RAPPORT DE PROJET H-419
CLASSIFICATION DES IMAGES DE CANCER DU SEIN

Etudiant: **MEFENYA TAKOUMBO Romeo Joel**;
Enseignant: **DEBEIR Olivier**

Table des matières

1 Résumé

Ce rapport présente un projet de classification supervisée d'images de cancer du sein. L'objectif est de distinguer les tumeurs bénignes des tumeurs malignes et de détecter le cancer du sein à un stade précoce pour améliorer les chances de guérison et de survie des patients. Nous avons utilisé un CNN pour construire notre modèle et avons évalué ses performances en termes de précision, rappel et f1-score pour les classes 'benin', 'malignant' et 'normal', avec une précision globale de 0,81. Cependant, notre modèle a eu des difficultés à distinguer les tumeurs bénignes des tumeurs malignes, ce qui est une préoccupation majeure dans le domaine médical. Pour améliorer les performances de notre modèle, il serait intéressant d'avoir une plus grande base de données pour améliorer la qualité des données et de recueillir davantage de données de qualité pour aider le modèle à mieux distinguer les tumeurs bénignes des tumeurs malignes. Enfin, une étude clinique plus approfondie serait nécessaire pour valider l'utilisation de ce modèle dans un environnement de soins de santé réel

2 Introduction

La classification est un processus d'attribution d'étiquettes ou de catégories à des données en fonction de caractéristiques communes ou de similarités. La classification peut être supervisée ou non supervisée : La classification supervisée implique l'utilisation de données d'entraînement étiquetées pour prédire l'étiquette d'une nouvelle donnée. La classification non supervisée, en revanche, ne suppose pas de données étiquetées et cherche plutôt à trouver des motifs ou des structures cachés dans les données non étiquetées L'importance de la classification supervisée et non supervisée dépend du problème que l'on cherche à résoudre. Si nous disposons de données étiquetées et que nous voulons prédire de nouvelles étiquettes, la classification supervisée est la méthode la plus appropriée. Si nous ne disposons pas de données étiquetées et que nous voulons découvrir des structures cachées dans les données, la classification non supervisée est plus appropriée.

Pour notre projet, nous avons à faire à une classification supervisée. La classification d'images de cancer du sein est d'une grande importance car elle peut aider à détecter et à diagnostiquer le cancer du sein à un stade précoce, ce qui peut améliorer les chances de guérison et de survie des patients. La classification d'images permet également de distinguer les tumeurs bénignes des tumeurs malignes, ce qui peut aider les médecins à planifier le traitement approprié pour chaque patient.

Ceci est le but de ce projet.

3 Réseau

Pour faire cela, nous avons utilisés le Convolutional Neural Network (CNN) de tensorflow qui est actuellement l'un des meilleurs réseaux de neurone de classification. C'est une architecture de réseau de neurones profonds qui a été spécifiquement conçue pour la reconnaissance d'images. Les réseaux de neurones traditionnels sont souvent inadaptés à la tâche de reconnaissance d'images, car ils ne sont pas capables de détecter les motifs spatiaux dans les don-

nées d'image. Le CNN utilise une opération appelée "convolution" pour extraire des caractéristiques importantes des images en les parcourant avec des filtres (kernels) qui détectent les motifs locaux

3.1 Architecture d'un CNN

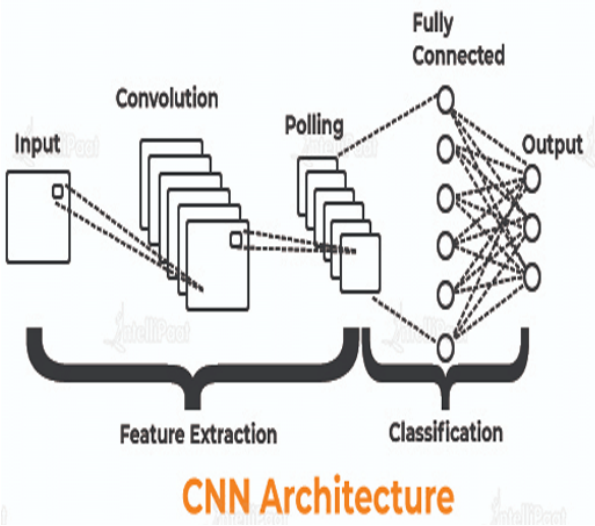

FIGURE 1 – Architecture CNN. Intellipaat. (2021). Convolutional Neural Network (CNN) Tutorial

