



Mémoire présenté en vue
de l'obtention du diplôme de Master

Intitulé

CLASSIFICATION SUPERVISEE D'IMAGES DE PEAU BASEE SUR DEUX CLASSIFIEURS DANS LE DOMAINE MEDICAL

Domaine : Mathématiques-Informatique

Filière : Informatique

Spécialité : Intelligence artificielle et traitement de
l'information

Par : Mr Naoufel Ahtchi ali

Jury d'évaluation

| Qualité | Nom et Prénom | Grade | Université |
|------------|-------------------|-------|----------------------|
| Président | Mme Nabihaz AZIZI | Prof | Badji Mokhtar-Annaba |
| Rapporteur | Mme Akila DJEBBAR | MCB | Badji Mokhtar-Annaba |
| Examineur | Mme NouhaKOBSSI | MAA | Badji Mokhtar-Annaba |

REMERCIEMENTS

Nous remercions tout d'abord Dieu le tout puissant de nous avoir donné la volonté et la persévérance pour réaliser ce travail. Nous tenons à saisir cette occasion et adresser nos profonds remerciements et nos profondes reconnaissances à madame DJEBBAR Akila, notre encadreur, pour ses précieux conseils et son orientation ficelée tout au long de notre recherche. Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail Et de l'enrichir par leurs propositions. Nous souhaitons adresser nos remerciements les plus sincères aux personnes qui nous ont apporté leur aide et qui ont contribué à l'élaboration de ce mémoire ainsi qu'à la réussite de cette formidable année universitaire. Nous remercions tous nos enseignants pour leur dévouement, leur patience et leur contribution à notre formation. Enfin, nous adressons nos plus sincères remerciements à tous nos proches et amis, qui nous ont toujours encouragés au cours de la réalisation de ce mémoire.

DEDICACES

Je dédie ce travail aux deux êtres les plus chères au monde, les bougies qui m'ont toujours guidé vers le bon chemin : Ma mère, la lumière de ma vie, qui a tout fait pour ma réussite et mon bonheur. Mon père, à lui, je devais offrir tout le respect et l'amour pour son soutien et sa tendresse. ^[L]_[SEP] A ma soeur et à tous les membres de la famille et à tous les amis.

TABLE DES MATIERES

| | |
|--|-----------|
| Table des matières | 4 |
| Liste des figures | 7 |
| Liste des tableaux | 8 |
| Liste des abréviations | 9 |
| Introduction générale | 10 |
| 1. Chapitre 1 : Apprentissage Artificiel | 12 |
| 1.1. Définition du Machine Learning..... | 12 |
| 1.2. Machine Learning et Intelligence Artificielle | 13 |
| 1.3.Types de Machine learning | 14 |
| 1.3.Apprentissage supervisé | 15 |
| 1.3.1. Classification | 15 |
| 1.3.1.1.1 Support Vector Machine(SVC) | 16 |
| 1.3.1.1.2 K-Nearest-Neighbour(KNN) | 16 |
| 1.3.1.1.3 Classifieur des forêts aléatoires | 17 |
| 1.3.1.2 Régression | 17 |
| 1.3.1.2.1 Support Vector Regression (SVR) | 18 |
| 1.3.1.2.2 Régression linéaire | 18 |
| 1.3.2 Apprentissage non supervisé | 19 |
| 1.3.2.1 Apprentissage non-supervisé vs. Supervisé | 20 |
| 1.3.2.2 Utilisation d`apprentissage non-supervisé | 20 |
| 1.3.2.3 Liste des algorithmes d'apprentissage non supervisé | 20 |
| 1.3.2.4 Méthodes d'apprentissage non supervisé | 21 |
| 1.3.2.5 L`algorithme K-Means pour le Clustering | 22 |
| 1.3.3 Apprentissage semi supervisé | 22 |
| 1.3.4 Apprentissage par renforcement | 23 |
| 1.4. Conclusion | 23 |
| 2. Chapitre 2 : Les réseaux de neurones convolutifs (CNN) | 24 |
| 2.1. Définition | 24 |

| | |
|--|-----------|
| 2.2. Architecture | 25 |
| 2.2.1 Couches de convolution | 25 |
| 2.2.2 Couches de pooling (POOL) | 26 |
| 2.2.3 Couches de correction (ReLU) | 28 |
| 2.2.4 Couche entièrement connectée (FullyConnected) | 28 |
| 2.2.5 Couche de perte (Loss) | 29 |
| 2.3. Choix des hyper paramètres | 29 |
| 2.3.1 Nombre de filtres | 29 |
| 2.3.2 Forme des Max Pooling | 30 |
| 2.4. Méthodes de régularisation | 30 |
| 2.4.1 Empirique | 30 |
| 2.4.1.1 Dropout | 30 |
| 2.4.1.2 DropConnect | 31 |
| 2.4.1.3 Pooling stochastique | 31 |
| 2.4.1.4 Données artificielles | 31 |
| 2.4.2 Explicite | 32 |
| 2.4.2.1 Taille du réseau | 32 |
| 2.4.2.2 Dégradation du poids | 32 |
| 2.5 Systèmes connexes | 33 |
| 2.5.1 Système1 « Skin lesions: HAM 10000 »..... | 33 |
| 2.5.2 Système2 « Decision Support System using CNN »..... | 34 |
| 2.5.3 Système3«SkinLesion Classification UsingCNNs»..... | 34 |
| 2.6 Conclusion | 35 |
| 3. Chapitre 3 : Conception du système | 36 |
| 3.1. Architecture détaillée du système | 36 |
| 3.2. Description de la base de données | 38 |
| 3.3. Prétraitement des données | 40 |
| 3.3.1 Equilibrer la distribution des images sur les classes de la base de données. | 42 |
| 3.3.2 Séparation des données | 43 |
| 3.3.3 Conversion des données 1D en images | 43 |
| 3.3.4 Normalisation des données | 43 |
| 3.3.5 Répartition de la BDD (Apprentissage/Test) | 43 |
| 3.3.6 Régénération des images dupliqués | 44 |

| | |
|---|-----------|
| 3.4. Extraction de caractéristiques et apprentissage CNN | 44 |
| 3.5. Classification avec Softmax | 47 |
| 3.6. Classification avec les Forêts Aléatoires | 47 |
| 3.7. Les mesures de performances | 47 |
| 3.8. Conclusion | 48 |
| 4. Chapitre 4 : Implémentation du système | 49 |
| 4.1. Environnement de travail | 49 |
| 4.2. Résultats et discussions | 50 |
| 4.2.1 Classification avec Softmax | 50 |
| 4.2.2 Autres essais d'architecture CNN | 53 |
| 4.2.3 Classification avec les forêts aléatoires..... | 54 |
| 4.2.4 Comparaison entre les classifieurs Softmax et Forêts Aléatoires | 56 |
| 4.3 Comparaison du système / Related Works | 56 |
| 4.4 Conclusion | 58 |
| Conclusion générale..... | 59 |
| Références | 60 |
| Annexe A | 63 |
| Annexe B | 65 |

LISTE DES FIGURES

- Fig. 1. Types d'apprentissage.
- Fig. 2. Apprentissage supervisé.
- Fig. 3. Représentation visuelle de la régression linéaire [8].
- Fig. 4. Exemple K-Means Clustering [10].
- Fig. 5. Représentation des couches d'un réseau de neurones convolutif. [11].
- Fig. 6. Max pooling avec un filtre 2×2 et un pas de 2.[13]
- Fig. 7. Architecture générale du système.
- Fig. 8. Echantillons de la base de données HAM10000.[23]
- Fig. 9. Les meta-données de la base HAM10000.[23]
- Fig. 10. Distribution des classes dans la base HAM10000.
- Fig. 11. Visualisation du fichier CSV de notre BDD
- Fig. 12. Schéma des étapes de prétraitement
- Fig. 13. Distribution des classes après duplication d'images
- Fig. 14. Architecture du modèle CNN avec LeNet Style
- Fig. 15. Sommaire architecture CNN- Trainable paramètres.
- Fig. 16. Les taux de précision durant l'entraînement avec 10 epochs.
- Fig. 17. Les taux d'erreur durant l'entraînement avec 10 epochs.
- Fig. 18. Matrice de confusion modèle CNN.
- Fig. 19. Rapport de classification pour le modèle CNN.
- Fig. 20. Taux de précision avec le classifieur des forêts aléatoires.
- Fig. 21. Matrice de confusion modèle Forêts Aléatoires.
- Fig. 22. Rapport de classification pour les Forêts Aléatoires.
- Fig. 23. Interface de l'application.
- Fig. 24. Exemple de classification avec les forêts aléatoires.
- Fig. 25. Exemple de classification avec Softmax.

LISTE DES TABLEAUX

Tab. 1. Architecture du modèle CNN

Tab. 2. Essais d'architectures CNN avec résultats obtenus.

Tab. 3. Comparaison du système/ Relatedworks.

Tab. 4. Résumé des performances du classifieur Softmax.

Tab. 5. Résumé des performances du classifieur des Forêts aléatoires.

LISTE DES ABREVIATIONS

- IA** Intelligence Artificielle^[1]_{SEP}
- ML** Machine Learning
- ADN** Acide désoxyribonucléique
- CNN** Convolutional Neural Network
- VGG** Visual Geometry Group
- KNN** K-Nearest-Neighbour
- SVM** Support Vector Machine
- SVC** Support Vector Machine FOR CLASSIFICATION
- SVR** Support Vector Machine FOR REGRESSION
- RF** forêts aléatoire
- ResNet** Residential Energy Services Network
- ReLU** Rectified Linear Unit
- FC** Fully Connected
- PIL** Python Imaging Library
- BDD** base de données
- HAM10000** Humain Against Machine 10000 images
- AKIEC** Actinickeratoses and intraepithelialcarcinoma / Bowensdisease.
- BCC** Basal cellcarcinoma.
- BKL** Benignkeratosis-likelesions.
- DF** Dermatofibroma.
- NV** Melanocyticnevi.
- VASC** Vascularlesions.
- MEL** Melanoma.

INTRODUCTION

Contexte de recherche

Le cancer de la peau est l'un des trois types de cancer les plus dangereux, causé par un ADN endommagé qui peut entraîner la mort. Cet ADN endommagé entraîne une croissance incontrôlable des cellules, qui s'accélère de nos jours. Il existe quelques recherches pour l'analyse informatisée de la malignité dans les images de lésions cutanées. Cependant, l'analyse de ces images est très difficile en raison de certains facteurs gênants tels que les réflexions lumineuses de la surface de la peau, les variations de l'éclairage des couleurs, les différentes formes et tailles des lésions. Par conséquent, la reconnaissance automatique du cancer de la peau est précieuse pour renforcer la précision et la compétence des pathologistes dans les premiers stades.

Problématique

Il existe certaines difficultés pour détecter le cancer de la peau qui peuvent être attribuées à la variation des types d'images et des sources. La variation de l'apparence de la couleur de la peau humaine rend la détection du cancer de la peau plus difficile et complexe. Ces défis sont illustrés sur les points suivants :

- Les principales difficultés du cancer de la peau sont les différentes tailles et formes des images qui ne peuvent pas donner un résultat précis d'identification. Dans cette perspective, un prétraitement est nécessaire pour une analyse précise.
- Parfois, le faible contraste des tissus voisins pose des difficultés supplémentaires et rend plus difficile l'analyse précise du cancer de la peau.
- L'illumination de la couleur pose également des difficultés en raison de facteurs tels que la texture de la couleur, les rayons lumineux et les réflexions.
- Il existe dans le corps humain des grains de beauté qui ne développent jamais de cellules cancéreuses, mais qui rendent difficile la détection précise du cancer de la peau à partir d'images cancéreuses.
- Le biais actuel qui déforme la performance des modèles pour obtenir un meilleur résultat est un autre défi concernant la détection du cancer de la peau.

Motivations

Il existe quatre types de cancer de la peau les plus fréquents comme les kératoses actiniques, le carcinome basocellulaire, le carcinome spinocellulaire et le mélanome. L'organisation mondiale de la santé prévoit qu'un cancer sur trois diagnostiqué est un cancer de la peau. Le nombre de personnes diagnostiquées comme étant atteintes d'un cancer de la peau a augmenté à un rythme assez constant au cours des dernières décennies aux États-Unis, au Canada et en Australie.

Objectifs

Notre objectif c'est d'essayer de faire une classification automatique qui nous permet de faire la différence de ces types de lésions pigmentés avec un taux de précision maximum pour cela nous avons utilisé deux classifieurs : « Softmax » et les forêts aléatoires « Randomforest ».

Contenu du mémoire

Ce mémoire est réparti sur quatre chapitres :

- Chapitre 1 : Apprentissage Machine : Sur ce chapitre nous avons définis l'apprentissage machine ainsi que ses types.
- Chapitre 2 : Les réseaux de neurones convolutionnels (CNN) : nous introduisons l'architecture et les différentes couches et hyper-paramètres d'un réseau CNN.
- Chapitre 3 : Conception du système : Ce chapitre est dédié pour expliquer l'architecture de notre système ainsi que les différentes étapes de sa conception.
- Chapitre 4 : Implémentation du système : C'est le dernier chapitre dans ce mémoire il illustre l'environnement du travail ainsi que les résultats obtenus avec une comparaison entre les deux classifieurs utilisés.

CHAPITRE 1

APPRENTISSAGE ARTIFICIEL

Pouvons-nous introduire l'apprentissage dans des choses non vivantes fabriquées par l'homme que l'on appelle des machines? Permettre à une machine capable d'apprendre comme les humains est un rêve dont la réalisation peut nous conduire à avoir des machines déterministes avec liberté (ou illusion de liberté en un sens). Pendant ce temps, nous pourrions nous vanter avec bonheur que nos humanoïdes ressemblent à l'image et à la vraisemblance des humains sous l'apparence de machines. [1]. Dans ce chapitre nous allons définir l'apprentissage machine ainsi que ses types.

1.1 Définition du Machine learning

«L'apprentissage automatique» fait référence à un sous-domaine de l'informatique concerné par les programmes informatiques capables d'apprendre de l'expérience et ainsi d'améliorer leurs performances au fil du temps. [2]. Comme nous le verrons, l'idée que les ordinateurs «apprennent» est en grande partie une métaphore et n'implique pas que les systèmes informatiques reproduisent artificiellement les systèmes cognitifs avancés censés être impliqués dans l'apprentissage humain.[1].

Nous pouvons considérer ces algorithmes comme un apprentissage au sens fonctionnel: ils sont capables de changer leur comportement pour améliorer leurs performances sur une tâche par l'expérience. Généralement, les algorithmes d'apprentissage automatique sont utilisés pour détecter des modèles dans les données afin d'automatiser des tâches complexes ou de faire des prédictions. Aujourd'hui, ces algorithmes sont utilisés dans diverses applications commerciales du monde réel, notamment les résultats de recherche sur Internet, la reconnaissance faciale, la détection de fraude et l'exploration de données. L'apprentissage automatique est étroitement associé à la grande entreprise «d'analyse prédictive», car les chercheurs utilisent souvent des méthodes d'apprentissage automatique pour analyser les données existantes afin de prédire la probabilité de résultats incertains. S'ils fonctionnent bien, les algorithmes d'apprentissage automatique peuvent produire des résultats automatisés qui se rapprochent de ceux qui auraient été obtenus par une personne dans une situation similaire. L'apprentissage automatique est donc souvent considéré comme une branche de l'intelligence artificielle, puisqu'un algorithme performant peut produire des résultats automatisés qui semblent «intelligents». [1]

Cependant, cela vaut la peine de passer quelques instants au départ sur des questions fondamentales, juste pour voir à quel point elles sont délicates, avant de retrouver nos manches et d'examiner l'apprentissage automatique en pratique. Les choses apprennent quand elles changent de comportement d'une manière qui les rend plus performantes à l'avenir. Cela lie l'apprentissage à la performance plutôt qu'à la connaissance. Vous pouvez tester l'apprentissage en observant le comportement présent et en le comparant avec le comportement passé. Il s'agit d'un type de définition beaucoup plus objectif et semble beaucoup plus satisfaisant.[1].

