
Master Spécialise en Physique Médicale

Mémoire de Projet de Fin d Études

Présenté

Par:

Ayoub EL MHAMDI

youssef MADANE

DÉTECTION DES NODULES PULMONAIRES CANCÉREUX SUR LES IMAGES DE TDM

Soutenu le 22/07/2023 devant la Commission d'Examen :

Pr RAJAE Sebihi

Encadrant

REMERCIEMENTS.

Nous tenons à remercier d'abord toutes les équipes pédagogiques du Master **Technique des Rayonnements et Physique Médicale** de la Faculté des Sciences à Meknès, ainsi que les professeurs ayant contribué activement à notre formation.

Nous profitons de cette occasion, pour remercier vivement notre Professeur **RAJAE SEBIHI** qui n'a pas cessé de nous encourager tout au long de l'exécution de notre Projet de Fin d'Études, ainsi que pour sa générosité et ses compétences en matière de formation et d'encadrement. Nous lui sommes reconnaissants pour ses aides et conseils précieux qui nous ont permis de mener à bien le présent projet.

Nos vifs remerciements vont aussi aux membres de jury pour avoir accepté de juger ce travail.

A la même occasion, nous voudrions également remercier chaleureusement nos parents qui nous ont toujours encouragés durant notre cursus de formation.

Enfin, nos vifs remerciements sont adressés à toutes ces personnes qui nous ont apporté leur aide précieuse et leur soutien inconditionnel. ■

TABLE DES MATIÈRES.

RÉSUMÉ.	1
INTRODUCTION GÉNÉRALE.	2
1. Étendue du problème.	2
2. Travaux connexes.	2
Chapitre I. ANATOMIE DES NODULES PULMONAIRES.	4
I.1. Introduction.	4
1. Définition.	4
2. Types de nodules pulmonaire.	4
3. Causes des nodule pulmonaire.	5
4. Importance de l'études des nodules pulmonaire.	5
5. Traitement des nodules pulmonaire.	6
I.2. Anatomie internes et externes du poumon.	7
1. Morphologie externe.	7
2. Morphologie interne.	8
3. Diagnostique des maladies pulmonaire.	9
I.3. Cancer du poumon et nodules pulmonaire.	9
1. Causes, types et stades des cancer de poumon.	9
2. Methodes de traitement du cancer du poumon.	10
I.4. Conclusion.	11
Chapitre II. APERÇU SUR DEEP LEARNING.	13
II.1. Introduction.	13
II.2. Fonctionnement de Deep Learning.	14
II.3. Applications.	14
II.4. Les réseaux de neurones artificiels.	15
1. Exemple de Deep learning dans la pratique.	16
2. Erreur quadratique moyenne E .	19
3. Algorithme de descente de gradient.	19
II.5. Images et Deep Learning.	21
1. Fonctions d'activation.	21
2. L'importance des matrices dans l'apprentissage profond.	23
II.6. Types de réseaux de neurones.	24
II.7. Conversion d'images en vecteurs pour l'analyse par réseaux neuronaux.	24
II.8. Conclusion.	24
Chapitre III. DÉTECTION DES NODULES PULMONAIRES DU CANCER.	26
III.1. Introduction	26
III.2. Méthode.	28

1. Ressources	28
2. Scans CT avec Nodules Pulmonaires.	28
3. Développement des l'algorithmes de détection des nodules.	30
III.3. Résultats.	33
1. Évaluation des performances du modèle.	33
2. Métriques d'évaluation : Précision, Rappel et Score F1.	33
III.4. Discussion.	35
III.5. Conclusion.	37
ANNEXE 1.	38
ANNEXE 2.	41
RÉFÉRENCES BIBLIOGRAPHIQUES.	44

RÉSUMÉ.

Les images médicales jouent un rôle important dans le diagnostic et la prise en charge des cancers. Les oncologues analysent des images pour déterminer les différentes caractéristiques de la tumeur, pour proposer un traitement adapté et suivre l'évolution de la maladie.

L'objectif de ce rapport est d'évaluer la détection automatique des nodules pulmonaire cancéreuse dans le contexte de la radiothérapie, à partir des images de CT scan.

Nous utilisons les nodules candidats qui sont segmentés par un réseau neuronal convolutif entraîné sur des images de scanner segmentées par des experts. Notant que celles-ci sont faiblement annotées, et sont souvent disponibles en quantités très limitées du fait de leur coût.

Nous créons un modèle basé sur un CNN (Convolutional Neural Network) pour diagnostiquer ces nodules. et les expériences réalisées sur des scanners pulmonaires de haute qualité ont démontré une détection assez bonne des nodules cancéreux du poumon.

Mots clés: Réseau neuronal convolutif, CNN, apprentissage profond, CT scan, TDM, radiothérapie, Caner, Nodules pulmonaire.■

INTRODUCTION GÉNÉRALE.

1. Étendue du problème.

Dans le monde entier, le cancer du poumon est la cause principale des décès liés au cancer [1]. Une découverte opportune grâce à des scanners thoraciques de dépistage peut augmenter considérablement les chances de survie [2]. Potentiellement, les nodules pulmonaires (des masses rondes ou ovales détectables dans les scanners thoraciques) peuvent indiquer un cancer du poumon [3]. L'efficacité des soins de santé pourrait bénéficier significativement d'un système informatisé capable d'identifier automatiquement ces nodules, permettant d'économiser du temps et des ressources pour les prestataires de santé et les patients.

Les algorithmes de détection des nodules se composent généralement de deux parties [4] :

- La première étape recherche une grande variété de nodules possibles avec une grande sensibilité; cependant, elle génère de nombreux faux positifs.
- L'étape suivante atténue ces faux positifs en utilisant des caractéristiques et des classificateurs améliorés, une tâche difficile en raison des variables englobant les formes, les tailles, les types de nodules et leur ressemblance potentielle avec d'autres composants thoraciques comme les vaisseaux sanguins ou les ganglions lymphatiques [3, 5].

Dans nos recherches, nous avons appliqué un réseau neuronal convolutionnel sur l'ensemble de données public LUNA16, composé de scanners thoraciques de 888 patients évalués par quatre experts médicaux [4].

Finalement, la résultat a montré une performance de classification de nodule. Cela signifie qu'elle peut identifier de manière un peu précise plus de nodules cancéreux et moins de nodules non cancéreux [6, 7].

2. Travaux connexes.

Les premières tentatives d'automatisation des dépistages du cancer du poumon se sont appuyées sur des algorithmes pour extraire les caractéristiques uniques des nodules pulmonaires. Les chercheurs ont mis l'accent sur les données volumétriques des nodules et les zones proches, mais ces méthodes ont souvent eu du mal à différencier correctement la gamme de variations des nodules, nécessitant une personnalisation pour chaque type de nodule distinct[8, 9, 10]. Avec le temps, grâce à l'avancée des réseaux neuronaux profonds, ces techniques ont été progressivement améliorées. Les innovations récentes, tout particulièrement les méthodes basées sur les réseaux neuronaux convolutionnels (Convolutional Neural Networks, CNN), ont montré qu'elles pourraient améliorer la classification des nodules[5, 4, 11].

Un changement significatif dans le paradigme de détection des nodules pulmonaires a été l'intégration d'informations contextuelles multi-échelles, en particulier avec l'ensemble de données Luna16. Cette approche tire profit des méthodologies d'apprentissage profond pour évaluer une vaste gamme de caractéristiques morphologiques et structurales à travers diverses échelles[12, 13]. Plusieurs techniques ont démontré leur efficacité.

Des chercheurs ont également suggéré plusieurs stratégies prometteuses pour la détection des anomalies pulmonaires, comme l'utilisation de patches 3D pour une précision accrue avec les données volumétriques et la réduction des faux positifs[4, 5]. L'extraction graduelle des caractéristiques, une méthode séquentielle qui fusionne l'information de contexte à différentes échelles, offre une alternative à la pratique conventionnelle de l'intégration radicale[12, 14].

La combinaison holistique de ces approches a permis d'obtenir des modèles plus fiables et robustes pour la détection des nodules pulmonaires. Les régions entourant les nodules pulmonaires potentiels ont été minutieusement examinées et comparées à d'autres organes ou tissus pour améliorer la différenciation des nodules[14]. Les améliorations futures pourraient inclure l'intégration de données contextuelles provenant des zones adjacentes aux nodules, renforçant ainsi potentiellement les performances et l'exactitude des modèles[13, 14]. ■

Chapitre I.

ANATOMIE DES NODULES PULMONAIRES.

I.1. Introduction.

I.1.1. Definition.

Les nodules pulmonaires, de petites masses arrondies formées dans les poumons, apparaissent lors des examens d'imagerie. Leurs caractéristiques déterminent le traitement : nature (bénin ou malin), taille, croissance, et symptômes éventuels. Pour les évaluer, on utilise souvent la surveillance régulière et la biopsie.

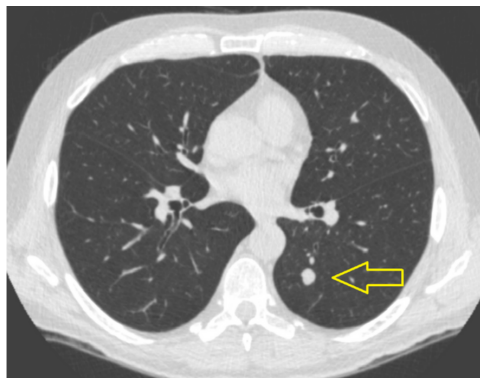


Figure I.1: *Nodule pulmonaire.*

I.1.2. Types de nodules pulmonaire.

Les nodules pulmonaires, petites masses arrondies apparues dans les poumons, sont détectés par des examens d'imagerie. Leur traitement dépend de leur nature, taille, croissance et symptômes éventuels. Dans l'évaluation de ces nodules, la surveillance régulière et la biopsie jouent un rôle clé.

Plusieurs types de nodules pulmonaires existent¹ : les infectieux, granulomateux, calcifiés et carcinoïdes sont les plus courants. Les nodules infectieux résultent d'infections pulmonaires, comme la tuberculose, la pneumonie ou les infections fongiques, avec des caractéristiques variant selon l'agent pathogène. Les nodules granulomateux se distinguent par une inflammation granulomateuse, souvent liée à des infections spécifiques comme la tuberculose ou la sarcoïdose. Les nodules calcifiés, qui résultent d'infections antérieures ou d'autres processus inflammatoires, contiennent des dépôts de calcification. Les carcinoïdes se développent à partir de cellules neuroendocrines dans les poumons et peuvent être bénins ou malins.

¹<https://www.ocean-imagerie.fr/fiche-conseil/nodule-pulmonaire/>

A noter qu'un diagnostic précis des nodules pulmonaires nécessite un examen médical détaillé, incluant des examens d'imagerie, des analyses de sang, et parfois une biopsie pour déterminer la nature du nodule.

En résumé, la classification des nodules pulmonaires se fait selon leurs causes et caractéristiques. Un examen médical détaillé est souvent nécessaire pour un diagnostic précis.

I.1.3. Causes des nodule pulmonaire.

Les nodules pulmonaires peuvent naître de diverses causes². Certaines, comme les infections pulmonaires, les maladies inflammatoires, les réactions à des substances inhalées et les tumeurs bénignes, sont plus courantes. Les nodules malins peuvent découler de tumeurs malignes, y compris le cancer du poumon ou des cancers métastatiques. Des examens détaillés sont nécessaires pour diagnostiquer précisément les nodules pulmonaires. Ils permettent de déterminer l'origine exacte des nodules et d'orienter le traitement adéquat.

I.1.4. Importance de l'étude des nodules pulmonaire.

L'étude des nodules pulmonaires est importante pour plusieurs raisons :

- **Détection précoce du cancer du poumon:** Les nodules pulmonaires peuvent être un signe précoce de cancer du poumon. Le cancer du poumon étant souvent asymptomatique dans ses premiers stades, la détection précoce des nodules pulmonaires peut permettre un diagnostic précoce du cancer et améliorer les chances de réussite du traitement.
- **Différencier les nodules bénins des nodules malins:** Tous les nodules pulmonaires ne sont pas cancéreux. Une étude approfondie des nodules nous permet de différencier les nodules bénins des nodules malins. Cette distinction est essentielle pour éviter toute intervention ou traitement inutile chez les patients présentant des nodules bénins, tout en assurant une prise en charge rapide des nodules malins.
- **Surveillance et évaluation des changements:** Les nodules pulmonaires peuvent changer de taille, de forme ou de caractéristiques au fil du temps. L'étude régulière des nodules nous permet de surveiller ces changements et de déterminer si d'autres mesures sont nécessaires, telles qu'une biopsie ou une intervention chirurgicale.
- **Évaluation du pronostic:** L'étude des nodules pulmonaires peut fournir des informations sur le pronostic d'un patient atteint d'un cancer du poumon. Par exemple, la taille, la localisation et les caractéristiques du nodule peuvent aider à prédire le stade de la maladie et à orienter les options thérapeutiques.
- **Planification du traitement:** Les résultats de l'étude des nodules pulmonaires aident les professionnels de santé à élaborer un plan de traitement approprié. Celui-ci peut inclure des options telles que la surveillance régulière, la chirurgie, la radiothérapie, la chimiothérapie ou l'immunothérapie, en fonction de la nature des nodules et de leur éventuelle malignité.

En résumé, l'étude des nodules pulmonaires est d'une grande importance pour la détection précoce, la différenciation des nodules bénins et malins, la surveillance des changements, l'é-

²<https://cancer.ca/fr/cancer-information/cancer-types/lung/what-is-lung-cancer/lung-nodules>

valuation du pronostic et la planification du traitement. Une approche précise et complète de l'étude des nodules pulmonaires permet une prise en charge optimale des patients et de meilleurs résultats cliniques.

I.1.5. Traitement des nodules pulmonaire.