3.1.1 La Convolution

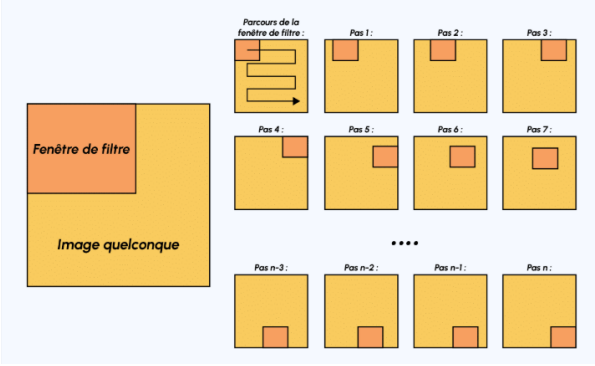

FIGURE 2 – Convolution. T. Surya. (2020)

La convolution est une opération mathématique simple généralement utilisée pour le traitement et la reconnaissance d'images. Sur une image, son effet s'assimile à un filtrage dont voici le fonctionnement

- Dans un premier temps, on définit la taille de la fenêtre de filtre située en haut à gauche.
- La fenêtre de filtre, représentant la feature, se déplace progressivement de la gauche vers la droite d'un certain nombre de cases défini au préalable (le pas) jusqu'à arriver au bout de l'image
- À chaque portion d'image rencontrée, un calcul de convolution s'effectue permettant d'obtenir en sortie une carte d'activation ou feature map qui indique où est localisées les features dans l'image : plus la feature map est élevée, plus la portion de l'image balayée ressemble à la feature.

3.1.2 Le Max-Pooling

Le Max-Pooling est un processus de discrétisation basé sur des échantillons. Son objectif est de sous-échantillonner une représentation d'entrée (image, matrice de sortie de couche cachée, etc.) en réduisant sa dimension. De plus, son intérêt est qu'il réduit le coût de calcul en réduisant le nombre de paramètres à apprendre et fournit une invariance par petites translations (si une petite translation ne modifie pas le maximum de la région balayée, le maximum de chaque région restera le même et donc la nouvelle matrice créée restera identique)

3.1.3 Fully connected

Couches entièrement connectées (fully connected) : les couches entièrement connectées sont généralement utilisées à la fin du réseau pour effectuer la classification finale. Elles prennent les caractéristiques apprises par les couches précédentes et les utilisent pour prédire les classes d'images.

4 Données

Notre projet ainsi que nos données ont été pris sur Kaggle et le lien du vers le code est annexé, (https ://www.kaggle.com/datasets/aryashah2k/breast-ultrasound-images-dataset.) qui est une plateforme en ligne populaire pour les scientifiques des données et les praticiens de l'apprentissage automatique.

5 Programmation

5.1 Téléchargement et exploitation des données

Comme mentionné plus haut, nos données viennent de Kaggle. Il est entièrement gratuit et fournit un accès gratuit à des ressources de calcul haute performance, y compris des GPU et des TPU. Il propose également un environnement de développement qui est utilisé pour la programmation de plusieurs langages (mais plus utilisé en Python et R). Kaggle est plus adapté aux projets de données en communauté et pour accéder à des ensembles de données publics :

- Accès à des ressources de calcul puissantes sans avoir à acheter ou à configurer son propre matériel.
- Sauvegarde automatique du code et des résultats
- Partage facile de fichiers et de notebooks avec d'autres utilisateurs.
- Grande puissance de calcul.

