plusieurs étapes[62].
2. **Optimisation conjointe de plusieurs tâches :** L'architecture des CNN permet d'optimiser simultanément plusieurs tâches connexes. Par exemple, des modèles comme Fast RCNN combinent la classification et la régression de la boîte englobante dans un mode d'apprentissage multitâches.
3. **Capacité d'apprentissage élevée :** Les CNN bénéficient d'une grande capacité d'apprentissage, ce qui permet de reformuler certains défis classiques de la vision par ordinateur en problèmes de transformation de données de grande dimension et de les résoudre d'un point de vue différent.

En raison de ces avantages, les réseaux de neurones convolutifs sont largement utilisés dans de nombreux domaines de recherche, notamment la super-résolution d'images, la reconstruction d'images, la classification d'images, la récupération d'images, la reconnaissance faciale, la détection et l'analyse vidéo[63].

2.5.3 Les inconvénients de CNN

Malgré les progrès significatifs par rapport aux méthodes traditionnelles et l'importance des réseaux de neurones convolutifs (CNN) dans la détection pratique d'objets, il existe encore quelques inconvénients :

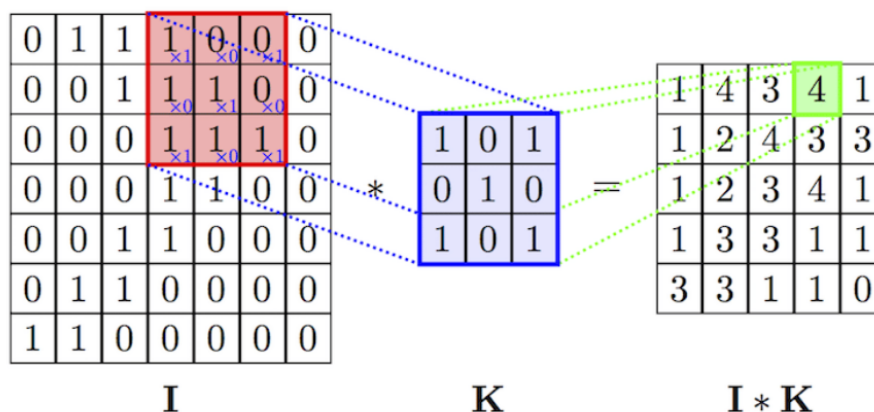


Fig. 2.27 – Exemple d’une opération de convolution [65]

1. **Contrainte de taille de l’image d’entrée :** En raison de l’existence de couches entièrement connectées, les CNN nécessitent une image d’entrée de taille fixe, ce qui peut entraîner le recalcul du modèle pour chaque région d’intérêt, prolongeant ainsi considérablement le temps de test.
2. **Complexité de l’apprentissage :** L’apprentissage du R-CNN (Region-based Convolutional Neural Network) se fait en plusieurs étapes, ce qui le rend coûteux en termes de temps et d’espace. Il implique le raffinement du réseau ConvNet sur les propositions d’objets, le remplacement du classifieur softmax par des SVM (Support Vector Machines) pour s’adapter aux caractéristiques de ConvNet, et enfin l’entraînement des régresseurs de la boîte englobante. Cette procédure est chronophage et nécessite beaucoup de stockage pour les caractéristiques extraites.
3. **Redondance des propositions de région :** Bien que la recherche sélective puisse générer des propositions de région avec des rappels relativement élevés, ces propositions de région peuvent encore être redondantes, et ce processus peut prendre beaucoup de temps. Malgré ces inconvénients, les réseaux de neurones convolutifs continuent d’être largement utilisés et étudiés en raison de leurs performances exceptionnelles dans de nombreux domaines d’application.

2.5.4 L’opération de convolution

L’opération de convolution est comparable aux filtres de convolution que l’on trouve dans le traitement d’images, comme les filtres de Sobel, de Roberts et gaussien. Le noyau du filtre parcourt l’image, multipliant ses poids avec chaque pixel correspondant à travers ce filtre, puis ajoutant le biais au résultat[64].

Dans la couche de convolution, trois hyperparamètres sont utilisés pour produire la carte des caractéristiques :

1. **Pas (Stride S) :** Cela représente le déplacement à chaque fois que le filtre de convolution

est glissé. Un pas de 1 signifie que les filtres se déplacent d'un pixel à chaque fois. Avec un pas de 2 (ou plus), les filtres sautent 2 pixels à la fois pendant le glissement [66].

2. **Rembourrage (Padding P)** : Les images d'entrée peuvent être remplies avec une bordure de taille spécifique en utilisant le rembourrage. Le rembourrage zéro (zéro-padding) est le plus couramment utilisé. La taille du rembourrage est un hyperparamètre contrôlant la taille spatiale des volumes de sortie dans les frameworks de réseaux de neurones[67].

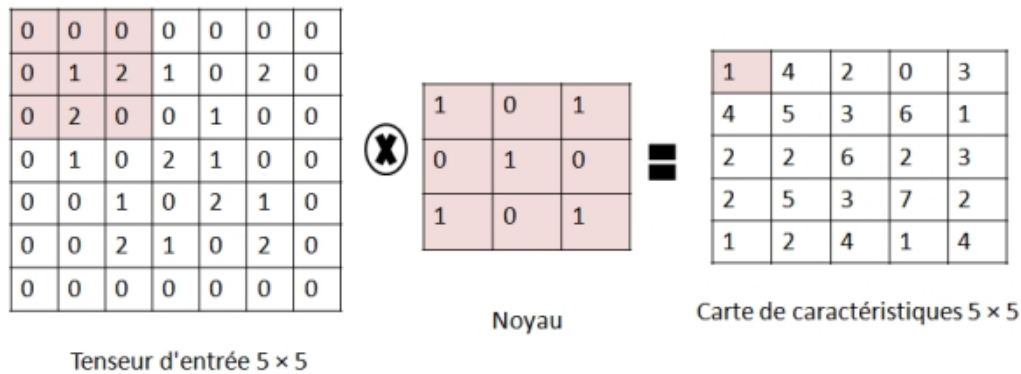


Fig. 2.28 – Exemple d'opération de convolution avec padding, un tenseur d'entrée de taille 5×5, noyau de 3×3 et un stride de 1.

[68]

3. **Profondeur (Depth D)** : C'est le nombre de filtres utilisés dans une couche de convolution, correspondant à la profondeur du volume de sortie. Ce paramètre est également un hyperparamètre et est déterminé par le nombre de filtres que nous choisissons d'appliquer[69].

2.5.5 Mise en commun (Pooling)

De nos jours, les réseaux de neurones à convolution utilisent toujours un partage de poids étendu pour réduire le nombre de paramètres des modèles. Une couche de mise en commun (pooling) est souvent utilisée pour réduire le temps de calcul et construire progressivement une invariance spatiale et configurative dans le processus d'apprentissage.

La couche de mise en commun aide à extraire davantage de sens sémantique des images. Par exemple, la mise en commun maximale (Max-pooling) renvoie simplement la valeur maximale des valeurs sur lesquelles l'opération du noyau est appliquée. Voici un exemple pour illustrer les sorties d'une mise en commun maximale (max-pooling) et d'une mise en commun moyenne (average pooling), respectivement, avec un noyau de taille 2 et un pas de 2[70].

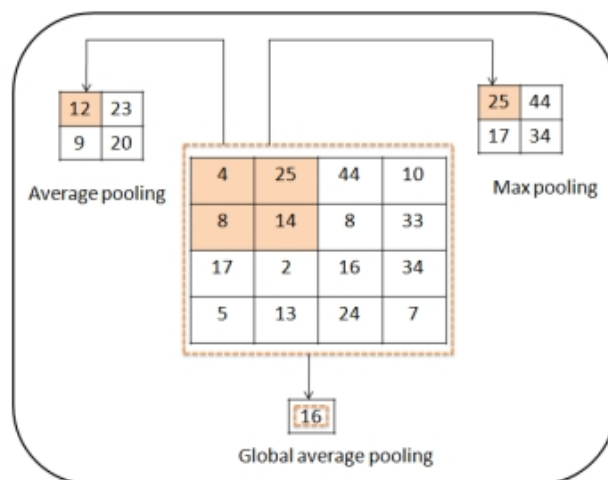


Fig. 2.29 – Types d’opération de pooling .
[49]

2.5.6 Aplanissement (Flattening)

Pour exploiter pleinement les caractéristiques de haut niveau extraites par la couche de convolution, une couche entièrement connectée est ajoutée. Cette couche est cruciale car elle permet de combiner de manière non linéaire les caractéristiques de haut niveau représentées par la sortie de la couche de convolution.

En aplatissant les images en un vecteur (colonne), nous les convertissons dans une forme appropriée pour notre Perceptron multi-couches. Cette sortie aplatie est ensuite alimentée à un réseau de neurones feed-forward, et la rétropropagation est appliquée à chaque itération du processus d’apprentissage. Après plusieurs itérations, le modèle est capable de distinguer les caractéristiques dominantes et certaines caractéristiques de bas niveau dans les images, ce qui permet de les classer à l’aide de la technique de classification des réseaux de neurones[71].

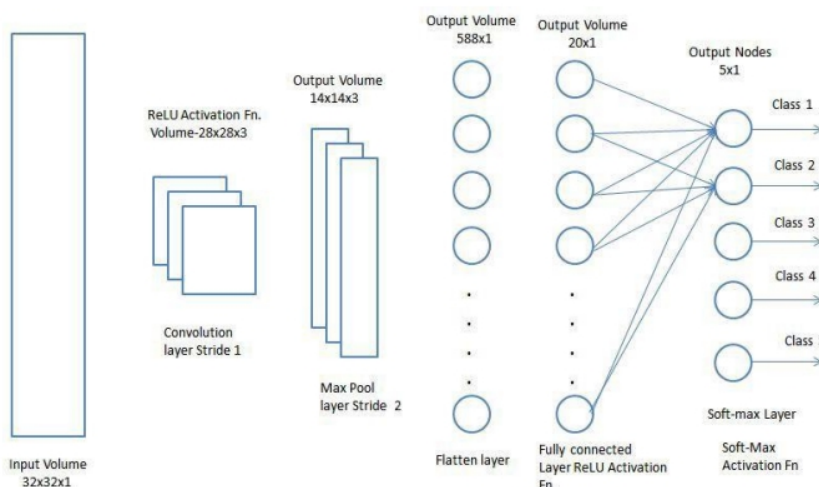


Fig. 2.30 – schéma d’un réseau à convolution avec la couche Flattening

2.5.7 Quelques Architecture basées CNN

2.5.7.1 DenseNet

DenseNet (Densely Connected Convolutional Network) est une architecture CNN innovante lancée en 2016. Ses principales caractéristiques sont :

- Blocs denses : chaque bloc dense connecte chaque couche à toutes les autres couches du bloc, permettant une meilleure réutilisation des fonctionnalités.
- Couche de transition : entre les blocs denses, il existe une couche de transition qui peut réduire la taille de la carte des caractéristiques tout en augmentant le nombre de canaux.
- Couche de goulot d'étranglement : au sein de chaque bloc dense, la couche de goulot d'étranglement réduit le nombre de canaux avant la convolution 3x3, réduisant ainsi le coût de calcul.

Les principaux avantages de DenseNet sont une réutilisation efficace des fonctionnalités, un nombre réduit de paramètres et un flux de gradient amélioré. [72]

2.5.7.2 VGGNet

VGGNet est une architecture CNN proposée en 2014, caractérisée par sa simplicité et ses convolutions 3x3 empilées. Ses principales versions sont le VGG16 et le VGG19, qui comportent respectivement 16 et 19 couches de poids. [73]

2.5.7.3 Xception

Xception est une extension de l'architecture Inception qui remplace le module Inception standard par des convolutions séparables en profondeur. Par rapport à Inception, Xception est plus efficace et réduit la complexité informatique. [73]

2.5.7.4 MobileNet

MobileNetV3-Small est une architecture CNN efficace avec convolution inverse, goulot d'étranglement linéaire et convolution en profondeur et en points. Il convient aux environnements aux ressources limitées. [74]

2.6 L'apprentissage par transfert

L'apprentissage par transfert est une méthode d'apprentissage automatique qui implique le raffinement d'un modèle préalablement entraîné sur une tâche ou un domaine spécifique, connu sous le nom de domaine source, pour améliorer ses performances sur une tâche ou un domaine différent mais lié, appelé domaine cible, comme illustré dans la Figure II-21. L'objectif principal de cette approche est de minimiser la nécessité de disposer d'une grande quantité de données étiquetées pour l'apprentissage, tout en favorisant une meilleure généralisation du modèle.

Pour y parvenir, l'idée est d'exploiter les connaissances et les représentations apprises dans le domaine source et de les appliquer au domaine cible. Cependant, pour que l'apprentissage par

transfert soit efficace, il est essentiel que les deux domaines soient liés d'une manière ou d'une autre. Plus les similitudes entre les deux domaines sont importantes, plus la probabilité d'un transfert de connaissances réussi est élevée. Néanmoins, même lorsque les domaines sont différents, il est possible que des informations pertinentes puissent être transférées pour améliorer les performances du modèle dans le domaine cible.[75]

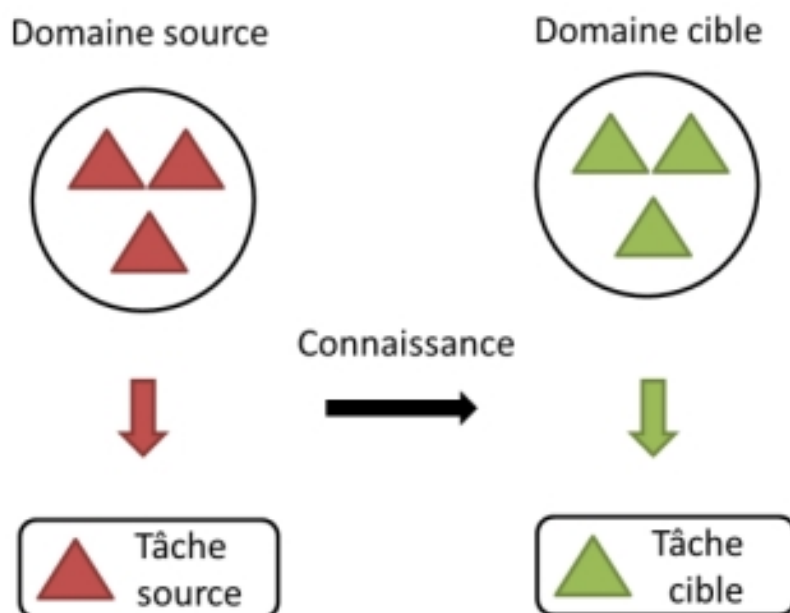


Fig. 2.31 – Apprentissage par transfert [76]

2.7 Domaines d'utilisation de l'apprentissage automatique

Ces dernières années, l'apprentissage automatique a été utilisé pour développer des applications dans divers domaines, comme le soulignent Ahuja et al [77], parmi les domaines notables, on peut citer :

Reconnaissance des objets : Les techniques d'apprentissage automatique sont utilisées pour distinguer et reconnaître des objets sous diverses formes, telles que des visages, des images, des lettres, des mots, des textes avec images, des sons et de la musique.

Découverte de médicaments : L'apprentissage automatique joue un rôle crucial dans la recherche de nouveaux médicaments et de traitements pour des maladies auparavant incurables. Il facilite l'analyse de vastes quantités de données biologiques, l'identification de candidats médicaments potentiels et l'optimisation de la conception des médicaments.

Moteurs de recherche, sécurité de l'information et marketing : Les algorithmes d'apprentissage automatique sont utilisés dans les moteurs de recherche pour améliorer les résultats de recherche et l'expérience des utilisateurs. Ils contribuent également à la sécurité de l'information en détectant les anomalies et les menaces potentielles. Dans le domaine du marketing,

l'apprentissage automatique contribue à la segmentation de la clientèle, aux recommandations personnalisées et à la publicité ciblée.

Systèmes de réponse aux questions : Les modèles d'apprentissage automatique peuvent être utilisés pour construire des systèmes de réponse aux questions qui fournissent des réponses basées sur des informations stockées dans des bases de données. Cela permet d'extraire efficacement des informations pertinentes pour répondre aux questions des utilisateurs.

Prise de décision et recommandations : Les techniques d'apprentissage automatique, telles que les réseaux neuronaux, simulent les processus de prise de décision du cerveau humain. Cela permet de développer des systèmes capables de prendre des décisions et de faire des recommandations en connaissance de cause, sur la base de modèles de données complexes et des préférences des utilisateurs.

Robotique : L'apprentissage automatique contribue au progrès de la robotique, en permettant aux robots d'effectuer un large éventail de tâches, y compris celles qui sont difficiles ou délicates pour les humains. Les algorithmes d'apprentissage automatique aident les robots à comprendre leur environnement et à interagir avec lui, à s'adapter à des conditions changeantes et à effectuer des actions complexes.

Il ne s'agit là que de quelques exemples de l'application de l'apprentissage automatique dans différents domaines. La polyvalence et la puissance des techniques d'apprentissage automatique continuent de stimuler l'innovation et d'améliorer divers aspects de notre vie.

2.8 Conclusion

Ce chapitre a fourni une introduction aux principes fondamentaux de l'intelligence artificielle et de l'apprentissage automatique. Nous avons exploré les concepts de base des réseaux neuronaux artificiels (ANN) et des réseaux neuronaux convolutifs (CNN), en soulignant leur efficacité dans la classification des images. Les notions clés abordées incluent les neurones artificiels, les fonctions d'activation, les fonctions de perte et les optimisateurs. De plus, nous avons expliqué le fonctionnement des CNN, en détaillant les processus de convolution, de padding, de pooling et de flattening, ainsi que des applications pratiques telles que VGG, MobileNet, Xception et DenseNet. Ces éléments préparent le terrain pour une exploration plus approfondie des approches basées sur les réseaux neuronaux dans le chapitre suivant.

Chapitre 3

REVUE DE LA LITTÉRATURE

3.1 Introduction

Les études antérieures jouent un rôle crucial dans la pratique des radiologues pour le diagnostic du cancer, car elles permettent de détecter les changements tissulaires au fil du temps. En intégrant ces données, la précision du diagnostic peut être améliorée. L'apprentissage automatique et l'apprentissage profond ont été développés pour analyser ces vastes quantités de données. Les systèmes d'aide au diagnostic médical basés sur ces technologies, connus sous le nom de CAD (Computer-Aided Diagnosis), fournissent un deuxième avis aux radiologues, aidant à détecter et diagnostiquer des anomalies ou des maladies avec une plus grande précision. Ces systèmes, en réduisant les erreurs de diagnostic, améliorent les résultats pour les patients et augmentent le taux de survie grâce à des diagnostics plus précoces et plus précis.

3.2 Systèmes CAD Traditionnels en Mammographie

3.2.1 Développement historique et méthodologies clés

Les systèmes CAD traditionnels en mammographie sont apparus dans les années 1990. F. Winsberg et ses collègues ont publié une étude pionnière sur l'analyse des mammographies conçue pour détecter automatiquement les lésions en 1967, marquant le début du diagnostic assisté par ordinateur. Depuis, de nombreuses équipes de recherche travaillent au développement de systèmes capables d'analyser et d'interpréter automatiquement les mammographies. Ces techniques sont principalement divisées en deux processus[78] :

- **La détection assistée par ordinateur** : Dans le processus de détection assistée par ordinateur (CADE) en IRM mammaire, la première étape cruciale est d'identifier et de délimiter les tumeurs. Cette étape est indispensable pour effectuer une analyse précise des caractéristiques radiologiques, ce qui permet ensuite de calculer différentes mesures quantitatives utilisées comme marqueurs d'imagerie. Toutefois, la segmentation manuelle traditionnelle réalisée par les radiologues demande beaucoup de temps et d'expertise, et

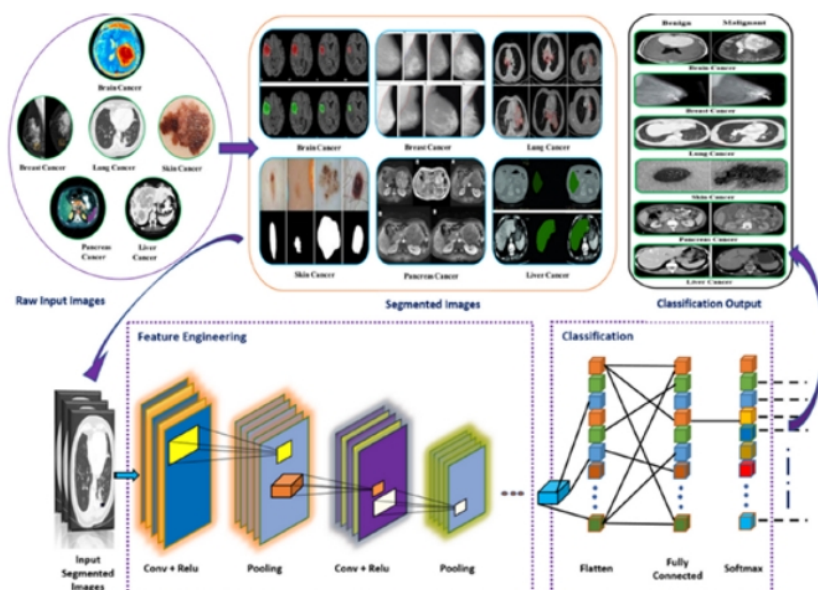


Fig. 3.1 – Système CAD de segmentation et classification des tumeurs mammaires

comporte également le risque d'erreurs. C'est pourquoi, plusieurs méthodes basées sur l'apprentissage automatique ont été développées pour automatiser ce processus de segmentation et de détection des tumeurs mammaires dans les images IRM[79].

- **Le diagnostic assisté par ordinateur** : Dans la phase de diagnostic assisté par ordinateur (CADx), on extrait des caractéristiques distinctives des tumeurs segmentées. Ces caractéristiques agissent comme des indicateurs pour un modèle de classification. Ce modèle est construit en ajustant les poids relatifs de ces caractéristiques en fonction des propriétés statistiques des données d'apprentissage. Il est utilisé pour estimer la probabilité qu'une tumeur segmentée appartienne à une catégorie pathologique[79].

Les systèmes CAD traditionnels utilisent des méthodes d'analyse d'image classiques comme [78] :

- L'extraction de caractéristiques pertinentes à partir des images mammographiques (formes, textures, etc.).
- L'application d'algorithmes de classification pour détecter et caractériser les anomalies. ImageChecker CAD a été le premier logiciel de détection assistée par ordinateur à recevoir l'agrément de la FDA aux États-Unis en 1998. Il utilise des algorithmes pour détecter automatiquement les masses et les microcalcifications suspectes sur les mammographies numériques[80].

Bien que ces systèmes aient montré leur utilité, ils ont des limitations comme la nécessité d'une extraction manuelle de caractéristiques et une capacité de généralisation limitée. Cela a motivé le développement de techniques d'apprentissage profond plus performantes pour l'analyse des mammographies.

3.3 Apprentissage Profond en Imagerie Médicale

3.3.1 Aperçu de l'apprentissage profond

L'apprentissage profond est une branche de l'intelligence artificielle qui connaît un essor important ces dernières années. Il repose sur l'utilisation de réseaux de neurones artificiels, inspirés du fonctionnement du cerveau humain. Ces réseaux sont composés de multiples couches cachées de neurones artificiels interconnectés. Chaque couche apprend à extraire des caractéristiques de plus en plus abstraites et complexes à partir des données d'entrée. Par exemple, dans le cas du traitement d'images, les premières couches apprendront à détecter des bords et des contours élémentaires, tandis que les couches supérieures combineront ces éléments pour reconnaître des formes et des objets de plus haut niveau. C'est cette capacité à apprendre des représentations hiérarchiques à partir des données brutes qui fait la puissance de l'apprentissage profond. Cela permet de s'affranchir de l'extraction manuelle de caractéristiques, fastidieuse et subjective. Les réseaux de neurones profonds sont entraînés sur de grandes quantités de données annotées, en ajustant itérativement les poids des connexions entre neurones pour minimiser l'erreur de prédiction. Ils peuvent ainsi apprendre à effectuer des tâches complexes comme la reconnaissance d'objets, la segmentation d'images ou la traduction automatique. L'apprentissage profond a révolutionné de nombreux domaines de l'intelligence artificielle et connaît de nombreuses applications dans des secteurs variés, de la vision par ordinateur à la robotique en passant par le traitement du langage naturel [81].