Mais il y a toujours un problème. L'apprentissage est un concept plutôt glissant. Beaucoup de choses modifient leur comportement de manière à les rendre plus performantes à l'avenir, mais nous ne voudrions pas dire qu'ils ont réellement appris. L'apprentissage implique une réflexion et un but. Quelque chose qui apprend doit le faire intentionnellement. C'est pourquoi nous ne dirions pas qu'une vigne a appris à pousser autour d'un treillis dans un vignoble - nous dirions qu'elle a été formée. Apprendre sans but n'est qu'une formation. Ou, plus précisément, dans l'apprentissage, l'objectif est celui de l'apprenant, alors que dans la formation, c'est le professeur. Ainsi, à y regarder de plus près, la deuxième définition de l'apprentissage, en termes opérationnels et axés sur la performance, a ses propres problèmes lorsqu'il s'agit de parler d'ordinateurs. [3]

1.2 Machine learning et Intelligence artificielle

L'apprentissage machine (ML) et l'intelligence artificielle (IA) influencent de plus en plus la vie, grâce à des augmentations significatives de la disponibilité des processeurs, de la vitesse, de la connectivité et du stockage de données bon marché. L'IA fait progresser la fourniture de soins médicaux et de santé, la livraison des transports, l'interaction avec Internet, les systèmes d'approvisionnement alimentaire et soutient la sécurité dans des structures géopolitiques en mutation. La société approche de l'ère des voitures autonomes, aidant les médecins à éviter les erreurs de diagnostic, la reconnaissance vocale précise et la réception de suggestions d'achat personnalisées. La plupart des applications sont bénéfiques, bien que des problèmes éthiques existent.[3]

Simultanément, les modes de vie en évolution doivent interagir en toute sécurité avec le changement climatique. On se rend de plus en plus compte que les impacts du changement climatique ne sont pas une menace isolée, mais nécessitent plutôt des réponses plus holistiques parallèlement à d'autres problèmes de société. Le changement climatique est une question scientifique complexe et multiforme, qui se prête à l'analyse de ML et d'IA, mais en général, cela ne s'est pas encore produit. De nombreux algorithmes de ML sont disponibles depuis des décennies, et peut-être plus particulièrement les réseaux de neurones. Cependant, jusqu'à récemment, les contraintes d'architecture et de puissance de calcul ont limité leur application, et en particulier pour des questions aussi gourmandes en données que le changement climatique. Divers noms décrivent de nouvelles méthodes de calcul, notamment le Big Data, le ML et l'IA. Le Big Data consiste à utiliser des ensembles de

données complexes, si volumineux que les techniques d'analyse traditionnelles ne conviennent pas. L'IA est une forme d'informatique, où le but est souvent d'apprendre à un ordinateur à accomplir des tâches qu'un humain ne peut pas faire, et implique généralement la prise de décision dans différents contextes. Le ML est un sous-domaine de l'IA où les ordinateurs apprennent les relations à partir de grands ensembles de données d'entraînement. [4]

1.3 Types de Machine learning

À un niveau élevé, l'apprentissage automatique est simplement l'étude de l'enseignement d'un programme informatique ou d'un algorithme comment améliorer progressivement une tâche définie qui lui est confiée. Du côté de la recherche, l'apprentissage automatique peut être envisagé à travers le prisme de la modélisation théorique et mathématique du fonctionnement de ce processus. Cependant, plus concrètement, c'est l'étude de la manière de créer des applications qui présente cette amélioration itérative. Il existe de nombreuses façons de formuler cette idée, mais il existe en grande partie trois grandes catégories reconnues: l'apprentissage supervisé, l'apprentissage non supervisé, l'apprentissage semi-supervisé et l'apprentissage par renforcement.

Dans un monde saturé d'intelligence artificielle, d'apprentissage automatique et de discussions trop zélées sur les deux, il est intéressant d'apprendre à comprendre et à identifier les types d'apprentissage automatique que nous pouvons rencontrer. Pour l'utilisateur moyen d'un ordinateur, cela peut prendre la forme d'une compréhension des types d'apprentissage automatique et de la manière dont ils peuvent se manifester dans les applications que nous utilisons. Et pour les praticiens qui créent ces applications, il est essentiel de connaître les types d'apprentissage automatique afin que, pour toute tâche donnée que vous puissiez rencontrer, vous puissiez créer l'environnement d'apprentissage approprié et comprendre pourquoi ce que vous avez fait a fonctionné.

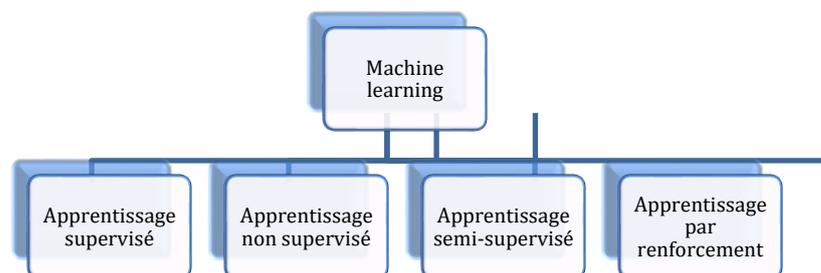


Fig.1.Types d`apprentissage.[4]

1.3.1 Apprentissage supervisé

Les algorithmes d'apprentissage automatique supervisé sont les algorithmes qui nécessitent une assistance externe. L'ensemble de données d'entrée est divisé en ensemble de données de train et de test. [3]

Il s'agit de la technique la plus courante pour la formation des réseaux neurones et des arbres de décision. Ces deux éléments dépendent des informations fournies par la classification prédéterminée.[3]

En outre, cet apprentissage est utilisé dans les applications où les données historiques prédisent les événements de fonctionnalité probables. Il existe de nombreux exemples pratiques de cet apprentissage, par exemple une application qui prédit l'espèce d'iris à partir d'un ensemble de mesures de sa fleur.[5]

Les tâches d'apprentissage supervisé sont divisées en deux catégories: la classification et la régression. En classification, l'étiquette est discrète, tandis qu'en régression, l'étiquette est continue. [5] Le flux de travail des algorithmes d'apprentissage automatique supervisé est présenté dans la figure (Fig.2).

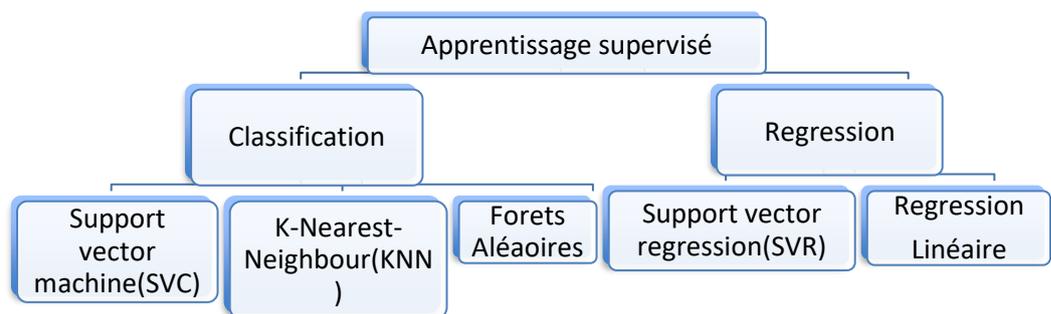


Fig.2. Apprentissage supervisé.[5]

1.3.1.1 Classification

Les algorithmes d'apprentissage automatique supervisé qui traitent davantage de la classification comprennent les éléments suivants: classificateurs linéaires, classificateur Naïve Bayes, K-Nearest-Neighbour (KNN), Support Vector Machine (SVC); Classificateurs quadratiques, arbre de décision, forêts aléatoire (RF); Réseaux de neurones, réseaux

bayésiens, etc. [6] Nous expliquerons les trois plus importants mentionnés dans la figure (Fig.2).

1.3.1.1.1 Support Vector Machine(SVC)

Il s'agit de la technique d'apprentissage automatique supervisé la plus récente. Les modèles SVM (Support Vector Machine) sont étroitement liés aux réseaux neuronaux de perception multicouche classiques. Les SVM tournent autour de la notion de «marge» - de chaque côté d'un hyper plan qui sépare deux classes de données. Il a été prouvé que la maximisation de la marge et ainsi la création de la plus grande distance possible entre l'hyperplan de séparation et les instances de chaque côté de celui-ci réduisaient une limite supérieure sur l'erreur de généralisation attendue. Il est principalement utilisé pour la classification. SVM fonctionne sur le principe du calcul de la marge. Il s'agit essentiellement de dessiner des marges entre les classes. Les marges sont dessinées de manière à ce que la distance entre la marge et les classes soit maximale et, par conséquent, minimise l'erreur de classification.[6].

1.3.1.1.2 K-Nearest-Neighbour(KNN)

Afin d'illustrer le concept d'algorithme de k-plus proche voisin, considérons un cas de données d'entrée bidimensionnelles comme le montre la figure 4, le graphique du haut de la figure montre la distribution des données. Qu'il y ait une relation entre ces données d'entrée et les données de sortie (non montrées ici). Pour le moment, nous pouvons ignorer cette relation. Considérons que nous utilisons la valeur de k comme 3. Comme le montre le graphique du bas de la figure, laissez un échantillon de test situé comme indiqué par un point rouge. Ensuite, nous trouvons les 3 voisins les plus proches du point de test à partir de la distribution d'apprentissage, comme indiqué. Maintenant, afin de prédire la valeur de sortie du point de test, tout ce que nous avons à faire est de trouver la valeur de la sortie pour les 3 voisins les plus proches et de faire la moyenne de cette valeur.[5].

1.3.1.1.3 Classifieur des Forêts aléatoires

L'algorithme des « forêts aléatoires » (ou Random Forest parfois aussi traduit par forêt d'arbres décisionnels) est un algorithme de classification qui réduit la variance des prévisions d'un arbre de décision seul, améliorant ainsi leurs performances. Pour cela, il combine de nombreux arbres de décisions dans une approche de type bagging.[7]

L'algorithme des « forêts aléatoires » a été proposé par Leo Breiman et Adèle Cutler en 2001. Dans sa formule la plus classique, il effectue un apprentissage en parallèle sur de multiples arbres de décision construits aléatoirement et entraînés sur des sous-ensembles de données différents. Le nombre idéal d'arbres, qui peut aller jusqu'à plusieurs centaines voire plus, ce dernier est un paramètre important : il est très variable et dépend du problème. Concrètement, chaque arbre de la forêt aléatoire est entraîné sur un sous-ensemble aléatoire de données selon le principe du bagging, avec un sous-ensemble aléatoire de features (caractéristiques variables des données) selon le principe des « projections aléatoires ». Les prédictions sont ensuite moyennées lorsque les données sont quantitatives ou utilisés pour un vote pour des données qualitatives, dans le cas des arbres de classification. L'algorithme des forêts aléatoires est connu pour être un des classifieurs les plus efficaces « out-of-the-box » (c'est-à-dire nécessitant peu de prétraitement des données). Il a été utilisé dans de nombreuses applications, y compris grand public, comme pour la classification d'images de la caméra de console de jeu Kinect* dans le but d'identifier des positions du corps. [7]

1.3.1.2 Régression

La régression est l'un des outils d'apprentissage automatique et de statistiques les plus importants et les plus largement utilisés. Il vous permet de faire des prédictions à partir de données en apprenant la relation entre les caractéristiques de vos données et une réponse à valeur continue observée. La régression est utilisée dans un grand nombre d'applications allant de la prévision des cours des actions dans les bourses à la compréhension des réseaux de régulation des gènes.[8]

1.3.1.2.1 Support Vector Regression (SVR)

La Support Vector Machine (SVM) est un algorithme d'apprentissage automatique proposé par Cortes et Vapnik basé sur la théorie de l'apprentissage statistique. La minimisation des risques structurels est le concept de base de cette méthode. Une version de SVM pour la régression a été proposée dans. La régression vectorielle de support (SVR) a été largement appliquée dans la prédiction de séries chronologiques ainsi que dans la prévision de la demande de charge électrique et la prédiction des défauts. [8]

Supposons qu'un ensemble de données chronologiques soit donné comme suit pour un système électrique

1.3.1.2.2 Régression linéaire

Le but de la régression linéaire, dans le cadre de la famille des algorithmes de régression, est de trouver des relations et des dépendances entre les variables. Il représente une relation de modélisation entre une variable dépendante scalaire continue y (également étiquette ou cible dans la terminologie d'apprentissage automatique) et une ou plusieurs variables explicatives (un vecteur D -dimensionnel) (également des variables indépendantes, des variables d'entrée, des caractéristiques, des données observées, des observations, attributs, dimensions, points de données, etc.) notés X à l'aide d'une fonction linéaire. Dans l'analyse de régression, l'objectif est de prédire une variable cible continue, alors qu'un autre domaine appelé classification est de prédire une étiquette à partir d'un ensemble fini. Le modèle pour une régression multiple qui implique une combinaison linéaire de variables d'entrée prend la forme:

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + e \quad (1)[8]$$

La régression linéaire appartient également à la catégorie des algorithmes d'apprentissage supervisé. Cela signifie que nous formons le modèle sur un ensemble de données étiquetées (données d'apprentissage), puis que nous utilisons le modèle pour prédire les étiquettes sur des données non étiquetées (données de test).[8]

Comme le montre la figure (Fig.3) , le modèle (ligne rouge) est calculé à l'aide de données d'apprentissage (points bleus) où chaque point a une étiquette connue (axe y) pour ajuster les points aussi précisément que possible en minimisant la valeur d'une fonction de perte choisie. Nous pouvons alors utiliser le modèle pour prédire les étiquettes inconnues (nous ne connaissons que la valeur x et voulons prédire la valeur y). [8]

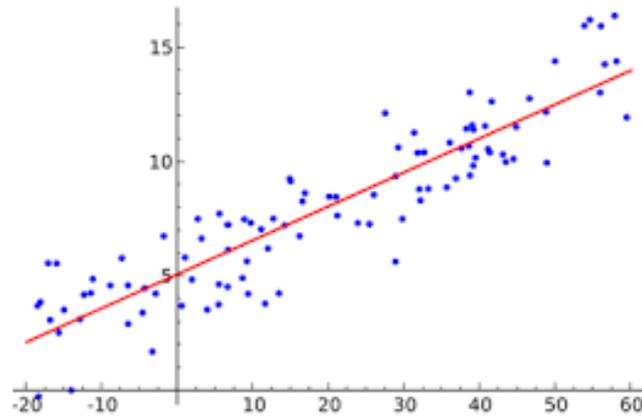


Fig.3.Représentation visuelle de la régression linéaire [8].

1.3.2 Apprentissage non supervisé

L'introduction dans un système d'une approche d'apprentissage non supervisé est un moyen d'expérimenter l'intelligence artificielle. En général, des systèmes d'apprentissage non supervisé permettent d'exécuter des tâches plus complexes que les systèmes d'apprentissage supervisé, mais ils peuvent aussi être plus imprévisibles. Même si un système d'IA d'apprentissage non supervisé parvient tout seul, par exemple, à faire le tri entre des chats et des chiens, il peut aussi ajouter des catégories inattendues et non désirées, et classer des races inhabituelles, introduisant plus de bruit que d'ordre [9].

1.3.2.1 Apprentissage non supervisé vs. Supervisé

L'apprentissage non supervisé consiste à apprendre sans superviseur. Il s'agit d'extraire des classes ou groupes d'individus présentant des caractéristiques communes. La qualité d'une méthode de classification est mesurée par sa capacité à découvrir certains ou tous les motifs cachés.

On distingue l'apprentissage supervisé et non supervisé. Dans le premier apprentissage, il s'agit d'apprendre à classer un nouvel individu parmi un ensemble de classes prédéfinies: on connaît les classes a priori. Tandis que dans l'apprentissage non-supervisé, le nombre et la définition des classes ne sont pas donnés a priori[9].

