République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur Et de La Recherche Scientifique
Université Kasdi Merbah Ouargla

Faculté des Nouvelles Technologies de L'Information et de la Communication
Département d'Electronique et de Télécommunication

Mémoire présenté en vue de l'obtention du diplôme de

Master Académique

Filière : *Electronique*
Spécialité : *Electronique des systèmes embarqués*

Par : Badreddine atallah

Thème

Etude statistique pour le choix des paramètres de classification des images médicale par des modèles d'intelligence artificielle

Soutenu publiquement le : 19 Juin 2023

Devant le jury :

BENSID Khaled	**MCB**	Univ. Kasdi merbah - Ouargla	**Président**
CHERGUI Abdelhakim	**MCB**	Univ. Kasdi merbah - Ouargla	**Examinateur**
LATI Abdelhai	**MCA**	Univ. Kasdi merbah - Ouargla	**Encadreur**

Année universitaire 2022/2023


بسم الله الرحمن الرحيم

إهداء

أهدي ثمرة جهدي بفضل الله تعالى إلى بلدي..... وإلى الوالدين الكريمين حفظهما الله وأدامهما نورا لدربي وإلى زوجتي التي ساندتني ولا تزال وإلى أبنائي "حسام الدين" و "المعتصم بالله" وأخوتي وأصدقائي ولجميع أولئك الذين يعملون من أجل الجزائر.

Dédicace

Je dédie le fruit de mes efforts, par la grâce de Dieu Tout-Puissant, à mon pays. . . . . et à mes honorables parents, que Dieu les protège et les garde comme une lumière sur mon chemin, à ma femme qui m'a soutenu et qui s'est toujours, à mes fils " Houssam Eddine " et" Moatasem Bellah ", mes frères et amis, et à tous ceux qui travaillent pour l'Algérie.

شكر وتقدير

اللهم لك الحمد والشكر .... أما بعد أتقدم بالشكر الجزيل الى كل من قدم لي يد العون في تحقيق هذا العمل المتواضع وخاصة الأستاذ الفاضل" **العاتي عبد الحي**" بجامعة قاصدي مرباح -ورقلة

الذي كان لي المشرف المشجع.

وإلى كل من ساهم بشكل مباشر أو غير مباشر في تحقيق هذا العمل.

Remerciement

Oh mon Dieu, louange et merci à toi.... Maintenant, je voudrais adresser mes sincères remerciements à tous ceux qui m'ont donné un coup de main dans la réalisation de cet humble travail, en particulier l'honorable professeur

"Lati Abdelhai "de l'Université de Kasdi Merbah - Ouargla ce qui m'a beaucoup encouragé

Et à tous ceux qui ont contribué directement ou indirectement à la réalisation de ce travail.

ملخص:

في هذا العمل نقدم تصنيف صور الاشعة السينية بهدف الكشف ما إذا كان الشخص مصابا بمرض الالتهاب الرئوي ام لا مما يسهل عملية التشخيص للأطباء ولذلك قمنا بتصميم مجموعة من نماذج الذكاء الاصطناعي وتم المفاضلة بينهم من خلال مقاييس الدقة وكانت دقة النماذج تتراوح بين 80.2 %أ الى 91.6 % وتم اختيار أفضل النماذج الا وهو نموذج الشبكة العصبية الالتفافية **CNN**بدقة تقدر 91.6 % ونشير الى ان النموذج يعتبر نموذج مساعد للطبيب المتخصص وليس بديلا عن الطبيب ولا يحل محله.

الكلمات المفتاحية: تصنيف الصور-التعلم الالي – نماذج التصنيف - الشبكة العصبية الالتفافية – تشخيص مرض الالتهاب الرئوي – SPSS Modeler.

Résumé:

Dans ce travail, nous présentons la classification des images radiographiques afin de détecter si une personne a une pneumonie ou non, ce qui facilite le processus de diagnostic pour les médecins par conséquent, nous avons conçu un ensemble de modèles d'intelligence artificielle, et des comparaisons ont été faites entre eux par des mesures de précision, et la précision des modèles variait de 80.2 % à 91.6 % , les meilleurs modèles ont été choisis, à savoir le modèle de réseau neuronal convolutif, avec une précision de 91.6 % Nous soulignons que le modèle est considéré comme un auxiliaire modèle pour le médecin spécialiste et ne se substitue pas au médecin et ne le remplace pas.

Mots clés : Classification d'images - apprentissage automatique - modèles de classification - réseau de neurones convolutifs - Diagnostic de pneumonie - SPSS Modeler.

Abstract:

In this work, we present the classification of x-ray images in order to detect whether a person has pneumonia or not, which facilitates the diagnosis process for doctors. Therefore, we designed a set of artificial intelligence models, and comparisons were made between them through accuracy measures, and the accuracy of the models ranged from 80.2 % to 91.6 %, the best models were chosen, which is the convolutional neural network model, with an accuracy of 91.6 % we point out that the model is considered an auxiliary model for the specialized doctor and is not a substitute for the doctor and does not replace him.

Keywords: Image classification - machine learning - classification models - convolutional neural network - Diagnosis of pneumonia - SPSS Modeler.

Sommaire

Sommaire

Chapitre II : Méthodologie du Travail

Chapitre III : Réalisation des modèles de classification

Liste des abréviations

ML: Machine Learning

DL: Deep Learning

AI: Artificial Intelligence

ReLu: Rectified Linear Units

AUC: Area Under Curve

KNN : k plus proches voisins

SVM : Machine à vecteurs de support

RB : Réseau Bayésien

AD : Arbre de Décision

LR : Régression de Logistique

NN : réseau de neurones

AA : Arbres aléatoire

CNN : Convolution neural network

Liste de figure

Liste de tableaux

Introduction générale

Introduction générale

La pneumonie est une forme d'infection respiratoire aiguë qui affecte les poumons. Les poumons sont constitués de petits sacs appelés alvéoles, et ces sacs se remplissent d'air lorsqu'une personne en bonne santé respire. Lorsqu'une personne souffre de pneumonie, les alvéoles pulmonaires se remplissent de pus et de liquide, ce qui rend la respiration douloureuse et limite l'apport d'oxygène.