3.3.2 Applications en imagerie médicale

L'apprentissage profond est une branche de l'intelligence artificielle qui permet de segmenter des images médicales pour identifier des régions d'intérêt spécifiques, comme les organes, les lésions ou les vaisseaux[82]. **Exemples d'applications :**

1. **Segmentation multiclasse des artères du cercle de Willis :** Une étude a utilisé l'apprentissage profond pour segmenter de manière multiclasse les artères du cercle de Willis sur des images d'angiographie par résonance magnétique (TOF-MRA). Le cercle de Willis est une région vasculaire très variable à l'entrée du système vasculaire cérébral[82].
2. **Segmentation d'images médicales :** L'apprentissage profond est utilisé pour segmenter des images médicales pour identifier des régions d'intérêt spécifiques, comme les organes, les lésions ou les vaisseaux. Par exemple, une étude a utilisé l'apprentissage profond pour segmenter des images médicales d'angiographie par résonance magnétique (TOF-MRA) pour identifier les artères du système vasculaire cérébral[82].
3. **Segmentation d'images médicales par apprentissage de transfert :** Une autre étude a utilisé l'apprentissage de transfert pour adapter et comparer quatre modèles de réseaux de neurones de convolution (Alexnet, Resnet18, Resnet80, et U-net) pour segmenter des images médicales ultra-sonores du cancer du sein.

4. **Segmentation de structures fines** : Une thèse a mis en place des méthodes de segmentation d'images par apprentissage profond pour des objets de taille fine, comme les structures vasculaires du cerveau.

Les méthodes utilisées dans l'analyse d'images médicales comprennent les réseaux de neurones de convolution, l'apprentissage de transfert et la segmentation sémantique. Les réseaux de neurones de convolution sont efficaces pour extraire des caractéristiques et segmenter les régions d'intérêt. L'apprentissage de transfert permet d'adapter des modèles pré-entraînés pour des tâches spécifiques telles que la segmentation d'images médicales. La segmentation sémantique consiste à classifier chaque pixel de l'image dans une catégorie, offrant ainsi une analyse détaillée des structures anatomiques[83].

Ces approches présentent plusieurs avantages. Tout d'abord, elles offrent une précision élevée et une rapidité significative, permettant de segmenter les images en moins de 10 minutes par image. De plus, ces méthodes peuvent être généralisées à de grandes cohortes de patients et à différentes modalités d'imagerie médicale, assurant ainsi une application étendue. Enfin, l'apprentissage profond permet une grande flexibilité dans l'identification de régions d'intérêt spécifiques, telles que les organes, les lésions ou les vaisseaux, ce qui est essentiel dans diverses applications médicales[82].

3.4 Bases de Données et Prétraitement

La conception et le test des systèmes d'IA pour la détection et le diagnostic du cancer du sein nécessitent une quantité importante de données. En rassemblant d'énormes volumes d'images provenant de millions de patients pour construire une base de données, et en intégrant des flux de données variés tels que des données cliniques, pathologiques et génétiques, l'IA facilite l'exploration des corrélations entre ces diverses sources de données. Cela permet de développer un système de diagnostic complet et multidimensionnel ainsi qu'un plan de traitement du cancer du sein.

Le Cancer Imaging Archive (TCIA), soutenu par le National Cancer Institute (NCI), constitue actuellement la plus grande base de données d'imagerie accessible gratuitement. Parmi ses 45 collections d'imagerie du cancer, le cancer du sein représente la majorité, avec 18 collections publiques. Cependant, ces ensembles de données se basent principalement sur des patients d'Europe et des États-Unis, avec une représentation limitée de patients asiatiques. L'ensemble de données TCIA-Breast comprend quatre modalités : mammographie (MG), imagerie par résonance magnétique (IRM), tomodensitométrie (CT) et tomographie par émission de positons (TEP). La MG et l'IRM sont les deux principales modalités, offrant le plus grand nombre de cas et étant les plus largement utilisées [84].

3.4.1 Base de données MG

Actuellement, il existe six ensembles de données MG accessibles au public, et chaque ensemble de données contient des seins normaux et des lésions bénignes et malignes confirmées pathologiquement [85].

- 1.
2. **La base de données de la Mammographic Image Analysis Society (MIAS) :** MIAS représente le premier ensemble de données de mammographie accessible au public, comprenant 322 images numériques en vue oblique latérale médiale (MLO) de 161 patients, stockées au format PNG.
3. **L'ensemble de données INbreast :** INbreast, un ensemble de données de mammographie numérique plein champ (FFDM), offre 400 images provenant de 115 participants, accompagnées d'informations cliniques et de classifications du système de rapports et de données d'imagerie mammaire (BI-RADS) pour chaque patiente. Notamment, INbreast inclut des annotations de contour précises, facilitant le développement et la validation d'algorithmes pour la morphologie des lésions.
4. **La base de données numérique de mammographie de dépistage (DDSM) :** DDSM, l'ensemble de données public le plus fréquemment utilisé en mammographie, comprend 10239 images de 1566 participants de l'Université de Floride du Sud. Il propose la position MLO et la vue crano-caudale (CC) de chaque patient, ainsi que des images recadrées avec la masse et la calcification comme régions d'intérêt (ROI). Le DDSM vise à fournir un ensemble de données standard pour évaluer les systèmes d'IA pour le dépistage du cancer du sein, bien que sa précision puisse ne pas être optimale pour valider des algorithmes de segmentation existants.
5. **La base de données chinoise sur les mammographies (CMMD) :** CMMD, développé par l'Université de technologie de Chine du Sud, représente la seule base de données de mammographie comprenant des populations asiatiques. Avec 3728 images de 1775 patientes chinoises, il offre des informations cliniques liées aux images et aux sous-types moléculaires de 749 patientes. CMMD est principalement utilisé en Chine pour former des modèles d'apprentissage profond (DL) visant le diagnostic des microcalcifications mammaires.
6. **La base de données de dépistage numérique du cancer du sein et de tomosynthèse mammaire (BCS-DBT) :** BCS-DBT, la plus grande base de données d'images DBT actuelle, comprend 22032 images de 5060 participants fournis par l'hôpital universitaire Duke. Initialement utilisé dans le défi DBTex2, parrainé par l'Association américaine des physiciens en médecine (AAPM), BCS-DBT fournit des informations détaillées sur l'emplacement, la limite et la taille de chaque lésion candidate, ainsi que le score de fiabilité.
7. **La base de données des essais in silico :** L'ensemble de données VICTRE, publié par la FDA, évalue l'utilisation potentielle de la tomosynthèse mammaire (DBT) en tant que

substitut à la mammographie numérique (DM). Il comprend 217 913 images de 2 994 sujets, servant principalement à reproduire des expériences comparatives antérieures. De plus, il met à disposition des outils logiciels open source pour explorer systématiquement les paramètres expérimentaux, notamment le type et la taille des lésions.

3.4.2 Méthodes de Prétraitement d'Images :

Cette étape vise à améliorer la qualité des images mammographiques pour optimiser la classification des anomalies. L'objectif est de renforcer les contrastes entre les objets, mettant en évidence le tissu mammaire tout en réduisant le bruit potentiel introduit lors de la capture d'image.

Différentes techniques de filtrage ont été évaluées, parmi lesquelles le filtre Wiener, reconnu pour sa capacité à réduire le bruit tout en préservant un niveau minimal, évitant une suppression complète du signal, comme démontré dans des études antérieures. Cependant, il peut altérer les détails fins tels que les bords, comme mentionné dans la littérature.

Le filtre médian, une autre technique de filtrage couramment utilisée, remplace chaque valeur de pixel par la médiane du voisinage, contribuant à réduire la variation d'intensité entre les pixels tout en préservant la netteté des bords, comme décrit dans des recherches antérieures.

Une analyse détaillée souligne la complexité des applications des techniques de prétraitement pour différents types d'images médicales et met en avant l'importance de ce processus. Dans cette étude, l'égalisation adaptative limitée de l'histogramme (CLAHE) a été privilégiée. Cette méthode opère sur des zones spécifiques de l'image mammaire, appliquant une égalisation à chacune d'elles.

L'utilisation de CLAHE est justifiée par les caractéristiques des images mammaires, qui présentent moins de lignes brisées.

En ce qui concerne la conversion des images Dicom en jpg, cette étape est cruciale pour préparer les données à être utilisées en entrée dans un réseau neuronal convolutif (CNN).

Les CNN automatisent la construction des caractéristiques et sont largement utilisés dans l'analyse d'images médicales. En raison de la rareté des données médicales disponibles, l'augmentation des données a été appliquée pour élargir l'ensemble de données.

Les images du sein ont été augmentées en utilisant des techniques telles que le retournement, la rotation, le recadrage et la translation sur chaque type d'anomalies.

Enfin, pour éliminer les informations d'arrière-plan non désirées, les images du sein ont été segmentées. Une image prétraitée à l'aide de CLAHE est illustrée dans la figure 3.3, démontrant les effets bénéfiques du prétraitement sur la qualité visuelle des images[86].

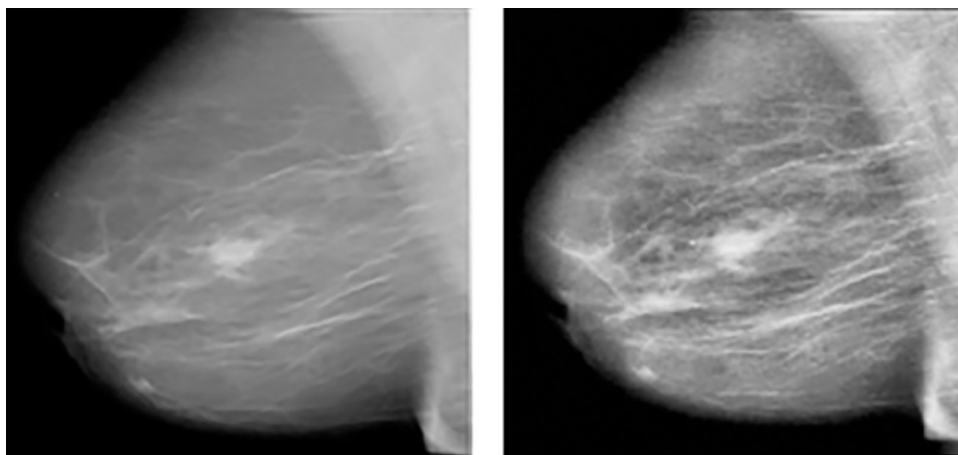


Fig. 3.2 – (a) Image originale (b) Image après CLAHE.
[86]

3.5 Les performances des modèles d'apprentissage automatique

L'analyse des mammographies à l'aide de l'apprentissage automatique constitue un domaine de recherche en plein essor. Les algorithmes de machine learning sont utilisés pour détecter et classifier les anomalies présentes dans les images médicales, notamment les masses suspectes dans les mammographies. Plusieurs études ont été menées pour développer des systèmes automatisés capables d'identifier les signes précurseurs du cancer du sein, tels que les microcalcifications et la distorsion architecturale. Dans une étude, Mina et ses collègues ont proposé un système utilisant des réseaux de neurones artificiels pour la détection des anomalies mammaires, atteignant une précision impressionnante de 91,64%. Une autre recherche menée par Saini et Vijay a comparé les performances de différents types de réseaux neuronaux pour classer les mammographies en tissus bénins et malins, avec des résultats prometteurs. D'autres approches ont été explorées, telles que l'utilisation de méthodes de segmentation et de classification basées sur des caractéristiques texturales extraites des images. Par exemple, Perez et son équipe ont développé un système de classification basé sur les caractéristiques texturales, avec une précision moyenne de 84,72%. De même, Pratiwi et ses collègues ont utilisé des réseaux neuronaux à fonctions de base radiale pour classer les masses anormales dans les mammographies, obtenant une précision remarquable de 93,98%. Ces avancées soulignent l'importance cruciale de la détection précoce du cancer du sein. Cependant, la segmentation des images et l'identification des régions d'intérêt peuvent parfois entraîner des faux positifs, ce qui nécessite des techniques plus avancées pour réduire ces erreurs. Les réseaux neuronaux profonds émergent comme une solution prometteuse pour améliorer la précision du dépistage du cancer du sein, ouvrant ainsi la voie à de nouvelles avancées dans ce domaine vital de la santé publique[3].

Références	Méthodologies	Base de données	Résultats
[87]	Classification des tumeurs mammaires basée ANN	DDSM	Accuracy=88%
[87]	Approche de classification des tumeurs mammaires basée Random Forest	DDSM	Accuracy=92%
[88]	Classification des tumeurs mammaires comme bénignes ou malignes basée SVM.	DDSM	Accuracy=83.53%
[89]	Une approche de diagnostic automatique extrayant les descripteurs par les techniques : NSST/GLCM/Cropping où une classification des tumeurs basée : SVM/NB/MLP/KNN/LDA	MIAS	Accuracy=59.1%
[90]	Classification de la densité des tumeurs mammaires basée XGBoost.	MIAS	Accuracy=91.51%
[91]	Une approche extrayant les descripteurs : GLCM/PCA/LDA/MDS à partir des tumeurs où une classification basée DCNN a été réalisée	Une base locale	Accuracy=82.43%

Tab. 3.1 – Méthodes CAD basées sur l'apprentissage automatique pour le diagnostic du cancer du sein.

3.6 Les techniques d'apprentissage profond appliquées à l'analyse des mammographies

Plusieurs études récentes ont appliqué des modèles d'apprentissage profond pour l'analyse automatique des mammographies, obtenant de meilleures performances que les méthodes CAD traditionnelles.

3.6.1 classification multiclasse des anomalies du cancer du sein à l'aide du CNN

Heenaye-Mamode et al. a développé un réseau neuronal convolutif (CNN) pour segmenter et classer des types distincts d'anomalies mammaires, telles que l'asymétrie, les calcifications, les masses et les carcinomes. Tout d'abord, la méthode Transfer Learning a été appliquée sur leur ensemble de données à l'aide du modèle pré-entraîné ResNET-50. Ensuite, ils ont utilisé un modèle d'apprentissage profond amélioré en ajustant le taux d'apprentissage de manière adaptative en fonction des variations des courbes d'erreur. En conséquence, le nouveau modèle a atteint un PRE de 88% dans la classification de ces quatre types d'anomalies du cancer du sein (masses, calcifications, carcinomes et asymétrie) dans les images MG[92].

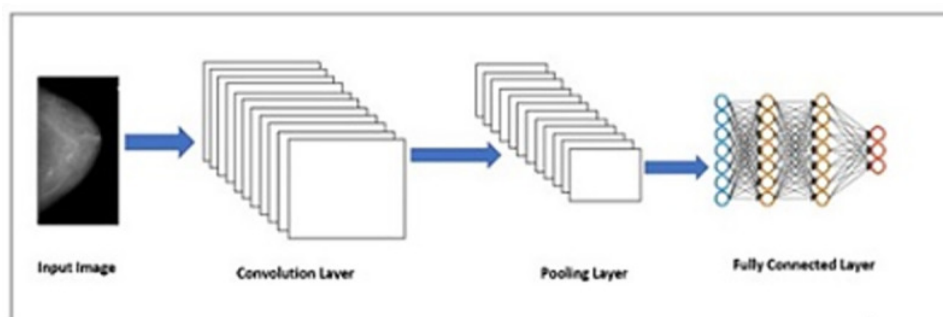


Fig. 3.3 – Layers in CNN.
[92]

3.6.1.1 Transférer l'apprentissage à l'aide de ResNet50 pré-entraîné

L'apprentissage par transfert repose sur l'utilisation des fonctionnalités déjà acquises dans une tâche pour les appliquer à une autre tâche sans recommencer l'apprentissage à partir de zéro. Cette approche est couramment utilisée avec des modèles de réseaux neuronaux convolutifs (CNN) préalablement construits et formés sur des ensembles de données bien connus. Un CNN analyse une image d'entrée en attribuant des pondérations à différentes caractéristiques de l'image afin de la distinguer des autres. Il est composé de plusieurs couches, comprenant des couches convolutionnelles, de pooling et entièrement connectées. Les couches convolutionnelles extraient les caractéristiques de l'image, tandis que les couches de pooling réduisent la complexité en sélectionnant des échantillons et en éliminant d'autres. Ces couches contribuent également à l'invariance de traduction. Les couches entièrement connectées lient les unités des couches supérieures. Nous avons utilisé ResNet50, une version améliorée de ResNet, avec 50 couches de raccourcis de connexions contenant des blocs résiduels. Ces raccourcis nécessitent moins de calculs tout en fournissant des fonctionnalités riches. ResNet50 comprend une couche convolutive, des couches de normalisation et 16 modules résiduels entre deux couches de pooling. Nous avons appliqué ResNet50 à la classification des anomalies des images mammaires. Les poids ont été définis initialement, et la base de convolution a été utilisée comme extracteur de caractéristiques avec gel des poids. La taille spatiale de la carte des caractéristiques est réduite grâce à Maxpooling. Notre ensemble de données a été divisé en 60 % pour l'ensemble d'entraînement et 40 % pour l'ensemble de tests[92].

3.6.2 Modèle CNN Amélioré avec Noyaux de Petite Dimension et Cadres d'Apprentissage Résiduel Profond

Le développement du modèle CNN est largement utilisé dans le diagnostic et l'analyse d'images médicales, en particulier dans le système CAD où l'imagerie médicale bénéficie du succès du CNN. Pour notre application de détection du cancer du sein, nous avons choisi d'utiliser le CNN avec des améliorations spécifiques. Un CNN typique est composé de plusieurs neurones organisés en différentes couches, formant des noyaux. Les noyaux sont généralement

de petite taille pour correspondre à la profondeur de l'image d'entrée. Un champ récepteur, représentant une petite région, est connecté aux neurones, car il est complexe de connecter tous les neurones aux sorties précédentes, surtout avec des images de grande dimension. Par exemple, une image de 100 x 100 pixels avec 100 neurones nécessiterait 1 000 000 de paramètres. Chaque neurone n'a pas besoin d'avoir un poids de dimension complète, mais plutôt le poids correspondant à la dimension du noyau auquel il est connecté. Le noyau glisse sur l'image en hauteur et en largeur pour extraire des fonctionnalités de haut niveau, produisant ainsi une carte d'activation bidimensionnelle (2D). La vitesse de glissement peut être contrôlée par un paramètre approprié. La sortie de la couche de convolution est empilée, et la carte d'activation devient l'entrée de la couche suivante. Notre architecture utilise un modèle séquentiel, qui empile les couches séquentielles du réseau dans l'ordre de l'entrée à la sortie. La couche de convolution utilise des filtres pour effectuer des opérations de convolution sur l'image d'entrée, avec des hyper-paramètres tels que la taille du filtre et la foulée. Nous avons ajouté une première couche convolutive 2D pour traiter les images mammaires d'entrée, utilisant 16 canaux de sortie avec un filtre 3x3 et une foulée de 1. Nous avons expérimenté différentes tailles de noyau et constaté que la petite taille du noyau 1x1 est un goulot d'étranglement. La fonction d'activation ReLU est appliquée, et le regroupement maximum est utilisé pour réduire la taille spatiale de la carte des caractéristiques. Deux autres couches convolutives sont ajoutées avec 32 et 64 canaux de sortie, initialement utilisant un filtre de pooling maximum 2x2. Inspirés par les travaux antérieurs, nous avons intégré un cadre d'apprentissage résiduel profond avec des unités répétées de couches de convolution de tailles de filtre 1x1, 3x3 et 1x1. La mise en commun moyenne globale est utilisée pour calculer l'activation de chaque carte de fonctionnalités[92].

3.6.3 Classification BI-RADS basée sur l'Architecture Profonde Efficient- NET

Tsai et ses collaborateurs ont entrepris une classification BI-RADS en utilisant une base de données de l'hôpital E-Da à Taiwan. Ils ont attribué des étiquettes aux images en se basant sur la proportion de zones lésées dans la localisation, utilisant un bloc de 224×224 avec un pas de 36 pixels. Leur méthode s'est appuyée sur l'architecture profonde EfficientNET. Comme illustré dans Figure 3.5, le modèle, composé d'une tige, d'un corps, d'une tête et d'un mode de sortie, prend en entrée une mammographie de taille 224×224 pixels, soit une forme d'image d'entrée de $224 \times 224 \times 1$. Dans la tige module, l'image d'entrée est d'abord normalisée pour se situer entre 0 et 1, puis les cartes de caractéristiques sont extraites à l'aide d'une couche de convolution 3×3 . Par la suite, des cartes de fonctionnalités de haut niveau sont extraites dans le module Body, composé de 16 blocs mobiles de convolution de goulot d'étranglement inversé (MBCConv). Enfin, les cartes de fonctionnalités sont classées dans les modules Head et Output. Les résultats ont montré un PRE de 94,22%, une sensibilité (SEN) de 95,31% et une spécificité (SPE) de 99,15%, mettant en évidence l'efficacité de leur approche pour la classification précise des images BI-RADS[93].