Le traitement des nodules pulmonaires dépend d'un certain nombre de facteurs, notamment de la nature des nodules (bénins ou malins), de leur taille, de leur croissance, de leurs caractéristiques et de la présence ou non de symptômes. Voici quelques options thérapeutiques possibles :

- **Surveillance régulière** : Les petits nodules pulmonaires bénins qui ne présentent aucun signe de malignité peuvent être surveillés régulièrement par des examens d'imagerie, tels que des radiographies ou des tomodensitométries du thorax, afin de suivre leur évolution. Si le nodule reste stable dans le temps, aucun autre traitement n'est généralement nécessaire.
- **Biopsie** : Dans certains cas, une biopsie peut être pratiquée pour déterminer la nature du nodule et exclure la présence d'un cancer. La biopsie peut être réalisée par différentes méthodes, telles que la biopsie percutanée sous guidage radiologique, la bronchoscopie ou la biopsie chirurgicale. La biopsie fournit un échantillon de tissu pour une évaluation pathologique plus approfondie.
- **Traitement médical** : Si les nodules pulmonaires sont causés par une infection ou une inflammation spécifique, un traitement médical approprié peut être prescrit. Il peut s'agir d'antibiotiques pour les infections bactériennes, d'antifongiques pour les infections fongiques, de corticostéroïdes pour l'inflammation, etc.
- **Chirurgie** : Si le nodule pulmonaire est suspecté d'être malin ou s'il présente des caractéristiques inquiétantes, une intervention chirurgicale peut être recommandée. La chirurgie peut consister en une ablation complète du nodule (lobectomie, pneumonectomie) ou en une résection partielle (segmentectomie, résection cunéiforme). L'intervention chirurgicale peut être réalisée à ciel ouvert ou avec une assistance vidéo.
- **Radiothérapie** : La radiothérapie peut être utilisée pour traiter les nodules pulmonaires malins lorsque la chirurgie n'est pas possible ou appropriée. Elle utilise des radiations de haute énergie pour détruire les cellules cancéreuses. La radiothérapie peut être administrée par voie externe ou interne (curiethérapie).
- **Chimiothérapie et immunothérapie** : Pour les nodules pulmonaires malins avancés ou métastatiques, des traitements systémiques tels que la chimiothérapie ou l'immunothérapie peuvent être utilisés pour cibler et détruire les cellules cancéreuses.

Le choix du traitement dépend de la nature spécifique du nodule pulmonaire, du stade de la maladie, de l'état de santé général du patient et de ses préférences personnelles. Il est essentiel de consulter un professionnel de la santé pour obtenir une évaluation précise et des recommandations appropriées concernant le traitement des nodules pulmonaires.

I.2. Anatomie internes et externes du poumon.

L'anatomie pulmonaire est l'étude de la structure des poumons. Les poumons sont des organes intrathoraciques qui permettent l'échange des gaz vitaux, notamment l'oxygène et le dioxyde de carbone. L'oxygène est nécessaire au métabolisme de l'organisme, et le dioxyde de carbone doit être évacué. L'appareil respiratoire humain est l'ensemble des organes de l'organisme qui permet d'acheminer le dioxygène (O_2) de l'extérieur (air ou eau) vers les cellules et d'éliminer le dioxyde de carbone (CO_2) produit durant la respiration cellulaire

On distingue trois régions dans le thorax :

- le médiastin: occupé par le cœur, les vaisseaux, trachée, œsophage voies lymphatiques... supérieure et a l'arc aortique
- les deux cavités pleuropulmonaires occupées par les poumons. Les deux poumons sont séparés par le médiastin. Le sommet du poumon dépasse légèrement au-dessus de la clavicule : le dôme du poumon se situe au niveau sus-claviculaire.

I.2.1. Morphologie externe.

1. **forme:** les poumons ont la forme d'un demi-cône irrégulier, mesure 20 cm de haut, 20 cm d'épaisseur et 10 cm de diamètre transversal.
1. **coloration:** rosée et brillant chez le sujet jeune, et tacheté de dépôts pigmentaires chez le sujet âgé. Cela est vrai chez le non-fumeur ; en effet, chez le fumeur, le phénomène d'anthracose fait que le poumon a un aspect noirâtre ou l'on voit des petits polygones roses (= lobules pulmonaires) entourés de noir.
2. **poids:** le poumon droit (700 g) est plus volumineux que le poumon gauche (600 g).
3. **consistance:** molle et spongieuse
4. **structure:** schématiquement le poumon apparaît constitué de :
 - la ramification de l'arbre bronchique depuis le hile jusqu'à la périphérie
 - trame vasculaire fonctionnelle(vx pulm.) et trophique (vx bronchiques et lymphatiques)
 - un tissu conjonctivo-élastique, constituant la charpente fibreuse

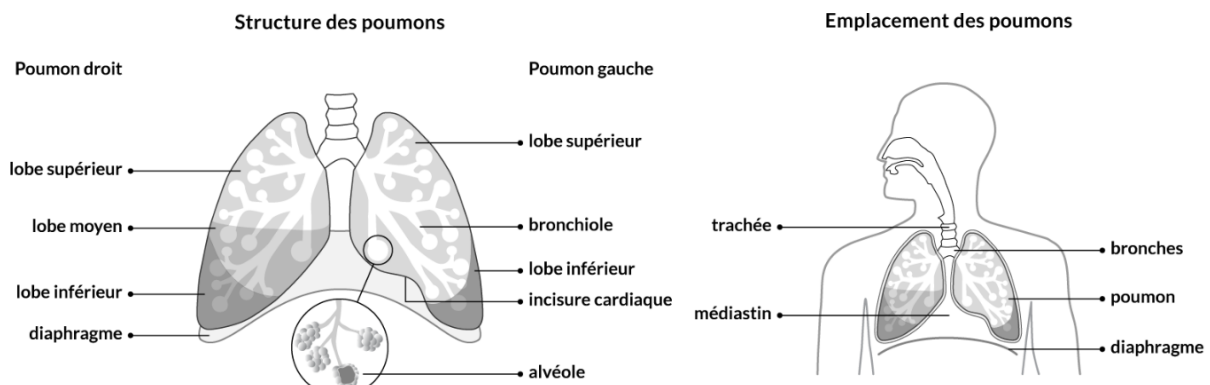


Figure I.2: *La structure de poumon*³.

5. **configuration externe:** on décrit au poumon :

- 3 faces : costale, médiastinale, diaphragmatique.
- 3 bords : antérieur , postérieur , et inférieur 1 base 1 sommet.

6. **Segmentation:** Chaque poumon se divise en lobe:

- Poumon droit : 3 lobes Le lobe supérieur Le lobe moyen Le lobe inférieur
- Poumon gauche : 2 lobes Le lobe supérieur Le lobe inférieur

I.2.2. Morphologie interne.

L'unité fonctionnelle: Les alvéoles sont enveloppées par des capillaires artérioveineux l'ensemble assurant les échanges gazeux à travers la membrane alvéolocapillaire. Cette surface estimée chez l'adulte de 70 à 90m² en moyenne, présente au cours du vieillissement une réduction de ses capacités d'échanges

I.2.2.1. Vascularisation.

La vascularisation du poumon désigne le réseau de vaisseaux sanguins qui assure l'apport d'oxygène et l'élimination du dioxyde de carbone dans les poumons. Voici un résumé de la vascularisation pulmonaire :

- Les artères pulmonaires transportent le sang désoxygéné provenant du ventricule droit du cœur vers les poumons.
- À l'intérieur des poumons, les artères pulmonaires se ramifient en artérioles pulmonaires de plus petite taille.
- Les artérioles pulmonaires se divisent en capillaires pulmonaires qui sont présents au niveau des parois des alvéoles pulmonaires.
- Les capillaires pulmonaires permettent les échanges gazeux : l'oxygène est absorbé par le sang des capillaires, tandis que le dioxyde de carbone est libéré du sang pour être expiré.
- Les capillaires pulmonaires se regroupent pour former des veinules pulmonaires, qui à leur tour se rejoignent pour former les veines pulmonaires.
- Les veines pulmonaires transportent le sang oxygéné depuis les poumons vers l'oreillette gauche du cœur.
- Le sang oxygéné est ensuite pompé par le ventricule gauche du cœur vers le reste du corps, fournissant ainsi de l'oxygène aux tissus et organes.

La vascularisation du poumon assure ainsi les échanges gazeux essentiels à la respiration, en fournissant de l'oxygène au sang et en éliminant le dioxyde de carbone. Ce processus permet de maintenir l'équilibre de l'organisme en oxygène et en gaz carbonique.

³<https://cancer.ca/fr/cancer-information/cancer-types/lung/what-is-lung-cancer/the-lungs>

I.2.3. Diagnostique des maladies pulmonaire.

Pour diagnostiquer les maladies pulmonaires, les médecins recourent à diverses méthodes. Celles-ci débutent par l'anamnèse, l'examen physique et les tests de la fonction pulmonaire, essentiels à l'évaluation de la santé pulmonaire. Les poumons sont ensuite visualisés et leurs anomalies détectées à l'aide de techniques d'imagerie comme la radiographie, la tomodensitométrie, l'échographie, la biopsie et la scintigraphie pulmonaire.

Le choix des procédures de diagnostic est déterminé par les caractéristiques spécifiques du patient et les soupçons cliniques du médecin. Ce dernier, en se basant sur les résultats des examens initiaux, peut suggérer des tests supplémentaires pour établir un diagnostic rigoureux. Ces tests permettent d'identifier la cause exacte de la pathologie pulmonaire et d'orienter le choix du traitement adéquat.

I.3. Cancer du poumon et nodules pulmonaire.

I.3.1. Causes, types et stades des cancer de poumon.

Le cancer du poumon, maladie qui voit des cellules anormales croître de manière incontrôlée dans le tissu pulmonaire, découle principalement de l'exposition à la cigarette, le tabagisme passif, l'exposition professionnelle à des substances cancérigènes comme l'amiante ou les produits chimiques aéroportés, ainsi qu'à des facteurs génétiques et environnementaux.

Deux principales versions de ce cancer existent⁴:

- Le cancer du poumon à petites cellules (CPPC) est le responsable d'environ 10 à 15 % des cas, souvent associé au tabagisme et se propage rapidement dans le corps.
- Le cancer du poumon non à petites cellules (CPNPC) est plus fréquent, représentant 85 à 90 % des cas. Il comprend des sous-types comme l'adénocarcinome, le carcinome épidermoïde et le carcinome à grandes cellules.

La stadification du cancer du poumon utilise couramment le système TNM (tumeur, ganglions lymphatiques, métastases) qui évalue la taille de la tumeur originelle (T), l'atteinte des ganglions lymphatiques (N) et la présence de métastases lointaines (M). Les stades se classent comme suit⁵:

- **Stade I** : tumeur limitée au poumon, sans atteinte des ganglions lymphatiques ni métastases à distance.
- **Stade II** : la tumeur a envahi les ganglions lymphatiques proches du poumon touché ou a infiltré les structures avoisinantes.
- **Stade III** : la tumeur est plus avancée localement et a envahi plus de ganglions lymphatiques ou de structures adjacentes.
- **Stade IV** : le cancer a migré à d'autres parties du corps, souvent éloignées des poumons.

⁴<https://www.e-cancer.fr/Patients-et-proches/Les-cancers/Cancer-du-poumon/Les-points-cles>

⁵<https://www.elsan.care/fr/radiotherapie-metz-iprm/nos-actualites/stades-du-cancer-du-poumon-et-esperance-de-vie>.

Il convient de noter que la classification et la stadification peuvent dépendre du système utilisé (par exemple, TNM ou la classification de l'American Joint Committee on Cancer - AJCC).

Un professionnel de santé, souvent un oncologue ou un pneumologue, à l'aide d'une série d'examens tels que biopsies, examens d'imagerie (comme le scanner thoracique et le PET scan), analyses de sang et tests de la fonction pulmonaire, peut déterminer le diagnostic précis, le type et le stade du cancer du poumon.

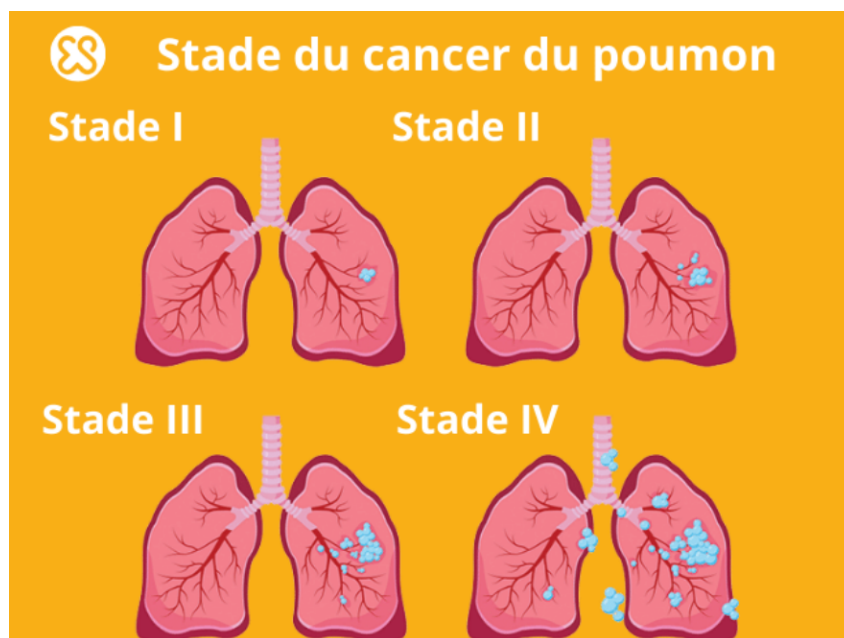


Figure I.3: *Stade du cancer du poumon.*

I.3.2. Methodes de traitement du cancer du poumon.

Le traitement du cancer du poumon, complexe, dépend de plusieurs facteurs : le type et le stade du cancer, la présence de métastases, et les caractéristiques spécifiques du patient⁶.

- La chirurgie permet d'ôter la tumeur et, si nécessaire, les ganglions lymphatiques proches. En fonction de l'extension de la maladie, différentes opérations, comme la lobectomie, la pneumonectomie ou la résection cunéiforme, peuvent être réalisées.
- La radiothérapie, utilisant des radiations de haute énergie pour détruire les cellules cancéreuses, peut être administrée avant ou après la chirurgie, ou être le traitement principal pour les patients inopérables ou afin de soulager les symptômes.