5.2 Prétraitement des données

Le prétraitement des données est une étape cruciale pour assurer la qualité des données d'entraînement et la performance des modèles d'apprentissage automatique. Il permet de nettoyer les données, de les normaliser, de les redimensionner, de les réduire et de les transformer, ce qui facilite l'entraînement des modèles d'apprentissage automatique et améliore la qualité des prédictions sur de nouvelles données. D'une part, nous avons effectué une normalisation redimensionnement de nos données :

Désignation	**Valeur**
Nombre de données	1578
Nombre de classe	**3** Malin Bénin Normal
Mask	798
Images	780
Type	RGB
Dimension moyenne	500*500
Redimensionne	256*256
Normalisation	0 à 1

TABLE 1 – Caractéristiques du dataset

D'autre part on a dû enlever tout ce qui était mask dans notre dataset afin de ne travailler qu'avec les images US. Cela ayant réduit notre dataset, nous avons augmenté les images à l'aide de la bibliothèque ImageDataGenerator pour avoir une base de données plus important.

5.3 Structure du CNN

Ce modèle est un réseau de neurones convolutif (CNN) pour la classification d'images en trois classes (malin, bénin et normal). Il est constitué de cinq couches de convolutions, chacune suivie d'une couche de max Pooling, ce qui permet de réduire la dimensionnalité de la sortie de la convolution. Le modèle se termine par deux couches entièrement connectées, une couche cachée avec une fonction d'activation ReLU et une

couche de sortie avec une fonction d'activation softmax pour produire des probabilités de classe. La taille de la sortie de la dernière couche convolutive est aplatie en un vecteur avant d'être passée à la couche cachée.

La configuration du processus d'apprentissage prend en entrée plusieurs paramètres :

-'optimizer' : c'est la méthode d'optimisation utilisée pour ajuster les poids du modèle et minimiser la fonction de perte. Dans ce cas, l'optimiseur utilisé est Adam.

-'loss' : c'est la fonction de coût utilisée pour calculer l'erreur de prédiction du modèle. Ici, nous utilisons la fonction 'SparseCategoricalCrossentropy' qui est utilisée pour la classification multi-classes lorsque les étiquettes sont des entiers.

La fonction de coût calcule la différence entre les prédictions du modèle et les étiquettes réelles, en utilisant la formule de l'entropie croisée pour mesurer l'erreur de classification. Le but de l'optimisation est de minimiser cette fonction de coût pour améliorer les performances du modèle.

-metrics' : ce sont les mesures de performance utilisées pour évaluer le modèle. Dans ce cas, nous utilisons la métrique de précision (accuracy).

En somme, cette fonction est nécessaire pour configurer le modèle avant l'entraînement en définissant les paramètres clés tels que l'optimiseur, la fonction de perte et les mesures de performance

6 Résultats et interprétation

6.1 Courbe d'évaluation de la précision et de perte

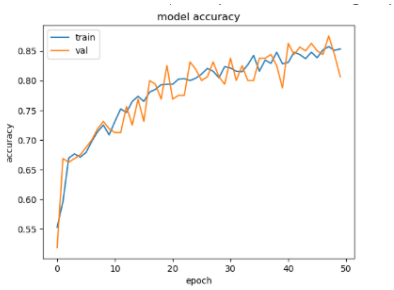

FIGURE 3 – Courbe d'évolution de la précision

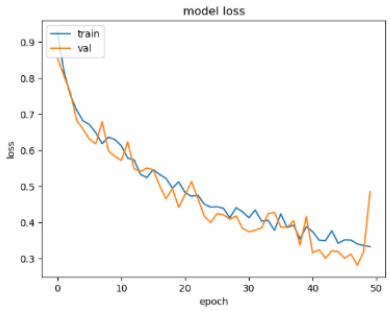

FIGURE 4 – Courbe d'évolution de la Perte

Les pertes et les précisions de formation et de validation sont indiquées pour chaque époque. Le modèle commence avec une précision et une perte faible, mais s'améliore progressivement avec le temps. La précision de la validation s'améliore également avec le temps, mais elle fluctue davantage que la précision de la formation.