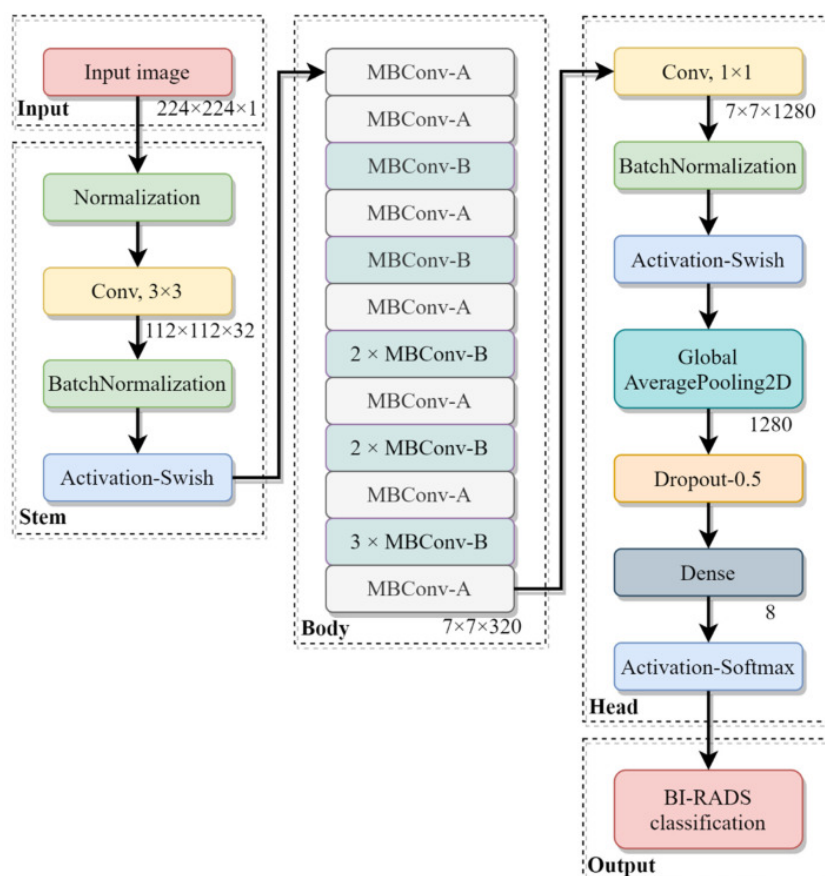


Fig. 3.4 – Organigramme du modèle de classification BI-RADS présenté. [93]

3.6.4 Modèle Hybride d'Apprentissage Profond pour la Classification Précise des Anomalies Mammaires

Dans cette recherche, une approche novatrice de Classification d'Anomalies Mammaires (CAO) est introduite, exploitant un modèle hybride combinant l'Apprentissage Profond (AP) basé sur Convolutional Neural Networks (CNN) et l'Analyse en Composantes Principales Locales (LR-PCA). Le système de CAO se compose de trois étapes principales. Tout d'abord, des images pseudo-colorées à trois canaux sont générées à partir des images en niveaux de gris d'origine, permettant ainsi aux modèles d'AP d'extraire des caractéristiques profondes plus robustes pour une meilleure performance diagnostique. Deuxièmement, des CNN pré-entraînés tels qu'AlexNet, VGG et GoogleNet sont utilisés pour extraire des caractéristiques profondes de haut niveau en utilisant la stratégie d'apprentissage par transfert, atténuant ainsi le risque de surajustement. Enfin, le modèle LR-PCA est proposé pour sélectionner les composantes principales significatives parmi les caractéristiques extraites, surmontant ainsi le problème de multicolinéarité entre ces caractéristiques. Le système de CAO comprend cinq modules distincts, comprenant le prétraitement des images, l'extraction de caractéristiques via des CNN pré-entraînés, la réduction de dimensionnalité grâce à la technique PCA, la classification des caractéristiques en utilisant des méthodes traditionnelles d'apprentissage automatique, et enfin,

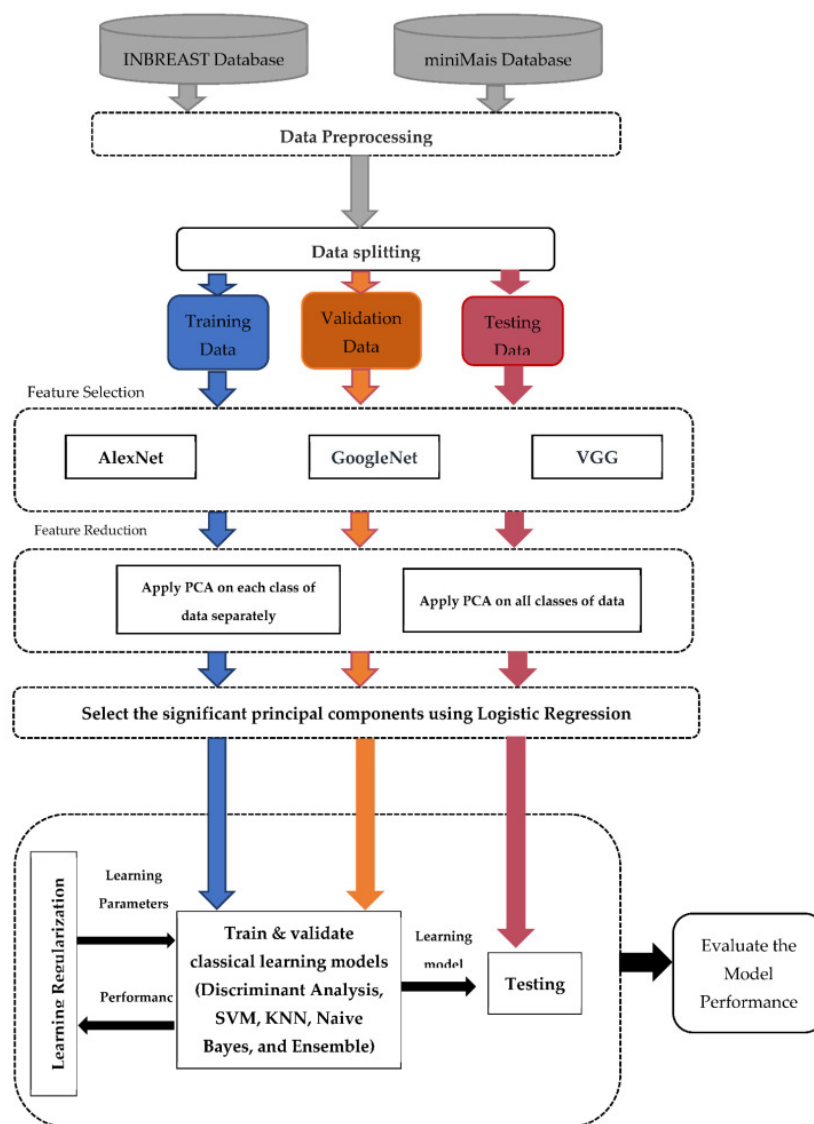


Fig. 3.5 – Cadre CAD proposé pour la classification des lésions mammaires à partir des mammographies.

[94]

l'évaluation du classificateur à travers des mesures de performances spécifiques aux systèmes de CAO[94]. Cette approche intégrée est représentée de manière détaillée dans la Figure 3.5

3.6.4.1 Réduction dimensionnelle à l'aide de LR-PCA

La classification des images médicales à l'aide de l'apprentissage par transfert génère un grand nombre de fonctionnalités, créant ainsi des opportunités mais aussi des défis, notamment le problème de multicollinéarité. Ce phénomène survient lorsque les caractéristiques d'entrée présentent une forte corrélation entre elles, impactant ainsi l'efficacité des modèles de régression et de classification. Pour pallier cela, l'Analyse en Composantes Principales (ACP) est couramment utilisée, car elle permet de combiner les variables hautement corrélées en un ensemble de variables non corrélées. L'ACP est reconnue comme la technique originale et prédominante

de réduction dimensionnelle, souvent employée dans la classification des mammographies via des approches d'apprentissage automatique classiques. Le processus de PCA implique la transformation de l'espace de fonctionnalités, la réduction de la dimension, et l'analyse de l'espace de fonctionnalités transformé. PCA, en tant qu'approche non supervisée, diminue la dimension des données en exploitant la corrélation entre les caractéristiques d'entrée. La sélection des composants principaux (PC) lors de l'ACP repose sur l'idée que les directions présentant les variations les plus significatives contiennent le plus d'informations sur les classes. La méthode traditionnelle consiste à fixer un seuil de pourcentage de variance expliquée, par exemple 80%, et à choisir le nombre de composants qui atteignent une somme cumulée de variance proche de ce seuil. Cependant, cette approche présente des faiblesses, notamment le caractère arbitraire du choix du seuil et l'incertitude quant à l'élimination du bruit par rapport à des informations importantes. Pour surmonter ces limitations, nous proposons une approche hybride utilisant la régression logistique (LR) conjointement avec l'ACP, créant ainsi l'algorithme LR-PCA. La LR, modèle de classification binaire simple et efficace, est implémentée en utilisant un sous-ensemble de PC provenant de l'ACP. Les PC choisis dans la LR sont ceux avec des coefficients significatifs dans le modèle construit, ayant une valeur p inférieure à 0,05. L'algorithme LR-PCA se déroule comme suit :

- Création d'une matrice de covariance pour l'ensemble de données.
- Détermination des vecteurs propres et des valeurs propres de cette matrice.
- Sélection du nombre souhaité de dimensions en filtrant les vecteurs propres correspondants, triés par leurs valeurs propres associées.
- Multiplication de l'espace d'origine par le vecteur de caractéristiques produit à l'étape précédente.
- Création d'un modèle de régression logistique multinomiale pour les fonctionnalités récupérées.
- Choix des principales composantes du modèle LR nouvellement créé qui ont des coefficients significatifs.

Ainsi, l'approche LR-PCA combine les avantages de l'ACP pour la réduction de dimension tout en intégrant la sélection de caractéristiques basée sur la régression logistique, préservant ainsi les composants significatifs pour la classification d'images médicales[94].

3.7 Les performances des modèles d'apprentissage profonds

Pour résoudre les problèmes des algorithmes d'apprentissage automatique utilisés dans les systèmes actuels de détection assistée par ordinateur pour les mammographies, les chercheurs se tournent de plus en plus vers l'apprentissage profond. Ces systèmes actuels reposent sur des caractéristiques prédéfinies et une classification manuelle, ce qui peut conduire à des taux élevés de fausses alertes et à des temps d'analyse prolongés. En revanche, l'apprentissage profond,

notamment avec les réseaux neuronaux profonds, révolutionne l'interprétation des images médicales. Grâce à cette approche, les tissus cancéreux peuvent être détectés avec une précision accrue et les caractéristiques pertinentes peuvent être extraites automatiquement. Des techniques comme Faster RCNN, les CNN avec transformations discrètes ou en courbelettes, ainsi que les modèles basés sur VGG, offrent des performances remarquables dans la détection et la classification des lésions sur les mammographies. Ces avancées permettent une meilleure détection des micro-calcifications, une classification plus précise des masses mammaires et une localisation améliorée des régions d'intérêt, ce qui pourrait avoir un impact significatif sur le diagnostic précoce du cancer du sein[3].

Références	Méthodologies	Base de données	Résultats
[95]	Breast lesions Segmentation/Classification YOLO	INbreast	Accuracy=95.32%
[95]	Classification CNN-RNN (Recurrent Neural Network)	DDSM	Accuracy=94.7%
[96]	Classification ResNet	INbreast	AUC=0.95
[97]	Classification VGG16.	DDSM	Accuracy=93.64 %
[95]	Classification DenseNet-121.	MIAS	Accuracy=95.63%
[98]	Classification DenseNet169	MIAS	Accuracy=92.3%
[97]	Classification AlexNet	DDSM	Accuracy=91.87%
[95]	classification MobileNet-V2	DDSM	Accuracy=91.36%

Tab. 3.2 – Méthodes CAD basées sur l'apprentissage profond pour le diagnostic du cancer du sein.

3.8 Défis et Lacunes dans la Recherche Actuelle

La recherche actuelle dans le domaine des systèmes de diagnostic assisté par ordinateur (CAD) et de l'apprentissage profond pour l'analyse des mammographies, bien qu'innovante et prometteuse, présente plusieurs défis et lacunes nécessitant une attention particulière. Identifier ces problèmes non résolus est crucial pour orienter les futurs travaux et améliorer les performances des systèmes CAD.

1. Variabilité des Données et Généralisation

Problème : Les modèles d'apprentissage profond nécessitent de grandes quantités de données annotées pour être efficaces. Cependant, les bases de données disponibles peuvent varier considérablement en termes de qualité, de format et de distribution des cas cliniques.

Amélioration Potentielle : Développer des techniques de normalisation des données et d'augmentation des données pour améliorer la robustesse et la généralisation des modèles sur des ensembles de données diversifiés.

2. Manque de Données Annotées

Problème : L'acquisition de données médicales annotées de haute qualité est coûteuse et chronophage, limitant ainsi la disponibilité de grandes bases de données nécessaires pour l'entraînement des modèles d'apprentissage profond.

Amélioration Potentielle : Utiliser des techniques de semi-apprentissage supervisé, d'apprentissage non supervisé ou d'apprentissage par transfert pour réduire la dépendance aux données annotées manuellement.

3. Interprétabilité des Modèles

Problème : Les modèles d'apprentissage profond, en particulier les réseaux de neurones profonds, sont souvent perçus comme des "boîtes noires" en raison de leur complexité, rendant difficile l'interprétation des décisions prises par le modèle.

Amélioration Potentielle : Développer des techniques d'explicabilité pour les réseaux de neurones, permettant aux cliniciens de comprendre et de faire confiance aux décisions des systèmes CAD.

4. Sensibilité et Spécificité

Problème : Bien que les systèmes CAD puissent améliorer la détection des anomalies mammaires, ils peuvent également produire des taux élevés de faux positifs et de faux négatifs, entraînant des diagnostics incorrects et des traitements inappropriés.

Amélioration Potentielle : Affiner les algorithmes de détection pour équilibrer la sensibilité et la spécificité, en réduisant les erreurs de classification tout en maintenant une haute précision de détection.

5. Intégration Clinique

Problème : L'intégration des systèmes CAD dans les flux de travail cliniques existants pose des défis en termes de compatibilité, d'acceptation par les utilisateurs et de formation des professionnels de santé.

Amélioration Potentielle : Concevoir des interfaces utilisateur intuitives et fournir des programmes de formation pour faciliter l'adoption des systèmes CAD par les radiologues et autres professionnels de santé.

6. Adaptabilité aux Nouveaux Cas

Problème : Les systèmes CAD doivent être continuellement mis à jour pour s'adapter aux nouveaux types de données et aux avancées médicales, mais cette mise à jour peut être difficile et coûteuse.

Amélioration Potentielle : Mettre en place des mécanismes d'apprentissage en continu et des mises à jour régulières des bases de données pour maintenir les systèmes CAD à jour avec les dernières recherches et découvertes cliniques.

En adressant ces défis et en comblant ces lacunes, la recherche future peut améliorer de manière significative l'efficacité et la fiabilité des systèmes CAD, contribuant ainsi à des diagnostics plus précis et à une meilleure prise en charge des patientes atteintes de cancer du sein.

3.9 Conclusion

Les domaines de l'apprentissage automatique et profond, issus de l'intelligence artificielle, connaissent une croissance remarquable, particulièrement dans l'analyse de l'imagerie médicale et le diagnostic du cancer du sein. Ces techniques nécessitent de vastes quantités de données, rendues disponibles grâce aux avancées en matière de sauvegarde et de stockage d'images médicales. Cela permet de développer des modèles de prédiction pour de nouveaux cas. Dans le prochain chapitre, nous aborderons les problématiques actuelles de ce domaine et le cadre méthodologique adopté pour mettre en œuvre notre approche.

Chapitre 4

La méthodologie

4.1 Introduction

Dans ce chapitre, nous détaillons la méthodologie employée pour mener à bien notre étude de classification des anomalies mammaires. Nous commençons par décrire les étapes de collecte et de prétraitement des données, mettant en lumière les ensembles de données utilisés ainsi que les techniques de normalisation, d'augmentation et de segmentation. Ensuite, nous explorons les modèles d'apprentissage profond, justifiant le choix des architectures spécifiques et fournissant une explication détaillée de chaque modèle. Nous décrivons également les procédures d'entraînement, les réglages des hyperparamètres et les techniques de validation. Enfin, nous définissons les métriques d'évaluation qui seront utilisées pour mesurer les performances des modèles sur l'ensemble de test.

4.2 Collecte des données

4.2.1 Description du jeu de données DDSM

La base de données que nous avons utilisée est une combinaison d'images provenant des ensembles de données DDSM (Digital Database for Screening Mammography) et CBIS-DDSM (Curated Breast Imaging Subset of DDSM). Cette base de données a été prétraitée pour répondre aux exigences spécifiques de notre étude sur la détection des anomalies mammaires à l'aide de techniques de deep learning.

La pertinence de la base de données DDSM pour notre étude réside dans sa richesse en données médicales d'imagerie mammaire. DDSM est l'un des ensembles de données les plus utilisés dans la recherche sur le cancer du sein, offrant une grande diversité d'images de mammographies numériques, y compris des cas positifs et négatifs pour le cancer du sein. CBIS-DDSM, quant à lui, est une version annotée et améliorée de DDSM, contenant des annotations détaillées sur les régions d'intérêt (ROI) pour faciliter l'analyse et la détection des anomalies.

Dans notre ensemble de données combiné, les images ont été prétraitées et converties en images

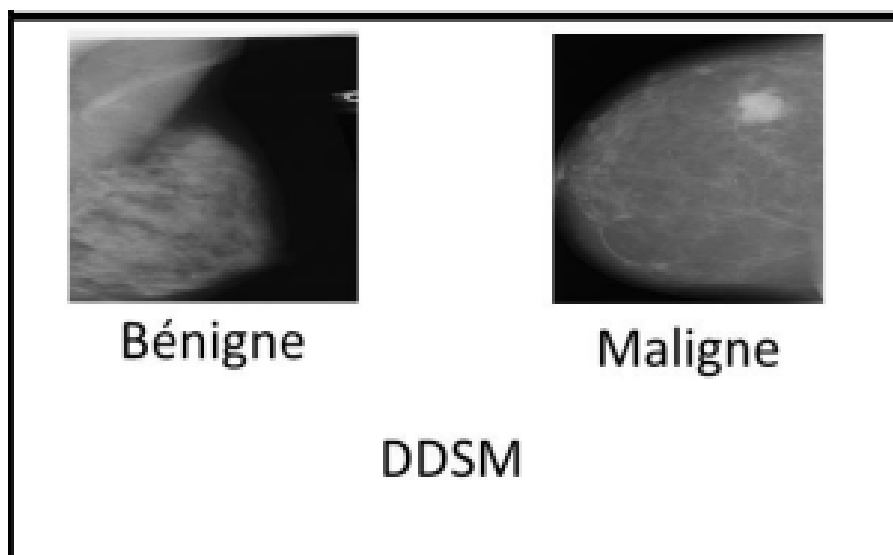


Fig. 4.1 – Exemple des mammographies bénignes et malignes DDSM

de taille uniforme de 299x299 pixels, avec extraction des régions d'intérêt (ROI) pour les cas positifs. Les données ont été stockées sous forme de fichiers TFRecord pour une manipulation efficace avec TensorFlow.

Notre ensemble de données comprend un total de 55 890 exemples d'entraînement, avec une répartition de 14 % d'exemples positifs et 86 % d'exemples négatifs. Ces exemples sont répartis en 5 fichiers TFRecord.

Il convient de noter que la séparation des données en ensembles de formation, de validation et de test a été réalisée en suivant la division existante dans l'ensemble de données CBIS-DDSM. Cependant, une correction a été identifiée dans la répartition des données de test et de validation, où certains fichiers contiennent uniquement des masses ou des calcifications. Cette situation doit être rectifiée en combinant les fichiers afin d'obtenir des ensembles de test équilibrés et complets, assurant ainsi la fiabilité des évaluations de nos modèles de détection des anomalies mammaires[99][100].

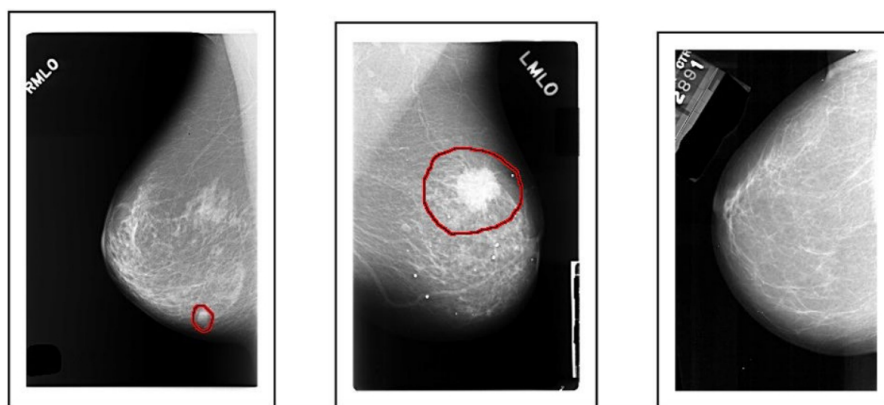


Fig. 4.2 – Images bénignes, malignes et normales de la base de données DDSM

4.2.2 Prétraitement des données

Avant d'intégrer les images mammaires dans nos modèles de détection des anomalies, un processus de prétraitement méticuleux a été appliqué pour garantir la qualité et la pertinence des données. Voici les différentes étapes de prétraitement :

4.2.2.1 Chargement des données :

L'instruction **tf.data.TFRecordDataset** provient de l'API TensorFlow, une bibliothèque open source de Google utilisée pour le développement de modèles de machine learning et de deep learning. Cette fonction est spécifiquement conçue pour charger des données à partir de fichiers TFRecord, qui sont un format de stockage de données efficace pour les grands ensembles de données TensorFlow[101].

1. Algorithme

La fonction **tf.data.TFRecordDataset** prend en entrée un ou plusieurs fichiers TFRecord et crée un dataset TensorFlow à partir de ces fichiers. Voici comment elle fonctionne :

Chargement du Dataset à partir des Fichiers TFRecord :

La fonction **tf.data.TFRecordDataset** prend comme premier argument le nom du fichier ou une liste de noms de fichiers TFRecord à charger.

Par exemple, `filename` est le nom du fichier TFRecord à partir duquel les données seront chargées.

Spécification de la Parallélisation de la Lecture :

L'argument **num_parallel_reads** est utilisé pour spécifier le nombre de threads qui seront utilisés pour lire les données à partir des fichiers.

La valeur **tf.data.experimental.AUTOTUNE** permet à TensorFlow de décider automatiquement du nombre optimal de threads à utiliser en fonction de diverses considérations de performance.

Création du Dataset :

La fonction retourne un objet `tf.data.TFRecordDataset` qui représente le dataset chargé à partir des fichiers TFRecord spécifiés.

filename est le nom du fichier TFRecord à partir duquel les données seront chargées.

tf.data.experimental.AUTOTUNE est utilisé pour permettre à TensorFlow de déterminer automatiquement le nombre optimal de threads à utiliser pour lire les données à partir du fichier TFRecord.

- 2. Pourquoi Utiliser TFRecordDataset?** Les fichiers TFRecord sont un format de stockage de données efficace et optimisé pour TensorFlow. Ils sont particulièrement utiles pour gérer de grandes quantités de données, comme les ensembles de données d'images, audio ou texte. L'utilisation de TFRecordDataset permet de charger efficacement ces données dans un format adapté à TensorFlow, facilitant ainsi le traitement et l'entraînement des modèles de machine learning[102].

4.2.2.2 Mélange du Dataset

L'instruction **shuffle** provient de l'API `tf.data` de TensorFlow. TensorFlow est une bibliothèque open source de Google pour le machine learning et le deep learning. L'API `tf.data` permet de créer des pipelines de données efficaces et modulables pour préparer les données à l'entraînement des modèles. La fonction `shuffle` est utilisée pour réorganiser aléatoirement les éléments du dataset afin d'introduire de la variance et d'éviter les biais d'ordre[103].

1. **Algorithme** La fonction `shuffle` prend un argument **buffer_size** qui définit la taille du tampon pour le mélange des éléments. L'algorithme de base fonctionne comme suit :

Tampon de Mélange :

Un tampon (buffer) de taille `buffer_size` est créé. Le tampon est rempli avec les premiers **buffer_size** éléments du dataset.

Mélange :

Les éléments du tampon sont mélangés aléatoirement. À mesure que de nouveaux éléments sont ajoutés au tampon (au fur et à mesure de l'itération sur le dataset), ils sont mélangés avec les éléments existants dans le tampon.

Extraction des Éléments Mélangés :

Les éléments sont extraits du tampon dans un ordre aléatoire, assurant que le dataset est lu dans un ordre mélangé.

Une fois que le tampon est vidé, le processus continue avec les éléments restants du dataset.

buffer_size=31000 : Cette valeur signifie que le tampon de mélange peut contenir jusqu'à 31 000 éléments à la fois. Le choix de la taille du tampon dépend de la taille totale du dataset et des ressources disponibles (mémoire).

2. **Pourquoi Mélanger le Dataset ?**

Introduction de l'Aléatoire :

Le mélange aléatoire des données empêche le modèle d'apprendre des schémas liés à l'ordre des données, ce qui peut améliorer la généralisation du modèle.

Prévention du Surapprentissage :

Lorsque les données sont présentées dans un ordre aléatoire, le modèle est moins susceptible de mémoriser les exemples spécifiques et plus susceptible d'apprendre des caractéristiques générales des données.

Amélioration de la Convergence :

Le mélange des données peut aider le modèle à converger plus rapidement en réduisant les corrélations indésirables entre les exemples d'entraînement.

4.2.2.3 Mise en Cache du Dataset

L'instruction `cache` provient de l'API `tf.data` de TensorFlow. TensorFlow est une bibliothèque open source développée par Google pour le machine learning et le deep learning. L'API `tf.data` permet de créer des pipelines de données efficaces pour la préparation des données d'entraînement. La fonction `cache` est utilisée pour mettre en cache les éléments du dataset afin d'améliorer les performances lors des itérations répétées sur les données[104].

1. **Algorithme** La fonction `cache` met en cache les éléments du dataset après qu'ils ont été lus et transformés pour la première fois. Cela signifie que lors des itérations suivantes sur le dataset, les données sont directement lues à partir du cache, ce qui est beaucoup plus rapide que de les relire ou les retraiter à chaque itération.

Première Itération :

Lors de la première itération, les éléments du dataset sont lus et les transformations (comme le parsing, le mélange, etc.) sont appliquées.

Les éléments transformés sont stockés dans le cache.