⁶<https://www.fondation-arc.org/cancer/cancer-poumon/traitement-cancer>



Figure I.4: *Accélérateur linéaire de radiothérapie⁷.*

- La chimiothérapie, basée sur des médicaments anticancéreux, peut être administrée avant ou après la chirurgie, ou comme traitement principal en cas de maladie avancée ou métastatique.
- Les thérapies ciblées sont des médicaments qui ciblent directement certaines altérations génétiques ou protéiques des cellules cancéreuses, utilisés pour traiter des sous-types spécifiques de cancer du poumon non à petites cellules.
- L'immunothérapie stimule le système immunitaire du patient pour combattre les cellules cancéreuses. Des inhibiteurs de points de contrôle immunitaire peuvent être utilisés pour traiter certains types de cancers du poumon avancés ou métastatiques.
- La thérapie photodynamique, qui combine l'utilisation d'un médicament photosensibilisant et d'une lumière laser, détruit principalement de petites tumeurs.
- Les soins palliatifs visent à soulager les symptômes, améliorer la qualité de vie et fournir un soutien aux patients atteints d'un cancer du poumon avancé ou métastatique.

Le choix du traitement est dicté par la situation individuelle du patient et doit être discuté avec une équipe médicale spécialisée dans le traitement du cancer du poumon.

I.4. Conclusion.

Nous avons approfondi dans ce chapitre l'anatomie des nodules pulmonaires, détaillant leur signification, causes, importance, variétés et options de traitement. L'anatomie interne et externe des poumons, les diverses maladies pulmonaires et la connexion entre les nodules pulmonaires et le cancer du poumon ont également été discutées.

Ces connaissances revêtent une importance pour diverses raisons : la détection précoce du cancer du poumon, la différenciation entre nodules bénins et malins, l'évaluation du pronostic et l'organisation du traitement. Un diagnostic précis associé à une surveillance constante sont vitaux pour prévenir des traitements ou interventions inutiles chez des patients avec des nodules bénins, et pour une prise en charge rapide des nodules malins.

Grâce à l'Intelligence Artificielle de type Deep Learning et l'ensemble de données Luna16, on peut bâtir des modèles capables de détecter et classer les nodules pulmonaires. Ces techniques autorisent une identification des nodules plus tôt et avec plus de précision que jamais,

⁷<https://en.wikipedia.org/wiki/Tomotherapy>

augmentant ainsi les chances de survie pour les patients. De plus, elles facilitent la distinction entre nodules bénins et malins, permettant une meilleure décision quant à la suite du traitement. ■

Chapitre II.

APERÇU SUR DEEP LEARNING.

II.1. Introduction.

L'apprentissage profond est une branche de l'apprentissage automatique⁸, qui est lui-même un domaine de l'intelligence artificielle. L'apprentissage profond permet de prédire ou d'analyser des données de haute dimension ou complexes, comme les images, les textes ou les sons, d'une manière similaire au cerveau humain. L'apprentissage profond utilise des réseaux de neurones artificiels multicouches⁹ qui peuvent extraire les sens et les motifs cachés dans les données sans avoir besoin d'intervention humaine. Ainsi, l'apprentissage profond acquiert une grande capacité d'adaptation et d'évolution avec le changement de données[15].

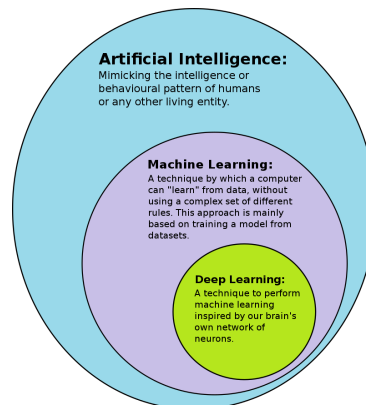


Figure II.5: *Comment l'apprentissage en profondeur est un sous-ensemble de l'apprentissage automatique et comment l'apprentissage automatique est un sous-ensemble de l'intelligence artificielle (IA).*

Par conséquent, l'apprentissage profond se distingue de l'apprentissage traditionnel par le fait qu'il ne repose pas sur des règles ou des algorithmes prédéfinis, mais qu'il peut générer ses propres règles et algorithmes par essai et erreur. De plus, l'apprentissage profond peut surmonter certains des problèmes rencontrés par l'apprentissage traditionnel, tels que le bruit, le manque ou le changement des données.

Ainsi, nous voyons que l'apprentissage profond est un domaine récent et prometteur en informatique qui mérite l'attention et la recherche, et qui peut contribuer à résoudre de nombreux problèmes dans différents domaines tels que la traduction, la reconnaissance d'images et de sons, etc.

⁸https://fr.wikipedia.org/wiki/Apprentissage_profond

⁹<https://www.ibm.com/topics/deep-learning>

II.2. Fonctionnement de Deep Learning.

Pour comprendre le principe de l'apprentissage profond, on peut utiliser des exemples de notre vie quotidienne. Lorsque nous voulons améliorer certains résultats, on change certains facteurs influençant ces résultats de manière cyclique, en se basant sur l'expérience et l'erreur. Par exemple, un vendeur de fruits essaie d'augmenter son revenu en changeant la quantité et les types de fruits offerts aux clients, en se référant aux ventes passées et actuelles. Il n'y a pas de règle fixe qui détermine la quantité de chaque fruit que le vendeur doit fournir, il doit donc expérimenter jusqu'à ce qu'il atteigne le point d'équilibre entre l'offre et la demande.

En apprentissage profond, on utilise une fonction mathématique appelée **fonction coût** pour mesurer la différence entre les résultats d'un modèle d'apprentissage et les résultats souhaités ou corrects. Puis on utilise une autre fonction appelée **fonction optimisation** pour ajuster la valeur de chaque cellule neuronale dans le réseau d'apprentissage afin de réduire la valeur de la **fonction coût**. Ces étapes sont répétées sur un grand ensemble de données jusqu'à ce que le modèle d'apprentissage soit capable d'accomplir les tâches demandées avec précision ou acceptabilité.

Cet exemple peut nous donner une idée qui nous aide à comprendre l'apprentissage en profondeur, mais il résume des concepts fondamentaux de l'apprentissage en profondeur tels que la fonction de coût ou la fonction de régression graduelle et l'optimisation, ce qui est clair pour nous dans des applications telles que la traduction automatique ou la vision par ordinateur.

II.3. Applications.

Dans la traduction automatique, un système d'apprentissage en profondeur utilise un réseau neuronal pour convertir une phrase d'une langue à une autre. Ce système utilise une fonction de coût pour mesurer la différence entre la traduction générée et la traduction cible. Ensuite, il utilise un algorithme d'optimisation tel que la descente de gradient pour ajuster le poids de chaque cellule neuronale dans le réseau afin de minimiser la valeur de la fonction de coût. Ce processus est répété sur un grand ensemble de phrases à traduire jusqu'à ce que le système soit capable de générer des traductions précises et naturelles.

En vision par ordinateur, un système d'apprentissage en profondeur utilise des réseaux neuronaux artificiels pour extraire des informations et des perspectives à partir d'images et de vidéos. Certaines applications dans ce domaine sont¹⁰ :

- La **classification d'images** consiste à attribuer une étiquette à une image ou une photographie entière.

Ce problème est également appelé « classification d'objets » et peut-être plus généralement « reconnaissance d'images », bien que cette dernière tâche puisse s'appliquer à un ensemble beaucoup plus large de tâches liées à la classification du contenu des images.

¹⁰<https://machinelearningmastery.com/applications-of-deep-learning-for-computer-vision/>

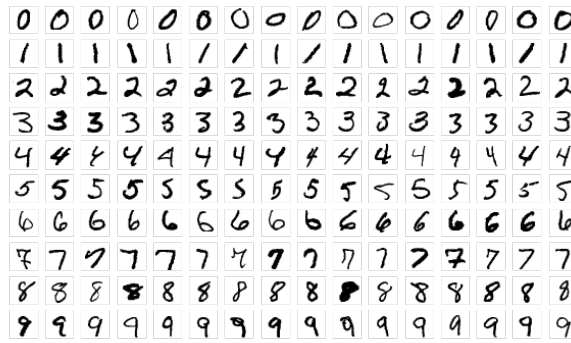


Figure II.6: Un exemple de chiffres de en jeu de données MNIST.

Un exemple populaire de classification d'images utilisé est le jeu de données(dataset) MNIST.

- **Surveillance du contenu:** pour supprimer automatiquement le contenu non sécurisé ou inapproprié des archives d'images et de vidéos.
- **Reconnaissance faciale:** pour identifier l'identité des personnes ou extraire des caractéristiques de leur visage, telles que l'ouverture ou la fermeture des yeux, le port de lunettes ou de moustaches.
- **Violation du droit d'auteur:** pour supprimer le contenu volé ou détourné d'images ou de vidéos protégées par des droits d'auteur.

Ces applications sont rendues possibles grâce à l'utilisation de réseaux neuronaux profonds qui peuvent apprendre à partir d'un grand nombre d'exemples et extraire des caractéristiques complexes à partir des données.

II.4. Les réseaux de neurones artificiels.

Les réseaux de neurones artificiels sont des modèles d'intelligence artificielle qui utilisent des cellules nerveuses artificielles pour convertir les entrées en sorties. Chaque cellule nerveuse reçoit des signaux d'autres cellules et envoie des signaux à d'autres cellules. Chaque signal est ajouté à une valeur de poids qui détermine sa force et son importance. Chaque cellule nerveuse calcule la somme des signaux reçus et applique une fonction d'activation pour produire un signal de sortie.

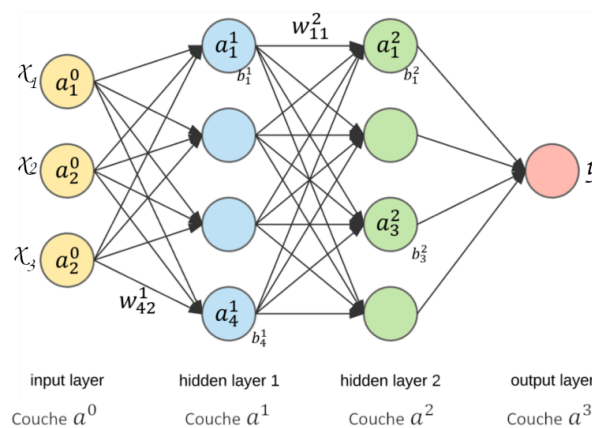


Figure II.7: *Architecture de modèle multicouche.*

II.4.1. Exemple de Deep learning dans la pratique.

Pour savoir la relation entre les réseaux de neurones artificiels et la fonction linéaire, on utilise l'exemple de calcul de température précédent.

La fonction linéaire est un type de fonctions mathématiques qui décrit une relation simple entre deux variables, où un changement dans l'une entraîne un changement proportionnel dans l'autre. Par exemple, si nous avons une fonction qui convertit la température de Celsius en Fahrenheit, cette fonction sera linéaire, car chaque augmentation d'un degré Celsius entraîne une augmentation constante de la température en Fahrenheit.

La formule utilisée pour convertir la température de Celsius en Fahrenheit est [16] :

$$F = \frac{9}{5}C + 32 \quad (1)$$

On peut le représenter par le schéma de réseau suivant:

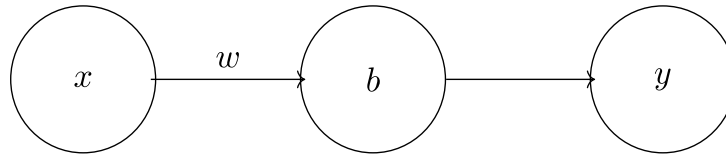


Figure II.8: *Structure mono-neurone..*

Dans cette formule, nous pouvons définir deux facteurs principaux : le poids des entrées et l'ordonnée à l'origine. Le poids des entrées est un nombre qui multiplie la valeur de la variable indépendante (°C) pour déterminer son effet sur la valeur de la variable dépendante (°F). Dans ce cas, le poids des entrées est 1.8. L'ordonnée à l'origine est un nombre qui est ajouté au produit du poids des entrées par la valeur de la variable indépendante pour déterminer la valeur de la variable dépendante lorsque la variable indépendante est égale à zéro. Dans ce cas, l'ordonnée à l'origine est 32.

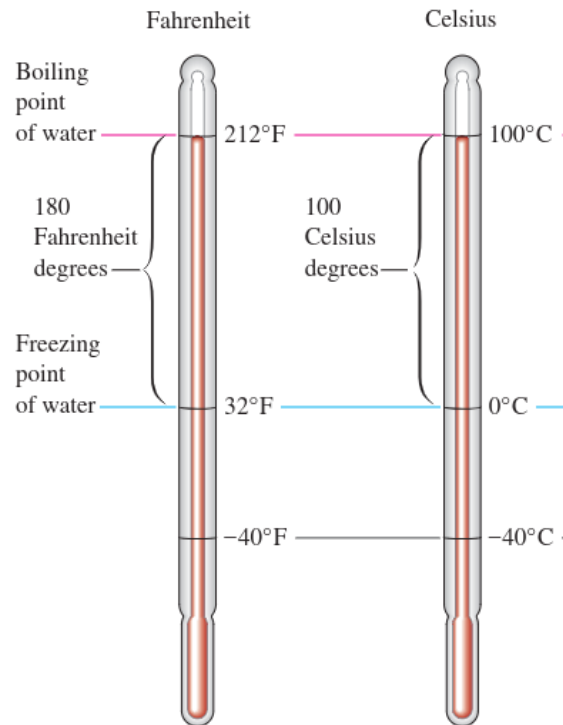


Figure II.9: Relation entre deux échelles de température Celsius et Fahrenheit.

Ces exemples montrent comment utiliser une fonction linéaire pour convertir la température de Celsius en Fahrenheit. Mais comment utiliser une fonction linéaire pour comprendre le concept de coût et d'optimisation dans l'apprentissage profond ?

Dans l'apprentissage profond, nous utilisons des fonctions linéaires pour produire des résultats basés sur un ensemble d'entrées. Chaque entrée a un poids qui ajuste la mesure de son impact sur le résultat produit. Ensuite, nous utilisons une fonction de coût pour mesurer la différence entre les résultats produits et les résultats cibles. Ensuite, nous utilisons un algorithme d'optimisation[15, 17] pour ajuster les poids des entrées afin de réduire la valeur de la fonction de coût.