Les fluctuations dans les mesures de la perte et de la précision (accuracy) lors de l'entraînement d'un modèle de réseau de neurones peuvent être dues à plusieurs raisons, dont :

— La taille de l'échantillon : comme l'échantillon utilisé pour l'entraînement est petit, les fluctuations

peuvent être plus importantes. Ce qui est le cas pour notre base de données. Donc pour remédier à cela, il nous faut une plus grande base de données.

— La complexité du modèle : Si le modèle est trop complexe, il peut être difficile à entraîner et les fluctuations peuvent être plus importantes. C'est raison pour laquelle notre modèle n'est pas complexe

— La régularisation : L'utilisation de techniques de régularisation, telles que la régularisation L1/L2 ou le dropout, peut entraîner des fluctuations dans les mesures de la perte et de la précision, car ces techniques ajoutent du bruit au modèle. Nous avons essayé cela, mais ça n'avait pas d'impact positif.

— La qualité des données : Si les données sont bruyantes ou contiennent des erreurs, cela peut entraîner des fluctuations dans les mesures de la perte et de la précision. Ce dernier paramètre est susceptible d'être présent dans notre base de données

6.2 Matrice de confusion

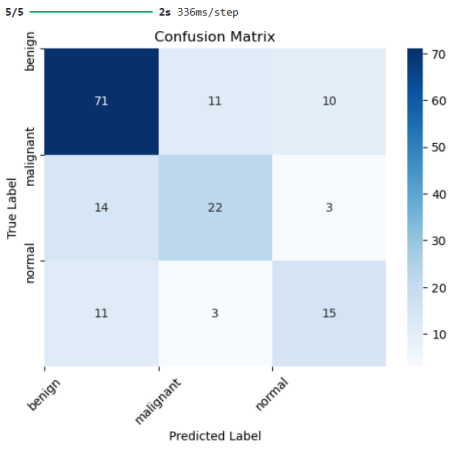

FIGURE 5 – Matrice de confusion

La matrice de confusion est un outil utilisé pour évaluer les performances d'un modèle de classification. Elle permet de voir la correspondance entre les prédictions du modèle et les vraies étiquettes de classe. En général, on souhaite avoir une diagonale de valeurs élevées (prédictions correctes) et des valeurs hors-diagonale faibles (confusions). Dans ce cas-ci, la plupart des prédictions sont correctes, mais on observe quelques confusions entre malin bénin et normal bénin. Il pourrait être intéressant de se pencher sur ces confusions pour comprendre pourquoi le modèle les fait et éventuellement les corriger

L'une des causes de cette confusion serait la qualité des images comme mentionné plus haut et une autre serait que le modèle ne s'est pas entrainé sur une plus grande taille de donnée afin de mieux généraliser les nouvelles.

6.3 Prédiction de quelques données

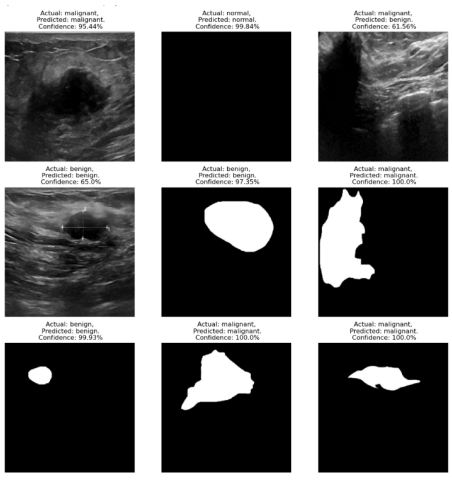

FIGURE 6 – Prédiction des données

La confiance (ou "score de confiance") est un indicateur de la certitude du modèle quant à sa prédiction. Elle est souvent exprimée sous forme de pourcentage et corres-

pond à la probabilité estimée par le modèle que l'image appartienne à la classe prédite

La confiance est importante car elle permet de quantifier la fiabilité de la prédiction. Si la confiance est élevée, cela indique que le modèle est très sûr de sa prédiction. Si la confiance est faible, cela peut indiquer que le modèle est incertain ou qu'il a du mal à distinguer les différentes classes.