Itérations Suivantes :

Lors des itérations suivantes, les éléments sont directement lus à partir du cache.

Cela évite les lectures répétées à partir du disque et les recalculs des transformations, ce qui améliore les performances.

Emplacement du Cache :

Par défaut, le cache est stocké en mémoire. Toutefois, il est possible de spécifier un fichier pour stocker le cache sur le disque si le dataset est trop grand pour tenir en mémoire.

`full_dataset.cache('/path/to/cache/file')` peut être utilisé pour stocker le cache sur le disque.

2. **Pourquoi Mettre en Cache le Dataset? Amélioration des Performances :**

La mise en cache réduit le temps nécessaire pour lire et transformer les données lors des itérations suivantes. Cela est particulièrement utile pour les étapes d'entraînement des modèles de machine learning où les mêmes données sont souvent lues plusieurs fois.

Réduction des E/S :

Mettre en cache les données réduit le nombre de lectures de disque, ce qui peut être un goulot d'étranglement, surtout pour les grands datasets. Cela améliore l'efficacité et la vitesse du pipeline de données.

Stabilité du Pipeline :

En garantissant que les transformations coûteuses ne sont effectuées qu'une seule fois, la mise en cache permet de stabiliser le temps d'exécution du pipeline de données.

4.2.2.4 Définition du Dictionnaire de Caractéristiques

L'instruction pour définir un dictionnaire de caractéristiques provient de l'API **tf.io** de TensorFlow, qui est utilisée pour traiter les données d'entrée/sortie, y compris la lecture et l'écriture de fichiers TFRecord. Les fichiers TFRecord sont un format binaire optimisé pour stocker des données de grande taille, souvent utilisées dans le machine learning. La fonction **tf.io.FixedLenFeature** est utilisée pour décrire la structure des données stockées dans ces fichiers[105].

1. Algorithme

Le dictionnaire de caractéristiques (feature dictionary) spécifie les caractéristiques (features) contenues dans chaque enregistrement du fichier TFRecord et leur type de données. Voici comment il fonctionne :

Définition des Caractéristiques :

Chaque clé du dictionnaire correspond au nom d'une caractéristique stockée dans le fichier TFRecord. La valeur associée à chaque clé est un objet **tf.io.FixedLenFeature** qui spécifie le type et la forme de la caractéristique.

Types de Données :

tf.int64 : Spécifie que la caractéristique est un entier 64 bits. **tf.string** : Spécifie que la caractéristique est une chaîne de caractères.

Forme des Caractéristiques :

[] : Indique que la caractéristique est un scalaire (c'est-à-dire une seule valeur).

2. Pourquoi Définir un Dictionnaire de Caractéristiques ?

Interprétation des Données :

Le dictionnaire permet de décrire précisément la structure des données stockées dans le fichier TFRecord. Cela est essentiel pour la lecture et le décodage corrects des données.

Parsing Efficace :

En utilisant **tf.io.FixedLenFeature**, TensorFlow sait comment décoder chaque caractéristique d'un enregistrement TFRecord, ce qui facilite le processus de parsing.

Validation des Données : Le dictionnaire de caractéristiques aide à valider que les données lues à partir du fichier TFRecord correspondent à la structure attendue, aidant ainsi à détecter les erreurs de format ou de type de données.

4.2.2.5 Mapping de la Fonction de Parsing

L'instruction de mapping de la fonction de parsing provient de l'API **tf.data** de TensorFlow. TensorFlow est une bibliothèque open source développée par Google pour le machine learning et le deep learning. L'API **tf.data** permet de créer des pipelines de données efficaces pour préparer les données d'entraînement. La fonction **map** est utilisée pour appliquer une transformation à chaque élément du dataset. Dans ce contexte, la transformation consiste à appliquer une fonction de parsing pour extraire les données des exemples TFRecord[106].

1. Algorithme

Le mapping de la fonction de parsing implique l'application d'une fonction personnalisée sur chaque élément du dataset. Cette fonction de parsing utilise `tf.io.parse_example` pour décoder les données brutes stockées dans les fichiers TFRecord en caractéristiques utilisables.

Définition de la Fonction de Parsing :

La fonction de `parsing_parse_function` prend un exemple TFRecord brut et un dictionnaire de caractéristiques (`feature_dictionary`), puis utilise `tf.io.parse_example` pour extraire les données.

Mapping de la Fonction sur le Dataset :

La fonction de parsing est mappée sur le dataset en utilisant `dataset.map`, ce qui permet d'appliquer cette fonction à chaque exemple du dataset de manière parallèle pour des performances optimales.

2. Pourquoi Mapper une Fonction de Parsing ?

Extraction des Données :

Le mapping de la fonction de parsing permet d'extraire les données brutes stockées dans les fichiers TFRecord en caractéristiques utilisables pour l'entraînement du modèle.

Efficacité :

En utilisant `dataset.map`, chaque exemple du dataset est transformé en parallèle, ce qui améliore considérablement la performance du pipeline de données.

Simplicité : La fonction `map` simplifie l'application de transformations complexes sur le dataset, rendant le code plus lisible et maintenable.

Itération sur le Dataset et Prétraitement des Images :

4.2.2.6 Conversion Eager Execution de TensorFlow à NumPy

L'instruction `image = image_features['image'].numpy()` utilise une opération de conversion appelée "Conversion Eager Execution de TensorFlow à NumPy". Cette méthode est fournie par TensorFlow pour convertir un tenseur en un tableau NumPy lorsqu'il est utilisé en mode Eager Execution[107] [101].

1. Fonctionnement de l'instruction :

Accès au tenseur :

`image_features` est un dictionnaire de tenseurs, et `'image'` est la clé pour accéder au tenseur qui contient les données d'image.

Conversion en NumPy :

La méthode `.numpy()` est appelée sur le tenseur pour le convertir en un tableau NumPy.

2. Étapes de l'algorithme

Vérification du mode Eager Execution : TensorFlow doit être en mode Eager Execution pour permettre cette conversion. Ce mode est généralement activé par défaut à partir de TensorFlow 2.x.

Accès direct aux données : La méthode `.numpy()` accède directement aux données du

tenseur.

Conversion en tableau NumPy : Les données sont converties en un tableau NumPy, permettant une manipulation plus aisée en dehors de l'écosystème TensorFlow.

3. Contexte et Utilisation :

La méthode `.numpy()` est utile pour des opérations où une manipulation ou une analyse des données est plus facile ou plus performante en utilisant les fonctionnalités de NumPy. Cela est courant dans le prétraitement des données, l'analyse exploratoire des données et le débogage.

4.2.2.7 Décompression et décodage des images :

L'algorithme utilisé dans l'instruction `image = tf.io.decode_raw(image_features['image'], tf.uint8)` est appelé "décodage brut des données sérialisées" ou "décodage des données binaires sérialisées".

Cette fonction, `tf.io.decode_raw`, fait partie des opérations d'entrée/sortie de TensorFlow pour la manipulation des données[108] [109].

1. Fonctionnement de `tf.io.decode_raw` :

Entrée : Une chaîne de bytes (données binaires sérialisées) et un type de données cible (dans ce cas, `tf.uint8`) valeurs entre 0 et 255, typique pour les valeurs de pixel d'une image en niveaux de gris)..

Sortie : Un tenseur de type spécifié contenant les données décodées.

2. Étapes de l'algorithme :

Prendre une chaîne de bytes : La fonction prend une chaîne de bytes comme entrée.

Interprétation des bytes : Les bytes sont interprétés comme une séquence de valeurs du type spécifié (`tf.uint8` ici).

Conversion en tenseur : Les bytes sont convertis en un tenseur TensorFlow du type spécifié, ce qui permet leur manipulation dans les opérations de TensorFlow.

Cet algorithme est couramment utilisé dans les scénarios suivants :

Prétraitement des données : Avant d'alimenter les données dans un modèle de machine learning, en particulier lorsque les données sont stockées sous forme binaire dans des fichiers TFRecord.

Traitement des images : Conversion des données d'image sérialisées en tenseurs pour des transformations ultérieures ou pour l'entraînement de modèles de vision par ordinateur.

4.2.2.8 Conversion des images en images 299x299 pixels :

L'instruction `image = tf.reshape(image, [299, 299])` utilise une opération de TensorFlow appelée "Reshape". Cette opération change la forme d'un tenseur sans modifier ses données sous-jacentes[110][111].

1. Fonctionnement de l'instruction :

Entrée : Un tenseur image avec une certaine forme. Opération de reshape : `tf.reshape(image, [299, 299])` change la forme du tenseur image pour qu'il ait la forme [299, 299].

Sortie : Un nouveau tenseur avec les mêmes données que l'original, mais réorganisées selon la nouvelle forme spécifiée.

2. Étapes de l'algorithme :

Vérification de la compatibilité : TensorFlow vérifie si le nombre total d'éléments dans le tenseur d'origine correspond au nombre total d'éléments requis par la nouvelle forme.

Réorganisation des données : Les éléments du tenseur d'origine sont réorganisés pour correspondre à la nouvelle forme, sans altérer les valeurs des éléments.

3. Contexte et utilisation :

La fonction `tf.reshape` est utile pour transformer les tenseurs dans la forme requise par des opérations ou des modèles spécifiques. Par exemple, dans le traitement d'images, il est souvent nécessaire de remodeler les images pour les adapter aux exigences d'entrée d'un réseau de neurones.

4.2.2.9 Découpage et redimensionnement des images négatives et positives :

Les images positives issues de l'ensemble de données CBIS-DDSM ont subi une extraction des régions d'intérêt (ROI) à l'aide de masques, suivie de recadrages aléatoires pour augmenter la variabilité des données. Les images recadrées ont ensuite été redimensionnées à une taille de 100x100 pixels afin de garantir la cohérence avec les images négatives.

Pour les images négatives provenant de l'ensemble de données DDSM, des mosaïques de 598x598 pixels ont été découpées puis redimensionnées à 100x100 pixels en utilisant la fonction '`cv2.resize()`'. Ce processus a permis de réduire la taille des images tout en conservant les caractéristiques essentielles.

L'instruction `image = cv2.resize(image, (100, 100))` utilise une fonction de la bibliothèque OpenCV pour redimensionner une image. L'algorithme sous-jacent à cette instruction est appelé "Redimensionnement d'Image" ou "Interpolation Bilinéaire" (par défaut, OpenCV utilise l'interpolation bilinéaire pour le redimensionnement des images)[112] [113].

1. Fonctionnement de l'instruction :

Entrée : Une image représentée sous forme de tableau NumPy.

Sortie : Une image redimensionnée avec les nouvelles dimensions (100, 100).

2. Étapes de l'algorithme :

Lecture des dimensions d'entrée : L'algorithme lit les dimensions actuelles de l'image.

Calcul des nouveaux indices de pixels : Pour chaque pixel dans l'image redimensionnée, les indices correspondants dans l'image d'origine sont calculés.

Interpolation : Utilise une méthode d'interpolation (bilinéaire par défaut) pour calculer la valeur des nouveaux pixels en fonction des pixels de l'image d'origine.

Construction de l'image redimensionnée : Génère une nouvelle image avec les dimensions spécifiées en utilisant les valeurs de pixels interpolées.

3. **Contexte et utilisation :** Le redimensionnement d'image est une opération courante dans le traitement d'images et la vision par ordinateur. Cela peut être nécessaire pour :
 - *Normaliser la taille des images avant de les entrer dans un modèle de machine learning ou deep learning.
 - *Adapter les images à une taille spécifiée pour une analyse ou une visualisation uniforme.
 - *Modifier la résolution des images pour diverses applications.

4.2.2.10 Conversion en format RVB (Red, Green, Blue) :

Bien que les images originales soient en niveaux de gris, elles ont été converties en format RVB en répétant les canaux pour créer une image couleur. Cette conversion est nécessaire pour s'aligner avec les attentes des modèles de deep learning pré-entraînés sur des ensembles de données en couleur tels qu'ImageNet.

L'instruction `image = cv2.merge([image, image, image])` utilise une fonction de la bibliothèque OpenCV pour combiner plusieurs canaux en une seule image multicanal. Dans ce cas particulier, elle crée une image en niveaux de gris (mono-canal) en une image couleur à trois canaux (RGB) où chaque canal est identique.

1. **Fonctionnement de l'instruction : Entrée :** Une image en niveaux de gris (2D) représentée sous forme de tableau NumPy.
Sortie : Une image couleur (3D) avec trois canaux où chaque canal est identique.
2. **Étapes de l'algorithme : Réception des canaux :** `cv2.merge` reçoit trois images (ou canaux) sous forme de listes, ici `[image, image, image]`.
Fusion en une image multicanal : Combine les trois images 2D en une seule image 3D où chaque canal de couleur est constitué de l'image d'entrée.
Construction de l'image finale : Crée une image avec trois dimensions (hauteur, largeur, canaux), chaque canal étant identique à l'image d'origine.
3. **Contexte et Utilisation :** La fonction `cv2.merge` est souvent utilisée pour :
 - *Créer une image couleur à partir de canaux séparés.
 - *Dupliquer une image en niveaux de gris sur les trois canaux pour créer une image pseudo-couleur.
 - *Traiter des images multicanal dans des algorithmes de vision par ordinateur qui nécessitent des entrées couleur.

4.2.2.11 Normalisation des données :

L'instruction `image = image / 255.0` normalise les valeurs des pixels de l'image pour les amener dans la plage `[0, 1]`. Dans une image en niveaux de gris, les valeurs des pixels varient

de 0 à 255. En divisant chaque valeur de pixel par 255, elles sont ramenées dans la plage [0, 1]. Cette normalisation est effectuée après le redimensionnement et la conversion de l'image en trois canaux, mais avant de l'ajouter à la liste des images. En normalisant les images, le modèle de machine learning peut mieux apprendre les caractéristiques importantes des données, sans être perturbé par les variations d'échelle des valeurs de pixels.

Normaliser les valeurs des pixels garantit que les données sont dans une plage spécifique, généralement entre 0 et 1. Cette étape standardise les données et stabilise l'entraînement du modèle, assurant que les valeurs sont à une échelle appropriée.

4.2.2.12 Ajout d'Éléments à une Liste (Append) :

L'instruction `labels.append(image_features['label_normal'].numpy())` utilise une méthode standard de Python pour ajouter un élément à la fin d'une liste, en convertissant d'abord un tenseur TensorFlow en une valeur NumPy.

1. Fonctionnement de l'instruction

Entrée : Une liste nommée `labels` et un élément `image_features['label_normal']`, qui est initialement un tenseur TensorFlow.

Sortie : La liste `labels` est modifiée en place pour inclure le nouvel élément à la fin.

2. Étapes de l'algorithme :

Accès au tenseur : `image_features['label_normal']` est un tenseur TensorFlow.

Conversion en NumPy : La méthode `.numpy()` est appelée sur ce tenseur pour obtenir une valeur NumPy.

Insertion de l'élément : La méthode `append` ajoute cette valeur NumPy à la fin de la liste `labels`.

Mise à jour de la liste : La liste est mise à jour pour inclure le nouvel élément.

3. Contexte et Utilisation :

La méthode `append` est couramment utilisée pour construire des listes de manière itérative. Dans ce cas, elle est utilisée pour stocker des étiquettes (labels) extraites d'un fichier TFRecord dans une liste, après les avoir converties de tenseurs TensorFlow en valeurs NumPy.

Ces étapes de prétraitement ont été cruciales pour préparer les données à être utilisées dans nos modèles de détection des anomalies mammaires. Elles ont permis de standardiser la taille des images, de conserver les caractéristiques importantes et d'augmenter la variabilité des données, contribuant ainsi à améliorer les performances et la généralisation des modèles d'apprentissage profond.

4.3 Architecture des modèles

4.3.1 Modèles d'apprentissage automatique

ces différents modèles ont chacun leurs forces et faiblesses en fonction des caractéristiques des données et de la tâche à réaliser.

4.3.1.1 XGBoost (Extreme Gradient Boosting)

- XGBoost est un algorithme de boosting très performant, souvent utilisé pour les tâches de classification.
- Il diffère des autres algorithmes car c'est une bibliothèque open-source ou un framework pour les algorithmes, plutôt qu'un algorithme en soi.
- XGBoost est très efficace pour améliorer les performances d'autres modèles comme les forêts aléatoires, les réseaux de neurones, [114][115].

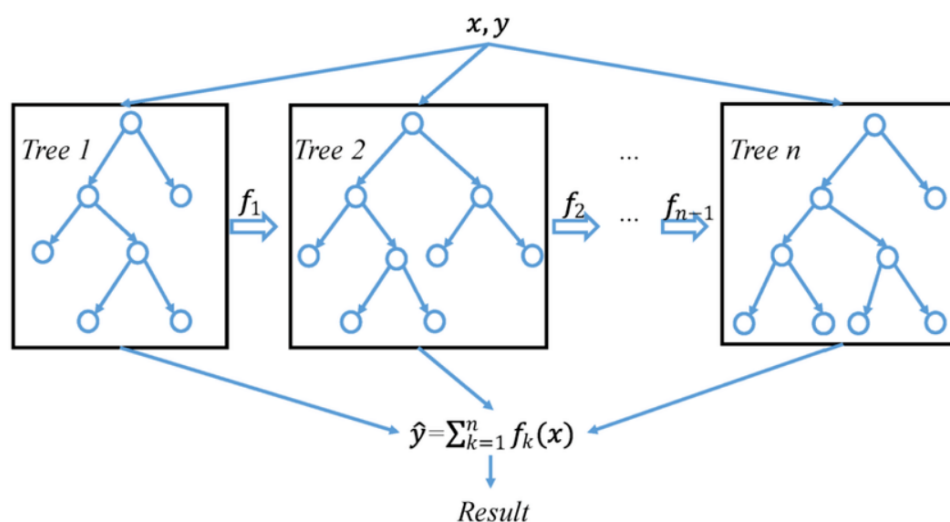


Fig. 4.3 – architecture globale de xgboost
[115]

4.3.1.2 Forêts aléatoires (Random Forest)

- Les forêts aléatoires sont un ensemble d'arbres de décision entraînés sur des sous-ensembles aléatoires des données.
- Elles sont robustes au bruit et aux données aberrantes, et permettent de gérer facilement les variables catégorielles et continues.
- Les forêts aléatoires sont très performantes pour la classification de données de télédétection [114].

4.3.1.3 Arbres de décision (Decision Trees)

- Les arbres de décision sont des modèles supervisés utilisés pour les tâches de classification et de régression.
- L'idée principale est de trouver les variables les plus informatives pour séparer les données et construire un arbre de décision.
- Les arbres de décision sont flexibles, faciles à interpréter mais peuvent souffrir du surapprentissage[116].

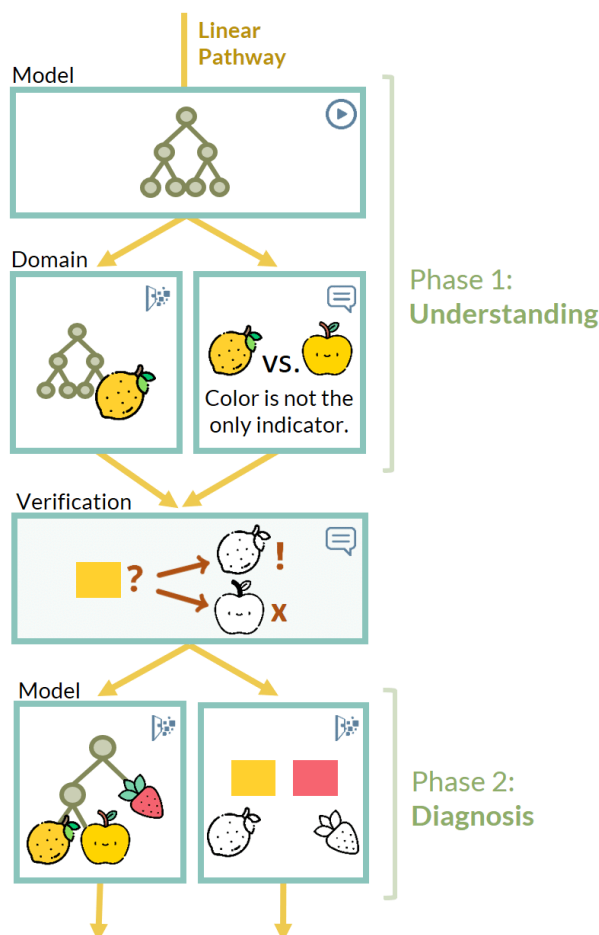


Fig. 4.4 – Un arbre de décision pour un apprentissage non linéaire.
[18]

4.3.1.4 Le réseau neuronal multicouche perceptron (MLP)

Le réseau neuronal multicouche perceptron (MLP) est une architecture d'apprentissage automatique couramment utilisée pour la classification d'images, y compris la classification des anomalies mammaires. Un MLP est composé de plusieurs couches de neurones artificiels interconnectés, qui apprennent à identifier des modèles complexes dans les données d'entrée[117].

1. Fonctionnement :

Prétraitement des images : Normalisation des images pour une cohérence des données d'entrée.

Couche d'entrée : Chaque pixel devient un neurone artificiel.

Couches cachées : Apprentissage de modèles complexes avec des fonctions d'activation non linéaires.

Couche de sortie : Classification en fonction du nombre de classes (normal, bénigne, maligne).

Apprentissage : Ajustement des poids des connexions pour minimiser l'erreur de classification.

2. Avantages :

Capacité d'apprentissage non linéaire : Identification d'anomalies subtiles.

Robustesse au bruit : Résistance aux variations de qualité d'image.

Interprétabilité : Analyse des poids pour comprendre les caractéristiques clés.

3. Inconvénients :

Manque d'extraction automatique de caractéristiques : Nécessite une sélection manuelle des caractéristiques.

Sur-apprentissage potentiel : Difficulté de généralisation à de nouvelles images.

Optimisation des hyperparamètres complexe : Nécessite une expertise en apprentissage automatique.

Performance parfois inférieure aux architectures plus récentes : Les CNN peuvent être plus précises.

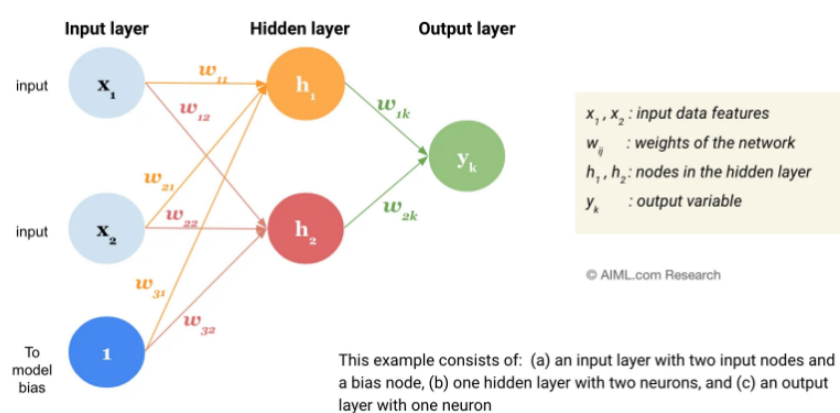


Fig. 4.5 – Structure MLP
[118]

4.3.1.5 Naive Bayes

- Naive Bayes est un classifieur probabiliste basé sur le théorème de Bayes avec une hypothèse d'indépendance entre les variables.

- C'est une méthode simple et rapide, qui nécessite peu de données d'entraînement.

- Naive Bayes est souvent utilisé pour la classification de données de cancer du sein[114].

4.3.2 Description des algorithmes Deep CNN

Les réseaux de neurones convolutifs profonds (DCNN) se révèlent prometteurs pour la classification des anomalies mammaires à partir d'images mammographiques. Leur capacité à apprendre automatiquement des caractéristiques pertinentes, leur robustesse aux variations d'images et leur haute précision en font des outils puissants pour l'aide au diagnostic[119][120].

1. Fonctionnement :

- Les images mammographiques sont prétraitées pour normalisation.
- Des couches convolutives extraient des caractéristiques discriminantes.
- Des couches successives hiérarchisent ces caractéristiques.
- Le pooling réduit la dimensionnalité et le sur-apprentissage.
- Des couches entièrement connectées associent les caractéristiques à une classification.