Comme première étape dans l'apprentissage profond, nous devons collecter un ensemble de données qui forment des paires d'entrées et de résultats cibles. Dans l'exemple de conversion de la température de Celsius en Fahrenheit, nous utilisons un tableau avec des mesures différentes de température dans les deux systèmes comme des paires d'entrées et de résultats cibles :

température en Celsius °C	température en Fahrenheit °F
-40	-40
-20	-4
0	32
20	68
40	104
60	140
80	176
100	212

Table II.1: Conversion de températures entre Celsius et Fahrenheit

Ce tableau nous permet de comparer la valeur de la température en Celsius avec sa valeur correspondante en Fahrenheit. Mais que se passe-t-il si nous voulons convertir une valeur qui n'est pas dans le tableau ? Pouvons-nous trouver le poids des entrées et la coupe de l'axe qui convient pour représenter ces données ?

La réponse est oui, mais pas facilement. Si nous essayons de deviner le poids des entrées et la coupe de l'axe au hasard, nous obtiendrons des résultats différents des résultats cibles. Par exemple, si nous supposons que le poids des entrées est de 2 et que la coupe de l'axe est de 0, notre fonction sera:

$$F = (C \times 1) + 0 \quad (2)$$

Cette fonction donne des résultats imprécis. Par exemple, si nous voulons convertir $20^{\circ}C$ en Fahrenheit, elle donne[18]:

$$F = (20 \times 1) + 0 = 20 \quad (3)$$

Et c'est une erreur car la valeur correcte est $68^{\circ}F$.

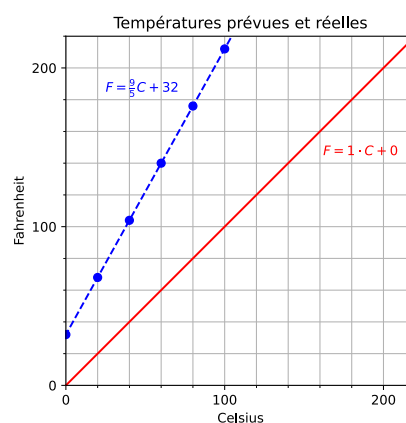


Figure II.10: L'écart entre les températures prévues et réelles.

Alors, comment trouvons-nous le poids des entrées et la coupe de l'axe corrects ? C'est là que l'apprentissage en profondeur intervient. L'apprentissage en profondeur utilise un algorithme appelé régression pour trouver les meilleures valeurs pour ces deux facteurs afin de réduire

l'écart entre les résultats générés et les résultats cibles. Cet écart est appelé fonction de coût ou fonction d'erreur.

II.4.2. Erreur quadratique moyenne E .

La fonction de coût est une fonction mathématique mesurée entre zéro et la valeur maximale possible. Plus la valeur de la fonction de coût est proche de zéro, plus les résultats générés sont proches des résultats cibles. Par exemple, nous utilisons la fonction de coût appelée erreur quadratique moyenne E , qui calcule la moyenne de toutes les mesures d'erreur quadratique entre chaque résultat généré et chaque résultat cible.

$$E = \frac{1}{n} \sum (y - y_0)^2 \quad (4)$$

Où n est le nombre de paires d'entrées et de résultats cibles, y (ou $^{\circ}F$) est la mesure du résultat généré et y_0 (ou $^{\circ}F_0$) est la mesure du résultat cible.

Ainsi, si nous utilisons la E pour mesurer l'écart entre une fonction linéaire et un tableau de conversion des températures Celsius en Fahrenheit, la valeur de la E sera :

$$\begin{aligned} E &= \frac{1}{n} \sum (y - y_0)^2 \\ &= \frac{1}{n} \sum ((wx + b) - y_0)^2 \\ &= 4480 \end{aligned} \quad (5)$$

Il s'agit d'une explication de l'algorithme de descente de gradient[15] qui est utilisé pour trouver les meilleurs poids d'entrée et les biais afin que la valeur de la fonction de coût soit réduite à zéro. Cela se fait en commençant par des poids et des biais aléatoires, puis en les mettant à jour fréquemment en se déplaçant dans la direction opposée du gradient de la fonction de coût.

II.4.3. Algorithme de descente de gradient.

Le gradient est un vecteur qui indique la direction dans laquelle la fonction de coût augmente. En se déplaçant dans la direction opposée, nous pouvons trouver le point le plus bas de la fonction de coût, qui correspond aux meilleures valeurs pour les poids et les biais. La règle de mise à jour pour les poids et les biais est donnée par[19]:

$$w_{n+1} = w_n - \varepsilon \frac{\partial E}{\partial w} \quad (6)$$

$$b_{n+1} = b_n - \varepsilon \frac{\partial E}{\partial b} \quad (7)$$

Où ε est appelé taux d'apprentissage et est un petit nombre positif qui contrôle la taille du pas que nous prenons à chaque itération pour réduire la différence entre les résultats attendus et initiaux. $\frac{\partial E}{\partial w}$ et $\frac{\partial E}{\partial b}$ sont les dérivées partielles de la fonction de coût par rapport aux poids et aux biais respectivement. Ces dérivées partielles nous disent dans quelle mesure la fonction de coût change lorsque le poids ou le biais change légèrement.

Pour calculer ces dérivées partielles, nous pouvons utiliser une technique appelée règle de chaîne[20], qui nous permet de décomposer une fonction complexe en fonctions plus simples et de multiplier leurs dérivées. Par exemple, si nous avons une fonction $f(x) = g(h(x))$, où g et h sont des fonctions plus simples, nous pouvons écrire:

$$\frac{\partial f}{\partial x} = \frac{\partial g}{\partial h} * \frac{\partial h}{\partial x} \quad (8)$$

En utilisant cette technique, nous pouvons trouver les dérivées partielles de E par rapport à w et b comme suit:

$$\begin{aligned} \frac{\partial E}{\partial w} &= \frac{1}{n} \sum -2x(y - y_0) \\ \frac{\partial E}{\partial b} &= \frac{1}{n} \sum -2(y - y_0) \end{aligned} \quad (9)$$

Soit $\varepsilon = 2\frac{\varepsilon}{n}$ car ε est un nombre arbitraire. Cela signifie que vous pouvez simplifier les formules en éliminant le facteur $\frac{2}{n}$, ce qui ne change pas le sens de l'algorithme dans ce cas.

$$\begin{aligned} w_{n+1} &= w_n + \alpha \sum x(y - y_0) \\ b_{n+1} &= b_n + \alpha \sum (y - y_0) \end{aligned} \quad (10)$$

Où x est la valeur d'entrée $^{\circ}C$, y_0 est la valeur cible de sortie $^{\circ}F$ et y est la valeur de sortie obtenue en utilisant notre fonction linéaire.

En utilisant ces formules, on peut mettre à jour notre poids et notre biais à chaque itération jusqu'à ce qu'on atteigne un point où la fonction de coût est réduite au minimum[19, 21].

Et on peut programmer un code simple en langage C qui effectue cette tâche (voir Annexe).

Dans l'entraînement de ce modèle, on a utilisé un ensemble de données d'entraînement (td) qui contient des paires de températures en Celsius et en Fahrenheit, et qui cherche à apprendre la formule de conversion entre ces deux unités. La formule exacte est $y = \frac{9}{5}x + 32$, où y est la température en Fahrenheit et x est la température en Celsius. On a initialisé les paramètres du modèle à des valeurs aléatoires proches de zéro, et on a lancé la descente de gradient pour 100000 itérations. A chaque 10000 itérations, on a affiché l'évolution de la fonction de coût et des paramètres du modèle.

Après 100000 itérations, on a obtenu les résultats suivants :

- Fonction de coût : 0.009562
- Poids : 1.801169
- Biais : 31.883127

On peut voir que les paramètres du modèle sont très proches des valeurs exactes de la formule de conversion. Pour tester la performance du modèle, on a utilisé une nouvelle température $50^{\circ}C$ qui n'est pas dans l'entraînement et on obtient une température de $121.94^{\circ}F$. On a calculé l'erreur moyenne absolue (E) entre les prédictions du modèle et les valeurs réelles. On a obtenu un E de $0.029^{\circ}C$, ce qui montre que le modèle est très précis et qu'il a bien appris la formule de conversion.

On peut visualiser les résultats du modèle sur un graphique qui montre la relation entre les températures en Celsius et en Fahrenheit. On peut voir que les points sont alignés sur une droite qui correspond à la formule $y = 1.8x + 32$. On peut également comparer le modèle avec un modèle aléatoire qui prédit des valeurs aléatoires entre -40°C et 100°C . On peut voir que le modèle aléatoire a un E beaucoup plus élevé que le modèle entraîné.

On peut conclure que le modèle de régression linéaire^{11, 12, 13} est capable de générer des prédictions très proches des valeurs réelles, et qu'il a réussi à apprendre la formule de conversion entre les températures en Celsius et en Fahrenheit. Ce modèle pourrait être utilisé pour convertir des températures dans d'autres unités, comme les kelvins ou les degrés Rankine.

II.5. Images et Deep Learning.

Les images tridimensionnelles qui portent des informations vitales sur la santé humaine sont connues sous le nom d'images scanner (CT scan) et sont utilisées dans le diagnostic et le traitement de nombreuses maladies telles que le cancer, les accidents vasculaires cérébraux et les fractures[22]. Ces images nécessitent un traitement spécial pour éliminer les distorsions, améliorer la qualité et extraire les caractéristiques importantes [23]. C'est pourquoi on utilise des réseaux de neurones artificiels avancés capables de traiter efficacement ces images. Les réseaux de neurones artificiels sont un système inspiré du fonctionnement des cellules du cerveau dans le traitement de l'information et se composent d'un ensemble de cellules nerveuses qui reçoivent et envoient des signaux en utilisant des fonctions d'activation, des poids et des biais.

Le réseau de neurones se compose de plusieurs couches, y compris la couche d'entrée(input layer), la couche de sortie (output layer) et les couches cachées (hidden layers). Chaque couche étant composée d'un ensemble de cellules nerveuses. La première couche joue le rôle de récepteur des données et la dernière joue le rôle de sortie des données du réseau, et entre elles il y a plusieurs couches qui jouent le rôle de récepteur des signaux des couches précédentes et de les envoyer aux couches suivantes.

Cette étude vise à passer en revue les principaux types de réseaux de neurones utilisés dans le domaine de l'imagerie médicale, en particulier les images scanner. Elle vise également à expliquer l'importance des matrices dans le stockage et la mise à jour des paramètres dans un réseau de neurones, en mettant en lumière un exemple d'un réseau de neurones écrit avec des matrices et des vecteurs.

II.5.1. Fonctions d'activation.

Dans cette exemple, nous avons vu comment l'apprentissage profond prédit la température Fahrenheit en utilisant une seule cellule nerveuse basée sur les données initiales uniquement. La solution requise était une équation pour une ligne droite qui pouvait être représentée par une ligne droite. Mais dans certains cas, la distribution des données est courbe comme une fonction sinus ou autre, et ne peut pas être représentée par une ligne droite. Nous avons donc

¹¹https://en.wikipedia.org/wiki/Linear_regression

¹²https://fr.wikipedia.org/wiki/R%C3%A9gression_lin%C3%A9aire

¹³https://fr.wikipedia.org/wiki/Algorithme_du_gradient

besoin d'ajouter des fonctions non linéaires appelées fonctions d'activation, qui aident à plier la courbe linéaire générée par les cellules nerveuses en utilisant la puissance de leur poids et de leur biais[15].

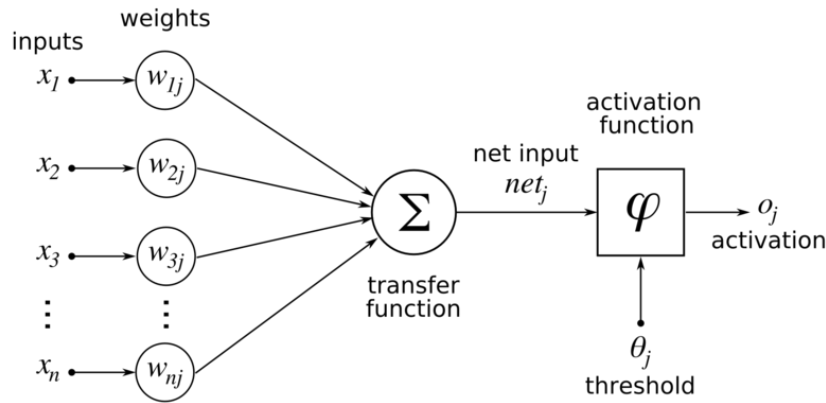


Figure II.11: Graphique de la fonction Unité Linéaire Rectifiée **ReLU**

L'une des fonctions d'activation les plus simples est l'unité linéaire rectifiée, ou fonction **ReLU**, qui est une fonction linéaire par morceaux qui renvoie zéro si son entrée est négative et renvoie directement l'entrée sinon:

$$f(x) = \max(0, x) \quad (11)$$

La fonction **ReLU** est souvent utilisée dans les réseaux d'apprentissage en profondeur pour ajouter de la non-linéarité au modèle. Cette non-linéarité permet au réseau neuronal d'apprendre des relations complexes entre les entrées et les sorties et d'éviter le surajustement des données d'entraînement.

ReLU Function

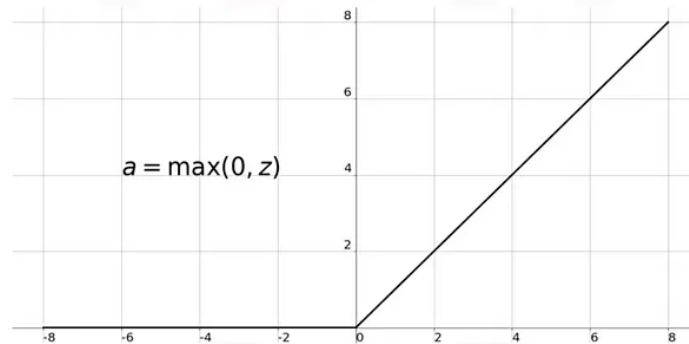


Figure II.12: Graphique de la fonction Unité Linéaire Rectifiée **ReLU**

Il existe plusieurs d'autres fonctions d'activation populaires en deep learning. Les plus courantes sont:

- **Sigmoid** : $f(x) = \frac{1}{1+e^{-x}}$
- **Tanh** : $f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$

II.5.2. L'importance des matrices dans l'apprentissage profond.