1.3.2.2 Utilisation d'apprentissage non-supervisé

Les techniques d'apprentissage non-supervisé peuvent être utilisées pour résoudre, entre autres, les problèmes suivants :

- le partitionnement de données (par exemple avec l'algorithme des k-moyennes, le regroupement hiérarchique),
- l'estimation de densité de distribution (distribution de mélange, estimation par noyau),
- la réduction de dimension (analyse en composantes principales, carte auto-adaptative).

L'apprentissage non supervisé peut aussi être utilisé en conjonction avec une inférence bayésienne pour produire des probabilités conditionnelles pour chaque variable aléatoire étant données les autres [9].

1.3.2.3 Liste des algorithmes d'apprentissage non supervisé

Dans cette section nous citons les algorithmes les plus connus pour l'apprentissage non supervisé [9] :

- K-meansclustering (K-moyenne)
- DimensionalityReduction (Réduction de la dimensionnalité)

- Neural networks (Réseaux de neurones) / Deep Learning
- Principal Component Analysis (Analyse des composants principaux)
- Singular Value Decomposition (Décomposition en valeur singulière)
- Independent Component Analysis (Analyse en composantes indépendantes)
- Distribution models (Modèles de distribution)
- Hierarchicalclustering (Classification hiérarchique).

1.3.2.4 Méthodes d'apprentissage non supervisé

Le clustering consiste à grouper des points de données en fonction de leurs similitudes, tandis que l'association consiste à découvrir des relations entre les attributs de ces points de données:

Les techniques de clustering cherchent à décomposer un ensemble d'individus en plusieurs sous ensembles les plus homogènes possibles

On ne connaît pas la classe des exemples (nombre, forme, taille)

Les méthodes sont très nombreuses, typologies généralement employées pour les distinguer Méthodes de partitionnement / Méthodes hiérarchiques

Avec recouvrement / sans recouvrement

Autre : incrémental / non incrémental

D'éventuelles informations sur les classes ou d'autres informations sur les données n'ont pas d'influence sur la formation des clusters, seulement sur leur interprétation⁷.

L'un des algorithmes le plus connu et utilisé en clustering est la K-moyenne. Cet algorithme va mettre dans des "zones" (Cluster), les données qui se ressemblent. Les données se trouvant dans le même cluster sont similaires.

L'approche de K-Means consiste à affecter aléatoirement des centres de clusters (appelés centroids), et ensuite assigner chaque point de nos données au centroid qui lui est le plus proche. Cela s'effectue jusqu'à assigner toutes les données à un cluster [9].

1.3.2.5 L'algorithme K-Means pour le Clustering

L'algorithme K-Means (K-moyennes) est **le plus connu** dans l'Unsupervised Learning. Il s'agit d'un algorithme de Clustering. Ce dernier va mettre dans des "zones" (**Cluster**), les données qui se ressemblent. Les données se trouvant dans le même cluster sont similaires.

L'approche de K-Means consiste à affecter aléatoirement des centres de clusters (appelés **centroids**), et ensuite assigner chaque point de nos données au centroid qui lui est le plus proche. Cela s'effectue jusqu'à assigner toutes les données à un cluster.

L'illustration par une image, du résultat d'exécution de l'algorithme K-Means vous permettra d'en appréhender le fonctionnement.[10]

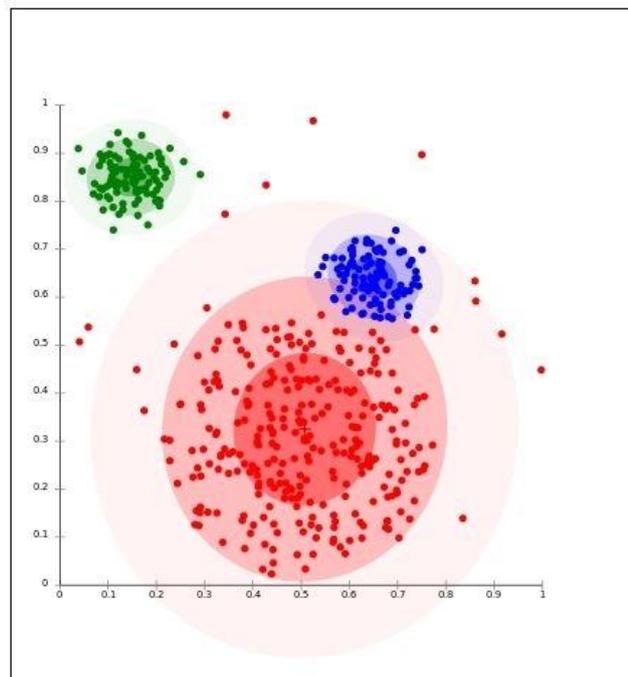


Fig.4.Exemple K-Means Clustering [10].

Dans la figure (Fig.4), on reconnaît trois clusters : (un en rouge, un bleu et un vert) [10].

1.3.3 Apprentissage semi-supervisé

L'apprentissage semi-supervisé est une classe de techniques d'apprentissage automatique qui utilise un ensemble de données étiquetées et non étiquetées. Il se situe ainsi entre l'apprentissage supervisé qui n'utilise que des données étiquetées et l'apprentissage non

supervisé qui n'utilise que des données non étiquetées. Il a été démontré que l'utilisation de données non étiquetées, en combinaison avec des données étiquetées, permet d'améliorer significativement la qualité de l'apprentissage.[10]

Un autre intérêt provient du fait que l'étiquetage de données nécessite souvent l'intervention d'un utilisateur humain. Lorsque les jeux de données deviennent très grands, cette opération peut s'avérer fastidieuse. Dans ce cas, l'apprentissage semi-supervisé, qui ne nécessite que quelques étiquettes, revêt un intérêt pratique évident.[10]

Un exemple d'apprentissage semi-supervisé est le coapprentissage, dans lequel deux classifieurs apprennent un ensemble de données, mais en utilisant chacun un ensemble de caractéristiques différentes, idéalement indépendantes. Si les données sont des individus à classer en hommes et femmes, l'un pourra utiliser la taille et l'autre la pilosité par exemple [10].

1.3.4 Apprentissage par renforcement

L'apprentissage par renforcement consiste, pour un agent autonome (robot, etc.), à apprendre les actions à prendre, à partir d'expériences, de façon à optimiser une récompense quantitative au cours du temps. L'agent est plongé au sein d'un environnement, et prend ses décisions en fonction de son état courant. En retour, l'environnement procure à l'agent une récompense, qui peut être positive ou négative. L'agent cherche, au travers d'expériences itérées, un comportement décisionnel (appelé stratégie ou politique, et qui est une fonction associant à l'état courant l'action à exécuter) optimal, en ce sens qu'il maximise la somme des récompenses au cours du temps [10].

Conclusion

Dans ce chapitre nous avons expliqué les différentes méthodes et types d'apprentissage machine, voyons dans le chapitre suivant une méthode d'apprentissage supervisé particulièrement l'apprentissage profond les réseaux de neurones convolutifs (CNN).

CHAPITRE 2

LES RESEAUX DE NEURONES CONVOLUTIFS (CNN)

Dans cet chapitre nous allons parler en détail sur une méthode d'apprentissage supervisée très utilisée dans l'intelligence artificielle qui est les réseaux de neurones convolutifs ou réseau de neurones à convolution (en anglais CNN ou ConvNet pour Convolutional Neural Networks). Tel que nous allons expliquer l'architecture et les différentes couches du réseau ainsi que les hyperparamètres et les méthodes de régularisation.

2.1 Définition

En apprentissage profond, le réseau de neurones à convolution (CNN) apparaît comme l'un des sujets importants qui nous aident dans la reconnaissance des visages, la détection d'un objet et la classification des images. CNN se compose de plusieurs couches. Ces couches sont des couches entièrement connectées comme les neurones dans un cerveau biologique. Les neurones aident à transporter le «message» d'une cellule à une autre. Un neurone est entouré de nombreux autres neurones.

CNN travaille également sur cette approche. Une couche est connectée à de nombreuses autres couches, de sorte que la sortie d'une couche devient une entrée pour de nombreuses autres couches qui sont directement connectées à cette couche particulière. CNN prend des entrées sous la forme de signaux audio, d'images ou de tout échantillon structuré [11]. La figure ci-dessous (Fig.5) représente une architecture générale d'un réseau de neurones convolutif.

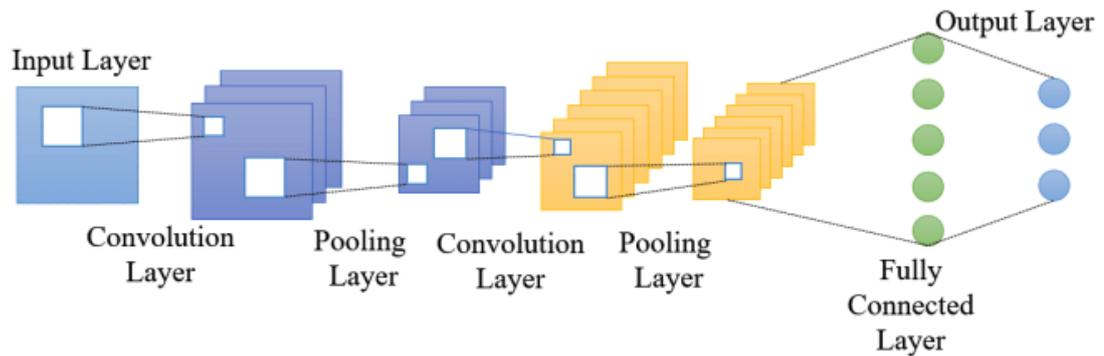


Fig.5.Représentation des couches d'un réseau de neurones convolutif [11].

2.2 Architecture

Un réseau neuronal convolutif se compose d'une couche d'entrée, de couches cachées et d'une couche de sortie. Dans tout réseau de neurones à réaction directe, toutes les couches intermédiaires sont appelées cachées car leurs entrées et sorties sont masquées par la fonction d'activation et la convolution finale. Dans un réseau neuronal convolutif, les couches cachées incluent des couches qui effectuent des convolutions. Cela inclut généralement une couche qui effectue un produit scalaire du noyau de convolution avec la matrice d'entrée de la couche. Ce produit est généralement le produit interne Frobenius et sa fonction d'activation est généralement ReLU. Au fur et à mesure que le noyau de convolution glisse le long de la matrice d'entrée de la couche, l'opération de convolution génère une carte d'entités, qui à son tour contribue à l'entrée de la couche suivante. Ceci est suivi par d'autres couches telles que les couches de regroupement, les couches entièrement connectées et les couches de normalisation.[11].

2.2.1 les couches de convolution

Dans un CNN, l'entrée est un tenseur de forme:

(Nombre d'entrées) x (hauteur d'entrée) x (largeur d'entrée) x (canaux d'entrée).

Après avoir traversé une couche convolutionnelle, l'image devient abstraite dans une carte d'entités, également appelée carte d'activation, avec la forme:

(nombre d'entrées) x (hauteur de la carte d'entités) x (largeur de la carte d'entités) x (canaux de carte d'entités).

Une couche convolutive dans un CNN a généralement les attributs suivants:

Filtres / noyaux convolutifs définis par une largeur et une hauteur (hyper-paramètres).

Le nombre de canaux d'entrée et de canaux de sortie (hyper-paramètres). Les canaux d'entrée d'une couche doivent être égaux au nombre de canaux de sortie (également appelés profondeur) de son entrée.

Hyper paramètres supplémentaires de l'opération de convolution, tels que: rembourrage, foulée et dilatation.

Les couches convolutives convolent l'entrée et transmettent son résultat à la couche suivante. Ceci est similaire à la réponse d'un neurone dans le cortex visuel à un stimulus spécifique. [12]

Chaque neurone convolutif traite les données uniquement pour son champ récepteur. Bien que des réseaux de neurones à réaction directe entièrement connectés puissent être utilisés pour apprendre les fonctionnalités et classer les données, cette architecture est généralement peu pratique pour des entrées plus volumineuses telles que des images haute résolution. Cela nécessiterait un nombre très élevé de neurones, même dans une architecture peu profonde, en raison de la grande taille d'entrée des images, où chaque pixel est une caractéristique d'entrée pertinente. Par exemple, une couche entièrement connectée pour une (petite) image de taille 100 x 100 a 10 000 poids pour chaque neurone de la deuxième couche. Au lieu de cela, la convolution réduit le nombre de paramètres libres, permettant au réseau d'être plus profond. [12]

Par exemple, quelle que soit la taille de l'image, l'utilisation d'une zone de mosaïque 5 x 5, chacune avec les mêmes poids partagés, ne nécessite que 25 paramètres apprenables. L'utilisation de poids régularisés sur moins de paramètres évite les gradients disparaissant et les problèmes de gradients explosifs observés lors du rétro propagation dans les réseaux de neurones traditionnels. [13] [14]

En outre, les réseaux de neurones convolutifs sont idéaux pour les données avec une topologie en forme de grille (telles que des images) car les relations spatiales entre des caractéristiques distinctes sont prises en compte pendant la convolution et / ou la mise en commun.[14]

2.2.2 Couches de pooling (POOL)

Un autre concept important des CNNs est le pooling (« mise en commun »), ce qui est une forme de sous-échantillonnage de l'image. L'image d'entrée est découpée en une série de rectangles de n pixels de côté ne se chevauchant pas (pooling). Chaque rectangle peut être vu comme une tuile. Le signal en sortie de tuile est défini en fonction des valeurs prises par les différents pixels de la tuile.[13] [14]

Le pooling réduit la taille spatiale d'une image intermédiaire, réduisant ainsi la quantité de paramètres et de calcul dans le réseau. Il est donc fréquent d'insérer périodiquement une couche de pooling entre deux couches convolutives successives d'une architecture de

réseau de neurones convolutifs pour réduire le sur-apprentissage. L'opération de pooling crée aussi une forme d'invariance par translation.[13] [14]

La couche de pooling fonctionne indépendamment sur chaque tranche de profondeur de l'entrée et la redimensionne uniquement au niveau de la surface. La forme la plus courante est une couche de mise en commun avec des tuiles de taille 2×2 (largeur/hauteur) et comme valeur de sortie la valeur maximale en entrée (voir Fig.6). On parle dans ce cas de « Max-Pool 2×2 » (compression d'un facteur 4).[13] [14]

Il est possible d'utiliser d'autres fonctions de pooling que le maximum. On peut utiliser un « averagepooling » (la sortie est la moyenne des valeurs du patch d'entrée), du « L2-norm pooling ». Dans les faits, même si initialement l'averagepooling était souvent utilisé il s'est avéré que le max-pooling était plus efficace car celui-ci augmente plus significativement l'importance des activations fortes. En d'autres circonstances, on pourra utiliser un pooling stochastique (voir « Méthodes de régularisation » plus bas dans ce document).[14]

Le pooling permet de gros gains en puissance de calcul. Cependant, en raison de la réduction agressive de la taille de la représentation (et donc de la perte d'information associée), la tendance actuelle est d'utiliser de petits filtres²⁹ (type 2×2). Il est aussi possible d'éviter la couche de pooling³⁰ mais cela implique un risque de sur-apprentissage plus important. [13] [14]

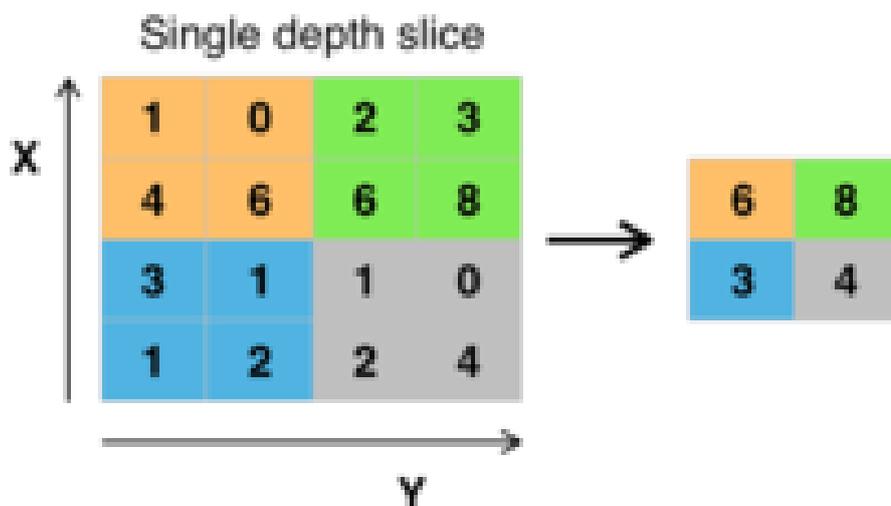


Fig. 6. Max pooling avec un filtre 2×2 et un pas de 2.[13]

2.2.3 Couches de correction (ReLU)

Souvent, il est possible d'améliorer l'efficacité du traitement en intercalant entre les couches de traitement une couche qui va opérer une fonction mathématique (fonction d'activation) sur les signaux de sortie. On a notamment :

La correction ReLU (abréviation de Unité Linéaire Rectifiée) :

$$f(x) = \max(0, x) \quad (2)$$

Cette fonction, appelée aussi « fonction d'activation non saturante », augmente les propriétés non linéaires de la fonction de décision et de l'ensemble du réseau sans affecter les champs récepteurs de la couche de convolution.