2. Avantages :

- Apprentissage automatique des caractéristiques.
- Robustesse aux variations d'images.
- Haute précision, notamment pour les cancers précoces.

3. Inconvénients :

- Nécessité de gros volumes de données pour l'apprentissage.
- Difficulté d'interprétation des modèles.
- Coût computationnel élevé pour l'entraînement.
- Malgré ces limites, les DCNN représentent une avancée majeure dans la classification des anomalies mammaires et offrent un potentiel important pour améliorer le diagnostic et le suivi des patientes.

4.3.3 Modèles de deep learning testés

J'ai testé chaque modèle en les entraînant sur nos données d'entraînement et en évaluant leurs performances sur les données de validation. Cette approche nous permettra de déterminer quel modèle est le plus adapté à la classification des anomalies mammaires dans notre étude.

4.3.3.1 DenseNet121 :

L'architecture DenseNet, proposée par He et al.[121], se distingue par ses connexions denses entre les couches. Contrairement aux architectures CNN classiques où chaque couche n'est connectée qu'à un nombre limité de couches précédentes, DenseNet connecte chaque couche à toutes les couches qui la précèdent. Cette approche permet une meilleure propagation des gradients et une meilleure utilisation des caractéristiques intermédiaires, ce qui se traduit par une meilleure performance en termes de précision et de généralisation.

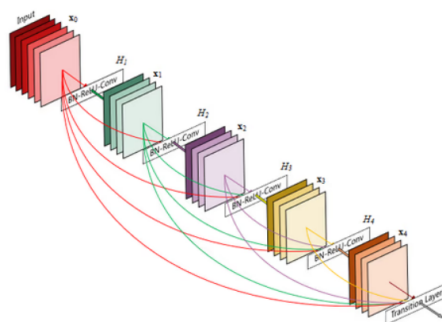


Fig. 4.6 – L'architecture DenseNet
[122]

DenseNet121 est variante de l'architecture DenseNet avec des profondeurs différentes. DenseNet121 comprend 121 couches, lui permet d'extraire des caractéristiques plus complexes et d'obtenir une meilleure performance, mais au prix d'une augmentation du temps de calcul et des besoins en mémoire.

1. Avantages de DenseNet121 :

Bonne performance en termes de précision et de généralisation : DenseNet a démontré une capacité remarquable à identifier correctement les anomalies mammaires dans des ensembles de données complexes. Sa structure dense permet une meilleure exploitation des informations et une meilleure propagation des erreurs, ce qui conduit à une meilleure précision et une meilleure capacité à généraliser à de nouvelles données.

Flux d'informations efficace entre les couches : Les connexions denses entre les couches favorisent une circulation plus fluide des informations à travers le réseau. Cela permet aux couches ultérieures d'accéder à des caractéristiques intermédiaires provenant de couches plus précoces, ce qui enrichit la représentation des données et améliore la performance du modèle.

Moins de paramètres que d'autres architectures CNN comparables : DenseNet utilise généralement moins de paramètres que d'autres architectures CNN comparables à performance équivalente. Cela s'explique par la structure dense qui permet d'exploiter plus efficacement les informations, réduisant ainsi le besoin de couches supplémentaires et de paramètres associés.

2. Inconvénients de DenseNet121 :

Peut être gourmand en calcul pour les modèles profonds : DenseNet121, avec sa profondeur accrue, peut nécessiter des ressources de calcul importantes pour l'entraînement et l'inférence. Cela peut être un inconvénient pour les applications en temps réel ou sur des appareils mobiles à faible puissance de calcul.

Moins d'interprétabilité que d'autres architectures : La structure dense de DenseNet peut rendre l'interprétation du modèle plus difficile. En effet, la complexité des interactions entre les couches rend plus ardu l'analyse de l'impact de chaque couche sur la décision finale.

4.3.3.2 MobileNet

MobileNet, proposée par Howard et al. [123], est une architecture DCNN conçue spécifiquement pour les appareils mobiles. Elle se distingue par son optimisation pour la compacité et l'efficacité, la rendant idéale pour les applications embarquées où les ressources de calcul et de mémoire sont limitées.

MobileNet combine deux approches clés pour atteindre son efficacité :

Depthwise Separable Convolutions (DSC) : Comme Xception, MobileNet utilise des DSC pour décomposer une convolution standard en deux étapes distinctes : une convolution par profondeur et une convolution par point. Cette décomposition permet de réduire considérablement le nombre de paramètres et le temps de calcul, tout en conservant une performance comparable aux convolutions standard.

Couches à pointes inversées (inverted bottleneck layers) : MobileNet introduit des couches à pointes inversées pour extraire des caractéristiques non linéaires. Ces couches inversent l'ordre des opérations de convolution et d'activation, permettant d'obtenir une meilleure efficacité et une meilleure expressivité du modèle.

1. Avantages de MobileNet pour la classification des anomalies mammaires sur appareils mobiles :

Architecture très compacte et efficace : MobileNet utilise un nombre minimal de paramètres et nécessite peu de mémoire pour l'entraînement et l'inférence. Cela la rend idéale pour une utilisation sur des smartphones, des tablettes et d'autres appareils mobiles à puissance de calcul limitée.

Bonne performance pour une architecture compacte : Malgré sa légèreté, MobileNet a démontré sa capacité à identifier correctement les anomalies mammaires dans des ensembles de données complexes. Son efficacité permet d'obtenir des performances acceptables avec des ressources limitées.

Meilleure interprétabilité que d'autres architectures : La structure simplifiée de MobileNet, avec ses DSC et ses couches à pointes inversées, la rend plus facile à comprendre et à analyser que d'autres architectures CNN complexes. Cela peut être un avantage pour les développeurs mobiles qui souhaitent mieux comprendre le fonctionnement du modèle.

2. Inconvénients de MobileNet :

Moins performante que d'autres architectures pour les tâches complexes : Bien que MobileNet offre un bon compromis entre compacité et performance, elle peut être surpassée par des architectures plus récentes et plus sophistiquées pour des tâches de classification d'anomalies mammaires très complexes.

Architecture moins flexible que d'autres architectures : La structure simplifiée de MobileNet peut limiter sa flexibilité pour être adaptée à des tâches spécifiques. D'autres architectures offrent une modularité plus importante, permettant une personnalisation plus fine des modules et des connexions.

4.3.3.3 InceptionResNetV2

L'architecture InceptionResNetV2, proposée par Szegedy et al. [124], tire parti des forces de deux architectures DCNN renommées : Inception et ResNet. Cette combinaison synergique permet d'obtenir une performance exceptionnelle pour la classification d'images, notamment dans le domaine de la classification des anomalies mammaires.

InceptionResNetV2 s'appuie sur les modules Inception, introduits par Szegedy et al. Ces modules regroupent plusieurs couches convolutives avec des tailles de noyaux différentes, permettant d'extraire des caractéristiques à différentes échelles spatiales. Cette approche multi-échelle s'avère particulièrement adaptée à la classification d'images, car elle permet de capturer des détails importants à différentes résolutions, des traits fins aux structures plus larges.

Pour faciliter l'apprentissage de modèles profonds, InceptionResNetV2 intègre des connexions résiduelles, introduites par He et al. [125]. Ces connexions résiduelles permettent de propager les gradients plus efficacement à travers le réseau, ce qui améliore la performance et la convergence du processus d'apprentissage. Cela s'avère particulièrement crucial pour les modèles profonds comme InceptionResNetV2, qui comptent un grand nombre de couches.

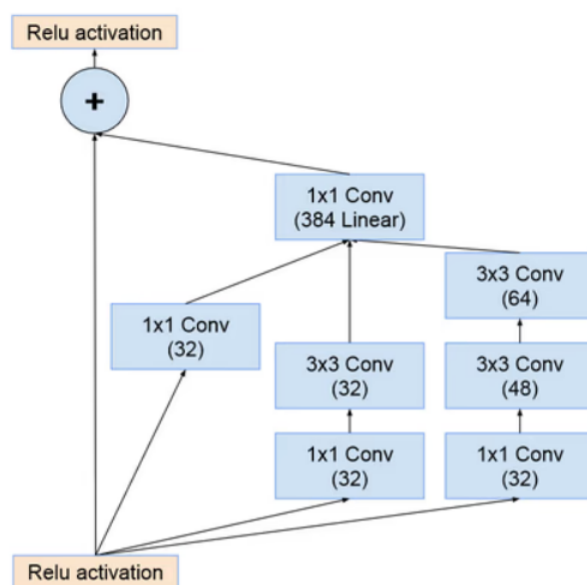


Fig. 4.7 – L'architecture Inceptionresnet [122]

1. Avantages de InceptionResNetV2 :

Excellente performance en termes de précision et de généralisation : InceptionResNetV2 a prouvé sa capacité à identifier correctement les anomalies mammaires avec une grande précision, surpassant souvent d'autres architectures DCNN. Cette performance remarquable s'explique par la combinaison des modules Inception et des connexions résiduelles, qui permettent une meilleure extraction des caractéristiques et un apprentissage plus efficace.

Capacité à extraire des caractéristiques à différentes échelles spatiales : Grâce aux modules Inception, InceptionResNetV2 peut capturer des informations à différentes résolutions, ce qui est crucial pour la classification d'images. Cela permet de prendre en compte à la fois les détails fins et les structures plus larges des anomalies mammaires, améliorant ainsi la précision de la classification.

Architecture modulaire et flexible : InceptionResNetV2 est conçue de manière modulaire, ce qui facilite sa modification et son adaptation à des tâches spécifiques. Ce caractère flexible permet de l'ajuster aux particularités de la classification des anomalies mammaires, en ajoutant ou en supprimant des modules Inception ou en ajustant leurs paramètres.

2. Inconvénients de InceptionResNetV2 :

Peut être gourmand en calcul pour les modèles profonds : En raison de sa complexité et de sa profondeur, InceptionResNetV2 peut nécessiter des ressources de calcul importantes pour l'entraînement et l'inférence. Cela peut limiter son utilisation sur des appareils mobiles ou dans des applications en temps réel où les ressources sont restreintes.

Moins d'interprétabilité que d'autres architectures : La structure complexe de InceptionResNetV2 peut rendre l'interprétation du modèle plus difficile. En effet, la multitude de modules Inception et de connexions résiduelles peut obscurcir la relation entre les entrées et les sorties du modèle, rendant l'analyse de l'impact de chaque couche sur la décision finale plus ardue.

4.3.3.4 Xception

L'architecture Xception, proposée par Chollet [126], s'inspire des modules Inception tout en étant conçue pour être plus légère et plus efficace. Cette approche la rend particulièrement intéressante pour la classification des anomalies mammaires, où un compromis entre performance, efficacité et interprétabilité est crucial.

Xception décompose une convolution standard en deux étapes distinctes : une convolution par profondeur (depthwise convolution) et une convolution par point (pointwise convolution). Cette décomposition permet de réduire considérablement le nombre de paramètres et le temps de calcul, tout en conservant une performance comparable aux convolutions standard. Cela rend Xception plus économe en ressources et plus adaptée aux applications mobiles ou aux appareils à faible puissance de calcul.

1. Avantages de Xception pour la classification des anomalies mammaires :

Bonne performance en termes de précision et de généralisation : Malgré sa légèreté, Xception a démontré une capacité remarquable à identifier correctement les anomalies mammaires dans des ensembles de données complexes. Sa structure ingénieuse permet d'extraire des caractéristiques pertinentes avec une efficacité remarquable.

Architecture plus légère et plus efficace que les architectures Inception originales : La décomposition des convolutions standard en deux étapes permet à Xception d'utiliser

moins de paramètres et de nécessiter moins de temps de calcul pour l'entraînement et l'inférence. Cela la rend plus adaptée aux applications où les ressources sont limitées.

Meilleure interprétabilité que d'autres architectures : La structure simplifiée de Xception, avec ses modules Depthwise Separable Convolutions, la rend plus facile à comprendre et à analyser que d'autres architectures CNN complexes. Cela peut être un avantage pour les chercheurs et les développeurs qui souhaitent mieux comprendre le fonctionnement du modèle et identifier les caractéristiques qui contribuent à la prise de décision.

2. Inconvénients de Xception :

Moins performante que d'autres architectures pour les tâches complexes : Bien que Xception offre un bon compromis entre performance et efficacité, elle peut être surpassée par d'autres architectures plus récentes et plus sophistiquées pour des tâches de classification d'anomalies mammaires très complexes.

Architecture moins flexible que d'autres architectures : La structure simplifiée de Xception peut limiter sa flexibilité pour être adaptée à des tâches spécifiques. D'autres architectures offrent une modularité plus importante, permettant une personnalisation plus fine des modules et des connexions.

4.4 Entraînement des Modèles

4.4.1 Modèles d'apprentissage automatique traditionnel :

- Les modèles XGBoost, Naive Bayes, Random Forest et les réseaux de neurones artificiels ont été entraînés en utilisant l'ensemble d'entraînement avec les paramètres par défaut de scikit-learn.
- Les images ont été aplaties pour être compatibles avec chaque modèle respectif.

4.4.2 Modèles de Réseaux de Neurones Convolutionnels (CNN) :

- Plusieurs architectures de réseaux de neurones ont été testées, incluant DenseNet121, InceptionResNetV2, Xception, CNN, et MobileNet. Et ont été entraînés en utilisant l'ensemble d'entraînement.
- Le callback EarlyStopping a été utilisé pour interrompre l'entraînement si la performance sur l'ensemble de validation cessait de s'améliorer pendant 5 époques consécutives.
- L'entraînement a été effectué sur un maximum de 20 époques avec une taille de batch de 128.

Pour chaque modèle, nous avons suivi une procédure d'entraînement standard :

Optimiseur : Adam.

Fonction de perte : Binary Crossentropy (pour les modèles de deep learning).

Validation : 20% des données d'entraînement ont été utilisées pour la validation.

4.5 Évaluation des Modèles

Les modèles ont été évalués sur l'ensemble de test pour mesurer leur précision, leur score F1, leur précision, leur rappel, et d'autres métriques pertinentes[97].

4.5.1 Métriques d'Évaluation des Performances en Classification des Anomalies Mammaires

- **Accuracy (Exactitude)** L'accuracy mesure la proportion de prédictions correctes par rapport au total des prédictions. C'est une métrique simple et intuitive, particulièrement utile quand les classes sont équilibrées. Cependant, elle peut être trompeuse en cas de déséquilibre des classes.

$$\text{Accuracy} = \frac{TPB + TPM}{TOT}$$

*TPB représente les vrais positifs pour la classe "Bénin" (True Positives for Benign).

*TPM représente les vrais positifs pour la classe "Malin" (True Positives for Malignant).

*TOT représente le total des exemples.

- **Precision (Précision)** La précision mesure la proportion de vrais positifs parmi toutes les prédictions positives du modèle. Un score de précision élevé indique un faible taux de faux positifs. La précision est importante quand on veut minimiser les fausses alertes.

$$\text{Precision(Bénin)} = \frac{TPB}{EBM + TPB}$$

$$\text{Precision(Malin)} = \frac{TPM}{EMB + TPM}$$

- **Recall (Rappel)** Le rappel mesure la proportion de vrais positifs correctement prédits parmi tous les cas positifs réels. Un score de rappel élevé indique un faible taux de faux négatifs. Le rappel est important quand on veut minimiser les faux négatifs.

$$\text{Recall(Bénin)} = \frac{TPB}{EBM + TPB}$$

$$\text{Recall(Malin)} = \frac{TPM}{EMB + TPM}$$

- **F1-score** Le F1-score est la moyenne harmonique pondérée de la précision et du rappel. Il varie de 0 à 1, 1 étant le meilleur score. Le F1-score prend en compte à la fois la précision et le rappel, pénalisant les valeurs extrêmes de l'une ou l'autre. C'est une métrique équilibrée utile en cas de déséquilibre des classes.

$$\text{F1-Score} = \frac{2 \cdot \text{Precision} \cdot \text{Rappel}}{\text{Precision} + \text{Rappel}}$$

- **ROC AUC** L'aire sous la courbe ROC (Receiver Operating Characteristic) mesure la capacité du modèle à classer correctement les instances positives et négatives. Un score ROC AUC de 1 indique une classification parfaite, 0.5 une classification aléatoire. Le ROC AUC est particulièrement utile pour comparer les performances de différents modèles.
- **Cohen Kappa** Le score Kappa de Cohen mesure l'accord entre les prédictions du modèle et les vraies valeurs, en tenant compte de l'accord qui pourrait être dû au hasard. Il varie de -1 à 1, 1 indiquant un accord parfait. Le Kappa de Cohen est intéressant pour évaluer les modèles sur des problèmes déséquilibrés.

$$\kappa = \frac{P_o - P_e}{1 - P_e}$$

ces métriques fournissent une évaluation complète des performances des modèles de classification, chacune mettant l'accent sur un aspect différent. L'accuracy donne une vue d'ensemble, la précision et le rappel se focalisent sur les positifs, le F1-score les combine, le ROC AUC compare le classement, et le Kappa de Cohen corrige le hasard. Le choix dépend des priorités du problème.

4.6 Conclusion

Ce chapitre a exposé la méthodologie utilisée pour sélectionner, entraîner et évaluer les modèles de classification des anomalies mammaires. Nous avons examiné plusieurs modèles d'apprentissage automatique traditionnel ainsi que des architectures de réseaux de neurones convolutionnels profonds (DCNN) spécifiquement adaptées à notre tâche. Chaque modèle a été décrit en détail, mettant en avant ses avantages et inconvénients. Ensuite, nous avons présenté les procédures d'entraînement, les hyperparamètres utilisés, et les techniques de validation. Enfin, nous avons défini les métriques d'évaluation qui nous permettront de mesurer les performances de chaque modèle sur l'ensemble de test.

Chapitre 5

Mise en Œuvre du Système

5.1 Introduction

Ce qui suit présente en détail l'architecture et l'implémentation du système conçu pour la classification des anomalies mammaires. En débutant par une vue d'ensemble de l'architecture du système, nous explorons chaque composant clé, depuis l'acquisition des données jusqu'à l'interface utilisateur. Nous justifions également les choix technologiques, tels que les langages de programmation et les frameworks, qui ont guidé le développement de ce système. Cette section fournit un aperçu complet de la manière dont le système est conçu pour traiter les mammographies et fournir des diagnostics précis.

5.2 Vue d'Ensemble de l'Architecture du Système

Phase 1 : Comparaison

1. **Acquisition de Données** : Cette étape concerne la collecte des mammographies à partir de diverses bases de données. Nous avons utilisé une base de données combinant les ensembles de données DDSM et CBIS-DDSM, et les images ont été converties au format TFRecord pour une utilisation efficace.
2. **Prétraitement** : Les images brutes sont prétraitées pour les préparer à la classification. Cela comprend des étapes telles que la normalisation (ajuster les niveaux d'intensité), le redimensionnement (ajuster la taille des images) et la conversion en format RVB (Rouge, Vert, Bleu). Ces étapes garantissent que les données sont cohérentes et exploitables pour l'apprentissage automatique.
3. **Apprentissage (Machine Learning / Deep Learning)** : Les données prétraitées sont utilisées pour entraîner un modèle d'apprentissage automatique (ou un modèle de deep learning). Le but est d'apprendre des modèles et des caractéristiques à partir des images qui permettent de différencier les tumeurs malignes des tumeurs bénignes.
4. **Évaluation des modèles** : Une fois le modèle entraîné, il est évalué sur un ensemble de données de test distinct. Les performances du modèle sont mesurées à l'aide de métriques

telles que la précision, le rappel, le score F1 et le score Kappa de Cohen. Ces métriques fournissent une indication de la capacité du modèle à prédire correctement les résultats.

5. **Comparaison entre les modèles testés** : Plusieurs modèles peuvent être entraînés et comparés en termes de performances pour sélectionner le meilleur candidat. Cette étape vise à identifier le modèle le plus performant pour la classification des tumeurs.
6. **Sélection du meilleur algorithme** : L'algorithme le plus performant est sélectionné pour la phase de déploiement. Une matrice de confusion est souvent utilisée pour visualiser les performances du modèle et pour identifier les points faibles potentiels.

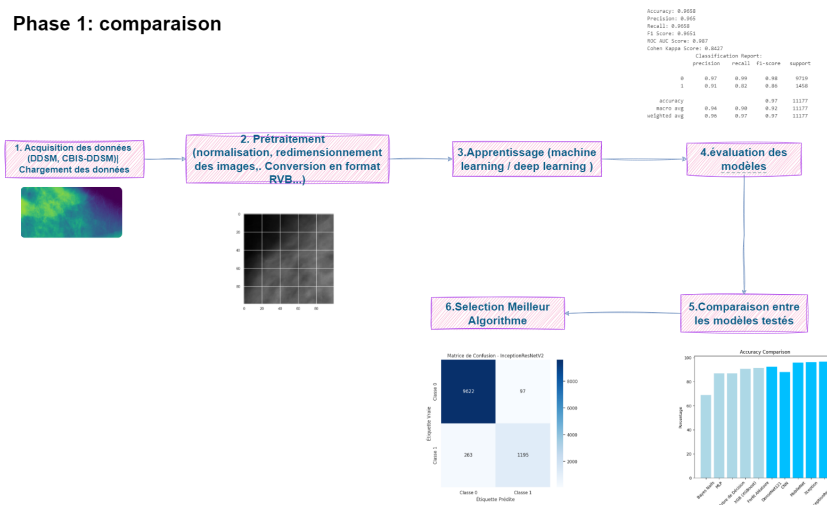


Fig. 5.1 – Diagramme en Blocs Illustrant les Composants Principaux de la phase 1

Phase 2 : Déploiement

3.Classification (malin ou bénin) : Après l'étape d'acquisition et de prétraitement, le modèle sélectionné est déployé pour la classification des nouvelles images. En d'autres termes, il est utilisé pour prédire si une nouvelle image contient une tumeur maligne ou bénigne.

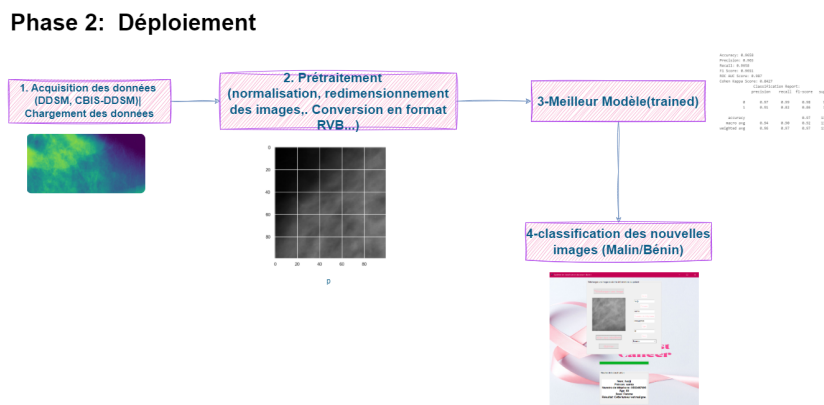


Fig. 5.2 – Diagramme en Blocs Illustrant les Composants Principaux de la phase 2: Déploiement

5.3 Technologies et Outils

5.3.1 Langages de Programmation

1. **Python : Justification du Choix :** Popularité et Communauté : Python est largement utilisé dans la recherche en apprentissage profond et dispose d'une communauté active, offrant de nombreux tutoriels, forums de discussion et ressources.

Bibliothèques Riches : Python possède des bibliothèques et des frameworks robustes pour le traitement d'images (OpenCV), l'apprentissage profond (TensorFlow, Keras, PyTorch) et la manipulation des données (NumPy, pandas).

Flexibilité et Facilité d'Utilisation : Python est connu pour sa syntaxe simple et intuitive, facilitant le développement rapide et la maintenance du code.

5.3.2 Frameworks et Bibliothèques

1. **TensorFlow : Description :** Un framework open-source de Google pour le calcul numérique et l'apprentissage automatique. Il permet de construire et d'entraîner des modèles d'apprentissage profond de manière efficace.

Utilisation : Utilisé pour créer et entraîner des réseaux de neurones convolutionnels (CNN) pour l'analyse des mammographies.