En pratique, la confiance peut être utilisée pour ajuster le seuil de décision du modèle, ce qui permet de contrôler le compromis entre la précision et le rappel du modèle. Par exemple, en augmentant le seuil de décision, on peut augmenter la précision (au détriment du rappel) en ne considérant que les prédictions les plus sûres. À l'inverse, en diminuant le seuil de décision, on peut augmenter le rappel (au détriment de la précision) en acceptant des prédictions moins sûres.

6.4 Rapport de classification

Le rappel mesure la capacité du modèle à identifier tous les exemples positifs dans l'ensemble de données de test. Le F1-score est une mesure pondérée de la précision et du rappel, qui prend en compte à la fois les faux positifs et les faux négatifs.

Ainsi, pour calculer le rappel et le F1-score, nous utilisons les données de test avec des étiquettes de classe connues pour pouvoir comparer les prédictions du modèle avec les vraies étiquettes de classe

Globalement, le modèle semble donner des résultats raisonnables avec une bonne précision. Cependant, il y a des déséquilibres dans la performance du modèle pour les différentes classes, avec une précision plus faible pour les classe normal et malin.

7 Limitation et discussion

Notre modèle semble relativement peu profond par rapport à certains modèles plus complexes, mais être suffisant pour résoudre des tâches de classification d'images simples (avec une base de données faible). Nous avons complexifié le modèle pour voir l'amélioration des résultats, mais cela à plutôt empirer, car on l'a fait sans toutefois agrandir la base de données. Nous pensons donc que la complexité du modèle dépend de la taille des données. Donc si on veut complexifier le modèle, faut faire cela avec une plus grande taille de donnée.

8 Conclusion

Ce projet portait sur la classification des images du sein (normal, cancer malin et benin). Nous avons utilisé un CNN pour construire notre modèle car il est approprié pour de tel problème. Après avoir évalué notre modèle de classification d'images de tumeurs mammaires à partir de données, nous pouvons conclure que le modèle a obtenu des résultats satisfaisants en termes de précision, rappel et f1-score pour les classes 'benin', 'malignant' et 'normal', avec une précision globale de 0,81. Cependant, il est important de noter que le modèle a eu des difficultés à distinguer les tumeurs bénignes des tumeurs malignes, ce qui est une préoccupation majeure dans le domaine médical. Cela pourrait expliquer pourquoi le rappel (recall) pour la classe "benign" est faible, ce qui signifie que le modèle a manqué de classer certaines images "benign" comme telles.

Dans le futur, il serait intéressant de continuer à améliorer la qualité des données en ayant une plus grande base de données,

afin d'améliorer les performances du modèle. Il serait également important de recueillir davantage de données de qualité pour aider le modèle à mieux distinguer les tumeurs bénignes des tumeurs malignes. Enfin, une étude clinique plus approfondie serait nécessaire pour valider l'utilisation de ce modèle dans un environnement de soins de santé réel.

9 Références

[1] OpenClassrooms. "Découvrez les méthodes factorielles et la classification non supervisée." https://openclassrooms.com/fr/courses/4525281-realisez-une-analyse-exploratoire-dedonnees/5291335-decouvrez-les-methodesfactorielles-et-la-classification-non-supervisee

[2]https://images.squarespacecdn.com/content/v1/519a7bc0e4b08ccdf8f31445/1524659432980-

[3] DataScientest. "Convolutional Neural Network." : https://datascientest.com/convolutional-neuralnetwork.

[4]Intellipaat. "Convolutional Neural Network." https://www.imaios.com/fr/ressources/blog/introduction-aux-architectures-d-apprentissage-automatique-profond-les-plus-couran

[5] StackLima. "ML : prétraitement des données en Python.

[6] "Neural Networks and Deep Learning" de Michael Nielsen http://neuralnetworksanddeeplearning.com/

[7] "Regularization in Deep Learning" par Jason Brownlee https://machinelearningmastery.com/how-toreduce-overfitting-with-dropout-regularizationin-keras/

.

10 Lien code, base de données originale

Code : cliquez ici pour voir la base de données

BDOriginale : cliquez ici pour voir la base de données