La correction par tangente hyperbolique :

$$f(x) = \tanh(x) \quad (3)$$

La correction par la tangente hyperbolique saturante :

$$f(x) = |\tanh(x)| \quad (4)$$

La correction par la fonction sigmoïde :

$$f(x) = (1 + e^{-x})^{-1} \quad (5)$$

Souvent, la correction Relu est préférable, car il en résulte la formation de réseau neuronal plusieurs fois plus rapide, sans faire une différence significative à la généralisation de précision. [13] [14]

2.2.4 Couche entièrement connectée (FullyConnected)

Après plusieurs couches de convolution et de max-pooling, le raisonnement de haut niveau dans le réseau neuronal se fait via des couches entièrement connectées. Les neurones dans une couche entièrement connectée ont des connexions vers toutes les sorties de la couche précédente (comme on le voit régulièrement dans les réseaux réguliers de neurones). Leurs fonctions d'activations peuvent donc être calculées avec une multiplication matricielle suivie d'un décalage de polarisation. [15].

2.2.5 Couche de perte (loss)

La couche de perte spécifie comment l'entraînement du réseau pénalise l'écart entre le signal prévu et réel. Elle est normalement la dernière couche dans le réseau. Diverses fonctions de perte adaptées à différentes tâches peuvent y être utilisées. La perte « Softmax » est utilisée pour prédire une seule classe parmi K classes mutuellement exclusives. La perte par entropie croisée sigmoïde est utilisée pour prédire K valeurs de probabilité indépendante dans $[0,1]$. La perte euclidienne est utilisée pour régresser vers des valeurs réelles dans $[-\infty, +\infty]$. [15].

2.3 Choix des hyper paramètres

Les réseaux de neurones convolutifs utilisent plus d'hyper paramètres qu'un perceptron multicouche standard. Même si les règles habituelles pour les taux d'apprentissage et des constantes de régularisation s'appliquent toujours, il faut prendre en considération les notions de nombre de filtres, leur forme et la forme du max pooling. [13]

2.3.1 Nombre de filtres

Comme la taille des images intermédiaires diminue avec la profondeur du traitement, les couches proches de l'entrée ont tendance à avoir moins de filtres tandis que les couches plus proches de la sortie peuvent en avoir davantage. Pour égaliser le calcul à chaque couche, le produit du nombre de caractéristiques et le nombre de pixels traités est généralement choisi pour être à peu près constant à travers les couches. Pour préserver l'information en entrée, il faudrait maintenir le nombre de sorties intermédiaires (nombre d'images intermédiaires multiplié par le nombre de positions de pixel) pour être croissante (au sens large) d'une couche à l'autre.

Le nombre d'images intermédiaires contrôle directement la puissance du système, dépend du nombre d'exemples disponibles et la complexité du traitement.. [13]

2.3.2 Forme des Max Pooling

Les valeurs typiques sont 2×2 . De très grands volumes d'entrée peuvent justifier un pooling 4×4 dans les premières couches. Cependant, le choix de formes plus grandes va considérablement réduire la dimension du signal, et peut entraîner la perte de trop d'information. [16]

2.4 Méthodes de régularisation

En apprentissage automatique, la régularisation est un procédé visant à améliorer les performances de généralisation d'un algorithme d'apprentissage, autrement dit à diminuer son erreur sur échantillons de test. Cela peut éventuellement être réalisé au détriment de l'erreur d'apprentissage. Un tel procédé a pour but d'éviter le sur-apprentissage qui résulte d'une adaptation trop forte du modèle aux données d'entraînement. Du point de vue du compromis biais/variance, le sur-apprentissage décrit un modèle capable de très bien s'adapter à tout ensemble d'apprentissage donné (faible biais) mais devrait fortement modifier ses paramètres (poids) pour s'adapter à un autre jeu de données d'apprentissage (forte variance).[17]

2.4.1 Empirique

2.4.1.1 Dropout

La méthode du dropout³³ consiste à « désactiver » des sorties de neurones aléatoirement (avec une probabilité prédéfinie, par exemple 0.5 pour les couches cachées et 0.8 pour la couche d'entrée) pendant la phase d'apprentissage. Cela revient à simuler un ensemble de modèles différents (bagging) et à les apprendre conjointement (bien qu'aucun ne soit appris de bout en bout). Chaque neurone étant possiblement inactif pendant une itération d'apprentissage, cela force chaque unité à « bien apprendre » indépendamment des autres et évite ainsi la « co-adaptation »³². Le dropout peut permettre une accélération de l'apprentissage.[17]

En phase de test, les auteurs proposent de pondérer chaque poids appris par sa probabilité d'activation pendant l'apprentissage. Pour un dropout avec un probabilité 0.5 par exemple, cela revient à diviser les poids par deux³². [17]

La technique du dropout est notamment utilisée dans les systèmes de reconnaissance d'image, de voix, le classement de documents et sur des problèmes de calculs en biologie. [17].

2.4.1.2 DropConnect

Le DropConnect³⁴ est une alternative au dropout consistant à inhiber une connexion (l'équivalent de la synapse), et ce de manière toujours aléatoire. Les résultats sont similaires (rapidité, capacité de généralisation de l'apprentissage) au dropout, mais présentent une différence au niveau de l'évolution des poids des connexions. Une couche « complètement connectée » avec un DropConnect peut s'apparenter à une couche à connexion « diffuse ». [18].

2.4.1.3 Pooling stochastique

Le pooling stochastique³⁵ reprend le même principe que le Max-pooling, mais la sortie choisie sera prise au hasard, selon une distribution multinomiale définie en fonction de l'activité de la zone adressée par le pool. [19].

Dans les faits, ce système s'apparente à faire du Max-pooling avec un grand nombre d'images similaires, qui ne varient que par des déformations localisées. On peut aussi considérer cette méthode comme une adaptation à des déformations élastiques de l'image³⁶. C'est pourquoi cette méthode est très efficace sur les images MNIST (base de données d'images représentant des chiffres manuscrits). La force du pooling stochastique est de voir ses performances croître de manière exponentielle avec le nombre de couches du réseau. [19].

2.4.1.4 Données artificielles

Pour limiter le sur-apprentissage (sur-rigidité du réseau de neurones), il est possible de légèrement modifier les données en entrée à partir des données déjà existantes. On va par exemple déformer légèrement une image ou la redimensionner pour recréer une nouvelle

image d'entraînement. Il peut être possible aussi de créer des données complètement artificielles. Le but est d'augmenter le nombre de données d'entraînement (souvent des images) pour améliorer les performances du réseau³⁷. [20].

2.4.2 Explicite

2.4.2.1 Taille du réseau

La manière la plus simple de limiter le sur apprentissage (problèmes de convergence du traitement) est de limiter le nombre de couches du réseau et de libérer les paramètres libres (connexions) du réseau. Ceci réduit directement la puissance et le potentiel prédictif du réseau. C'est équivalent à avoir une « norme zéro ».[21].

2.4.2.2 Dégradation du poids

Le concept est de considérer le vecteur des poids d'un neurone (liste des poids associés aux signaux entrants), et de lui rajouter un vecteur d'erreur proportionnel à la somme des poids (norme 1) ou du carré des poids (norme 2 ou euclidienne). Ce vecteur d'erreur peut ensuite être multiplié par un coefficient de proportionnalité que l'on va augmenter pour pénaliser davantage les vecteurs de poids forts.[21].

La régularisation par norme 1 : La spécificité de cette régulation est de diminuer le poids des entrées aléatoires et faibles et d'augmenter le poids des entrées « importantes ». Le système devient moins sensible au bruit.

La régularisation par norme 2 : (norme euclidienne) La spécificité de cette régulation est de diminuer le poids des entrées fortes, et de forcer le neurone à plus prendre en compte les entrées de poids faible.

Les régularisations par norme 1 et norme 2 peuvent être combinées : c'est la « régularisation de réseau élastique » (Elastic net regulation). [22]

2.5 Systèmes connexes

2.5.1 Système1 « Skin lesions: HAM 10000 » :

L'objectif est de proposer un système qui détecte le cancer de la peau et de faire une classification dans différentes classes en utilisant le réseau neuronal à convolution. La méthodologie de diagnostic utilise un modèle de traitement d'image et d'apprentissage profond. L'image de dermoscopie du cancer de la peau prise, a été soumise à diverses techniques pour éliminer le bruit et la résolution de l'image. Le bruit et la résolution de l'image. Le nombre d'images est également augmenté en utilisant diverses techniques d'augmentation de l'image. Enfin, la méthode d'apprentissage par transfert est utilisée pour augmenter la précision de la classification des images. L'exactitude de la classification des images. le modèle CNN utilisé a donné une moyenne pondérée de précision de 0,88, une moyenne pondérée de rappel de 0,5 et une moyenne pondérée de précision de 0,5..0,88, un rappel moyen pondéré de 0,74 et un score f1 pondéré de 0,77. L'approche d'apprentissage par transfert appliquée en utilisant le modèle ResNet a donné une précision de 90,51%.[22].

2.5.2 Système2 « Decision Support System for Detection and Classification of Skin cancer using CNN

Dans ce système de classification l'auteur a introduit les techniques suivantes pour aborder le problème du diagnostic des lésions cutanées avec l'apprentissage profond :

- augmentation des données mixUp
- 2 types d'apprentissage auto-supervisé pour le pré-entraînement des données,
- 2 étapes d'EfficientNet et l'optimisation de la perte focale en tant que proxy pour un meilleur rappel moyen. Nous trouvons que l'EfficientNetpré-formé avec 2 étapes de prédictionet la perte focale avec $\gamma = 1$ pour être la configuration la plus efficace pour maximiser le rappel moyen entre les 7 classes dans le cas d'un projet de recherche. Le rappel moyen entre les 7 classes dans ce problème.[23]

2.5.3 Système 3 « Skin Lesion Classification Using CNNs with Patch-Based Attention and Diagnosis-Guided Loss Weighting » :

Ce travail aborde deux problèmes clés de la classification des lésions cutanées. Le premier problème est l'utilisation efficace d'images à haute résolution avec des architectures standard pré-entraînées pour la classification d'images. Le second problème est le déséquilibre élevé des classes rencontré dans les ensembles de données multi-classes du monde réel.

Méthodes : Pour utiliser les images à haute résolution, nous proposons une nouvelle architecture d'attention basée sur les patches qui fournit un contexte global. Nouvelle architecture d'attention basée sur les patches qui fournit un contexte global entre de petits patches à haute résolution. Nous modifions trois architectures pré-entraînées et étudions la performance de l'attention basée sur les patches. L'attention basée sur les patches. Pour contrer les problèmes de déséquilibre des classes, nous comparons sur échantillonnage, l'échantillonnage par lot équilibré et la pondération de la perte spécifique à la classe. En outre, nous proposons une nouvelle méthode de pondération des pertes guidée par le diagnostic qui prend la méthode de pondération des pertes de la classe. Qui prend en compte la méthode utilisée pour l'annotation de la vérité du terrain. Résultats : Notre mécanisme d'attention basé sur les patches surpasse les méthodes précédentes et améliore la sensibilité avec une moyenne de 7 %. L'équilibrage des classes améliore significativement la sensibilité moyenne et nous montrons que notre méthode de pondération des pertes guidée par le diagnostic.

Nous montrons que notre méthode de pondération des pertes guidée par le diagnostic améliore la sensibilité moyenne de 3 % par rapport à l'équilibrage normal des pertes.

par rapport à l'équilibrage normal des pertes. Conclusion : Le nouveau mécanisme d'attention basé sur les patches peut être intégré dans des architectures préformées et fournit un contexte global entre les patches locaux tout en surpassant les autres méthodes basées sur les patches. Tout en surpassant les autres méthodes basées sur les patches. Par conséquent, les architectures prétraitées préformées peuvent être facilement utilisées avec des images à haute résolutionsans sous-échantillonnage. La nouvelle méthode de pondération des pertes guidée par le diagnostic de perte guidée par le diagnostic surpasse les autres méthodes et permet une formation efficace en cas de déséquilibre des classes. Importance : Les méthodes proposées améliorent la classification automatique des lésions cutanées. Elles peuvent être étendues à d'autres applications cliniques où les données haute résolution et le déséquilibre des classes sont pertinents.[24]

2.6 Conclusion

Pour conclure on peut résumer que les réseaux des neurones convolutionnelles (CNN) est une technique d'apprentissage profond et bien efficace pour une extraction automatique des caractéristiques d'un problème donné pour faire ensuite une classification. Son architecture se diffère cas par cas et elle est conçue d'une façon expérimentale avec une grande variabilité de choix des hyper-paramètres. Voyons dans les deux chapitres suivants une application d'un modèle CNN.

CHAPITRE 3

CONCEPTION DU SYSTEME

Dans les deux chapitres précédents, nous avons présenté plusieurs techniques de classification en intelligence artificiel ainsi que nous avons discuté les différents types d'apprentissages et de classification.

Ce chapitre est structuré en plusieurs sections, en premier lieu, nous décrivons l'architecture détaillé du système qui est basés sur l'approche de l'apprentissage profond (DeepLearning) de plus une combinaison d'un autre classifieur à savoir les forêts aléatoires (Random Forest).

Dans la deuxième section, nous présentons les différentes étapes de la phase d'apprentissage avec des explications.

La troisième section, sera dédiée pour les techniques et les mesures de performances utilisées pour évaluer le système.

3.1 Architecture générale du système

L'objectif de ce travail est de réaliser une classification d'images des types de cancers de peau. La base de données contient 7 types de cancers de peau nommés comme suit :

1:AKIEC - Actinickeratoses and intraepithelialcarcinoma / Bowensdisease.

2:BCC - Basal cellcarcinoma.

3:BKL - Benignkeratosis-likelesions.

4:DF – Dermatofibroma.

5:NV - Melanocyticnevi.

6:VASC - Vascularlesions.

7:MEL – Melanoma.

Pour faire différencier les 7 types nous allons utiliser deux classifieurs (SOFTMAX, RF) que nous allons les détailler dans cette section. Ci-dessous le schéma général du système proposé de la classification (Fig.7).