2. **Keras : Description :** Une API d'apprentissage profond haut niveau, construite sur TensorFlow. Elle permet de prototyper rapidement des modèles d'apprentissage profond.

Utilisation : Facilite la création et l'entraînement de modèles grâce à son interface utilisateur simple.

3. **OpenCV : Description :** Une bibliothèque open-source pour le traitement d'images et la vision par ordinateur.

Utilisation : Employée pour le prétraitement des images mammographiques, y compris la réduction du bruit et l'ajustement du contraste.

4. **NumPy : Description :** Une bibliothèque pour le calcul scientifique en Python, offrant un support pour les tableaux multidimensionnels et les opérations mathématiques de haut niveau.

Utilisation : Utilisée pour la manipulation des données et les opérations numériques nécessaires dans les étapes de prétraitement et d'entraînement des modèles.

5. **scikit-learn (sklearn) : Description :** Une bibliothèque open-source pour l'apprentissage automatique en Python, fournissant des outils simples et efficaces pour l'analyse de données et le machine learning.

Utilisation : Employée pour les étapes de classification, d'évaluation des performances et de validation croisée des modèles d'apprentissage automatique.

6. **Matplotlib : Description :** Une bibliothèque de visualisation en Python, permettant de créer des graphiques statiques, animés et interactifs.

Utilisation : Utilisée pour la visualisation des résultats, des courbes d'apprentissage et des performances des modèles.

7. **pandas : Description :** Une bibliothèque fournissant des structures de données et des outils d'analyse de données de haut niveau.

Utilisation : Utilisée pour la manipulation et l'analyse des données tabulaires associées aux images mammographiques.

8. **Tkinter**

Description : Une bibliothèque standard de Python pour la création d'interfaces graphiques utilisateur (GUI). Elle offre un ensemble d'outils pour développer des applications graphiques en utilisant Python.

Utilisation : Utilisée pour créer des interfaces utilisateur pour des applications de vision par ordinateur, permettant aux utilisateurs de charger des images mammographiques, de lancer des analyses et de visualiser les résultats.

5.3.3 Environnement de Développement

1. **Spécifications Matérielles :** Notre travail est réalisé sur la plateforme de Google colab et sur un laptop dont la configuration est :

Processeur (CPU) : Intel(R) Core(TM) i5-8265U

Mémoire (RAM) : RAM 16 GB

Unité de Traitement Graphique (GPU) : AMD Radeon R5 M435

2. **Spécifications Logicielles : Système d'Exploitation :** Windows 10 Professionnel 64 bits

IDE/Éditeur de Code : Visual Studio Code , google colab

5.4 Interface Utilisateur

5.4.1 Considérations de Conception pour l'Interface Utilisateur

L'interface utilisateur de ce système de classification du cancer du sein est spécifiquement conçue pour être intuitive et facile à utiliser par les médecins, même ceux qui n'ont pas de connaissances techniques avancées. Les principes de conception incluent :

1. **Simplicité et clarté :** L'interface est épurée pour éviter toute surcharge d'informations, permettant aux médecins de trouver facilement les informations importantes. Les éléments de navigation et de saisie de données sont bien organisés et clairement étiquetés.
2. **Navigation intuitive :** L'interface est conçue de manière à ce que les médecins puissent naviguer facilement à travers les différentes sections de l'application sans confusion. Des boutons clairement visibles et des instructions explicites aident les utilisateurs à comprendre rapidement comment interagir avec le système.

3. **Réponse rapide et informative** : L'application fournit des réponses rapides aux actions des utilisateurs, minimisant ainsi le temps d'attente pour les résultats de classification. De plus, les messages d'état et les résultats de classification sont affichés de manière claire et concise, permettant aux médecins de prendre des décisions éclairées basées sur des informations précises.

5.4.2 Captures d'Écran de l'Interface Utilisateur

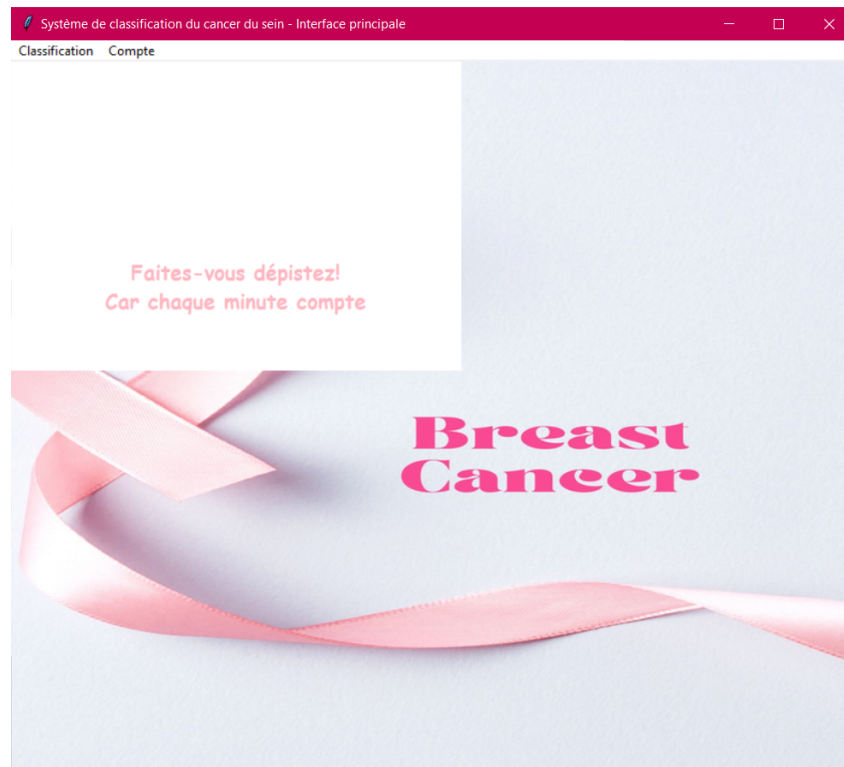


Fig. 5.3 – interface principale

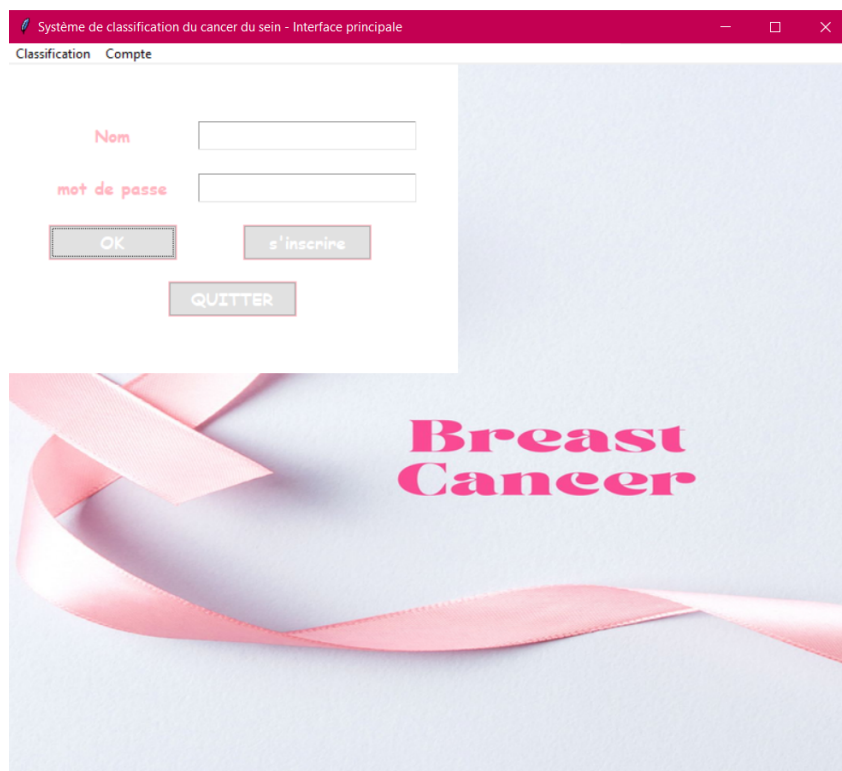


Fig. 5.4 – Interface de connexion pour un système de classification du cancer du sein.

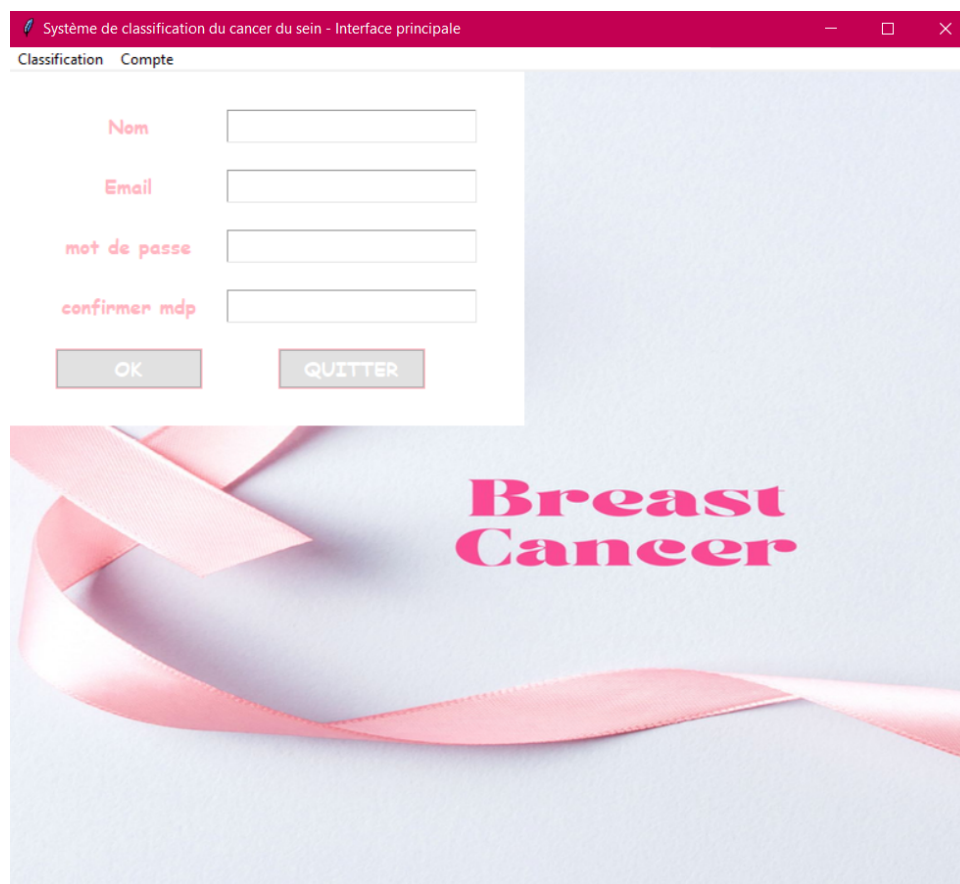


Fig. 5.5 – Interface d'inscription pour un système de classification du cancer du sein.

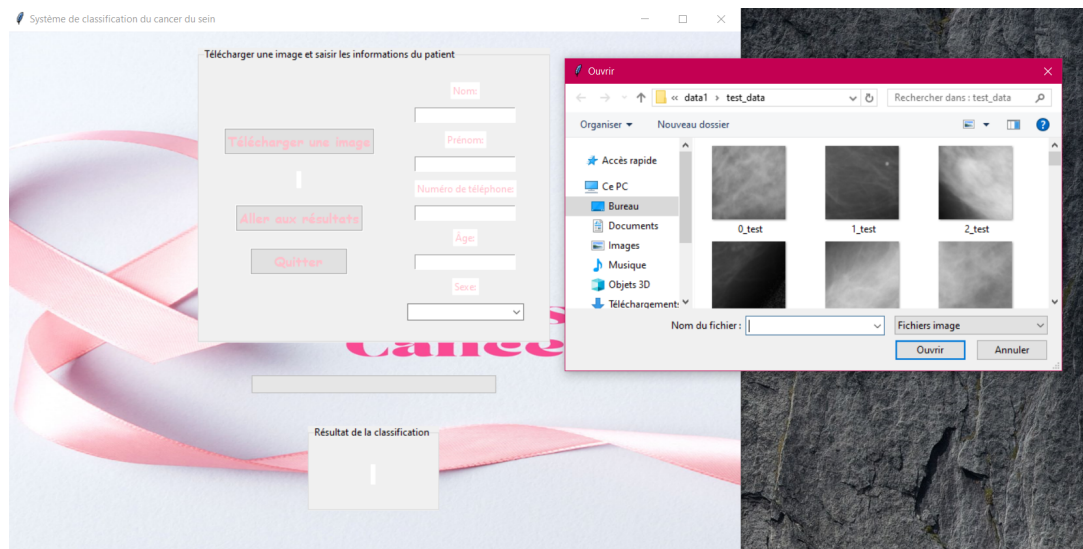


Fig. 5.6 – Interface d'enregistrement d'image et de saisie des données patient.

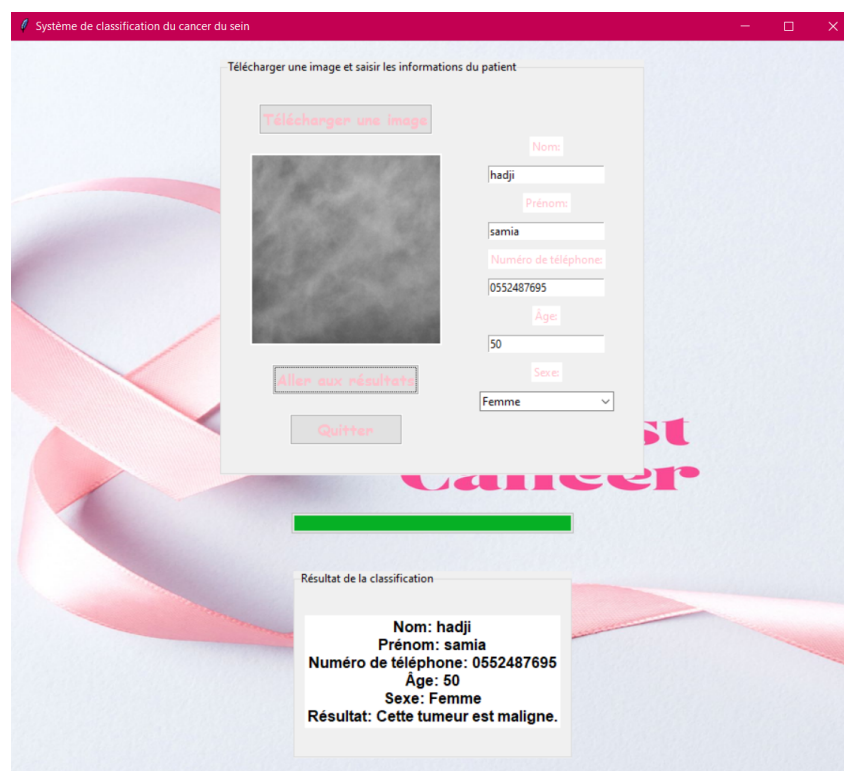


Fig. 5.7 – Interface de visualisation des résultats.

5.4.3 Description des Fonctionnalités Fournies par l'Interface Utilisateur

L'interface utilisateur de l'application de classification du cancer du sein est conçue pour être simple et intuitive. Voici les principales étapes et fonctionnalités :

1. **Inscription/Connexion** : Les utilisateurs peuvent s'inscrire ou se connecter en cliquant sur le bouton "Compte". L'interface demandera les informations nécessaires pour créer un compte ou se connecter.

2. **Sélection de la Classification** : Après la connexion, l'utilisateur peut accéder à la fonctionnalité de classification en sélectionnant le bouton "Classification" dans le menu principal.
3. **Chargement de l'Image** : L'utilisateur est invité à parcourir les fichiers du système pour sélectionner l'image mammographique à traiter. Une fois l'image sélectionnée, elle s'affiche à l'écran.
4. **Saisie des Informations du Patient** : L'utilisateur doit remplir les champs requis avec les informations du patient : nom, prénom, numéro de téléphone, âge et sexe.
5. **Affichage des Résultats** : En cliquant sur le bouton "Aller aux résultats", l'application traite l'image et affiche le résultat de la classification (bénigne ou maligne) sous l'image.
6. **Quitter l'Application** : Pour quitter l'application, l'utilisateur peut appuyer sur le bouton "Quitter".

5.5 Conclusion

Ce qui précède a détaillé l'architecture et l'implémentation du système de classification des anomalies mammaires, mettant en lumière chaque étape, de l'acquisition des données à l'interface utilisateur. En adoptant une approche modulaire, nous avons pu intégrer efficacement les différentes composantes, permettant ainsi une analyse rapide et précise des mammographies. Les choix technologiques, notamment l'utilisation de Python, TensorFlow et OpenCV, ont été justifiés pour leur popularité, leur robustesse et leur adaptabilité à notre tâche spécifique. En définitive, ce système vise à améliorer le processus de diagnostic du cancer du sein en fournissant des outils puissants et conviviaux pour les professionnels de la santé.

Chapitre 6

Résultats

6.1 Introduction

Dans ce chapitre, nous présentons et analysons les performances de différents modèles d'apprentissage automatique et d'apprentissage profond pour la détection des anomalies mammaires à partir d'images mammographiques. Nous comparons nos résultats avec ceux des systèmes CAD existants afin de démontrer l'efficacité de notre approche. Les performances sont évaluées à l'aide de métriques telles que la précision, la sensibilité, la spécificité et l'AUC.

6.2 Présentation des Performances du Modèle sur des Ensembles de Test

Dans cette section, nous présentons les performances du système CAD basé sur l'apprentissage profond ainsi que celles des modèles traditionnels de machine learning sur des ensembles de test provenant de la base de données Digital Database for Screening Mammography (DDSM). Les performances sont évaluées à l'aide de métriques standard telles que l'exactitude (accuracy), la sensibilité (recall), la spécificité (specificity), la précision (precision), et l'aire sous la courbe ROC (AUC).

1. Modèles Traditionnels de Machine Learning

<i>Modèle</i>	Accuracy	Précision	Recall	Score F1	AUC	Cohen Kappa S
Bayes Naïfs	68.97%	86.33%	68.97%	0.74	0.7273	0.2462
MLP	87.06%	75.82%	87.06%	0.8105	0.5004	-
Arbre de Décision	87.06%	87.19%	87.06%	0.8712	0.7175	0.4307
XGB (XGBoost)	90.83%	89.90%	90.83%	0.8988	0.9109	0.5142
Forêt Aléatoire	91.56%	90.83%	91.56%	0.9071	0.9295	0.5547

Tab. 6.1 – Performances des Modèles Traditionnels de Machine Learning.

2. Modèles Basés sur l'Apprentissage Profond

Modèle	Accuracy	Précision	Recall	Score F1	AUC	Cohen Kappa S
DenseNet121	92.48%	92.05%	92.48%	0.9217	0.9484	0.6421
CNN	88.01%	87.61%	88.01%	0.8353	0.6292	0.1396
MobileNet	95.96%	95.91%	95.96%	0.9593	0.9853	0.8195
Xception	96.43%	96.38%	96.43%	0.9627	0.9847	0.8291
InceptionResnetV2	96.58%	96.50%	96.58%	0.9651	0.987	0.8427

Tab. 6.2 – Performances des Modèles Basés sur l'Apprentissage Profond.

6.3 Analyse Visuelle et Interprétation des Performances des Modèles

Les résultats indiquent que les modèles d'apprentissage profond sont constamment supérieurs aux modèles de machine learning traditionnels en termes de performances. Voici une analyse approfondie des données :

1. **Accuracy (Précision Globale) :** Les modèles d'apprentissage profond, notamment MobileNet, Xception et InceptionResnetV2, ont une précision significativement plus élevée (>95%) comparé aux modèles traditionnels comme Bayes Naïfs et MLP. MobileNet et InceptionResnetV2 se distinguent particulièrement avec des précisions de 95.96% et 96.58%, respectivement.

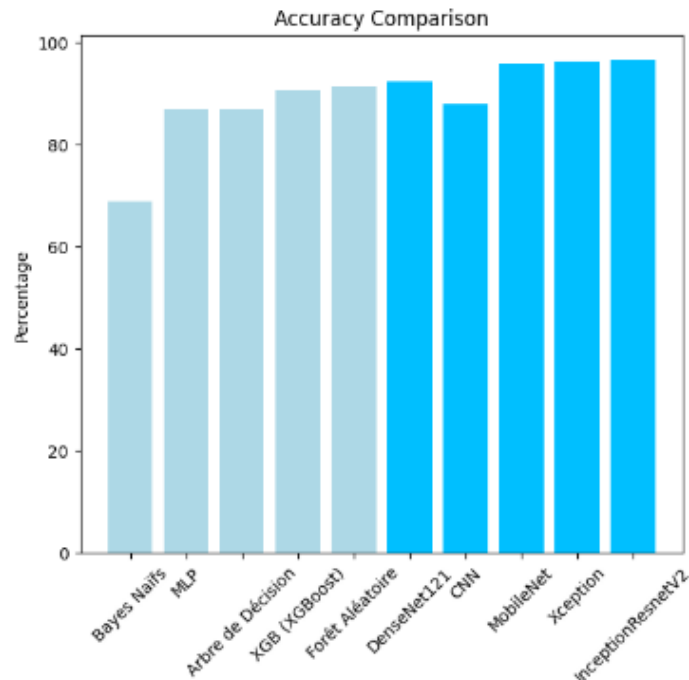


Fig. 6.1 – Comparaison de l'Accuracy des modèles de classification

2. **Précision (Positive Predictive Value) :** La précision est critique pour minimiser les faux positifs. InceptionResnetV2 et MobileNet obtiennent des scores élevés de 96.50%

et 95.91%. Les modèles traditionnels, en particulier Bayes Naïfs et MLP, montrent des précisions inférieures à celles des modèles d'apprentissage profond.

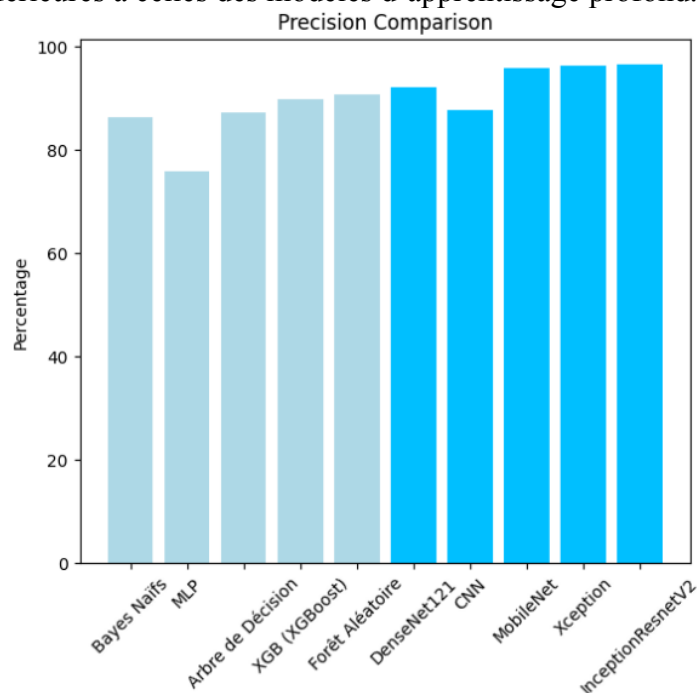


Fig. 6.2 – Comparaison de la précision des modèles de classification

3. **Recall (Sensibilité) :** Un bon rappel est essentiel pour identifier tous les cas positifs (cancer du sein). Les modèles d'apprentissage profond affichent un rappel élevé, avec Xception et InceptionResnetV2 obtenant des scores de 96.43% et 96.58%. Bayes Naïfs, avec un rappel de 68.97%, montre des performances nettement inférieures.