Les matrices sont un moyen efficace de stocker et mettre à jour les paramètres dans un réseau de neurones. Plus le nombre de cellules et de liens dans un réseau de neurones est élevé, plus le nombre de paramètres à modifier pour améliorer les performances du réseau est élevé. Nous utilisons donc des matrices pour éviter d'écrire des lignes de code répétitives pour corriger chaque paramètre individuellement. Au lieu de cela, nous utilisons des boucles itératives et des opérations algébriques sur les matrices pour mettre à jour les paramètres plus rapidement et plus simplement[24].

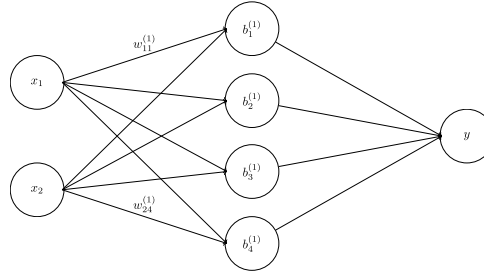


Figure II.13: Un réseau de neurones simple.

Soit un réseau de neurones avec deux entrées, une sortie et une couche cachée. Ce réseau contient 4 neurones et 12 paramètres. Si on ajoute des couches cachées supplémentaires, on augmente le nombre de variables. Cela complexifie les équations et les matrices. On utilise pas des fonctions d'activation pour simplifier le calcul. Le but est de faciliter la traduction du réseau en matrices, avant de le coder dans un langage de programmation.

L'équation de prédiction y c'est comme suit:

$$y = a_1^{(1)} + a_2^{(1)} + a_3^{(1)} + a_4^{(1)} \quad (12)$$

Avec

$$\begin{aligned} a_1^{(1)} &= w_{11}^{(1)}x_1 + w_{21}^{(1)}x_2 + b_1^{(1)} \\ a_2^{(1)} &= w_{12}^{(1)}x_1 + w_{22}^{(1)}x_2 + b_2^{(1)} \\ a_3^{(1)} &= w_{13}^{(1)}x_1 + w_{23}^{(1)}x_2 + b_3^{(1)} \\ a_4^{(1)} &= w_{14}^{(1)}x_1 + w_{24}^{(1)}x_2 + b_4^{(1)} \end{aligned} \quad (13)$$

et finalement:

$$y = \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} w_{11}^{(1)} & w_{12}^{(1)} & w_{13}^{(1)} & w_{14}^{(1)} \\ w_{21}^{(1)} & w_{22}^{(1)} & w_{23}^{(1)} & w_{24}^{(1)} \end{pmatrix} + \begin{pmatrix} b_1^{(1)} & b_2^{(1)} & b_3^{(1)} & b_4^{(1)} \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \quad (14)$$

L'ajout de couches cachées dans les réseaux de neurones peut compliquer la fonction de prédiction y en augmentant le nombre de matrices de dimensions différentes. Les couches cachées sont considérées comme des activations qui reçoivent des signaux des couches précédentes et les modifient. Cela peut donner lieu à des fonctions imbriquées

II.6. Types de réseaux de neurones.

Les réseaux de neurones peuvent être classés en plusieurs types selon la structure et les applications utilisées. Les réseaux de neurones peuvent être multicouches (MLP), récurrents (RNN), convolutionnels (CNN), et d'autres types. Le type de réseau de neurones est choisi en fonction du problème posé et des données disponibles[21].

II.7. Conversion d'images en vecteurs pour l'analyse par réseaux neuronaux.

La détection des nodules pulmonaires à l'aide du deep learning repose sur le traitement d'images, spécifiquement des scans de poumons. Ces images, représentées en matrices de pixels, constituent la matière première de notre apprentissage automatique. Chaque pixel de l'image correspond à une valeur qui détermine son intensité ou sa couleur.

En exploitant les caractéristiques matricielles de l'image, nous effectuons des opérations comme la rotation ou le filtrage. Par exemple, une rotation de 90 degrés s'obtient en transposant la matrice. Pour filtrer, il suffit de multiplier la matrice de l'image par un masque, aussi appelé noyau, contenant des coefficients définissant l'effet recherché, tels qu'une amélioration de la netteté ou une détection de contours.

Nos réseaux neuronaux nécessitent des vecteurs en entrée et en sortie - des matrices en une dimension. Cette configuration simplifie les calculs et garantit la compatibilité entre les couches du réseau. Pour transformer notre image en un tel vecteur, nous la redimensionnons en fonction de la taille attendue par le réseau, ensuite nous l'aplatissons en une ligne ou une colonne, et enfin, nous la normalisons, pour que ses valeurs soient allant de 0 à 1. Ces étapes permettent de réduire le nombre de paramètres à prendre en compte et d'améliorer ainsi l'efficacité de notre réseau.

Représenter les images en tant que matrice, opérer sur elles grâce à ces caractéristiques matricielles et les transformer en vecteurs pour les intégrer à notre réseau neuronal, sont des étapes cruciales dans l'exploitation du Deep Learning pour la détection de nodules pulmonaires avec l'ensemble de données Luna16.

II.8. Conclusion.

Après avoir exploré les principes fondamentaux du deep learning, la façon dont les réseaux neuronaux artificiels fonctionnent et le rôle des matrices dans cet apprentissage profond, nous avons pu comprendre l'importance de transformer les données d'images en vecteurs pour leur analyse par les réseaux de neurones. Ce processus s'avère d'une grande importance pour la détection des nodules pulmonaires sur l'ensemble de données Luna16.

Nous avons discuté en profondeur du fonctionnement des réseaux de neurones artificiels, moteurs du Deep Learning, capables d'extraire et d'interpréter des motifs complexes dans les données. Les matrices jouent un rôle central, stockant et mettant à jour les paramètres dans les réseaux neuronaux.

Le processus de transformation des images en vecteurs, pour leur analyse par les réseaux neuronaux, a également été souligné. Cette transformation facilite l'intégration des images dans le réseau et optimise son efficacité.

En résumé, le Deep Learning, grâce à sa capacité à traiter et analyser des images et sa grande adaptabilité, s'impose comme un outil essentiel dans la détection des nodules pulmonaires. L'utilisation de l'ensemble de données Luna16 présente un potentiel énorme pour améliorer la précision et la rapidité des diagnostics, contribuant de manière significative à la lutte contre le cancer du poumon. ■

Chapitre III.

DÉTECTION DES NODULES PULMONAIRES DU CANCER.

III.1. Introduction

Le cancer du poumon figure parmi les principales causes de mortalité liées au cancer dans le monde entier [25]. La reconnaissance et le diagnostic précoces des nodules pulmonaires, petites masses de tissu dans les poumons, peuvent considérablement augmenter les taux de survie et le succès du traitement pour les individus atteints de cancer du poumon. Cependant, la détection et la classification de ces nodules pulmonaires représentent un défi de taille en raison de leur taille, forme, emplacement et caractéristiques physiques variables [26]. De plus, la majorité des nodules pulmonaires sont bénins ou non cancéreux, avec seulement un faible pourcentage classé comme malin ou cancéreux [27]. Ces conditions créent des complications pour la détection et la classification automatisées des nodules pulmonaires par des modèles d'apprentissage automatique.

Un des défis majeurs dans le dépistage du cancer du poumon est de distinguer les nodules pulmonaires *bénins* et *malins* à partir des images de scanner. Les systèmes de détection assistée par ordinateur (CAO) peuvent aider les radiologues à identifier et à caractériser les nodules en fonction de leur taille, leur forme, leur évolution et leur *risque de malignité* [12, 28]. Ces systèmes utilisent des techniques d'intelligence artificielle, notamment des réseaux de neurones profonds, pour analyser les images et fournir une classification automatique des nodules. Cette approche peut réduire le temps de lecture, augmenter le taux de détection, harmoniser les pratiques cliniques et éviter des examens inutiles ou invasifs.

Dans cet étude, nous proposons d'utiliser le deep learning pour améliorer la prise en charge des nodules pulmonaires. Le deep learning permet d'apprendre à partir de grandes quantités de données et de réaliser des tâches complexes comme la classification ou la segmentation d'images. Nous utilisons le dataset **LIDC-IDRI**¹⁴[29], qui contient 1018 scanners thoraciques annotés par **quatre radiologues experts**. Chaque nodule pulmonaire est décrit par un fichier XML qui contient son *identifiant*, ses *caractéristiques* et sa *région d'intérêt*. Par exemple, voici le fichier XML correspondant au nodule numéro 4 :

¹⁴sous licence «Creative Commons Attribution 3.0 Unported License».

```

<unblindedReadModule>
  <moduleID>4</moduleID>
  <characteristics>
    <subtlety>4</subtlety>
    <internalStructure>1</internalStructure>
    <calcification>6</calcification>
    <sphericity>4</sphericity>
    <margin>4</margin>
    <lobulation>1</lobulation>
    <spiculation>2</spiculation>
    <texture>5</texture>
    <malignancy>3</malignancy>
  </characteristics>
  <roi>
    <imageZposition>1487.5</imageZposition>
    <imageSOP_UID>1.3.6.1.4.1.14519.5.2.1.6279.6001.270602739536521934855332163694</
imageSOP_UID>
    <inclusion>TRUE</inclusion>
    <edgeMap><xCoord>322</xCoord><yCoord>303</yCoord></edgeMap>
    <edgeMap><xCoord>322</xCoord><yCoord>304</yCoord></edgeMap>
    <edgeMap><xCoord>322</xCoord><yCoord>305</yCoord></edgeMap>
    <edgeMap><xCoord>323</xCoord><yCoord>303</yCoord></edgeMap>
    <edgeMap><xCoord>322</xCoord><yCoord>303</yCoord></edgeMap>
  </roi>
</unblindedReadModule>

```

Ce fichier XML contient des informations importantes sur le nodule, telles que sa *taille*, sa *forme*, sa *structure*, sa *calcification*, sa *texture* et son **risque de malignité**. Il contient également la *position* et le *contour* du nodule sur l'image.

Nous commençons par développer un modèle de classification qui peut identifier le **type** de nodulaire à partir des images CT scan(nodule ou lésion). Nous avons utilisé le dataset **LUNA16**¹⁵ [26], qui est un sous-ensemble du dataset *LIDC-IDRI*, pour entraîner et évaluer notre modèle de classification. Nous avons comparé les performances de notre modèle de classification avec celles des radiologues et avec d'autres études dans la tâche de classification des nodules.

Ensuite, nous créons un nouveau dataset appelé **TRPMLN**, qui extrait **les nodules qui ont une moyenne de malignité égale à 3 ou plus dans les annotations des quatre experts**. Nous avons inclus une annexe qui présente l'implémentation du code *PYTHON* pour créer cet ensemble de données. Nous développons un autre modèle de classification qui peut **classer les nodules en fonction de leur risque de malignité** à partir des images CT scan. Nous comparons les performances de notre modèle avec celles des radiologues et avec d'autres études dans le même tâche.

Le plan de l'étude est le suivant : dans la section Méthode, nous présentons le dataset *LIDC-IDRI*, le dataset *LUNA16*, le dataset *TRPMLN*, les modèles de deep learning et les critères d'évaluation. Dans la section Résultats, nous montrons les résultats obtenus par nos modèles

¹⁵sous licence «Creative Commons Attribution 4.0 International License».

sur les datasets *LUNA16* et *TRPMLN*. Dans la section Discussion, nous analysons les forces et les limites de notre approche, ainsi que les implications cliniques. Dans la section Conclusion, nous résumons nos contributions et proposons des perspectives futures. ■

III.2. Méthode.

Notre étude comprenait trois étapes principales : le prétraitement des données, le développement de l'algorithme de détection des nodules et l'évaluation des performances [27, 30, 31].

III.2.1. Ressources

Les ressources de notre étude étaient des scans CT et des annotations provenant du dataset *LIDC-IDRI*, du dataset *LUNA16* et *TRPMLN*. Le dataset *LIDC-IDRI* est une base de données publique qui contient 1018 scans thoraciques annotés par quatre radiologues experts. Chaque nodule pulmonaire est décrit par un fichier XML qui contient son identifiant, ses caractéristiques et sa région d'intérêt¹⁶.

Voici un tableau des points clés du résumé :

Pointer	Résumé
Origine	Partenariat public-privé ¹⁷
Taille	1018 CT scans
Nodules	Annoté par 4 radiologues
Annotations	Deux phases
Disponibilité	Disponible publiquement
Utiliser	Développement et évaluation CAO

III.2.2. Scans CT avec Nodules Pulmonaires.

Pour lire, traiter et représenter visuellement les scans CT montrant des nodules pulmonaires, nous avons mis en œuvre trois bibliothèques Python : *SimpleITK*¹⁸, *Pylidc*¹⁹, et *matplotlib*²⁰.

- Avec ***SimpleITK***[32, 33, 34] et ***Pylidc***[35], nous avons lu les fichiers de scan CT de l'ensemble de données *LUNA16* ou *LIDC-IDRI*, convertissant ces images de leur format *raw*, *mhd* ou *DICOM* en tableaux numériques multidimensionnels manipulables, appelés tableaux *numpy*.
- Nous avons utilisé ***Matplotlib***[36] pour tracer et afficher les tranches de scan CT contenant des nodules, complétant ces images par des lignes blanches marquant les limites autour de chaque nodule pour souligner leur emplacement et leurs dimensions.

¹⁶Dataset Wiki: <https://wiki.cancerimagingarchive.net/pages/viewpage.action?pageId=1966254>

¹⁷L'origine de l'ensemble de données *LIDC-IDRI* est un partenariat public-privé car il a été financé à la fois par le secteur public (le National Cancer Institute) et le secteur privé (la Fondation pour les National Institutes of Health et les sociétés d'imagerie médicale qui ont participé à le projet).

¹⁸sous licence «Apache License 2.0.».

¹⁹sous licence «MIT License.».

²⁰sous licence «PSF License.».

La Figure III.15 offre un exemple d'une tranche de scan CT, où le nodule est mis en évidence par des lignes blanches.



Figure III.15: Exemples d'une tranche de scan CT avec un nodule mis en évidence par des lignes blanches.

III.2.2.1. Dataset LUNA16.