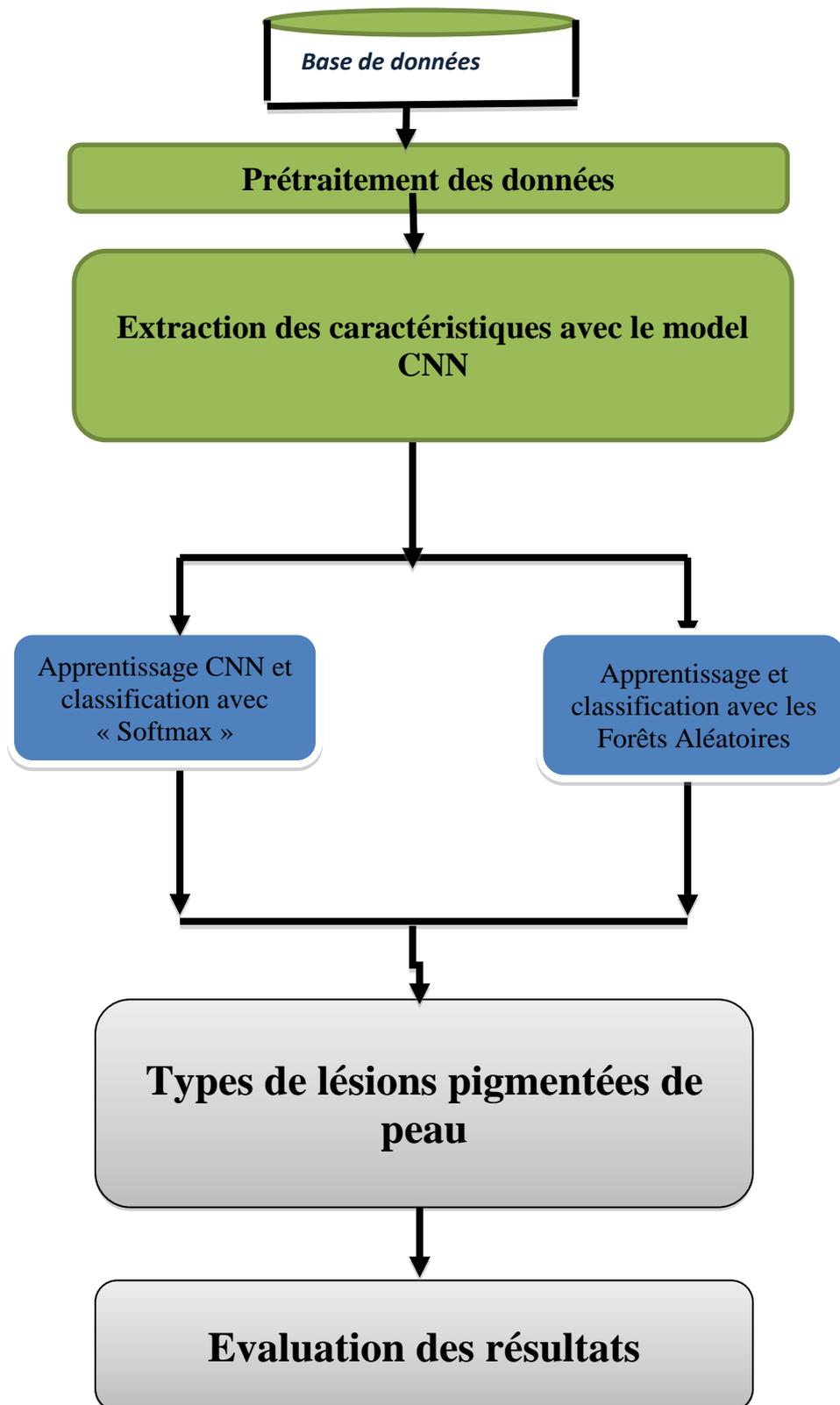


Fig. 7. Architecture générale du système.

3.2 Description de la base de données

La base donnée utilisée dans ce mémoire est HAM10000 (« HumanAgainst Machine with 10000 images »). Cette base de données multi classes comportant 10015 images dermatoscopiques de différentes populations, acquises et stockées selon différentes modalités (voir exemple dans Fig.8). Les cas comprennent une collection représentative de toutes les catégories diagnostiques importantes dans le domaine des lésions pigmentées.

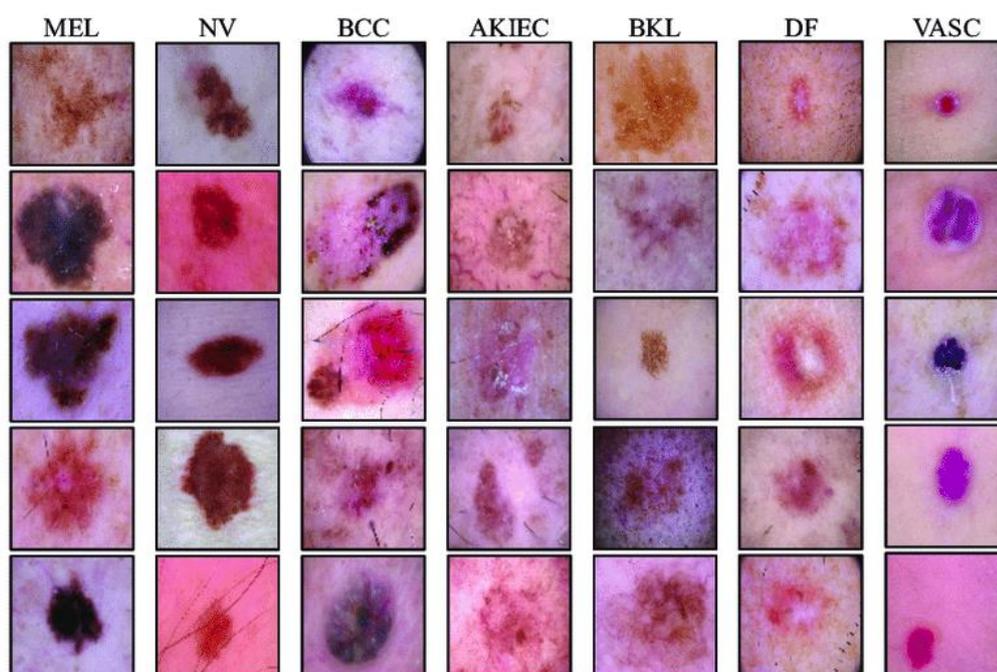


Fig. 8. Echantillons de la base de données HAM10000.[25]

Nous avons plus de 50% des lésions sur cette base de données sont confirmées par histopathologie.

Nous présentons également les informations de l'auteur de la base HAM10000

Dataset Persistent ID : doi:10.7910/DVN/DBW86T

Date de Publication : 2018-06-04

Titre : The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions.

Auteur : Tschandl, Philipp (Medical University of Vienna)

- ORCID: 0000-0003-0391-7810

Langue : Anglais

Sujet / Domaine : Médecine, santé et sciences de la vie; Informatique et sciences de l'information

Producteur : ViDIR Group (MedicalUniversity of Vienna). [25]

Avec les 10015 images que nous avons, on a également les métadonnées de nos cas (images) tel que : Image_ID, Type de lésion (DX), Age, Sexe, Localisation. (voir Fig.9)

| Index | lesion_id | image_id | dx ▲ | dx_type | age | sex | localization |
|-------|-------------|--------------|------|---------|-----|--------|-----------------|
| 216 | HAM_0003265 | ISIC_0025180 | bkl | histo | 35 | female | chest |
| 217 | HAM_0003265 | ISIC_0026107 | bkl | histo | 35 | female | chest |
| 218 | HAM_0004458 | ISIC_0033853 | bkl | histo | 80 | male | back |
| 219 | HAM_0004458 | ISIC_0033453 | bkl | histo | 80 | male | back |
| 220 | HAM_0004458 | ISIC_0034126 | bkl | histo | 80 | male | back |
| 221 | HAM_0003050 | ISIC_0033744 | bkl | histo | 70 | male | face |
| 222 | HAM_0003050 | ISIC_0033659 | bkl | histo | 70 | male | face |
| 223 | HAM_0003927 | ISIC_0032567 | bkl | histo | 60 | female | lower extremity |

Fig. 9. Les métadonnées de la base HAM10000.[25]

Dans ce mémoire nous intéressent uniquement à la classification des types de lésions pigmentées (les 7 classes) donc on va ignorer les autres informations sur la base de données et on va garder uniquement les images avec leurs étiquettes (Type de lésion).

La figure ci-dessous (Fig.10) représente la distribution des 10015 images par type de lésion.

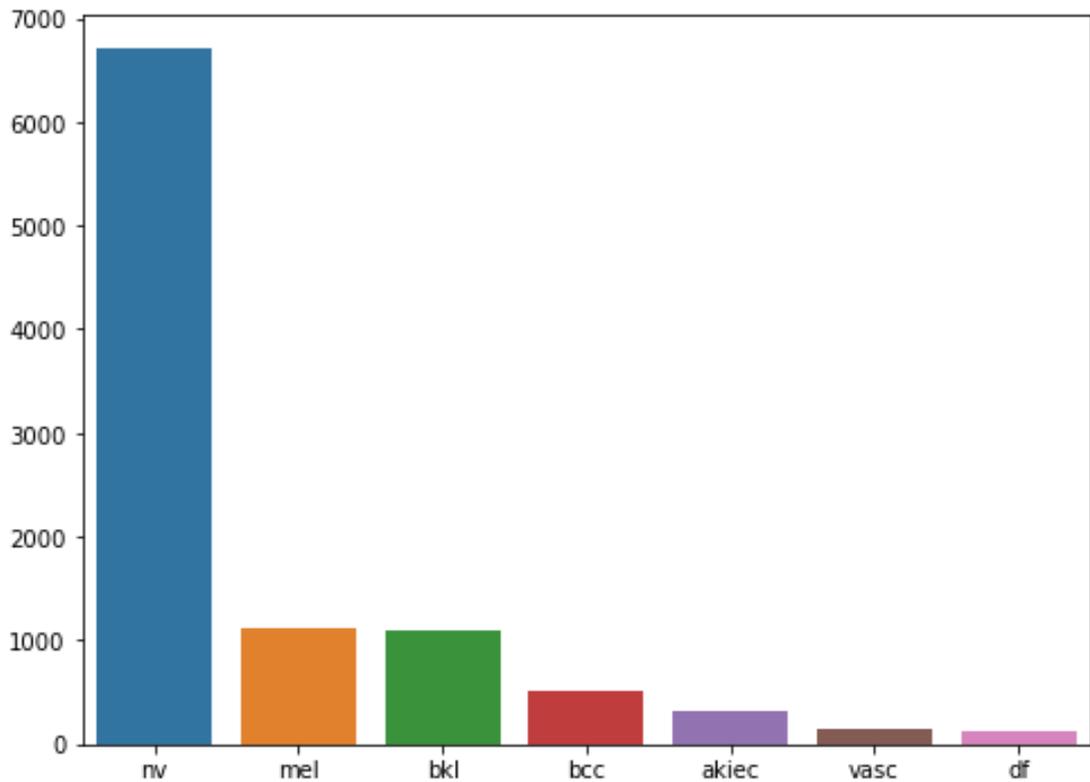


Fig. 10. Distribution des classes dans la base HAM10000.

3.3 Prétraitement des données

Les bons résultats qu'un classifieur automatique peut fournir reposent en grande partie, sur la phase de prétraitement. Les données issues d'un mauvais prétraitement vont mettre en péril la qualité du classifieur. [26]

Après la visualisation de notre base de données nous avons commencer d'abord par le choix de type de notre base de données qu'on va utiliser dans notre système de classification c.-à-d. comme il existe plusieurs versions de la base HAM10000 nous avons choisis pour notre système une base de données sous forme d'un fichier « .CSV » qui contient les valeurs des pixels des 10015 images redimensionnés en 28x28 pixels avec leurs type de lésion (étiquette). (Voir Fig.11) Ce choix nous gagnera un temps important pour le traitement sur notre système.

| Index | pixel0000 | pixel0001 | pixel0002 | pixel0003 | pixel0 |
|-------|-----------|-----------|-----------|-----------|--------|
| 0 | 192 | 153 | 193 | 195 | 155 |
| 1 | 25 | 14 | 30 | 68 | 48 |
| 2 | 192 | 138 | 153 | 200 | 145 |
| 3 | 38 | 19 | 30 | 95 | 59 |
| 4 | 158 | 113 | 139 | 194 | 144 |

Fig. 11. Visualisation du fichier CSV de notre BDD

Le schéma ci-dessous (Fig.12) illustre les étapes du prétraitement utilisé pour notre système de classification que nous allons les voir en détails par la suite.

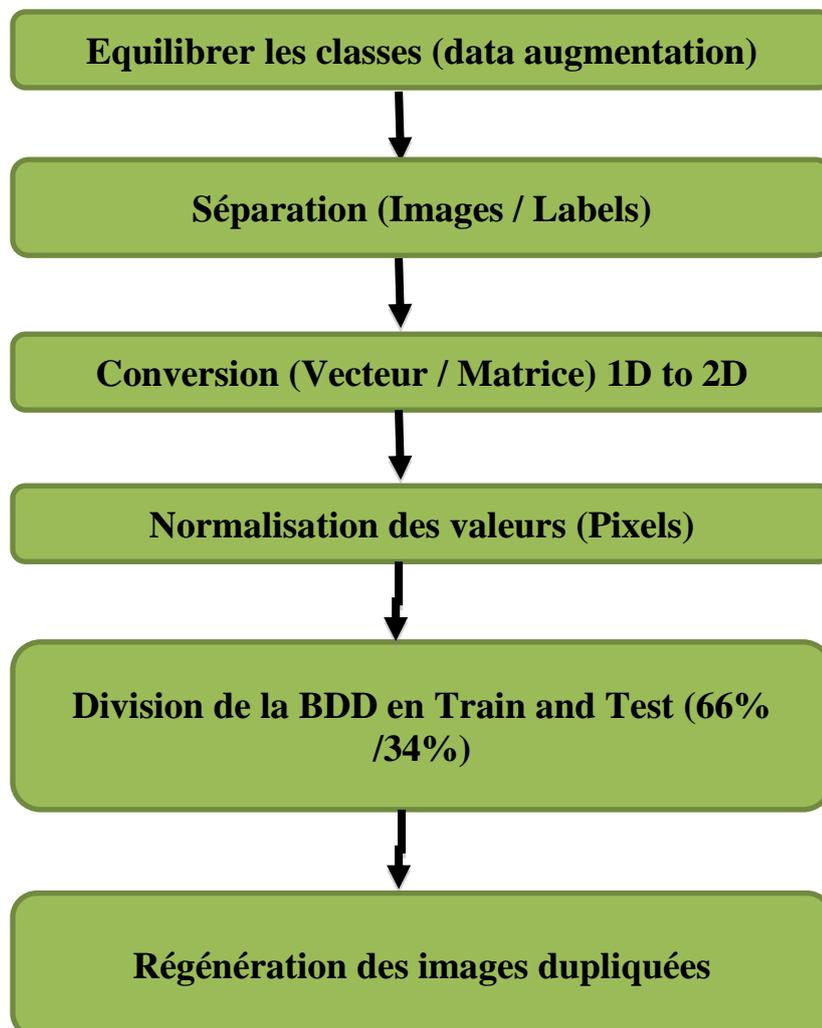


Fig. 12. Schéma des étapes de prétraitement

3.3.1 Equilibrer la distribution des classes de la base de données

La distribution des classes de la base de données est très mal équilibrée tel que nous avons 10015 images réparties comme suit :

NV-6705 images

MEL – 1113 images

BKL – 1099 images

BCC – 514 images

AKIEC- 327 images

VASC – 142 images

DF – 115 images.

Pour le moment nous allons dupliquer nos images pour faire équilibrer les classes. Après cette étape de prétraitement on obtient le graphe descriptif des classes (Fig.13).