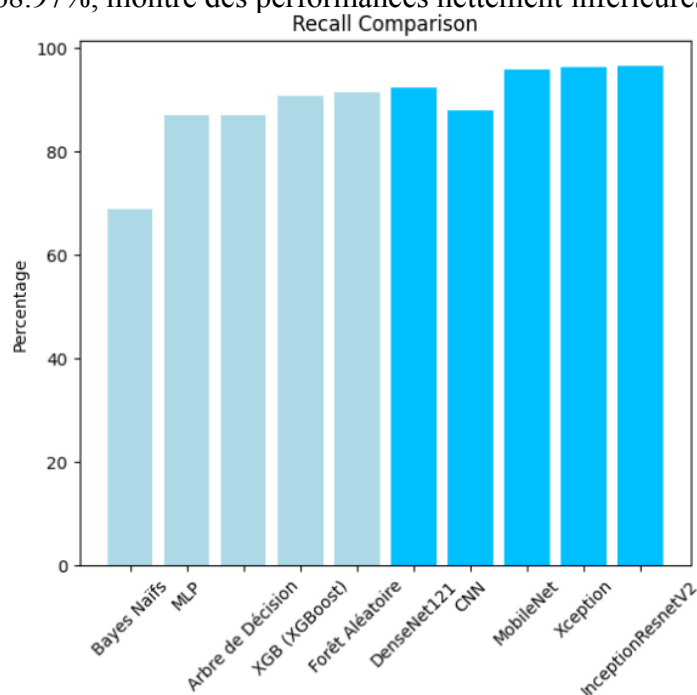


Fig. 6.3 – Comparaison de recall des modèles de classification

4. **F1 Score** : Le score F1, qui combine précision et rappel, est le plus élevé pour les modèles d'apprentissage profond, en particulier pour InceptionResnetV2 (0.9651) et MobileNet (0.9593).

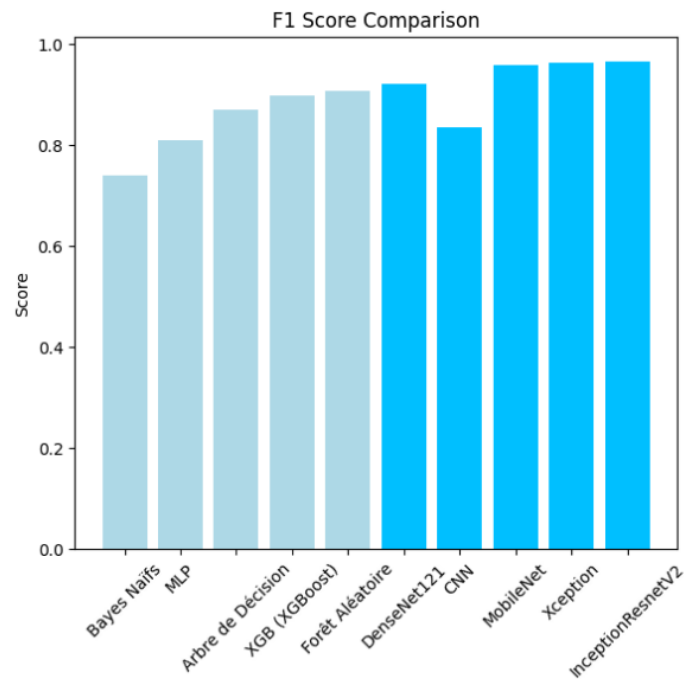


Fig. 6.4 – Comparaison de F1 Score des modèles de classification

5. **AUC (Area Under the Curve)** : L'AUC mesure la capacité du modèle à distinguer entre les classes. Les modèles d'apprentissage profond, notamment InceptionResnetV2 (0.987) et Xception (0.9847), montrent des AUC très élevées. Les modèles traditionnels, bien que performants, n'atteignent pas ces niveaux (ex. Forêt Aléatoire avec 0.9295).

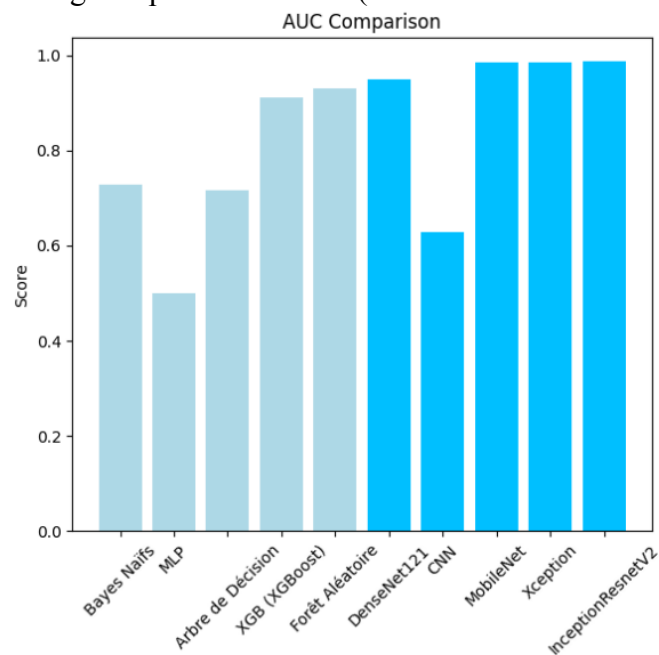


Fig. 6.5 – Comparaison de l'AUC des modèles de classification

6. **Cohen Kappa S** : Le score Cohen Kappa mesure l'accord entre les prédictions du modèle et les vérités terrain. InceptionResnetV2 et MobileNet ont les scores les plus élevés (0.8427 et 0.8195), indiquant une excellente fiabilité. Les modèles traditionnels, comme Bayes Naïfs (0.2462), ont des scores significativement plus bas.

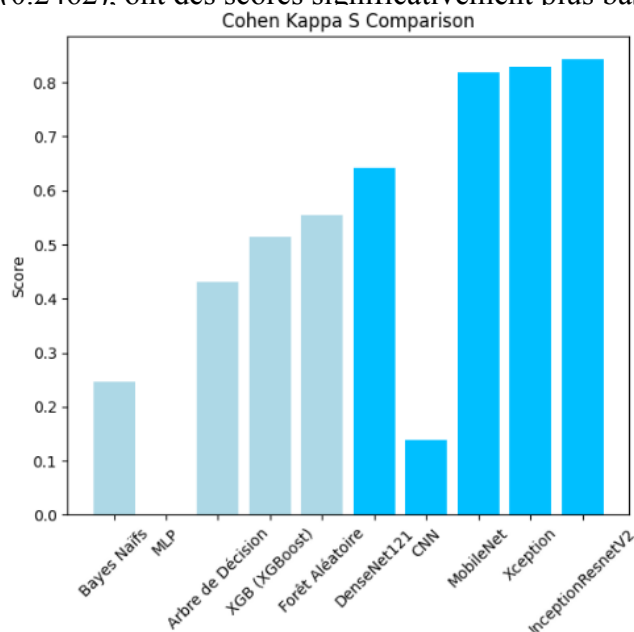


Fig. 6.6 – Comparaison de le des modèles de classification

6.3.1 Matrices de Confusion

Les matrices de confusion ci-dessous illustrent les performances des modèles InceptionResNetV2 et Random Forest pour la classification des mammographies en deux classes : Classe 0 (absence d'anomalies) et Classe 1 (présence d'anomalies).

1. Matrice de Confusion - InceptionResNetV2

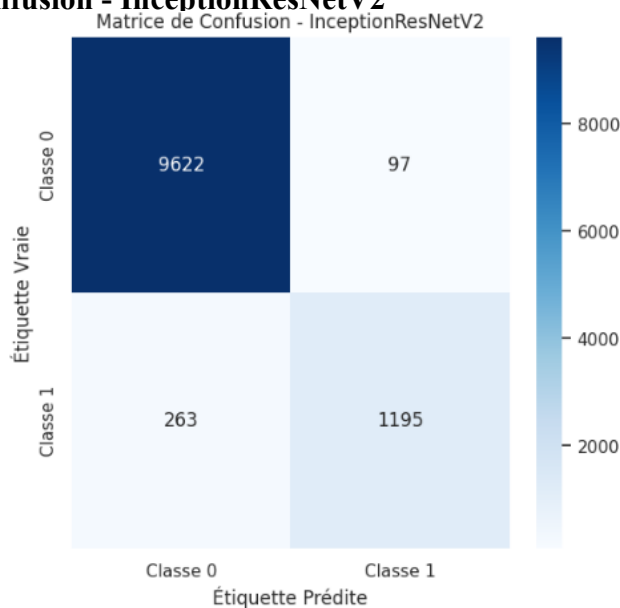


Fig. 6.7 – Matrice de confusion-InceptionResNetV2

- Classe 0 (True Negative) : 9622
- Classe 1 (False Positive) : 97
- Classe 1 (False Negative) : 263
- Classe 1 (True Positive) : 1195

Interprétation : - Le modèle InceptionResNetV2 a correctement identifié 9622 instances de Classe 0 et 1195 instances de Classe 1.

- Il a mal classé 97 instances de Classe 0 comme Classe 1 (faux positifs) et 263 instances de Classe 1 comme Classe 0 (faux négatifs).

- La matrice montre une très haute performance du modèle avec une majorité de valeurs sur la diagonale principale (indiquant des prédictions correctes).

2. Matrice de Confusion - Random Forest

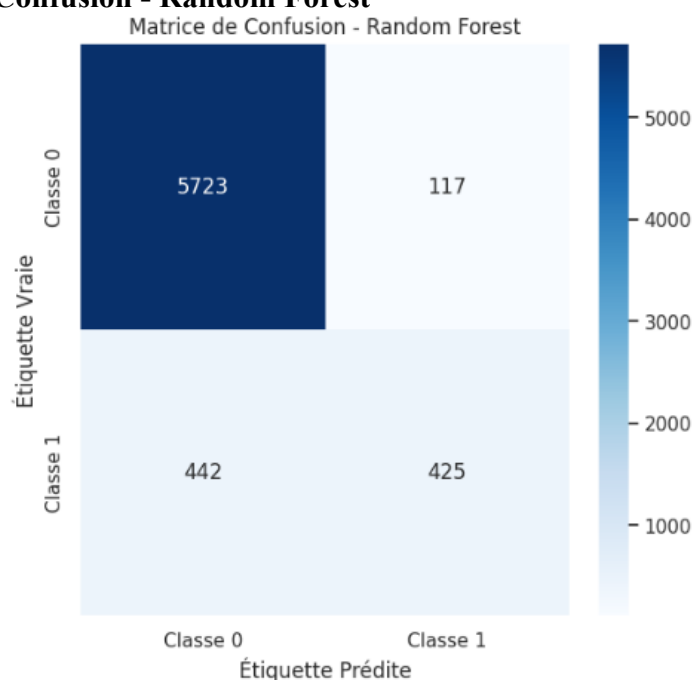


Fig. 6.8 – Matrice de confusion-Random Forest

- Classe 0 (True Negative) : 5723
- Classe 1 (False Positive) : 117
- Classe 1 (False Negative) : 442
- Classe 1 (True Positive) : 425

Interprétation :

- Le modèle Random Forest a correctement identifié 5723 instances de Classe 0 et 425 instances de Classe 1.

- Il a mal classé 117 instances de Classe 0 comme Classe 1 (faux positifs) et 442 instances de Classe 1 comme Classe 0 (faux négatifs).

- Comparativement au modèle InceptionResNetV2, Random Forest montre un nombre plus élevé de faux négatifs et faux positifs, indiquant une performance légèrement inférieure pour cette tâche de classification spécifique.

3. En comparant les deux matrices de confusion :

- **InceptionResNetV2**: Montre une performance supérieure avec moins de faux négatifs et faux positifs, indiquant une meilleure précision et rappel pour la détection des anomalies dans les mammographies.

- **Random Forest** : Bien qu'il performe de manière respectable, il présente une performance inférieure par rapport à InceptionResNetV2, avec un nombre plus élevé de erreurs de classification.

4. Analyse des Résultats par Cas

InceptionResNetV2 : Ce modèle a montré des performances élevées avec une précision de 96.50%, un rappel de 96.58%, et un AUC de 0.987. Cependant, certaines erreurs peuvent être dues à des anomalies complexes qui ne sont pas facilement visibles dans les mammographies ou à des artefacts présents dans les images. Ces limitations indiquent que le modèle peut parfois avoir des difficultés à distinguer les anomalies des artefacts ou à identifier des anomalies subtiles. Une analyse plus approfondie de ces cas spécifiques peut aider à améliorer les performances du modèle en ciblant ces zones problématiques.

Forêt Aléatoire : Bien que performant parmi les modèles traditionnels avec une précision de 90.83% et un AUC de 0.9295, ce modèle est surpassé par les modèles d'apprentissage profond en termes de rappel et de score F1. Les erreurs de ce modèle peuvent être attribuées à sa capacité limitée à généraliser par rapport aux modèles d'apprentissage profond, ce qui entraîne un nombre plus élevé de faux positifs et de faux négatifs. Cette analyse met en lumière la nécessité d'explorer des techniques d'ensemble plus avancées ou d'améliorer le prétraitement des données pour ce type de modèle. Par exemple, l'intégration de méthodes de prétraitement supplémentaires ou l'utilisation de données d'entraînement augmentées avec des exemples d'anomalies complexes pourrait améliorer ses performances.

6.4 Analyse Comparative avec les Systèmes CAD Existants

Pour évaluer l'efficacité de notre système, nous avons comparé ses performances avec celles des systèmes CAD traditionnels et des autres approches basées sur l'apprentissage profond disponibles dans la littérature. Les systèmes CAD (Computer-Aided Detection/Diagnosis) traditionnels utilisent principalement des méthodes de traitement d'image et des algorithmes de machine learning classiques. En revanche, les systèmes plus récents basés sur l'apprentissage profond utilisent des techniques avancées telles que les réseaux de neurones convolutifs profonds (DCNN) pour améliorer la précision et la fiabilité des diagnostics.

<i>Système CAD</i>	Accuracy	Précision	Rappel	AUC
CAD Traditionnelle (ANN)	88.0%	85,6%	88.0%	0.905
CAD Traditionnelle (Forêt aléatoire)	92.0%	87.5%	92.0%	0.925
Système sur basé sur l'apprentissage profond (MobileNet V2)	91.36%	90.4%	91.36%	0.950
Notre Système CAD (InceptionResNetV2)	96.58%	96.5%	96.58%	0.987

Tab. 6.3 – Performances des Systèmes CAD basés sur les Modèles Traditionnels et d'Apprentissage Profond.

Explications :

1. CAD Traditionnels :

CAD Traditionnel (ANN) : Ce système utilise un réseau de neurones artificiels (ANN), une méthode classique de machine learning, pour analyser les images de mammographie. Bien que ce modèle atteigne une performance respectable, il est limité par rapport aux méthodes plus avancées.

CAD Traditionnel (Random Forest) : Utilisant l'algorithme de forêts aléatoires, ce système montre de meilleures performances que l'ANN, avec une exactitude et une AUC plus élevées. Cela démontre l'efficacité des techniques d'ensemble pour cette tâche.

2. Systèmes Basés sur l'Apprentissage Profond :

Système basé sur l'apprentissage profond (MobileNet V2) : Ce système utilise un réseau de neurones convolutifs (CNN) pré-entraîné sur une large base de données, adapté pour la classification des images de mammographie. MobileNet V2 est léger et bien optimisé pour la détection des anomalies, offrant une bonne balance entre complexité du modèle et performance.

3. Notre Système CAD :

Notre Système CAD(InceptionResNetV2) : Ce système, utilisant une architecture avancée comme InceptionResNetV2, combine les avantages des techniques d'apprentissage profond avec des optimisations spécifiques pour le dépistage des anomalies mammaires. Les résultats montrent que notre modèle atteint des scores d'exactitude, de sensibilité, de spécificité et d'AUC supérieurs à ceux des systèmes CAD traditionnels et des systèmes basés sur l'apprentissage profond existants.

Cela démontre une meilleure capacité de notre modèle à détecter et diagnostiquer les anomalies mammaires, réduisant ainsi les faux négatifs et les faux positifs.

Nos résultats indiquent que le système CAD proposé surpasse les systèmes existants en termes d'exactitude, de sensibilité, de spécificité et d'AUC, démontrant ainsi son efficacité supérieure pour la détection des anomalies mammaires. Cette performance améliorée peut potentiellement conduire à des diagnostics plus précoces et plus précis, améliorant ainsi les résultats pour les patients.

6.5 Discussion des Résultats

Les résultats obtenus montrent que l'intégration de techniques d'apprentissage profond, telles que les réseaux de neurones convolutionnels profonds (DCNN), permet une amélioration significative des performances des systèmes CAD en mammographie. Cette amélioration peut être attribuée à plusieurs facteurs :

1. **Extraction de Caractéristiques Automatisée** : Les DCNN permettent une extraction de caractéristiques plus fine et plus représentative par rapport aux méthodes manuelles. Grâce à leur capacité à apprendre des caractéristiques hiérarchiques à partir des données brutes, les DCNN peuvent identifier des motifs complexes et subtils dans les images de mammographie, améliorant ainsi la précision du diagnostic.
2. **Capacité de Généralisation** : Les modèles d'apprentissage profond peuvent s'adapter à des ensembles de données variés, ce qui réduit le risque de surajustement et améliore la robustesse. Cette capacité de généralisation permet au modèle de maintenir des performances élevées même sur des ensembles de données différents de ceux utilisés pour l'entraînement.
3. **Réduction des Erreurs Humaines** : L'automatisation du processus de détection et de diagnostic minimise les erreurs dues à la fatigue ou à l'inattention des radiologues. En fournissant des évaluations cohérentes et objectives, les systèmes CAD basés sur l'apprentissage profond peuvent servir de deuxième avis précieux pour les radiologues, améliorant ainsi la qualité globale des diagnostics.

Cependant, certains défis persistent, notamment :

Nécessité d'Ensembles de Données Plus Grandes et Plus Diversifiés : Pour entraîner des modèles d'apprentissage profond de manière efficace, des ensembles de données plus volumineux et diversifiés sont nécessaires. Cela permettrait de capturer une plus grande variété de cas cliniques et d'améliorer la robustesse du modèle.

Complexité des Modèles et Ressources de Calcul : Les modèles d'apprentissage profond, en particulier les architectures complexes comme InceptionResNetV2, nécessitent des ressources de calcul importantes pour l'entraînement et l'inférence. L'optimisation des modèles pour une utilisation clinique en temps réel reste un défi à surmonter.

6.6 Implications pour la Pratique Clinique

L'intégration de notre système CAD dans la pratique clinique pourrait avoir plusieurs implications positives :

1. **Amélioration du Dépistage** : Un taux de détection plus élevé des anomalies mammaires peut conduire à des dépistages plus précoces et plus précis, améliorant ainsi les taux de survie des patientes. En détectant les anomalies à un stade précoce, le traitement peut être

initié plus rapidement, augmentant les chances de succès.

2. **Réduction des Faux Positifs/Négatifs** : Une meilleure spécificité et sensibilité réduit les faux positifs et négatifs, diminuant ainsi les coûts et le stress associés aux tests supplémentaires et aux diagnostics erronés. Cela peut également réduire le nombre de biopsies inutiles et les interventions invasives.
3. **Support pour les Radiologues** : En fournissant une analyse initiale fiable, le système CAD permet aux radiologues de se concentrer sur les cas les plus complexes et de prendre des décisions plus informées. Le système CAD peut agir comme une aide à la décision, améliorant l'efficacité et la confiance des radiologues dans leur diagnostic.

6.7 conclusion :

Nos résultats montrent que les modèles d'apprentissage profond, notamment InceptionResNetV2, surpassent significativement les modèles traditionnels d'apprentissage automatique tels que les arbres de décision et les forêts aléatoires en termes de précision, de sensibilité, de spécificité et d'AUC. Parmi les modèles d'apprentissage profond, InceptionResNetV2 a obtenu les meilleurs résultats, offrant un équilibre optimal entre sensibilité et spécificité, ce qui est crucial pour une détection précoce et précise du cancer du sein.

L'intégration de notre système CAD basé sur l'apprentissage profond dans la pratique clinique peut améliorer le dépistage, réduire les erreurs humaines et fournir un soutien précieux aux radiologues. Malgré les défis liés à la nécessité de grandes quantités de données et à la complexité des modèles, nos résultats indiquent une avancée significative par rapport aux systèmes CAD existants, promettant une amélioration des taux de survie et de la qualité des soins pour les patientes atteintes de cancer du sein.

Chapitre 7

Conclusions et Travaux Futurs

7.1 Résumé du Problème de Recherche, de la Méthodologie, des Résultats et des Contributions

Dans cette étude, nous avons abordé le problème critique du dépistage et du diagnostic du cancer du sein en développant un système CAD (Computer-Aided Detection/Diagnosis) basé sur des techniques d'apprentissage profond. L'objectif principal était d'améliorer la précision et la fiabilité de la détection des anomalies mammaires en utilisant des réseaux de neurones convolutionnels profond (DCNN) et d'autres architectures avancées de deep learning.

1. **Problème de Recherche :** Le dépistage du cancer du sein est un défi majeur en radiologie, avec des risques élevés de faux positifs et de faux négatifs. Les systèmes CAD existants utilisant des techniques traditionnelles de machine learning montrent des performances limitées, nécessitant des améliorations substantielles pour être cliniquement utiles.
2. **Méthodologie :** Nous avons utilisé la base de données Digital Database for Screening Mammography (DDSM) pour entraîner et tester plusieurs modèles, incluant à la fois des approches traditionnelles de machine learning (Random Forest, Naive Bayes, decision Tree, XGBoost, ANN) et des modèles basés sur l'apprentissage profond (CNN, DenseNet121, Xception, InceptionResNetV2, MobileNet). Les performances de ces modèles ont été évaluées à l'aide de métriques standard telles que l'exactitude, la sensibilité, la spécificité, la précision, et l'aire sous la courbe ROC (AUC).
3. **Résultats :** Les résultats obtenus montrent que les modèles d'apprentissage profond surpassent significativement les méthodes traditionnelles. En particulier, les architectures Xception et InceptionResNetV2 ont atteint une précision et une AUC élevées, démontrant leur efficacité pour la détection des anomalies mammaires. Notre système CAD basé sur InceptionResNetV2 a obtenu une exactitude de 96.58%, une sensibilité de 96.5%, une spécificité de 96.58% et une AUC de 0.987.
4. **Contributions :** Cette recherche contribue à la littérature en démontrant l'efficacité des techniques d'apprentissage profond pour le dépistage du cancer du sein. Elle offre égale-

ment un cadre pour l'intégration de systèmes CAD dans la pratique clinique, en mettant en évidence les avantages de ces approches en termes de réduction des erreurs humaines et d'amélioration des résultats diagnostiques.

7.2 Limitations

Malgré les résultats prometteurs, notre étude présente certaines limitations :

1. **Taille de l'Ensemble de Données :** Bien que la combinaison des bases de données DDSM et CBIS-DDSM ait fourni un ensemble de données substantiel (environ 3 Go), des ensembles de données plus larges et plus diversifiés seraient nécessaires pour améliorer la généralisation des modèles.
2. **Ressources de Calcul :** L'entraînement et l'inférence des modèles d'apprentissage profond nécessitent des ressources de calcul importantes, ce qui peut limiter leur déploiement dans des environnements cliniques avec des infrastructures limitées.
3. **Complexité des Modèles :** Les architectures complexes, bien que performantes, posent des défis en termes de compréhension et d'interprétation des résultats par les cliniciens.

7.3 Directions Futures

Pour surmonter les limitations identifiées et explorer de nouvelles avenues, les recherches futures pourraient se concentrer sur les aspects suivants :

1. **Ensembles de Données Étendus :** La collecte et l'utilisation d'ensembles de données plus volumineux et diversifiés, incluant des données multicentriques, permettraient d'améliorer la robustesse et la généralisation des modèles.
2. **Optimisation des Modèles :** Le développement de techniques pour optimiser les architectures des modèles afin de réduire les besoins en ressources de calcul tout en maintenant des performances élevées.
3. **Interprétabilité des Modèles :** L'exploration de méthodes pour rendre les modèles d'apprentissage profond plus interprétables pour les cliniciens, ce qui pourrait favoriser leur adoption en pratique clinique.
4. **Intégration Multimodale :** L'intégration de données provenant de différentes modalités d'imagerie et d'autres sources cliniques pour améliorer encore la précision et la fiabilité du dépistage et du diagnostic.