Le dataset *LUNA16* est un sous-ensemble du dataset *LIDC-IDRI*, qui contient 1186 nodules annotés dans 888 scans thoraciques. Ce dataset fournit également deux fichiers CSV distincts contenant les détails des candidats et des annotations. Dans le fichier *candidates.csv*, quatre colonnes sont illustrées : *seriesuid*, *coordX*, *coordY*, *coordZ*, et *classe*. Ici, le *seriesuid* fonctionne comme un identifiant unique pour chaque scan ; *coordX*, *coordY*, et *coordZ* représentent les coordonnées spatiales pour chaque candidat en millimètres, et *classe* fournit une catégorisation binaire, dépeignant si le candidat est un nodule (1) ou non (0).

<i>seriesuid</i>	<i>coordX</i>	<i>coordY</i>	<i>coordZ</i>	<i>class</i>
1.3.6...666836860	68.42	-74.48	-288.7	0
1.3.6...666836860	68.42	-74.48	-288.7	0
1.3.6...666836860	-95.20936148	-91.80940617	-377.4263503	0

Table III.2: Coordonnées des candidats détectés dans le dataset Luna16 avec diamètres

Le fichier *annotations.csv* est composé de cinq colonnes : *seriesuid*, *coordX*, *coordY*, *coordZ*, et *diamètre_mm*, commandant l'identifiant unique du scanner, les coordonnées d'annotation spatiales en millimètres, et le diamètre de chaque annotation en millimètres, respectivement. Ces annotations ont été marquées manuellement en se basant sur l'identification des nodules de plus de 3 mm de diamètre par quatre radiologistes indépendants [27, 30, 31].

<i>seriesuid</i>	<i>coordX</i>	<i>coordY</i>	<i>coordZ</i>	<i>diameter_mm</i>
1.3.6.1....6860	-128.6994211	-175.3192718	-298.3875064	5.65147063
1.3.6.1....6860	103.7836509	-211.9251487	-227.12125	4.224708481
1.3.6.1....5208	69.63901724	-140.9445859	876.3744957	5.786347814

Table III.3: Annotations des nodules détectés dans le dataset Luna16

III.2.2.2. Dataset TRPMLN.

La création du jeu de données *TRPMLN* implique plusieurs étapes. Initialement, un environnement virtuel Python est établi et activé, suivi par l'installation des packages Python nécessaires à partir d'un fichier

```
requirements.txt
```

. Un fichier de configuration pour

```
pyl IDC
```

est ensuite généré, qui spécifie le chemin vers le jeu de données LIDC-IDRI.

Le jeu de données LIDC-IDRI est par la suite interrogé en utilisant

```
py IDC
```

, en filtrant les scans avec une épaisseur de tranche supérieure à 3 mm et un espacement de pixels supérieur à 1 mm. Pour chaque scan dans le jeu de données filtré, les annotations sont regroupées en nodules. Un score moyen de malignité est calculé pour chaque nodule sur la base des scores fournis par différents experts. Si ce score moyen est de 3 ou plus, le nodule est classé comme cancéreux ; sinon, il est considéré comme normal.

Les données pour chaque nodule sont enregistrées, y compris le nom du nodule (qui indique s'il est cancéreux ou non) et l'objet d'annotation. Ces données sont utilisées pour créer un fichier CSV avec deux colonnes : « roi_name » (le nom du nodule) et « cancer » (indiquant s'il est cancéreux ou non).

Pour chaque ligne de données, une région d'intérêt (ROI) est extraite du volume du scan en fonction de la boîte englobante de l'annotation. La ROI est normalisée à une plage de 0 à 255 pour les images en 8 bits et sauvegardée en tant qu'image TIFF dans un répertoire spécifié. Pour gérer l'utilisation de la mémoire, le *garbage collector* de Python est invoqué toutes les 10 itérations pour nettoyer la mémoire inutilisée.

Le jeu de données *TRPMLN* final comprend des images TIFF de nodules et un fichier CSV contenant des informations sur ces nodules. Ce jeu de données peut être utilisé pour entraîner des modèles d'apprentissage profond pour classer les nodules malins.

Ces étapes sont démontrées dans l'ANNEXE 2.

III.2.3. Développement des algorithmes de détection des nodules.

Après avoir préparé les images de scans CT, l'ensemble de données (*LUNA16* ou *TRPMLN*) a été divisé en ensembles d'entraînement et de test. L'ensemble d'entraînement comprenait 67% des données, tandis que l'ensemble de test comprenait les 33% restants.

Deux modèles ont été entraînés sur l'ensemble d'entraînement et évalués sur l'ensemble de test : un modèle *CNN* et un modèle *ResNet50*. Les performances des modèles ont été mesurées en utilisant l'exactitude, le rappel et le score F1.

La construction des algorithmes de détection des nodules a été divisée en plusieurs étapes impératives. À leur base, les algorithmes reposaient sur un modèle de réseau neuronal convo-

lutf (CNN) et un modèle *ResNet50*, chargés d'identifier les nodules à partir d'images de scans CT [37]. Le modèle *CNN* était utilisé pour détecter la présence de nodules pulmonaires ou de lésions pulmonaires, tandis que le modèle *ResNet50* était utilisé pour classification des nodules selon l'approximation du risque de cancer, malin ou non malin.

III.2.3.1. Model 1: Detection de type de Nodule.

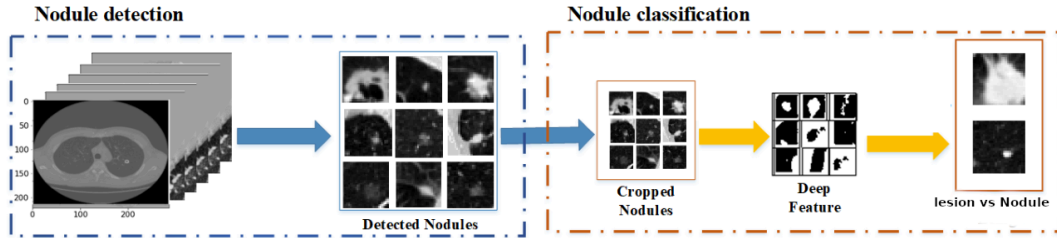


Figure III.16: La structure du modèle.

Utilisant le dataset *LUNA16*, qui contient des images de nodules et de lésions pulmonaires, nous avons expérimenté différentes combinaisons de couches et de filtres pour trouver la meilleure architecture pour notre problème. Nous avons trouvé que le meilleur modèle pour nous était conçu comme suit :

Stage	Output	Param
Conv2D	(None, 64, 64, 64)	4160
Conv2D	(None, 64, 64, 64)	16448
MaxPooling2D	(None, 8, 8, 64)	0
Conv2D	(None, 8, 8, 64)	16448
Conv2D	(None, 8, 8, 64)	16448
MaxPooling2D	(None, 4, 4, 64)	0
Conv2D	(None, 4, 4, 64)	16448
Conv2D	(None, 4, 4, 64)	16448
Conv2D	(None, 4, 4, 64)	262208
MaxPooling2D	(None, 2, 2, 64)	0
GlobalAveragePooling2	(None, 64)	0
Flatten	(None, 64)	0
Dense	(None, 2)	130
Total params: 348,738		

Table III.4: Architecture du modèle CNN.

Pour entraîner le modèle, l'optimiseur *Adam*[17] a été utilisé avec un taux d'apprentissage de 0,001, et la fonction de perte *d'entropie croisée binaire*. Cette fonction de perte mesure la divergence entre la probabilité prédite par le modèle et la vérité terrain pour chaque image. Elle est adaptée aux problèmes de classification binaire, comme celui de détecter la présence ou l'absence de nodules. L'entropie croisée binaire pénalise les prédictions erronées plus fortement que les prédictions correctes, ce qui encourage le modèle à apprendre à distinguer les nodules des non-nodules avec une grande confiance. L'entraînement du modèle s'est étendu

sur 100 époques, 6691 nodules ont été utilisés, dont 4165 de classe 0 et 2526 de classe 1. Les nodules recadrés détectés sont de taille $64 \times 64 \times 1$.

III.2.3.2. Model 2: Détection de nodule à risque de cancer.

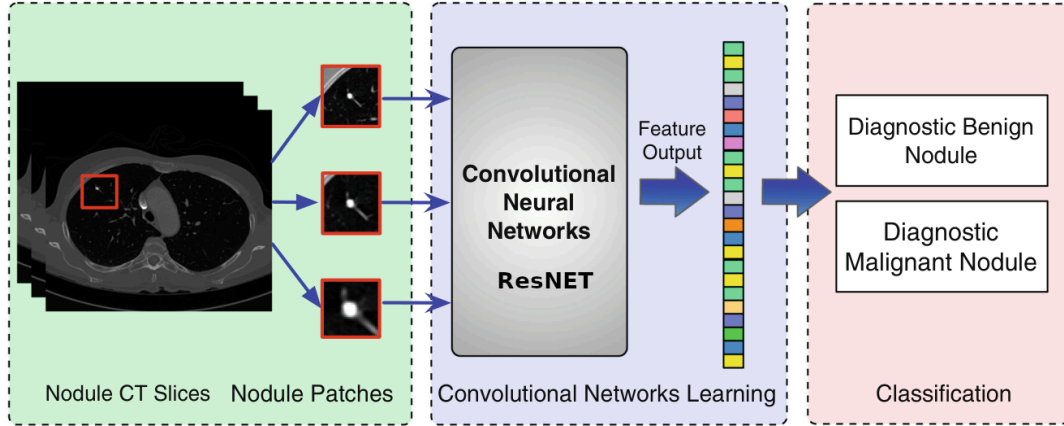


Figure III.17: Un aperçu de ResNET. Notre approche extrait d'abord plusieurs patches de nodules pour capturer le large éventail de variabilité des nodules à partir des images CT d'entrée. Enfin, notre approche applique un classificateur pour étiqueter la malignité du nodule d'entrée.

Nous développons aussi un modèle pour **approximer la probabilité de risque de malignité des nodules pulmonaires** à partir d'images CT scan, basé sur l'ensemble de données qui a été créé (TRPMLN) en se basant sur le dataset original **LIDC-IDRI** et en nous appuyant sur les annotations de quatre radiologues experts. Nous avons expérimenté différentes combinaisons de couches et de filtres pour trouver la meilleure architecture pour notre problème. Nous avons trouvé que le meilleur modèle pour nous était conçu comme suit :

Layer type	Output Shape	Param
InputLayer	(None, 64, 64, 1)	0
Concatenate	(None, 64, 64, 3)	0
UpSampling2D	(None, 192, 192, 3)	0
ZeroPadding2D	(None, 198, 198, 3)	0
ResNet	-	23587712
GlobalAveragePooling2D	(None, 2048)	0
Flatten	(None, 2048)	0
Dense	(None, 2)	4098
Total params: 23591810		

Table III.5: Architecture du modèle ResNET.

Les performances de classification des nodules du système conçu ont été évaluées sur TRPMLN. Pour la formation à la classification, 1568 nodules ont été utilisés, dont 801 de classe 0 et 767 de classe 1. Les nodules recadrés détectés sont de taille $64 \times 64 \times 1$. Entraînés sur 118 époques, un taux d'apprentissage de 0,001 a été utilisé, et comme optimiseur on a utilisé Adam et la fonction de perte d'entropie croisée binaire.

III.3. Résultats.

III.3.1. Évaluation des performances du modèle.

Nous avons évalué le succès du modèles à travers son **exactitude**²¹ sur les ensembles de données d'entraînement et de validation. L'**exactitude** du modèle sur les données d'entraînement et de validation a été documentée à chaque étape du processus d'apprentissage [26].

Le terme **exactitude** fait référence à la capacité du modèle à prévoir correctement les résultats sur les données d'entraînement, tandis que l'**exactitude de validation** signifie la capacité du modèle à généraliser ses prédictions à de nouvelles données inédites, c'est-à-dire les données de validation.

III.3.2. Métriques d'évaluation : Précision, Rappel et Score F1.

La performance du modèle peut a été évaluée à partir de *la matrice de confusion*, qui permet de calculer des métriques comme la **précision**, le **rappel (sensibilité)** et le **score F1**, en plus de l'**exactitude**. Ces mesures fournissent un aperçu plus large des performances du modèle, notamment quand il y a un déséquilibre des classes.

- La **précision** représente la fraction des prédictions positives correctes (plus précisément, lorsque le modèle identifie correctement un nodule) sur toutes les prévisions positives faites par le modèle. Une précision élevée indique un faible taux de faux positifs du modèle.

$$\text{précision} = \frac{VP}{VP + FP} \quad (15)$$

- Le **Rappel (Sensibilité)**, synonyme de sensibilité ou de taux de vrais positifs, est le rapport des prédictions positives correctes à tous les positifs réels. Un rappel élevé indique que le modèle a correctement identifié la majorité des cas positifs réels.

$$\text{rappel} = \frac{VP}{VP + FN} \quad (16)$$

- Le **F1-score** est la moyenne harmonique de la précision et du rappel, fournissant une seule mesure qui équilibre ces métriques.

•

$$F_1 = \frac{2 VP}{2VP + FP + FN} \quad (17)$$

III.3.2.1. Performances du modèle 1 sur LUNA16 en utilisant CNN

²¹exactitude: "accuracy" en Anglais.