Après cette étape nous avons obtenu un total de 50024 images réparties comme suit :

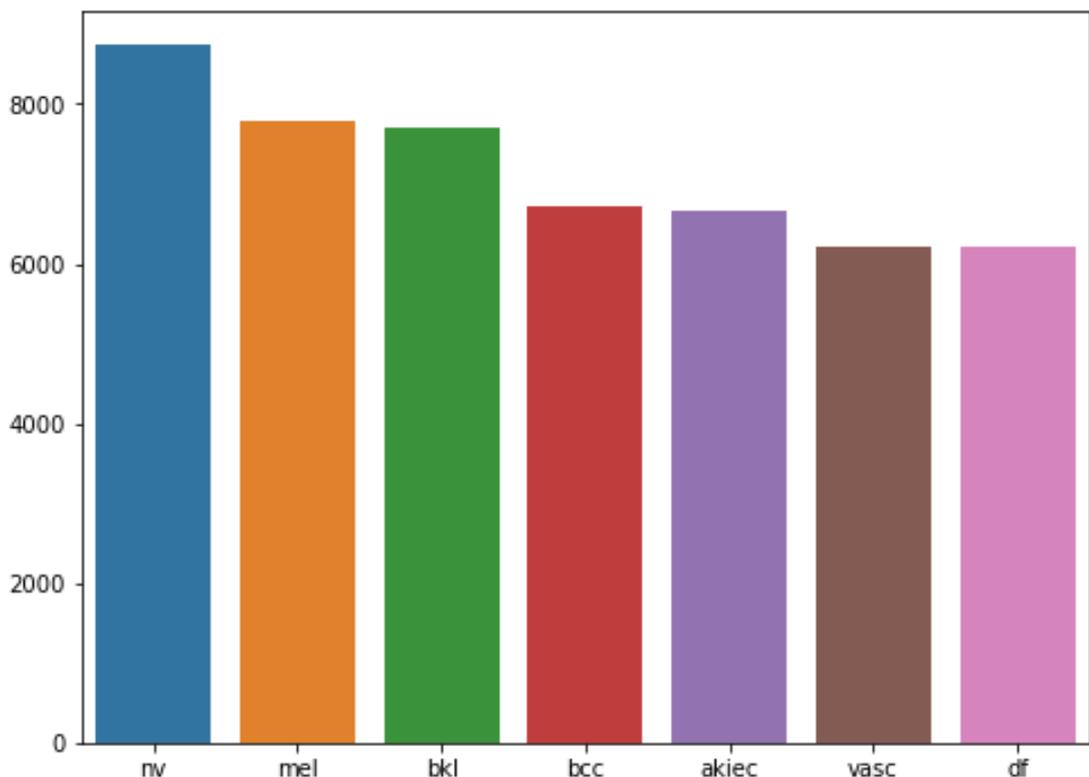


Fig. 13. Distribution des classes après duplication d'images

3.3.2 Séparation des données

Cette étape est incontournable elle s'agit de séparer les étiquettes (Labels) des images.

X : IMAGES.

Y : LABELS « étiquettes ».

3.3.3 Conversion des données 1D en images

Comme nous avons le fichier « .CSV » sous la structure suivante :

(Index, Pixel values, Labels).

Les images sont de taille 28x28 pixels en RGB.

Cette étape consiste à la construction des images.

3.3.4 Normalisation des données

Nos images sont de dimensions de 28x28 pixels en RGB donc les valeurs de pixels sont dans un intervalle de [0...255]. Nous pouvons donc normaliser ces valeurs dans un intervalle de [0...1] en nombres réels. Le but de la normalisation est de faciliter les calculs et le traitement.

Donc on peut appliquer simplement la formule :

$$Px = Px/255.00 \quad (6) [27]$$

Px : valeur du pixel

3.3.5 Répartition de la BDD (Apprentissage/Test)

Voyons maintenant le partage de notre BDD pour la phase d'entraînement et la phase de test. En utilisant la technique de « Random Split ».

Pour cela on prend 2/3 pour Apprentissage et 1/3 pour Test. Soit :

-33516 images pour l'Apprentissage.

-16508 images pour le Test.

3.3.6 Régénération des images dupliqués

Dans cette étape on fait appelle a l`algorithme de « ImageDataGenerator » pour régénérer les images qu`on a dupliqué. Cet algorithme fera les effets suivants sur les images :

- Rotation
- Zoom avant
- Zoom arrière
- Déplacement haut/bas
- Déplacement gauche/droite

3.4 Extraction de caractéristiques et apprentissage CNN

Parmi les points forts de l`apprentissage profond, c`est l`extraction automatique des caractéristiques des données. Ces derniers on va les utiliser pour entrainer notre modèle de classification. Après plusieurs essaies nous avons choisi le modèle CNN résumé sur le tableau ci dessous (Fig.14) :

| Couche (Type) | Taille | Paramètres |
|---------------|--------|------------------------|
| CONV2D | 16 | Activation: Relu |
| CONV2D | 32 | Activation: Relu |
| MaxPooling2D | 2 | |
| CONV2D | 32 | Activation: Relu |
| CONV2D | 64 | Activation: Relu |
| MaxPooling2D | 2 | |
| CONV2D | 64 | Activation: Relu |
| CONV2D | 64 | Activation: Relu |
| MaxPooling2D | 2 | |
| Flatten | | |
| Dense | 64 | Activation: Relu |
| Dense | 64 | Activation: Relu |
| Dense | 32 | Activation: Relu |
| Dense | 32 | Activation: Relu |
| Dense | 7 | Activation: Softmax |

Tab. 1. Architecture du modèle CNN

Avec cette architecture nous allons entrainer notre modèle pour la classification et extraire les caractéristiques pertinentes que nous allons les utiliser par la suite pour entrainer les deux classifieur «Softmax» et «Les Forêts aléatoires (RF)».

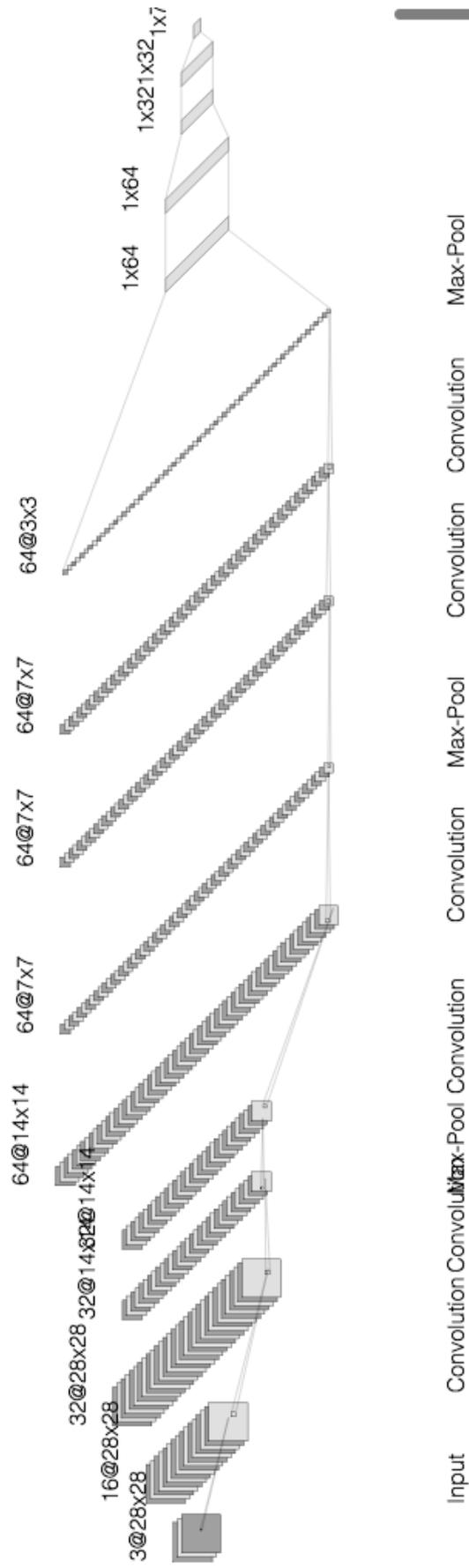


Fig.14. Architecture du modèle CNN avec LeNet Style

3.5 Classification avec Softmax

Nous pouvons maintenant prédire une classification des images, pour cela on utilise le classifieur « Softmax » pour les 7 classes des lésions.

3.6 Classification avec les forêts aléatoires

Prenons les caractéristiques générés de notre modèle de réseaux de neurones convolutionnels(CNN), qu'on va utiliser pour entrainer notre classifieur des forêts aléatoires.

3.7 Les mesures de performances

Dans la classification supervisée, les performances mesurés pour évaluer le succès du système sont : le taux de classification, le taux d'erreur, la matrice de confusion, ainsi le recall et la précision dont l'objectif de prendre le meilleur taux de classification obtenu. Nous proposons ici l'explication des mesures utilisés :

❖ **Le taux d'erreur :**

Est la proportion de mal classés, il estime la probabilité de mal classer un individu pris au hasard dans la population.

❖ **La précision :**

C'est le rapport entre le nombre de vrais positifs et la somme des vrais positifs et des faux positifs.

❖ **Recall:**

Combien de positifs le modèle réponde-t-il?

Un Recall de 1 signifie que tous les exemples positifs ont été trouvés.

❖ **Taux de classification :**

Le nombre d'exemples bien classés, en valeur absolue, puis en pourcentage du nombre total d'exemples.

3.8 Conclusion

Dans ce chapitre nous avons présenté la conception réalisée du système, où les différentes phases ont été détaillées, ainsi l'approche CNN et les classifieurs supervisés « Softmax et Forêts Aléatoires » pour la détection de types de lésions pigmentées, avec les mesures de performances appliqué pour évaluer notre système de classification.

Le chapitre suivant sera consacré pour l'implémentation du système.

CHAPITRE 4

IMPLEMENTATION DU SYSTEME

Dans le chapitre précédent, nous avons présenté la conception de notre système de classification des images de peau ainsi que nous avons montré les différentes étapes du prétraitement ainsi que l'architecture détaillée de notre système avec une présentation détaillée de la base de données utilisée et aussi les explications de chaque étape de notre système conçu.

Dans ce chapitre nous exposerons les résultats obtenus par les deux méthodes d'apprentissages appliquées, et il est structuré en plusieurs sections. Dans la première section, nous décrivons l'environnement utilisé pour notre système ainsi que quelques pseudo codes implémentés dans notre application.

Dans la deuxième section, nous présentons la discussion des résultats obtenus par l'approche CNN et les forêts aléatoires.

La troisième section, sera dédiée pour les autres bases de données utilisées avec des comparaisons entre eux.

4.1 Environnement de travail

Dans notre étude, nous avons utilisé l'environnement de programmation Python à travers la distribution Anaconda-Navigator ceci à cause de la flexibilité et du nombre conséquent de ressources disponibles à leur sujet. Nous avons utilisé la librairie Scikit-learn[29] pour partager la base de données en (Train/Test), l'implémentation de l'algorithme de classification des forêts aléatoires et application des fonctions des mesures de performances décrites dans le **chapitre 3 – section 7**. La librairie pour python Keras[30] nous a servi pour application de l'algorithme de ImageDataGenerator pour le Data Augmentation et l'implémentation des architectures des réseaux profonds et l'extraction des fonctionnalités décrites dans le **chapitre 3 – section 3 et 4**. A tout cela, s'ajoute les librairies Numpy, Matplotlib, Pillow, Pandas et Seaborn qui ont été mises à notre disposition pour nos implémentations dans le prétraitement, calcul et affichage des résultats et mesures de performances que nous allons les voir par la suite.

4.2 Résultats et discussions

Cette section représente les résultats obtenus après l'implémentation du système. Comme mentionné précédemment, nous avons réparti notre base en prenant 2/3 pour apprentissage et 1/3 pour validation.

4.2.1 Classification avec Softmax

Notre architecture CNN proposée nous a permis de récupérer 130 663 paramètres prêts à être entraînés. Voir (Fig.15).

```
=====  
Total params: 130,663  
Trainable params: 130,663  
Non-trainable params: 0  
=====
```

Fig. 15. Sommaire architecture CNN- Trainable paramètres.

Après le lancement d'entraînement du réseau CNN proposé en appliquant 10 Epochs et un batch size de 64 images par itération. On vous présente les courbes qui représentent les taux de classification dans la figure (Fig.16) et les taux d'erreur dans la figure (Fig.17) durant notre entraînement.

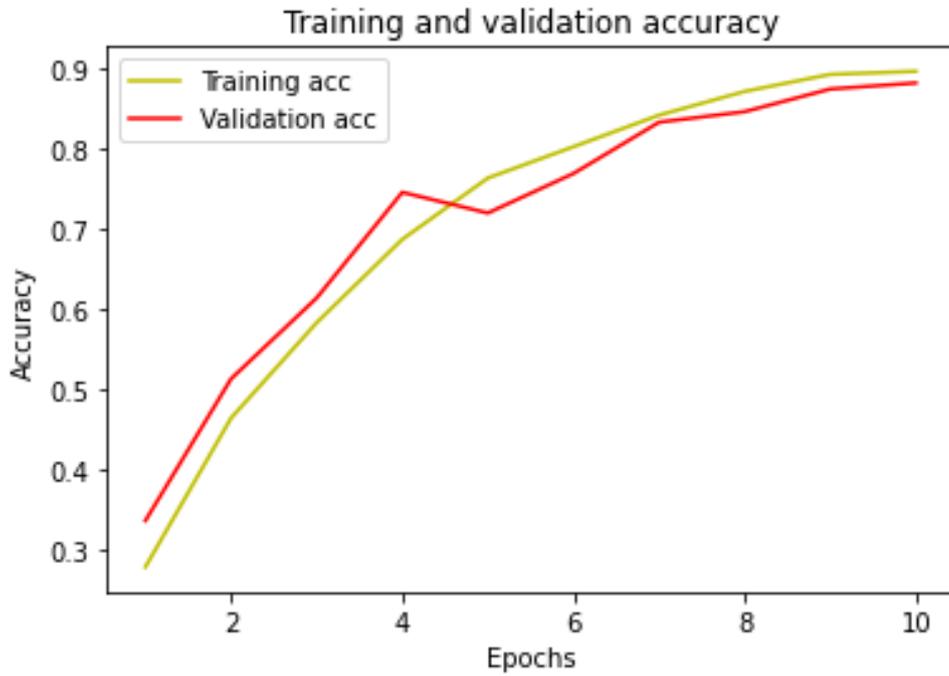


Fig. 16. Les taux de précision durant l'entraînement avec 10 epochs.

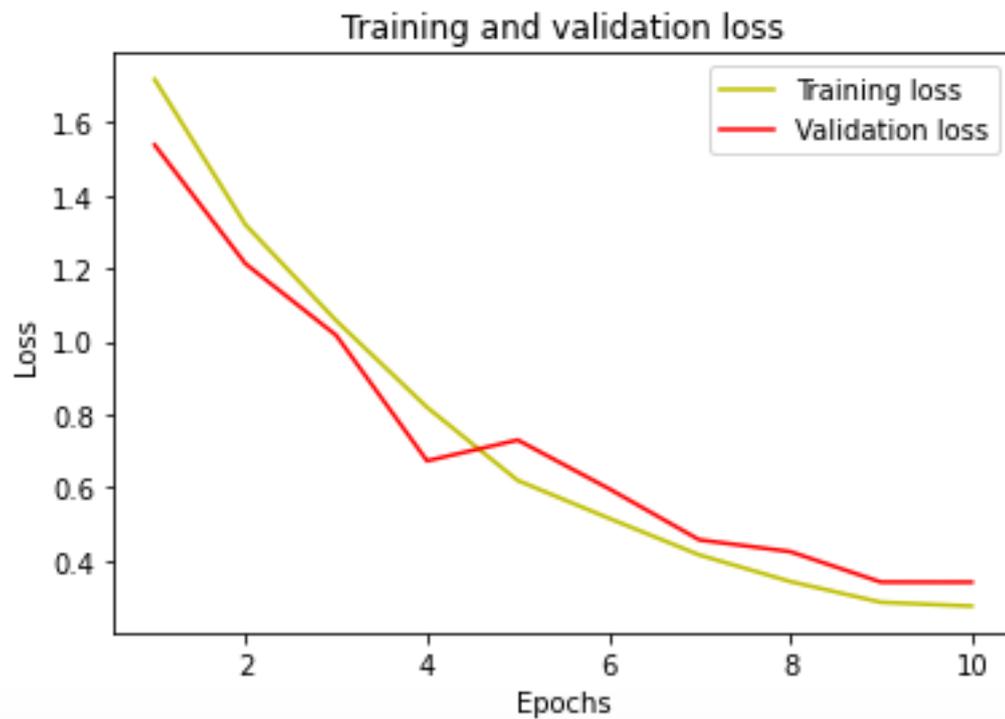


Fig. 17. Les taux d'erreur durant l'entraînement avec 10 epochs.