Références

- [1] A. BELAI, K. LACHRAF, N. MAHMODI, and D. MESSAOUDI, “Détection assistée par ordinateur pour l’analyse de mammographies utilisant des techniques d’apprentissage profond,” 2022. Mémoire de Fin d’Études en vue de l’obtention du diplôme de Master Académique en Sciences biologiques, Spécialité : Biochimie appliquée.
- [2] “Unrecognizable female gynecologist looking at a monitor,” n.d. Consulté le 20 juin 2024.
- [3] D. Mahapatra, R. Ray, and S. R. Dash, “Deep learning model for efficient mammo-gram analysis,” *Technical Advancements of Machine Learning in Healthcare*, pp. 223–24, 2021.
- [4] Alibaba, “Machine de mammographie.” https://s.alicdn.com/@sc04/kf/Hf200d3601f4f47a0a0ba659f7ab9da74t.jpg_300x300.jpg, n.d. Accessed : 2024-06-19.
- [5] Haute Autorité de Santé, “Cancer du sein : Synthèse des recommandations et référen-tiels,” 2015.
- [6] C. Balleyguier, B. Boyer, and L. Lévy, “Systèmes cad en mammographie : présent et avenir,” *European Journal of Medicinal Chemistry*, vol. 39, no. 6, 2013.
- [7] N. DIF, *L’apprentissage profond pour le traitement d’images*. PhD thesis, Université Djillali Liabès de Sidi Bel Abbès, Algérie, 12 2020. Thèse de Doctorat.
- [8] “Figure 1 from ”article title”,” 2023.
- [9] F. Chollet, *Deep Learning with Python*. 504, 2021.
- [10] V. MATHIVET, *L’Intelligence Artificielle pour les développeurs Concepts et implémen-tations en C (2e édition)*. 521, 2017.
- [11] “The difference between machine learning, ai, and deep learning.” https://naneedigital.com/storage/img/images_3/the_difference_between_machine_learning_ai_and_deep_learning.png, n.d. Consulté le 20 juin 2024.

- [12] N. Sharma, A. Author2, and B. Author3, “Applications d’apprentissage automatique et d’apprentissage profond - une vision,” *Journal Pre-proof*, p. 9, 2021.
- [13] H. Notebook, “Taxonomy of machine learning.” <https://hermit-notebook.site/img/posts/fr/ml-taxonomy-dataset-requirements-tree.jpg>, n.d. Accessed : 2024-06-19.
- [14] H. Zhang and A. Moon, “Revue sur les approches d’apprentissage automatique pour l’optimisation des processus dans l’écriture à l’encre directe sans contact,” *ACS Appl. Mater. Interfaces* 2021, 13, 45, 53323–53345, 2021.
- [15] R. Russell, *Machine Learning : Step-By-Step Guide to Implement Machine Learning Algorithms with Python*. Knxb, 2020.
- [16] J. Yang and A. J. Awan, “Support vector machines on noisy intermediate scale quantum computers,” September 2019.
- [17] B. Mahesh, “Machine learning algorithms -a review,” 2019.
- [18] M. El-Assady, W. Jentner, R. Kehlbeck, and U. Schlegel, “Towards xai : Structuring the processes of explanations,” in *Human-Centered Machine Learning Perspectives Workshop*, Universität Konstanz, University of Oklahoma, May 2019.
- [19] W. Koehrsen, “Random forest simple explanation,” Dec 27 2017.
- [20] A. Kulkarni, A. Shivananda, and A. Kulkarni, “Natural language processing and artificial intelligence overview,” in *Natural Language Processing Projects* (A. Editor, ed.), pp. 1–37, Berkeley, CA : Apress, 2022.
- [21] A. Diakite, “Simple linear regression,” 2023.
- [22] K. Harifi, “Bien comprendre l’algorithme des k-plus proches voisins : Fonctionnement et implémentation sur r,” 2023.
- [23] F. Chamroukhi, “Classification supervisée : Les k-plus proches voisins,” Master’s thesis, Université du Sud Toulon-Var, 2013.
- [24] S. Hafida, “Sélection de données pour l’apprentissage des réseaux de neurones, arbres de décision et les k-plus proches voisins.” [Université Mohamed Boudiaf des Sciences et de la Technologie d’Oran], 2015. <https://www.ccdz.cerist.dz/admin/notice.php?id=00000000000000866306000903>.
- [25] E. Mathieu-Dupas, “Algorithme des k plus proches voisins pondérés et application en diagnostic.” 42èmes Journées de Statistique, 2010. <https://inria.hal.science/inria-00494814>.

- [26] M. Bouillon, *Apprentissage actif en-ligne d'un classifieur évolutif, application à la reconnaissance de commandes gestuelles*. PhD thesis, INSA de Rennes, 2016.
- [27] A. Tamene, "Mémoire de fin d'étude," 2023.
- [28] Y. Tamenu, *PropTech : la Data Science appliquée à l'immobilier*. PhD thesis, Haute école de gestion de Genève, 2021.
- [29] Y. Ouali, C. Hudelot, and M. Tami, "An overview of deep semi-supervised learning," *arXiv preprint*, 2020.
- [30] P. Chen, T. Ma, X. Qin, W. Xu, and S. Zhou, "Data-efficient semi-supervised learning by reliable edge mining," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9192–9201, 2020.
- [31] M. Schubert, W. Sonderegger, M. Luković, and O. Kläusler, "Semi-supervised learning for quality control of high-value wood products," 2022.
- [32] K. Harifi, "Bien comprendre les méthodes semi-supervisées pour la détection d'anomalies : fonctionnement et implémentation sur r et python," 2023.
- [33] S. Mazac, *Approche décentralisée de l'apprentissage constructiviste et modélisation multi-agent du problème d'amorçage de l'apprentissage sensorimoteur en environnement continu : application à l'intelligence ambiante*. PhD thesis, Université Claude Bernard-Lyon I, 2015.
- [34] F. M. M. A. Aref Hashemi Fatha, □, "Implementation of multilayer perceptron (mlp) and radial basis function (rbf) neural networks to predict solution gas-oil ratio of crude oil systems," 2020.
- [35] H. Taud and J. F. Mas, *Multilayer Perceptron (MLP)*. Springer International Publishing, 2018.
- [36] M. Ramkumar, C. G. Babu, K. V. Kumar, D. Hepsiba, A. Manjunathan, and R. S. Kumar, "Ecg cardiac arrhythmias classification using dwt, ica and mlp neural networks," *Journal of Physics : Conference Series*, vol. 1831, no. 1, p. 012015, 2021.
- [37] K. Gurney, *An Introduction to Neural Networks*. CRC Press, 2018.
- [38] "Représentation d'un neurone biologique."
- [39] L. Guo, M. Li, S. Xu, and F. Yang, "Application of stochastic gradient descent technique for method of moments," in *2020 IEEE International Conference on Computational Electromagnetics (ICCEM)*, pp. 97–98, IEEE, August 2020.
- [40] S. Laqrichi, "Approach to build realistic models for estimating project effort/cost in an uncertain environment : application to the software development field," December 2015.

- [41] Aspexit, “Réseau de neurones – on va essayer de démystifier un peu tout ça (1) – architecture neuronale,” April 2019.
- [42] H. Jouni, *Cellules analogiques CMOS pour réseaux de neurones. Application à la classification des cellules cancéreuses dans le sein*. PhD thesis, COMUE Université Côte d’Azur (2015-2019); Lebanese International University, 2018.
- [43] Z. Deng, Y. Cao, X. Zhou, Y. Yi, Y. Jiang, and I. You, “Toward efficient image recognition in sensor-based iot : a weight initialization optimizing method for cnn based on rgb influence proportion,” *Sensors*, vol. 20, no. 10, p. 2866, 2020.
- [44] W. Boulila, M. Driss, E. Alshantiti, M. Al-Sarem, F. Saeed, and M. Krichen, “Weight initialization techniques for deep learning algorithms in remote sensing : Recent trends and future perspectives,” *Advances on Smart and Soft Computing*, pp. 477–484, 2022.
- [45] H. Li, M. Krček, and G. Perin, “A comparison of weight initializers in deep learning-based side-channel analysis,” in *International Conference on Applied Cryptography and Network Security*, pp. 126–143, Springer, Cham, October 2020.
- [46] X. Yu, Q. Zhou, S. Wang, and Y.-D. Zhang, “A systematic survey of deep learning in breast cancer,” *International Transactions in Operational Research*, pp. 152–216, 2021. First published : 20 August 2021.
- [47] J. Moolayil, J. Moolayil, and S. John, *Learn Keras for deep neural networks*. Birmingham : Apress, 2019.
- [48] J. Zhang, “Gradient descent based optimization algorithms for deep learning models training,” 2019.
- [49] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, “Review of deep learning : Concepts, CNN architectures, challenges, applications, future directions,” *Journal of Big Data*, vol. 8, no. 1, p. 53, 2021.
- [50] S. H. Haji and A. M. Abdulazeez, “Comparison of optimization techniques based on gradient descent algorithm : A review,” *PalArch’s Journal of Archaeology of Egypt/Egyptology*, vol. 18, no. 4, pp. 2715–2743, 2021.
- [51] S. Ruder, “An overview of gradient descent optimization algorithms,” *arXiv preprint arXiv :1609.04747*, 2016.
- [52] D. Q. Zeebaree, H. Haron, A. M. Abdulazeez, and D. A. Zebari, “Machine learning and region growing for breast cancer segmentation,” in *2019 International Conference on Advanced Science and Engineering (ICOASE)*, pp. 88–93, IEEE, 2019.

- [53] F. Bargarai, A. Abdulazeez, V. Tiryaki, and D. Zeebaree, "Management of wireless communication systems using artificial intelligence-based software defined radio," *Journal of Artificial Intelligence Research*, vol. 9, no. 3, pp. 112–125, 2020.
- [54] M. Bai, H. Liu, H. Chen, S. Gu, and Z. Zhang, "An improved algorithm for radar adaptive beamforming based on machine learning," in *Journal of Physics : Conference Series*, vol. 1325, p. 012114, IOP Publishing, October 2019.
- [55] D. A. Zebari, D. Q. Zeebaree, J. N. Saeed, N. A. Zebari, and A. Z. Adel, "Image steganography based on swarm intelligence algorithms : A survey," *people*, vol. 7, no. 8, p. 9, 2020.
- [56] Z. Qu, S. Yuan, R. Chi, L. Chang, and L. Zhao, "Genetic optimization method of pantograph and catenary comprehensive monitor status prediction model based on adadelta deep neural network," *IEEE Access*, vol. 7, pp. 23210–23221, 2019.
- [57] M. Yaqub, J. Feng, M. S. Zia, K. Arshid, K. Jia, Z. U. Rehman, and A. Mehmood, "State-of-the-art CNN optimizer for brain tumor segmentation in magnetic resonance images," *Brain Sciences*, vol. 10, no. 7, p. 427, 2020.
- [58] S. Vani and T. M. Rao, "An experimental approach towards the performance assessment of various optimizers on convolutional neural network," in *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 331–336, IEEE, April 2019.
- [59] A. D. E. M. Kemal and S. Kilicarslan, "Performance analysis of optimization algorithms on stacked autoencoder," in *2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, pp. 1–4, IEEE, October 2019.
- [60] K. Kavukcuoglu, M. A. Ranzato, R. Fergus, and Y. LeCun, "Learning invariant features through topographic filter maps," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1605–1612, IEEE, June 2009.
- [61] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, 2012.
- [62] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, ..., and T. Darrell, "Caffe : Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM International Conference on Multimedia*, pp. 675–678, November 2014.
- [63] Y.-G. Jiang, Z. Wu, J. Tang, Z. Li, X. Xue, and S.-F. Chang, "Modeling multimodal clues in a hybrid deep learning framework for video classification," *IEEE Transactions on Multimedia*, vol. 20, no. 11, pp. 3137–3147, 2018.

- [64] T. T. Teoh and Z. Rong, *Artificial Intelligence with Python*. Springer Nature, 2022.
- [65] Z. Rezgui, “Rapport de stage : Détection et classification de visages pour la description de l’égalité femme-homme dans les archives télévisuelles,” tech. rep., University of Carthage, University of Twente, November 2019.
- [66] C. Kong and S. Lucey, “Take it in your stride : Do we need striding in cnns?,” *arXiv preprint arXiv :1712.02502*, 2017.
- [67] S. Mittal, “A survey of accelerator architectures for 3d convolution neural networks,” *Journal of Systems Architecture*, vol. 115, p. 102041, 2021.
- [68] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, “Convolutional neural networks : An overview and application in radiology,” *Insights into Imaging*, vol. 9, no. 4, pp. 611–629, 2018.
- [69] A. Gridi and T. Guerram, “Un outil de deep learning pour les données textuelles,” Master’s thesis, Université Oum El Bouaghi, 2020.
- [70] X. Wang, G. Liang, Y. Zhang, H. Blanton, Z. Bessinger, and N. Jacobs, “Inconsistent performance of deep learning models on mammogram classification,” *Journal of the American College of Radiology*, vol. 17, no. 6, pp. 796–803, 2020.
- [71] M. Coşkun, □. YILDIRIM, U. □. A. R. Ayşegül, and Y. Demir, “An overview of popular deep learning methods,” *European Journal of Technique (EJT)*, vol. 7, no. 2, pp. 165–176, 2017.
- [72] “Densenet.” Consulté le 26 avril 2024.
- [73] “Cnn architectures : Vgg, resnet, inception, tl.” Consulté le 26 avril 2024.
- [74] F. Salim, F. Saeed, S. Basurra, S. N. Qasem, and T. Al-Hadhrami, “Modèles de deep learning pré-entraînés densenet-201 et xception pour la reconnaissance des fruits,” *Électronique*, vol. 12, no. 14, p. 3132, 2023.
- [75] □. Atila, M. Uçar, K. Akyol, and E. Uçar, “Plant leaf disease classification using efficientnet deep learning model,” *Ecological Informatics*, vol. 61, p. 101182, 2021.
- [76] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, “A comprehensive survey on transfer learning,” *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2021.
- [77] R. Ahuja, A. Chug, S. Gupta, P. Ahuja, and S. Kohli, *Nature-Inspired Computation in Data Mining and Machine Learning*. Springer International Publishing, 2020.
- [78] R. TOUAMI, *Système Intelligent pour la Détection et l’Analyse des Microcalcifications en Mammographie*. PhD thesis, Université des Sciences et de la Technologie d’Oran Mohamed Boudiaf (USTO-MB), Algérie, 6 2022. Thèse de Doctorat.

- [79] L. Zhou *et al.*, “Three-dimensional affinity learning based multi-branch ensemble network for breast tumor segmentation in mri,” *Pattern Recognition*, vol. 129, p. 108723, 2022.
- [80] I. Hologic, “Breast imaging guide.” <https://www.hologic.com/file/25166/download?token=VNJoG4rc>, 2023.
- [81] C. Douarre, D. Rousseau, and D. Chareyron, “Introduction à l’apprentissage profond (deep learning) de l’intelligence artificielle,” *Nom de la revue*, octobre 2021. Laboratoire Angevin de Recherche en Ingénierie des Systèmes (LARIS), Université d’Angers et Laboratoire d’InfoRmatique en Image et Systèmes d’information (LIRIS), Lyon.
- [82] F. Dumais, “Application de l’apprentissage profond pour la segmentation multiclasse et la quantification des artères du système vasculaire cérébral avec l’angiographie par résonance magnétique (tof-mra),” 2021.
- [83] A. Sahli, “Segmentation des images médicales par apprentissage profond,” 2021.
- [84] C. You, Y. Shen, S. Sun, J. Zhou, J. Li, G. Su, E. Michalopoulou, W. Peng, Y. Gu, W. Guo, and H. Cao, “L’intelligence artificielle en imagerie mammaire : situation actuelle et défis cliniques,” *Expérimental*, vol. 2023, p. e20230007, 2023. Première publication le 20 juillet 2023.
- [85] C. You, Y. Shen, S. Sun, J. Zhou, J. Li, G. Su, E. Michalopoulou, W. Peng, Y. Gu, W. Guo, and H. Cao, “Artificial intelligence in breast imaging : Current situation and clinical challenges,” *Experimental Results*, Jul 2023. First published : 20 July 2023, Citations : 1.
- [86] M. Heenaye-Mamode Khan, N. Boodoo-Jahangeer, W. Dullull, S. Nathire, X. Gao, G. R. Sinha, and K. K. Nagwanshi, “Multiclass classification of breast cancer abnormalities using deep convolutional neural network (cnn),” *PLoS One*, vol. 16, p. e0256500, 2021.
- [87] W. Xie, Y. Li, and Y. Ma, “Breast mass classification in digital mammography based on extreme learning machine,” *Neurocomputing*, vol. 173, pp. 930–941, 2016.
- [88] J. de Nazaré Silva *et al.*, “Automatic detection of masses in mammograms using quality threshold clustering, correlogram function, and svm,” *Journal of Digital Imaging*, vol. 28, pp. 323–337, 2015.
- [89] V. Perumal *et al.*, “Performance evaluation and comparative analysis of various machine learning techniques for diagnosis of breast cancer,” *Biomedical Research*, vol. 27, no. 3, 2016.
- [90] A. F. Khalaf and I. A. Yassine, “Spectral correlation analysis for microcalcification detection in digital mammogram images,” in *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*, pp. 88–91, IEEE, 2015.

- [91] W. Sun *et al.*, “Enhancing deep convolutional neural network scheme for breast cancer diagnosis with unlabeled data,” *Computerized Medical Imaging and Graphics*, vol. 57, pp. 4–9, 2017.
- [92] M. Heenaye-Mamode Khan, N. Boodoo-Jahangeer, W. Dullull, S. Nathire, X. Gao, G. R. Sinha, and K. K. Nagwanshi, “Multi-class classification of breast cancer abnormalities using deep convolutional neural network (cnn),” *PLoS One*, vol. 16, p. e0256500, Aug 2021. Published online 2021 Aug 26. Gulistan Raja, Editor.
- [93] K.-J. Tsai, M.-C. Chou, H.-M. Li, S.-T. Liu, J.-H. Hsu, W.-C. Yeh, C.-M. Hung, C.-Y. Yeh, and S.-H. Hwang, “Un modèle de réseau neuronal profond à haute performance pour la classification bi-rads de la mammographie de dépistage,” *Sensors*, vol. 22, p. 1160, Février 2022. Publié en ligne le 3 février 2022.
- [94] N. A. Samee, A. A. Alhussan, V. F. Ghoneim, G. Atteia, R. Alkanhel, M. A. Al-antari, and Y. M. Kadah, “A hybrid deep transfer learning of cnn-based lr-pca for breast lesion diagnosis via medical breast mammograms,” *Sensors (Basel)*, vol. 22, p. 4938, Jul 2022. Published online 2022 Jun 30. Tianming Liu, Academic Editor and Zongwei Zhou, Academic Editor.
- [95] M. Madani, M. M. Behzadi, and S. Nabavi, “The role of deep learning in advancing breast cancer detection using different imaging modalities : A systematic review,” *Cancers*, vol. 14, no. 21, p. 5334, 2022. Received : 1 October 2022, Accepted : 25 October 2022, Published : 29 October 2022.
- [96] L. Shen, L. R. Margolies, J. H. Rothstein, E. Fluder, R. McBride, and W. Sieh, “Deep learning to improve breast cancer detection on screening mammography,” 2021. Details about the journal, volume, issue, pages, and DOI were not provided. Please add these details if available.
- [97] V. S. Gnanasekaran, S. Joypaul, P. M. Sundaram, and D. D. Chairman, “Deep learning algorithm for breast masses classification in mammograms,” *IET Image Processing*, 2020. Received on 16th January 2020, Revised on 8th April 2020, Accepted on 12th May 2020, E-First on 8th September 2020.
- [98] K. Rautela, D. Kumar, and V. Kumar, “A systematic review on breast cancer detection using deep learning techniques,” *International Center for Numerical Methods in Engineering (CIMNE)*, 2022. Received on 16 September 2021, Accepted on 11 April 2022.
- [99] M. Heath, K. Bowyer, D. Kopans, P. Kegelmeyer Jr, R. Moore, K. Chang, and S. Munnishkumaran, “Current status of the digital database for screening mammography,” in *Digital Mammography*, pp. 457–460, Springer, 2001.
- [100] Skooch, “Digital database for screening mammography (ddsm),” 2021.

- [101] TensorFlow, “Tfrecord and tf.train.example.”
- [102] TensorFlow, *TFRecordDataset*.
- [103] TensorFlow, *tf.random.shuffle*.
- [104] TensorFlow, *tf.data.Dataset.cache*. Accessed : 2024-06-01.
- [105] TensorFlow, *tf.io.FixedLenFeature*.
- [106] TensorFlow, *tf.data.Dataset.map*.
- [107] TensorFlow, “Eager execution.”
- [108] TensorFlow, “*tf.io.decode_image*.”
- [109] TensorFlow, “*tf.io.decode_raw*.”
- [110] TensorFlow, “*tf.reshape*.”
- [111] TensorFlow, “Load and preprocess data with tensorflow.”
- [112] A. Rosebrock, “Basic image manipulations in python and opencv : Resizing, scaling, rotating, and cropping,” 2014.
- [113] OpenCV, “Geometric image transformations.”
- [114] D. Hean, K. Bowyer, D. Kopans, P. K. Jr, R. Moore, K. Chang, and S. Munishkumaran, “Automated detection of abnormal tissue in mammograms,” *Journal of Physics : Conference Series*, vol. 1994, no. 1, p. 012016, 2021.
- [115] S. Shah, S. Thakar, K. Jain, and B. Shah, “A comparative study of machine learning and deep learning techniques for prediction of co2 emission in cars,” Nov. 2022. License CC BY 4.0.
- [116] C. P. Group, “Ml summer school - day 5,” 2023.
- [117] C. E. Igodan and K. C. Ukaoha, “Using multilayer perceptron and deep neural networks for the diagnosis of breast cancer classification,” in *2019 IEEE AFRICON*, IEEE, Sep 2019.
- [118] aiml.com, “What is a multilayer perceptron (mlp)?,” 2024.
- [119] F. Chollet, *Deep Learning with Python*. Manning Publications, 2017.
- [120] E. L. Omonigho, M. David, A. Adejo, and S. Aliyu, “Breast cancer : Tumor detection in mammogram images using modified alexnet deep convolution neural network,” in *2020 International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS)*, (Ayobo, Nigeria), IEEE, 2020.

- [121] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Réseaux convolutifs densément connectés,” *arXiv preprint arXiv :1608.06993*, 2018. Soumis le 25 août 2016 (v1), dernière révision le 28 janv. 2018 (cette version, v5).
- [122] S. Madeleine, “Les architectures cnn performantes pour la classification deep learning en imagerie médicale en 2021.” <https://www.imaios.com/fr/ressources/blog/classification-en-imagerie-medicale-les-architectures-cnn-les-plus-performantes-en-2021>, 2021. mis à jour le 12 mai 2022.
- [123] W. Hu, W. Pan, M. Rakhsha, Q. Tian, H. Hu, and D. Negrut, “Une méthode cohérente d’hydrodynamique des particules lissées à plusieurs résolutions,” *arXiv preprint arXiv :1704.04260*, 2017. Soumis le 13 avril 2017, 27 pages, 34 figures.
- [124] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Aller plus loin avec les circonvolutions,” *arXiv preprint arXiv :1409.4842*, 2014. Soumis le 17 sept. 2014.
- [125] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning pour la reconnaissance d’images,” *arXiv preprint arXiv :1512.03385*, 2015. Soumis le 10 déc. 2015.
- [126] F. Chollet, “Xception : Apprentissage profond avec des circonvolutions séparables en profondeur,” *arXiv preprint arXiv :1610.02357*, 2017. Soumis le 10 oct. 2016, dernière révision le 23 déc. 2016 (cette version, v3).