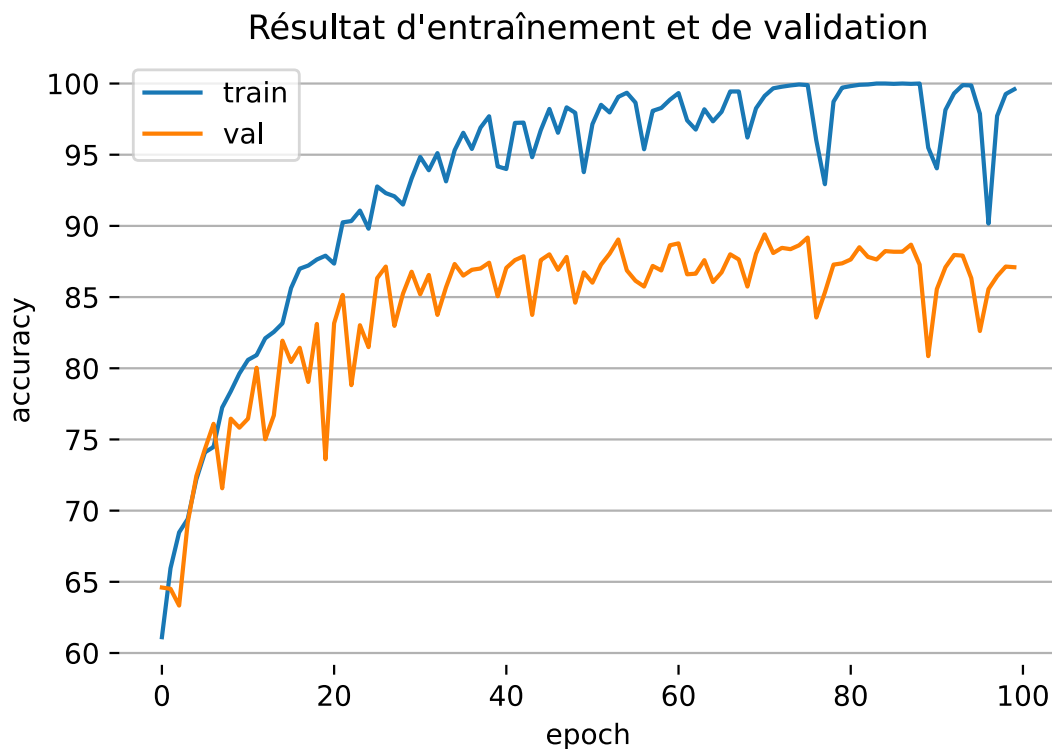


Figure III.18: Évolution des précisions d'entraînement et de validation de model 1 au cours de l'apprentissage.

En examinant les valeurs d'**exactitude** et d'**exactitude de validation** tout au long des étapes d'apprentissage, il est indiqué que le modèle acquiert des connaissances, comme on peut le voir à travers l'amélioration progressive des exactitudes d'entraînement et de validation. Le modèle commence avec des exactitudes relativement plus faibles, autour de 64%, avant d'augmenter à plus de 89% et de terminer avec un score de 87% à la fin de l'entraînement. Cela démontre la capacité raffinée du modèle à catégoriser correctement un ratio considérable de cas.

	<i>Prédiction négative</i>	<i>Prédiction positive</i>
<i>Réel Négatif</i>	771 VN	51 FP
<i>Réel Positif</i>	113 FN	404 VP

Table III.6: La matrice de confusion.

	<i>Précision</i>	<i>Rappel (sensibilité)</i>	<i>F1-score</i>
<i>Model CNN</i>	88.79%	75.23%	81.14%

Table III.7: Précision, rappel et F1-score du modèle 1

III.3.2.2. Performances du modèle 2 sur TRPMLN en utilisant RESNET.

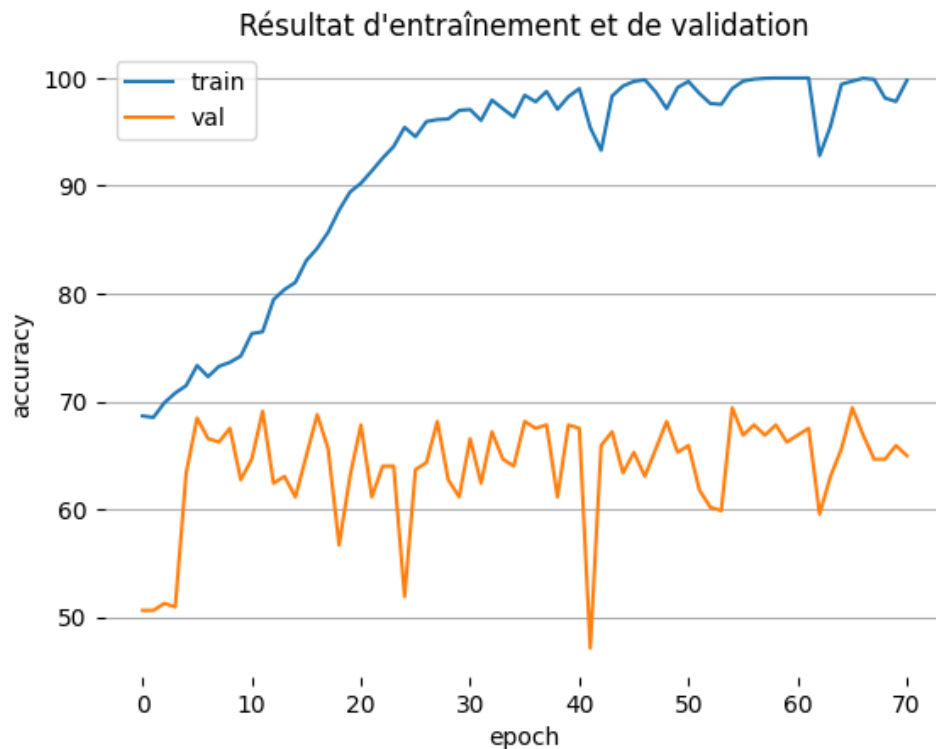


Figure III.19: Évolution des précisions d'entraînement et de validation de model 2 au cours de l'apprentissage.

Le modèle commence avec une exactitude d'environ 68,66 % à la première époque et s'améliore progressivement. À la 60ème époque, le modèle atteint une précision d'entraînement de 100% et la maintient pendant plusieurs époques.

La exactitude de validation commence à environ 50,63 % et fluctue tout au long du processus de formation, atteignant un pic d'environ 69,43 % mais ne se rapprochant jamais de la précision d'entraînement.

Cela suggère que le modèle a très bien appris les données d'entraînement, mais qu'il présente un surapprentissage.

	Prédiction négative	Prédiction positive
Réel Négatif	119 VN	40 FP
Réel Positif	70 FN	85 VP

Table III.8: La matrice de confusion.

	Précision	Rappel (sensibilité)	F1-score
Model ResNET	68.00%	54.83%	60.71%

Table III.9: Précision, rappel et F1-score du modèle 2

III.4. Discussion.

Dans les deux modèles, nous avons un overfitting et des fluctuations de précision.

- **Overfitting** : la exactitude de l'entraînement atteint 100 %, ce qui est un signe clair de surajustement, surtout par rapport à la exactitude de validation qui est bien inférieure. Le

surajustement signifie que le modèle a trop bien appris les données d'entraînement, y compris leur bruit et leurs valeurs aberrantes, ce qui le rend peu performant sur les données invisibles.

- **Fluctuations de la exactitude de la validation** : la exactitude de la validation fluctue beaucoup, ce qui peut suggérer que le modèle est instable ou que l'ensemble de validation n'est peut-être pas suffisamment représentatif.

	Précision	Rappel (sensibilité)
Song et al. [38]	82%	83%
Nibali et al. [39]	89%	91%
Zhao et al. [40]	82%	
Nos modèles	88.79%	75.25%

Table III.10: Comparaison avec d'autres études dans la tâche de classification des nodules ou des lésions.

Comparaison avec nos résultats, notre modèle de classification de nodule ou lésion est performant de manière compétente dans l'identification des deux classes. En général, le modèle a performé de manière impressionnante en termes de précision, de rappel et de score F1.

Models	Accuracy (%)	Year
Multi-scale CNN [12]	86.84	2015
Nodule level 2D CNN [41]	87.30	2016
Slice level 2D CNN [41]	86.70	2016
Multi-crop CNN [14]	87.14	2017
Vanilla 3D CNN [41]	87.40	2016
Deep 3D DPN [42]	88.74	2017
Deep 3D DPN + GBM [42]	90.44	2017
3D MixNet [43]	88.83	2019
3D MixNet + GBM [43]	90.57	2019
3D CMixNet + GBM [28]	91.13	2019
3D CMixNet + GBM + Biomarkers [28]	94.17	2019
Our Model ResNET	69,43	2023

Table III.11: Comparaison avec d'autres études dans le cadre de la classification des nodules malins ou bénins [38, 39, 40].

Mais la comparaison de notre modèle de classification de nodule ou de lésion avec d'autres modèles n'est pas performante de manière compétitive, lorsque l'on veut approximer la détection des nodules malins.

Plusieurs facteurs peuvent expliquer pourquoi le modèle a tendance à identifier la classe 0 plutôt que la classe 1. L'une des stratégies pourrait être de renforcer la présence de la classe 1, ce qui pourrait aider le modèle à mieux distinguer la classe dominante et à améliorer légèrement sa performance pour cette classe.

De plus, la détection des nodules peut être une tâche plus complexe que celle des non-nodules. Les nodules sont souvent de petite taille, flous ou masqués par d'autres structures pulmonaires, ce qui peut compliquer leur détection. Ainsi, l'exploration d'autres techniques d'optimisation du modèle pourrait être bénéfique pour atténuer le biais du modèle en faveur de la classe 0.

En outre, les variations de caractéristiques entre les classes peuvent également entraîner des taux de détection différents. Des analyses plus approfondies, comme un examen détaillé des caractéristiques des données d'entrée, pourraient aider à comprendre précisément pourquoi ces différences de performances sont observées. L'utilisation de techniques de prétraitement des images, comme la normalisation, pourrait améliorer la qualité et la diversité des données d'entrée.

Pour résoudre ce problème, une stratégie d'entraînement raffinée pour notre modèle est nécessaire, ainsi qu'un indicateur de performance plus robuste que la simple précision. Les solutions potentielles pourraient inclure :

- L'application de techniques spécifiques pour accroître le nombre d'échantillons malins dans notre ensemble de données.
- L'utilisation de techniques d'équilibrage des classes pour obtenir une distribution équilibrée des classes dans notre ensemble de données.

Dans notre travail ultérieur, nous visons à incorporer certaines de ces solutions et nous nous attendons à améliorer les performances de notre modèle par rapport à la classification des nodules pulmonaires, pour maîtriser la classification des sous-types de nodules, tels que solides, non-solides, partiellement solides, périfissuraux, calcifiés et spiculés. Différents traitements sont nécessaires pour différents types de nodules, ce qui rend leur détection précise encore plus pertinente pour un traitement réussi.

III.5. Conclusion.

Nous avons utilisé le Deep-Learning pour détecter et classer les nodules pulmonaires dans l'ensemble de données *LUNA16* et *TRPMLN*. Les modèles ont affronté des défis liés à la diversité des nodules en termes de taille, de forme et d'emplacement, ainsi qu'à une distribution inégale dans l'ensemble de données. Malgré ces difficultés, ils ont performé de manière satisfaisante, produisant des scores élevés, un bon rappel et un F1 score convaincant pour les nodules, qu'ils soient nodule ou lésion. Ils ont également performé de manière passable, produisant des scores passables pour le rappel et un *F1 score* convaincant pour les nodules, qu'ils soient bénins ou malins.

Les modèles ont montré un léger avantage dans l'identification des non-nodules, probablement en raison de la limitation du nombre de classe 1 dans l'ensemble de données.

Les résultats de notre étude soulignent que le Deep-Learning est efficace pour la détection et la classification des nodules pulmonaires. Il a le potentiel pour faciliter le diagnostic précoce du cancer du poumon, ce qui peut améliorer les chances de survie et l'efficacité du traitement.

Nous cherchons à améliorer nos modèles pour perfectionner leur performance, en particulier dans la détection des malignités des nodules pulmonaires. Pour cela, des recherches supplémentaires sont nécessaires.

ANNEXE 1.

CODE SOURCE EN C

```
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
// Train Data
float td[][2] = {
    // C    F
    {-40, -40},
    {-20, -4 },
    {0,   32 },
    {20,  68 },
    {40, 104},
    {60, 140},
    {80, 176},
    {100, 212},
};
#define N 8 // Number of item in Train Data
#define ALPHA 0.00001 // Define the learning alpha
#define EPOCS 100 * 1000 // Define the number of iterations

// Define a function to compute the mean squared error
double cost(double w, double b) {
    double error = 0.0;
    for (int i = 0; i < N; ++i) {
        double x = td[i][0];
        double y = td[i][1];
        double d = y - (w * x + b);
        error += d * d;
    }
    return error / (int) N;
}

// Define a function to perform gradient descent
void gradient_descent(double *w, double *b) {
    // Derivative of cost function with respect to w or b
    double dw = 0.0;
    double db = 0.0;
    for (int i = 0; i < N; i++) {
        double x = td[i][0];
        double y0 = td[i][1];
        double y = *w * x + *b;
        dw += x * (y - y0);
        db += (y - y0);
    }
}
```

```

    // Update w and b using the learning rate and the derivatives
    *w = *w - ALPHA * dw;
    *b = *b - ALPHA * db;
}

// Define a function to train the neuron using gradient descent
void train(double *w, double *b) {
    for (int i = 0; i < EPOCHS; i++) {
        gradient_descent(w, b);
        if (i % 101000 == 0)
            printf("Iteration: %d, Cost:%3.3f w=%.6lf b=%.6lf\n", i, cost(*w, *b),
                *w, *b);
    }
}

// Define a function to predict the output using the neuron
double predict(double x, double w, double b) {
    return w * x + b;
}

```

```

int main() {
    // Initialize w and b randomly
    double w = (double) rand() / RAND_MAX;
    double b = (double) rand() / RAND_MAX;

    // Train the neuron using gradient descent
    train(&w, &b);

    // Print the final values of w and b
    printf("\nFinal values are: w = %.6f and b = %.6f\n\n", w, b);

    // Test the neuron with some new inputs
    double x_new = 50; // Celsius
    double y_new = predict(x_new, w, b); // Fahrenheit
    printf("Fahrenheit of 50C: 122F\n");
    printf("Prediction of 50C: %.6fF\n", y_new);

    return 0;
}

```

```

Iteration: 10000, Cost:227.982250 w=1.980511 b=13.953147
Iteration: 20000, Cost: 74.392002 w=1.903113 b=21.691058
Iteration: 30000, Cost: 24.274565 w=1.858902 b=26.111201
Iteration: 40000, Cost:  7.920939 w=1.833647 b=28.636129
Iteration: 50000, Cost:  2.584651 w=1.819220 b=30.078449
Iteration: 60000, Cost:  0.843387 w=1.810979 b=30.902348
Iteration: 70000, Cost:  0.275202 w=1.806272 b=31.372986
Iteration: 80000, Cost:  0.089800 w=1.803583 b=31.641830
Iteration: 90000, Cost:  0.029302 w=1.802046 b=31.795402
Iteration: 100000, Cost:  0.009562 w=1.801169 b=31.883127

```

Final values are: $w = 1.801169$ and $b = 31.883127$

Fahrenheit of 50C: 122F

Prediction of 50C: 121.94F



ANNEXE 2.