D'après nos deux graphes obtenus on peut déjà préjuger que notre modèle est en bonne voie pendant son entraînement. Vu que la courbe est de forme exponentielle. On remarque également que les courbes d'entraînement et de validation sont les deux de la même direction.

Cela reflète un très bon apprentissage.

Le taux de classification pour cet apprentissage est de : 89,35 %.

Ci-dessous la matrice de confusion de classification avec « Softmax » (Fig.18) qui nous illustre le nombre de images bien classés et les images mal classés dans chaque une de nos 7 classes.

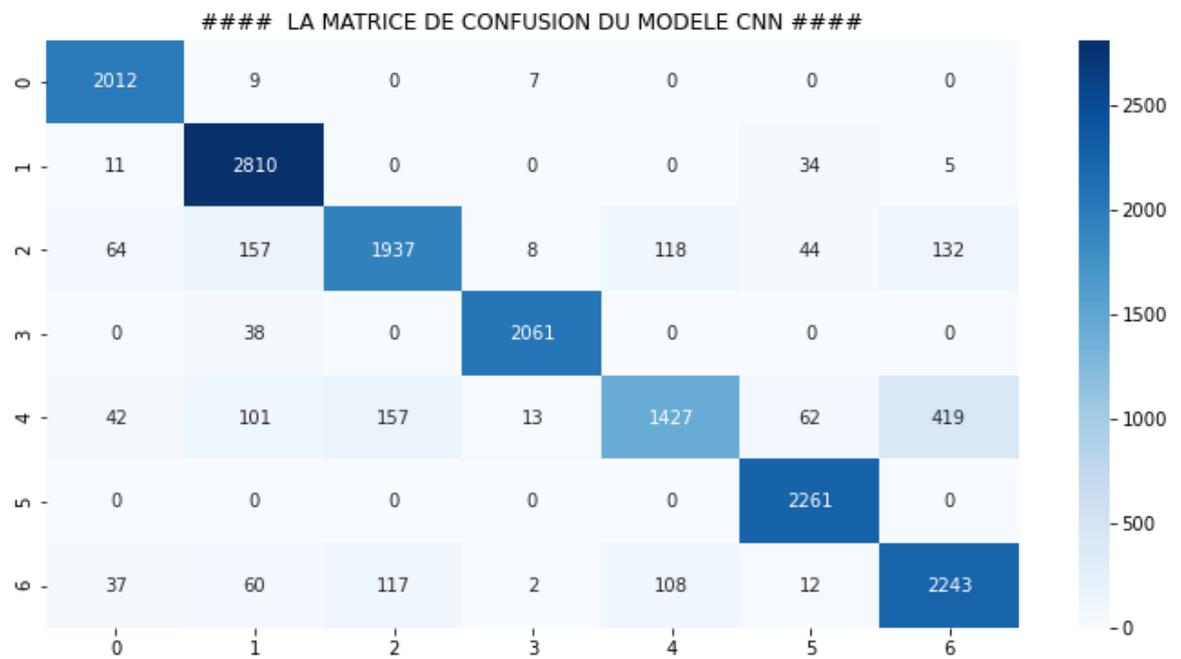


Fig. 18. Matrice de confusion avec classification Softmax.

D'après la matrice de confusion obtenue illustrée dans la figure (Fig.18) on constate qu'il y a 1746 images mal classés sur un total de 11061 images de notre base de test.

Voilà également le rapport de classification qui regroupe l'ensemble des résultats obtenus (Fig.19).

| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| nv | 0.93 | 0.99 | 0.96 | 2028 |
| mel | 0.89 | 0.98 | 0.93 | 2860 |
| bkl | 0.88 | 0.79 | 0.83 | 2460 |
| bcc | 0.99 | 0.98 | 0.98 | 2099 |
| akiec | 0.86 | 0.64 | 0.74 | 2221 |
| vasc | 0.94 | 1.00 | 0.97 | 2261 |
| df | 0.80 | 0.87 | 0.83 | 2579 |
| accuracy | | | 0.89 | 16508 |
| macro avg | 0.90 | 0.89 | 0.89 | 16508 |
| weighted avg | 0.89 | 0.89 | 0.89 | 16508 |

Fig. 19. Rapport de classification pour le modèle CNN.

4.2.2 Autres essais d'architecture CNN

Afin d'arriver à notre architecture utilisée dans notre système de classification, nous avons procédé plusieurs essais. Parmi eux sont résumés dans le tableau ci-dessous qui contient la configuration utilisée :

| | Configuration | Accuracy |
|-----------------------------|--|-------------------------------|
| Train 66 % Test 34 % | -Taille de l'image: 28x28 px -Réseau CNN : 32 filtres 2D – activ : ReLu 16 filtres 2D – activ : ReLu 8 filtres 2D – activ : ReLu Dense 8 – activ : Softmax -Apprentissage : 10 epochs ; batch size : 64 | CNN : 68,57 % RF : 64,04 % |
| | -Taille de l'image: 28x28 px -Réseau CNN : 64 filtres 2D – activ : ReLu 32 filtres 2D – activ : ReLu | CNN : 83,20 % RF : 87,56 % |

| | | |
|--|---|-------------------------------|
| | 16 filtres 2D – activ : ReLu Dense 32 – activ : Softmax -Apprentissage : 10 epochs ; batch size : 64 | |
| | -Taille de l`image: 28x28 px -Réseau CNN : 64 filtres 2D – activ : ReLu 32 filtres 2D – activ : ReLu 16 filtres 2D – activ : ReLu 8 filtres 2D – activ : ReLu Dense 8 – activ : Softmax -Apprentissage : 10 epochs ; batch size : 64 | CNN : 80,44 % RF : 76,31 % |

Tab. 2. Essais d`architectures CNN avec résultats obtenus.

4.2.3 Classification avec les forêts aléatoires

En utilisant les caractéristiques extraites du modèle CNN, on va les passer avec le classifieur des forêts aléatoires avec un nombre des arbres égale a 50.

Après le lancement d`entrainement, le classifieur nous a donné un taux de précision de 97,68 % (Fig.20)

```

...: from sklearn import metrics
...: print ("Accuracy R.F = ", metrics.accuracy_score(Y_test,
...: prediction_RF))
...:
- Accuracy R.F = 0.9768597043857523
In [15]:

```

Fig. 20. Taux de précision avec le classifieur des forêts aléatoires.

Ci-dessous la matrice de confusion pour bien voir les résultats de chaque classe (Fig.21)

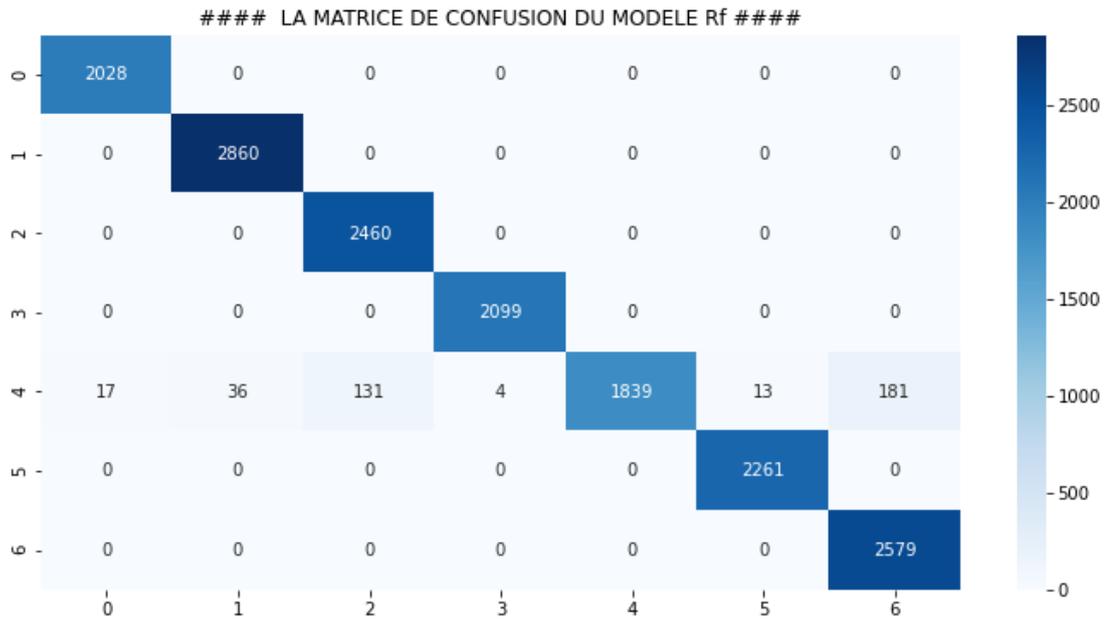


Fig. 21. Matrice de confusion modèle Forêts Aléatoires.

D'après la matrice de confusion montrée ci-dessus, on remarque que nous avons 382 images mal classées. Par ailleurs la classe numéro 4 avait un taux de classification équivalent à 100%. En outre le total des échantillons mal classés sont répartis dans la classe 4. Cela est due à une grande ressemblance de quelques lésions avec la classe numéro 4, à travers lesquels le classifieur de forêts aléatoires n'a pas pu faire la discrimination entre les caractéristiques de ces images extraites par notre modèle CNN adopté.

Ci-dessous le rapport de classification qui regroupe les mesures de performances du classifieur des forêts aléatoires (Fig.22).

| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| nv | 0.99 | 1.00 | 1.00 | 2028 |
| mel | 0.99 | 1.00 | 0.99 | 2860 |
| bkl | 0.95 | 1.00 | 0.97 | 2460 |
| bcc | 1.00 | 1.00 | 1.00 | 2099 |
| akiec | 1.00 | 0.83 | 0.91 | 2221 |
| vasc | 0.99 | 1.00 | 1.00 | 2261 |
| df | 0.93 | 1.00 | 0.97 | 2579 |
| accuracy | | | 0.98 | 16508 |
| macro avg | 0.98 | 0.98 | 0.98 | 16508 |
| weighted avg | 0.98 | 0.98 | 0.98 | 16508 |

Fig. 22. Rapport de classification pour les Forêts Aléatoires.

4.2.4 Comparaison entre les classifieurs Softmax et Forêts Aléatoires

D'après les résultats obtenus, on peut dire que le RF est plus performant que le softmax en terme d'efficacité de classification. Tel que l'amélioration de ce taux nous a permis de gagner plus de 10% par rapport a celui de Softmax.

| Classifieur | Taux de classification (en moyenne) |
|---------------|-------------------------------------|
| Softmax | 89,64 % |
| Random Forest | 98,13 % |

A notre avis, on justifie le taux de classification des forêts aléatoires grâce à son efficacité de classification sur les bases de données de multi classes

4.3 Comparaison du système / Related Works

Suite à des recherches sur net on a trouvé quelques travaux réalisés pour la classification de la base de données HAM10000. Le tableau ci dessous résume les techniques et approches utilisés pour la classification (Tab.3) :

| Related Works | Configuration | Taux de classification |
|----------------------|--|--|
| | <p>-Taille de l'image: 128x128 px</p> <p>-Apprentissage : 80 %</p> <p>- Test : 20%</p> <p>-Réseau CNN :</p> <p>EfficientNet, $\gamma = 1$</p> <p>2 stage EfficientNet, $\gamma = 1$</p> | <p>EfficientNet: 90,10 %</p> <p>EfficientNet 2 stage: 90,50 %</p> <p>[22]</p> |
| | <p>-Taille de l'image: 512x512 px</p> <p>-Réseau CNN :</p> <p>32 filtres 2D – activ : ReLu</p> <p>32 filtres 2D – activ : ReLu</p> <p>32 filtres 2D – activ : ReLu</p> <p>64 filtres 2D – activ : ReLu</p> <p>256 filtres 2D – activ : ReLu</p> <p>256 filtres 2D – activ : ReLu</p> <p>Dense 4096 – activ : ReLu</p> <p>Dense2 5 – activ : Segmoid</p> <p>-Apprentissage :</p> <p>50 epochs ;</p> | <p>CNN : 90,51 %</p> <p>Random Forest :65.90%</p> <p>XGBoost :65.15%</p> <p>Support Vector Classifier 65.86%</p> <p>[23]</p> |
| | | |

| | | | |
|--|-----------------------|--|---|
| | | <p>-Taille de l'image: 224x224 px</p> <p>-Réseau CNN :</p> <p>Inception-V3</p> <p>Densenet-121</p> <p>SE-Resnext50</p> | <p>Inception-V3 83.9 %</p> <p>Densenet-121 85.4 %</p> <p>SE-Resnext50 (SE-RX) 84.5 %</p> <p>[24]</p> |
| | Notre Approche | <p>-Taille de l'image : 28x28 px.</p> <p>-Réseau CNN :</p> <p>architecture détaillée dans le tableau (Tab.1) (Page 45)</p> | <p>Softmax 89,64 % (en moyenne)</p> <p>Randomforests 98,13 % (en moyenne)</p> |

Tab. 3. Notre système avec les systèmes connexes.

4.4 Conclusion

Dans ce chapitre, nous avons implémenté notre base de données en deux phases : la première phase est conçue pour la classification en utilisant le classifieur Softmax et la deuxième phase pour la classification en utilisant les forêts aléatoires, ensuite nous avons validé notre système à partir des mesures appliquées pour choisir le meilleur taux de classification et faire une étude comparative entre elles en basant sur les mêmes mesures de performances appliquées pour choisir le meilleur classifieur.

A travers ces expérimentations, nous pouvons dire que les deux classifications ont donné de bons résultats pour notre base de données.

CONCLUSION ET PERSPECTIVES

Chaque personne est exposée aux nombreuses maladies. Certaines maladies nécessitent un diagnostic immédiat par le médecin, L'objectif principal de notre projet est la classification des lésions cutanées de cancer de peau dans l'objectif de faire différencier 7 types de lésions avec un taux de précision maximal.

Résumé des contributions

Dans ce mémoire, le modèle CNN proposé atteint un meilleur taux de classification par rapport aux autres modèles d'apprentissage par transfert. De plus, la méthode proposée a été évaluée sur la base de données nommé « HAM 10000 » dans lequel nous avons obtenu une bonne précision pour l'apprentissage et pour le test. En outre, la base de données déséquilibrée et l'absence d'un grand nombre d'images ont empêché le modèle d'acquérir une meilleure précision. Par conséquent, nous avons équilibré l'ensemble de données pour les deux niveaux, ce qui améliore la précision de la classification. Nous avons également utilisé un deuxième classifieur qui est des forêts aléatoires ce qui nous a donné un taux de précision excellent par rapport à celui de Softmax du CNN.

Perspectives de recherche

À l'avenir, nous travaillerons sur une base de données avec plus de lésions cutanées étiquetées qui est en cours de construction au niveau de l'hôpital Ibn Rochd –Annaba-service Oncologie. Tel qu'on construisant une architecture CNN, y compris les étapes de prétraitement pour obtenir une meilleure précision de prédiction et de classification.

REFERENCES

[1] Surden, Harry, Machine Learning and Law (March 26, 2014). Washington Law Review, Vol. 89, No. 1, 2014, Available at SSRN: <https://ssrn.com/abstract=2417415>

[2] Artificial Intelligence A Modern Approach - 2010, 2003 by Pearson Education, Inc

[3] Ian Witten, Eibe Frank, Mark Hall, "Data Mining: Practical Machine Learning Tools and Techniques 3rd Edition". eBook ISBN: 9780080890364

[4] Chris Huntingford, Elizabeth S Jeffers, (2019). "Machine learning and artificial intelligence to aid climate change research and preparedness". <https://doi.org/10.1088/1748-9326/ab4e55>

[5] Vladimir Nasteski, (2017). "An overview of the supervised machine learning methods". DOI: [10.20544/HORIZONS.B.04.1.17.P05](https://doi.org/10.20544/HORIZONS.B.04.1.17.P05)

[6] Alsharif, M.H.; Kelechi, A.H.; Yahya, K.; Chaudhry, (2020). "S.A. Machine Learning Algorithms for Smart Data Analysis in Internet of Things Environment: Taxonomies and Research Trends". <https://doi.org/10.3390/sym12010088>

[7] Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, (2013). "Real-time human pose recognition in parts from single depth images". *Communications of the ACM*, 56(1), 116-124.