TRPMLN IMPLANTATION

```
# This script generates the Dataset TRPMLN for lung nodules from LIDC-IDRI.

import os
import sys
import argparse

import gc # clean ram used by garbage collector
import random

import cv2 # for normalise image and save in specific format
import pandas as pd # for save some information to scv
import pylidc as pl # we need pylidc to query lcidi-idri datasete
from tqdm.auto import tqdm # progress bar

class ScanData:
    def __init__(self, path=None):
        if path is None:
            raise KeyError("please provied path of LIDCI-IDRI")
        self.scans = self.create_pylidcrc(path)
        self.extract_data()

    def create_pylidcrc(self, path):
        config_file = "/root/.pylidcrc"
        config = f"[dicom]\npath={path}"
        with open(config_file, "w") as f:
            f.write(config)
        # scans += pl.query(pl.Scan).all() # for query all slices
        return pl.query(pl.Scan).filter(pl.Scan.slice_thickness <= 3,
pl.Scan.pixel_spacing <= 1)

    def extract_data(self):
        self.data = []
        total_scans = self.scans.count()
        for i, scan in tqdm(enumerate(self.scans), total=total_scans):
            # if i > 4: # deactivate the test for the 5 first items.
            #     break
            nodules = scan.cluster_annotations()
            # Note: for each scan.id we have many nodules, each nodules
            # has many anns from diffrent experts.
            for anns in nodules:
                malignancies = 0
```

```

        for ann in anns:
            malignancies += ann.malignancy
        avg_malignancy = malignancies / len(anns)
        cancer = 1 if avg_malignancy >= 3 else 0
        cancer_name = "cancer" if cancer else "normal"

        #ann = random.choice(anns) # ROI extracting depend the ann
        #selected.

        ann = anns[0]
        roi_name = f"{cancer_name}_{scan.patient_id}_{scan.id}_{ann.id}.tiff"

        row = {
            "roi_name": roi_name,
            "ann": ann,
            # "scan_id": scan.id,
            # "rand_nodule_id": ann.id,
            "cancer": cancer,
        }
        self.data.append(row)

    return self

def write_to_csv(self, filename):
    if filename is None or filename == "":
        raise KeyError("you miss name of csv file to store into data info")
    df = pd.DataFrame(self.data, columns=["roi_name", "cancer"])
    df.to_csv(filename, index=False)
    return self

def save_roi_to_tiff(self, dir=None):
    if dir is None or dir == "":
        raise KeyError("you miss name of dir to store images")
    # padding = [(0, 0), (0, 0), (0, 0)] # for no padding
    padding = [(30, 10), (10, 25), (0, 0)]
    for i, row in tqdm(enumerate(self.data), total=len(self.data)):
        vol, roi, bbox, ann = None, None, None, None
        ann = row["ann"]
        bbox = ann.bbox(pad=padding)
        try:
            vol = ann.scan.to_volume()
        except Exception as e:
            print(f'Warning: {e}')
            continue

        for region in range(vol[bbox].shape[2]):
            roi = vol[bbox][:, :, region]
            # Rescale the ROI image to the range of 0 to 255 for 8-bit
            images
            roi = cv2.normalize(roi, None, 0, 255, cv2.NORM_MINMAX,
                               dtype=cv2.CV_8U)

```

```

        # Save the image as a TIFF file in the patient directory
        filename = row["roi_name"]
        cv2.imwrite(f"{dir}/{filename}", roi)
    if i % 10: # clean some ram usage, use more cpu and time.
        gc.collect()
    return self

if __name__ == '__main__':
    parser = argparse.ArgumentParser(description="generate the Dataset TRPMLN
    for lung nodules from LIDC-IDRI.")
    parser.add_argument("-d", "--dataset", type=str, help="The path for Dataset
    LIDC-IDRI")
    parser.add_argument("-r", "--roi", type=str, help="The path for the ROI
    directory extracted.")
    parser.add_argument("-c", "--csv", type=str, help="The path for the csv
    file generated.")

    args = parser.parse_args(sys.argv[1:])

    if args.dataset is None:
        raise ValueError("Please provide the path for LIDC-IDRI Dataset.")

    if not os.path.exists(args.dataset):
        raise ValueError(f"Dir {args.dataset} does not exist.")

    if args.roi is None:
        raise ValueError("Please provide the path for ROI Directory output.")

    if not os.path.exists(args.roi):
        raise ValueError(f"Dir {args.roi} does not exist.")

    if args.csv is None:
        raise ValueError("Please provide the path to store the csv filename
    generated.")

    scan_data = ScanData(path=args.dataset)
    scan_data.write_to_csv(filename=args.csv)
    scan_data.save_roi_to_tiff(args.roi)

```

■

RÉFÉRENCES BIBLIOGRAPHIQUES.

- [1] R. L. Siegel, K. D. Miller, and A. Jemal, “Cancer Statistics, 2017,” *CA: Cancer J. Clinicians*, vol. 67, no. 1, pp. 7–30, 2017.
- [2] National Lung Screening Trial Research Team, D. R. Aberle, et al., “Reduced Lung-Cancer Mortality with Low-Dose Computed Tomographic Screening,” *New England J. Medicine*, vol. 365, no. 5, pp. 395–409, 2011.
- [3] M. K. Gould, J. Fletcher, et al., “Evaluation of Patients with Pulmonary Nodules: When is it Lung Cancer? ACCP Evidence-Based Clinical Practice Guidelines (2nd edition),” *Chest*, vol. 132, no. 3, p. 108, 2007.
- [4] A. A. A. Setio, F. Ciompi, et al., “Pulmonary Nodule Detection in CT Images: False Positive Reduction Using Multi-View Convolutional Networks,” *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1160–1169, 2016.
- [5] H. R. Roth, L. Lu, et al., “Improving Computer-Aided Detection using Convolutional Neural Networks and Random View Aggregation,” *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1170–1181, 2016.
- [6] T.-Y. Lin, P. Dollár, et al., “Feature Pyramid Networks for Object Detection,” *Arxiv Preprint Arxiv:1612.03144*, 2016.
- [7] K. Kamnitsas, C. Ledig, et al., “Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation,” *Med. Image Anal.*, vol. 36, pp. 61–78, 2017.
- [8] C. Jacobs, E. M. van Rikxoort, et al., “Automatic Detection of Subsolid Pulmonary Nodules in Thoracic Computed Tomography Images,” *Med. Image Anal.*, vol. 18, no. 2, pp. 374–384, 2014.
- [9] T. Okumura, T. Miwa, et al., “Automatic Detection of Lung Cancers in Chest CT Images by Variable N-Quoit Filter,” vol. 2, pp. 1671–1673, 1998.
- [10] Q. Li, S. Sone, and K. Doi, “Selective Enhancement Filters for Nodules, Vessels, and Airway Walls in Two- and Three-Dimensional CT Scans,” *Med. Phys.*, vol. 30, no. 8, pp. 2040–2051, 2003.
- [11] J. Ding, A. Li, Z. Hu, and L. Wang, “Accurate Pulmonary Nodule Detection in Computed Tomography Images Using Deep Convolutional Neural Networks,” *Arxiv Preprint Arxiv:1706.04303*, pp. 1–9, 2017.
- [12] W. Shen, M. Zhou, F. Yang, C. Yang, and J. Tian, “Multi-scale Convolutional Neural Networks for Lung Nodule Classification,” pp. 588–599, 2015, doi: 10.1007/978-3-319-19992-4_46. [Online]. Available: https://doi.org/10.1007/978-3-319-19992-4_46

- [13] Q. Dou, H. Chen, L. Yu, J. Qin, and P. A. Heng, "Multi-level Contextual 3D CNNs for False Positive Reduction in Pulmonary Nodule Detection," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 7, pp. 1558–1567, 2017.
- [14] W. Shen, M. Zhou, et al., "Multi-crop convolutional neural networks for lung nodule malignancy suspiciousness classification," *Pattern Recognit.*, vol. 61, pp. 663–673, Jan. 2017, doi: 10.1016/j.patcog.2016.05.029. [Online]. Available: <https://doi.org/10.1016/j.patcog.2016.05.029>
- [15] I. Goodfellow, Y. Bengio, and A. Courville, "Deep learning," 2016.
- [16] S. S. Zumdahl, and S. A. Zumdahl, *Chemistry*, Cengage Learning, 2013.
- [17] D. P. Kingma, and J. Ba, "Adam: a method for stochastic optimization," *Arxiv Preprint Arxiv:1412.6980*, 2014.
- [18] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "A comprehensive survey and performance analysis of activation functions in deep learning," *Corr*, 2021. [Online]. Available: <https://arxiv.org/abs/2109.14545>
- [19] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," in *Nature*, vol. 323, 1986, pp. 533–536.
- [20] M. Spivak, *Calculus*, Cambridge University Press, 1967.
- [21] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [22] C. C. Aggarwal, *Neural Networks and Deep Learning: A Textbook*, Springer, 2018. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-94463-0>
- [23] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into deep learning," 2023.
- [24] C. M. Bishop, *Pattern Recognition and Machine Learning*, springer, 2006.
- [25] N. Team, "Reduced lung-cancer mortality with low-dose computed tomographic screening," *New England J. Medicine*, vol. 365, no. 5, pp. 395–409, 2011.
- [26] A. Setio, A. Traverso, T. de~Bel, and etc., "Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the luna16 challenge," *Corr*, 2016.
- [27] Q. Dou, H. Chen, et al., "Automated pulmonary nodule detection via 3D convnets with online sample filtering and hybrid-loss residual learning," in *Miccai*, 2017.
- [28] N. Nasrullah, J. Sang, et al., "Automated lung nodule detection and classification using deep learning combined with multiple strategies," *Sensors*, vol. 19, no. 17, p. 3722, Aug. 2019, doi: 10.3390/s19173722. [Online]. Available: <https://doi.org/10.3390/s19173722>
- [29] K. Clark, B. Vendt, et al., "The cancer imaging archive (TCIA): maintaining and operating a public information repository," *J. Digit. Imag.*, vol. 26, no. 6, pp. 1045–1057, Jul. 2013, doi: 10.1007/s10278-013-9622-7. [Online]. Available: <https://doi.org/10.1007/s10278-013-9622-7>

- [30] J. Ding, A. Li, Z. Hu, and L. Wang, “Accurate pulmonary nodule detection in computed tomography images using deep convolutional neural networks,” in *Miccai*, 2017.
- [31] S. Armato, G. McLennan, L. Bidaut, and etc., “The lung image database consortium (lidc) and image database resource initiative (idri),” *Med. Eng. & Phys.*, vol. 38, no. 2, 2011.
- [32] R. Beare, B. Lowekamp, and Z. Yaniv, “Image segmentation, registration and characterization in \LaTeX with \LaTeX SimpleITK,” *J. Statistical Softw.*, vol. 86, no. 8, 2018, doi: 10.18637/jss.v086.i08. [Online]. Available: <https://doi.org/10.18637/jss.v086.i08>
- [33] Z. Yaniv, B. C. Lowekamp, H. J. Johnson, and R. Beare, “SimpleITK image-analysis notebooks: a collaborative environment for education and reproducible research,” *J. Digit. Imag.*, vol. 31, no. 3, pp. 290–303, Nov. 2017, doi: 10.1007/s10278-017-0037-8. [Online]. Available: <https://doi.org/10.1007/s10278-017-0037-8>
- [34] B. C. Lowekamp, D. T. Chen, L. Ibáñez, and D. Blezek, “The design of SimpleITK,” *Frontiers Neuroinformatics*, vol. 7, 2013, doi: 10.3389/fninf.2013.00045. [Online]. Available: <https://doi.org/10.3389/fninf.2013.00045>
- [35] M. C. Hancock, and J. F. Magnan, “Lung nodule malignancy classification using only radiologist-quantified image features as inputs to statistical learning algorithms: probing the lung image database consortium dataset with two statistical learning methods,” *J. Med. Imag.*, vol. 3, no. 4, p. 44504, Dec. 2016, doi: 10.1117/1.jmi.3.4.044504. [Online]. Available: <https://doi.org/10.1117/1.jmi.3.4.044504>
- [36] 2007, doi: 10.1109/MCSE.2007.55.
- [37] T. Lin, P. Dollár, et al., “Feature pyramid networks for object detection,” in *Cvpr*, vol. 1, 2017, p. 4.
- [38] Q. Song, L. Zhao, X. Luo, and X. Dou, “Using deep learning for classification of lung nodules on computed tomography images,” *J. Healthcare Eng.*, vol. 2017, pp. 1–7, 2017, doi: 10.1155/2017/8314740. [Online]. Available: <https://doi.org/10.1155/2017/8314740>
- [39] A. Nibali, Z. He, and D. Wollersheim, “Pulmonary nodule classification with deep residual networks,” *Int. J. Comput. Assisted Radiol. Surgery*, vol. 12, no. 10, pp. 1799–1808, May 2017, doi: 10.1007/s11548-017-1605-6. [Online]. Available: <https://doi.org/10.1007/s11548-017-1605-6>
- [40] X. Zhao, L. Liu, et al., “Agile convolutional neural network for pulmonary nodule classification using CT images,” *Int. J. Comput. Assisted Radiol. Surgery*, vol. 13, no. 4, pp. 585–595, Feb. 2018, doi: 10.1007/s11548-017-1696-0. [Online]. Available: <https://doi.org/10.1007/s11548-017-1696-0>
- [41] S.-H. Lai, V. Lepetit, K. Nishino, and Y. Sato, Eds., *Computer Vision – ACCV 2016*, vol. 10111, Springer International Publishing, 2017, p. 456. [Online]. Available: <https://doi.org/10.1007/978-3-319-54181-5>
- [42] W. Zhu, Q. Lou, Y. S. Vang, and X. Xie, “Deep multi-instance networks with sparse label assignment for whole mammogram classification,” Dec. 2016, doi: 10.1101/095794. [Online]. Available: <https://doi.org/10.1101/095794>

- [43] N. Nasrullah, J. Sang, M. S. Alam, and H. Xiang, “Automated detection and classification for early stage lung cancer on CT images using deep learning,” in *Pattern Recognit. Tracking Xxx*, May 2019, doi: 10.1117/12.2520333. [Online]. Available: <https://doi.org/10.1117/12.2520333>