[8] Richard A. Berk, (2016). "Statistical Learning from a Regression Perspective", eBook ISBN: 978-3-319-44048-4.

[9] Guillaume Cleuziou, (2004). "Une méthode de classification non-supervisée pour l'apprentissage de règles et la recherche d'information". Autre [cs.OH]. Université d'Orléans.

[10] Pierre-Louis GONZALEZ, (2008). "MÉTHODES DE CLASSIFICATION", Cnam.

- [11] Swain, Debabala, Pattnaik, PrasantKumar, Athawale, Tushar, (2020). "Machine Learning and Information Processing". Proceedings of ICMLIP 2020
- [12] LISA Lab,(2013), "Convolutional Neural Networks (LeNet) – DeepLearning 0.1 documentation". DeepLearning 0.1
- [13] Habibi, Aghdam, Hamed (2017). "Guide to convolutional neural networks : a practical application to traffic-sign detection and classification". ISBN 9783319575490. OCLC 987790957.
- [14] Venkatesan, Ragav; Li, Baoxin, (2017). "Convolutional Neural Networks in Visual Computing: A Concise Guide. CRC Press". ISBN 978-1-351-65032-8.
- [15] MasakazuMatsugu, KatsuhikoMori, YusukeMitari, YujiKaneda, (2003). « Subjectindependent facial expression recognition withrobust face detectionusing a convolutional neural network ». [https://doi.org/10.1016/S0893-6080\(03\)00115-1](https://doi.org/10.1016/S0893-6080(03)00115-1)
- [16] Anastasia Butskova, (2020). "Skin lesions: HAM 10000 dataset.BIODS220"Project.arXiv:1512.03385v1.
- [17] Trevor Hastie, Robert Tibshirani et Jerome Friedman, (2009). "The Elements of Statistical Learning : Data Mining, Inference, and Prediction, 2e éd".ISBN-13978-0387848570.
- [18] Li Wan, Matthew Zeiler, Sixin Zhang, Yann Le Cun, Rob Fergus, (2013). « Regularization of Neural Networks using DropConnect » PMLR 28(3):1058-1066.
- [19] Matthew D. Zeiler, Rob Fergus, (2013), "Stochastic Pooling for Regularization of Deep Convolutional Neural Networks". [arXiv:1301.3557v1](https://arxiv.org/abs/1301.3557v1).
- [20] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever et Ruslan R. Salakhutdinov, (2012). « Improving neural networks by preventingco-adaptation of feature detectors ».arXiv:1207.0580.

- [21] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, RuslanSalakhutdinov, (2014). « Dropout: A Simple Way to Prevent Neural Networks fromOverfitting ».15(56):1929–1958.
- [22] RishuGarg, SaumilMaheshwari and Anupam Shukla, (2019)."Decision Support System for Detection and Classificationof Skin Cancer using CNN". arXiv :1912.03798
- [23] Anastasia Butskova, (2020)."Skin lesions: HAM 10000 dataset.BIODS220" Project.arXiv:1512.03385v1.
- [24] Nils Gessert, ThiloSentker, FredericMadesta, (2019)."Skin Lesion Classification UsingCNNswith Patch-Based Attention and Diagnosis-GuidedLossWeighting". arXiv:1905.02793v2.
- [25] Tschandl, Philipp, 2018, "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of commonpigmented skin lesions", <https://doi.org/10.7910/DVN/DBW86T>.
- [26] Nils Gessert, ThiloSentker, FredericMadesta, (2019)."Skin Lesion Classification UsingCNNswith Patch-Based Attention and Diagnosis-GuidedLossWeighting". arXiv:1905.02793v2.
- [27] Marref Nadia, (2013). "Apprentissage Incrémental & Machines à Vecteurs Supports ", Magister en Informatique, Université Hadj Lakhdar – Batna, pp .10-60.
- [28] Rafael C. González, Richard Eugene Woods, (2007). "Digital Image Processing. Prentice Hall". p. 85. ISBN 978-0-13-168728-8.
- [29] Pedregosa, F., (2011)."Scikit-learn: Machine learning in Python. Journal of machine learning research": p. 2825-2830. ^[1]_[SEP]
- [30] Chollet, F, (2015)."Keras". ^[1]_[SEP]link : <https://github.com/fchollet/keras>

ANNEXE A

| Tests | Train_Accuracy | Val_Accuracy | Train_Loss | Val_Loss |
|----------------|----------------|----------------|----------------|----------------|
| 1 | 91,26 % | 90,31 % | 23,71 % | 27,69 % |
| 2 | 90,77 % | 90,56 % | 24,55 % | 25,75 % |
| 3 | 91,38 % | 91,29 % | 22,45 % | 25,13 % |
| 4 | 90,55 % | 88,95 % | 25,55 % | 29,78 % |
| 5 | 89,52 % | 90,62 % | 28,07 % | 25,30 % |
| 6 | 90,79 % | 87,76 % | 24,12 % | 34,47 % |
| 7 | 90,87 % | 87,02 % | 24,76 % | 35,36 % |
| 8 | 90,27 % | 88,98 % | 26,21 % | 29,82 % |
| 9 | 91,47 % | 91,39 % | 22,53 % | 24,35 % |
| 10 | 91,88 % | 89,52 % | 21,84 % | 27,23 % |
| Moyenne | 90,87 % | 89,64 % | 24,37 % | 28,48 % |
| Max | 91,88 % | 91,39 % | 28,07 % | 35,36 % |

Tab. 4. Résumé des performances du classifieur Softmax.

| Tests | Taux de classification |
|----------------|------------------------|
| 1 | 98,23 % |
| 2 | 98,27 % |
| 3 | 98,48 % |
| 4 | 97,77 % |
| 5 | 98,34 % |
| 6 | 98,01 % |
| 7 | 98,08 % |
| 8 | 97,98 % |
| 9 | 98,11 % |
| 10 | 98,06 % |
| Moyenne | 98,13 % |
| Max | 98,48 % |

Tab. 5. Résumé des performances du classifieur des forêts aléatoires.

ANNEXE B



Fig. 23. Interface de l'application.

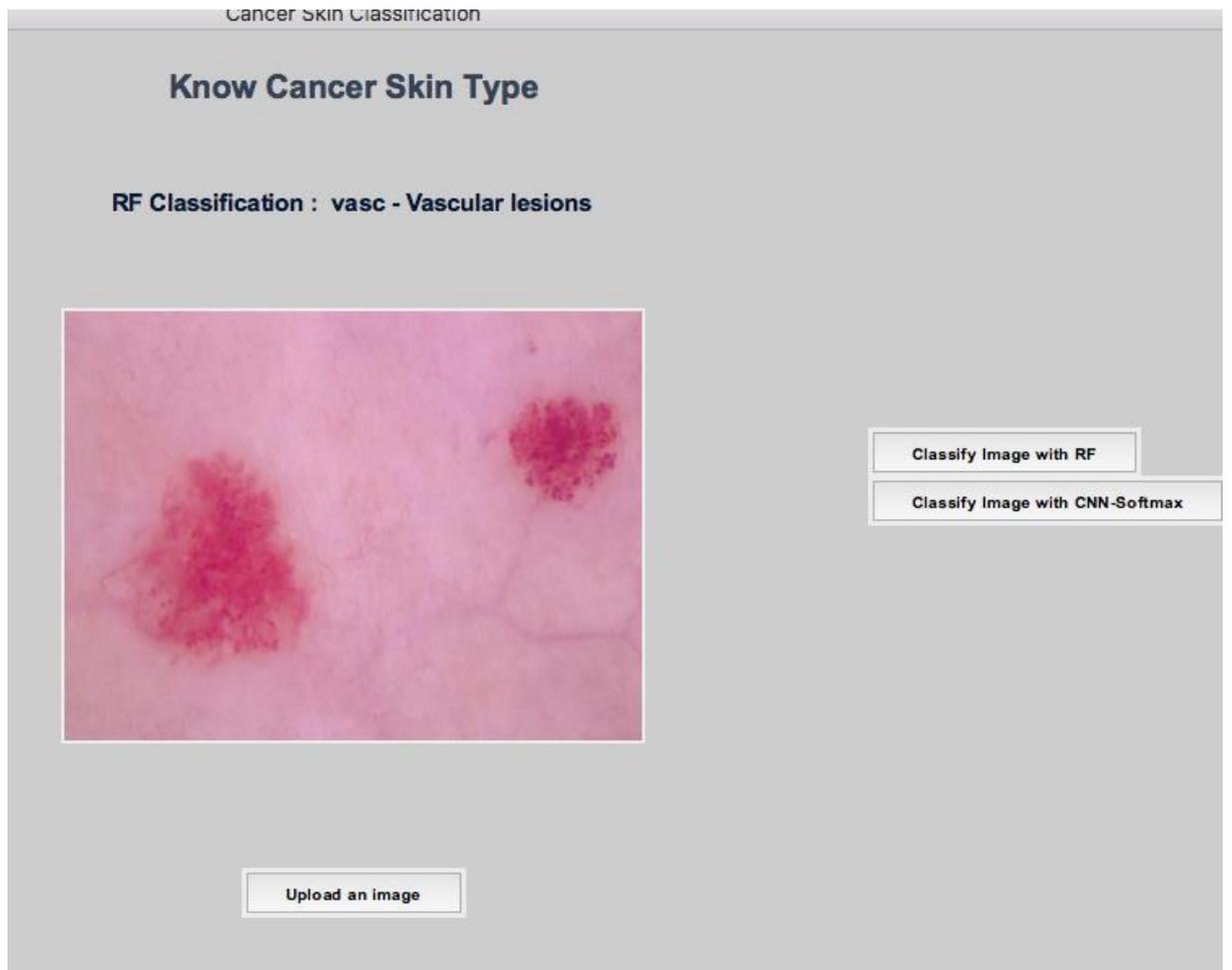


Fig. 24. Exemple de classification avec les forêts aléatoires.

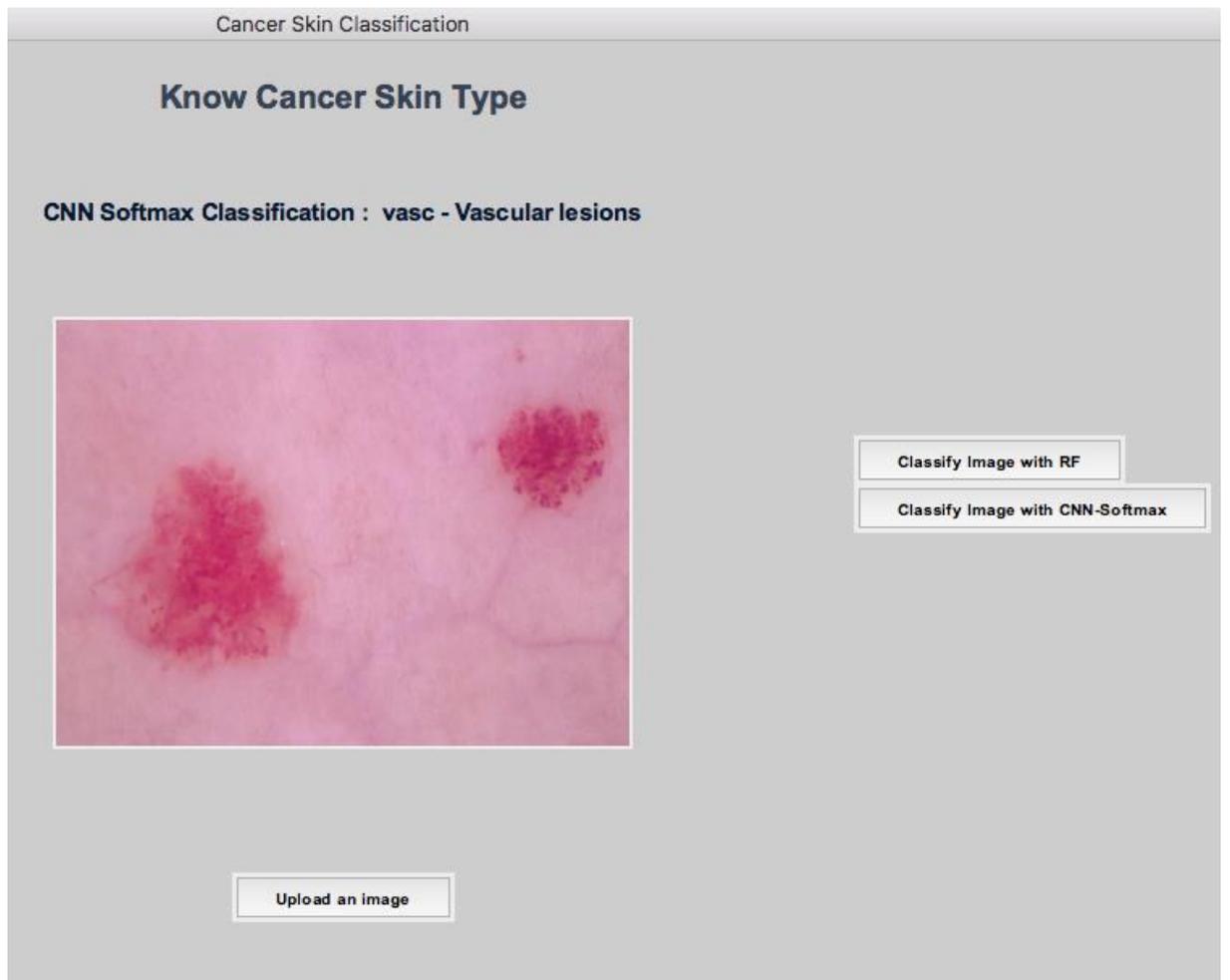


Fig. 25. Exemple de classification avec Softmax.

Résumé

Ce mémoire traite un problème de classification des images de peau de lésions cutanées (cancer de peau) sur une base de données de 10000 images appelée « HAM10000 » qui contient 7 types de lésions. Nous avons fait appel à l'algorithme « ImageDataGenerator » pour faire équilibrer et augmenter les images de la BDD (Data Augmentation). Concernant la classification, nous avons utilisé l'apprentissage profond avec le classifieur « Softmax », ainsi que les forêts aléatoires « Randomforests » comme un deuxième classifieur. Nous avons obtenu de très bons résultats de classification avec un taux moyen de 89,64 % par le classifieur Softmax, ainsi qu'un taux moyen équivalent à 98,13 % par les forêts aléatoires.

Mots clés. Classification d'images, CNN, Cancer de peau, Forêts Aléatoires, Data Augmentation.

ملخص

في هذه الرسالة سوف نقوم بمعالجة مشكلة تصنيف لصور لأفات جلدية مسرطنة وهذه في قاعدة بيانات المسماة HAM10000 و التي تحتوي على سبعة أنواع من الأفات الجلدية. قمنا بمعالجة قاعدة البيانات من اجل توازن وتضخيم الصور حسب نوع سرطان الجلد باستعمال الخوارزمية ImageDataGenerator. أما بالنسبة لتصنيف الصور الجلدية فقد استعملنا التعلم العميق من أجل تدريب النظام و قمنا باستعمال المصنف Softmax وأيضا استعملنا مصنف الغابات العشوائية Forests Aleatoires كمصنف ثاني. أما بالنسبة للنتائج فقد تحصلنا على معدل 89,64 % للمصنف Softmax وعلى معدل 98,13 % بالنسبة لمصنف الغابات

العشوائية

المصنف

كلمات مرشدة. تصنيف الصور، التعلم العميق، سرطان الجلد، الغابات العشوائية، توازن وتضخيم الصور،