La pneumonie est la principale cause de décès chez les enfants dans le monde. Elle a coûté la vie à 740 180 enfants de moins de cinq ans en 2019, ce qui représente 14 % de tous les décès enregistrés dans ce groupe et 22 % de tous les décès d'enfants âgés de 1 à 5 ans. La maladie affecte les enfants et leurs familles dans toutes les régions du monde, mais le taux de mortalité le plus élevé est enregistré en Asie du Sud et en Afrique subsaharienne.

Alors que la plupart des enfants en bonne santé peuvent repousser les infections grâce à leurs défenses naturelles, les enfants dont le système immunitaire est affaibli sont plus à risque de développer une pneumonie. Le système immunitaire d'un enfant peut être affaibli par la malnutrition ou son absence, en particulier chez les nourrissons qui ne sont pas nourris exclusivement au lait maternel.

Le Plan d'action mondial intégré OMS-UNICEF pour la prévention et le contrôle de la pneumonie et de la diarrhée vise à accélérer le contrôle de la pneumonie grâce à une combinaison d'interventions visant à protéger, prévenir et traiter les enfants contre la pneumonie.

Ce travail est une modeste contribution ajoutée à tous ces efforts, qui visent principalement à aider à la classification des images radiographiques afin de découvrir si une personne est atteinte de pneumonie. Pour cela, nous avons utilisé une base de données d'images radiographiques d'enfants atteints. Cette maladie et les enfants sans infection, puis nous avons transmis ces images à différents modèles de classification (qui est l'intelligence artificielle) pour les former et les utiliser plus tard pour déterminer si une personne a une pneumonie ou non, ce qui facilite le diagnostic pour les médecins. La base de données d'images radiographiques consistait en 5856 radiographies thoraciques (antéropostérieures) sélectionnées à partir de cohortes rétrospectives de patients pédiatriques âgés de 1 à 5 ans du Guangzhou Women and Children Medical Center, Guangzhou. Toutes les radiographies pulmonaires ont été prises dans le cadre des soins cliniques de routine des patients.

Dans ce travail, nous avons d'abord commencé par créer un ensemble de différents modèles de classification pour l'apprentissage automatique, puis nous les avons comparés par un ensemble de mesures de précision et la préférence absolue était le modèle de réseaux de neurones convolutifs.

Cette mémoire est organisée en trois chapitres organisés comme suit :

- Dans le premier chapitre, nous présenterons les concepts de base de l'intelligence artificielle, ses domaines d'utilisation, les méthodes d'apprentissage les plus couramment utilisées, et leur intérêt dans le domaine de la classification d'images.
- Le deuxième chapitre est consacré à la présentation de la méthodologie de travail.
- Dans le dernier chapitre, nous avons présenté la conception d'un groupe de différents modèles de classification, et ils ont été comparés, et nous avons choisi le meilleur modèle capable de détecter la pneumonie, qui est le modèle des réseaux de neurones convolutifs.

Enfin, nous terminons par une conclusion générale et des points de vue.

Chapitre I

Apprentissage automatique et
Méthodes de classification

I.1. Introduction

Dans ce chapitre, nous verrons d'abord la définition de l'apprentissage automatique, et nous discuterons des types d'apprentissage automatique, ainsi que de l'apprentissage profond, et de la relation de l'apprentissage automatique à l'apprentissage profond, et enfin nous discuterons de certaines méthodes de classification.

I.2. Apprentissage automatique

I.2.1. Définition et Histoire

L'apprentissage automatique ou Machine Learning (ML) est une sous-section du domaine de l'intelligence artificielle ou IA en informatique, qui vise à apprendre aux machines à effectuer une tâche sans être explicitement programmé, et cela en utilisant un des algorithmes qu'on appellera modèles et de données.

Le concept d'intelligence artificielle est apparu dans les années 50 dans une assemblée rassemblant toute une flopée de savants célèbres en informatique et en mathématique dont Alan Turing. Ce savant de génie a aussi prédit le développement du Machine Learning tel qu'on le connaît.

Le ML a refait surface entre les années 70 à 80, l'idée derrière le concept était de créer des algorithmes ayant la capacité d'accumuler de l'expérience, et de la connaissance à partir de données sans être explicitement programmé pour effectuer cette tâche, et c'est vers la fin des années 80 qu'on a le retour des réseaux de neurones (créé plutôt 1943 par Walter Pitts et Warren McCulloch inventeurs du premier Perceptron qu'on appellera par la suite Neurone artificiel. L'idée était de reproduire un neurone biologique d'un cerveau humain de façon artificielle en utilisant d'opérations mathématiques, malheureusement cette approche fut très limitée dans la résolution des problèmes. [1]

Vers la fin des années 80, on a vu le retour des réseaux de neurones, avec une réinvention de l'algorithme de Rétropropagation (Back propagation), mais sans succès car le domaine sera finalement laissé à l'abandon faute de capacité de calcul et de stockage suffisantes.

Il fallait attendre le milieu des années 2000 pour le grand retour des réseaux de neurones avec le Deep Learning (3).

I.2.2. Types de système d'apprentissage automatique

Il y a de nombreux sous domaines ou sous sections. On dénote 3 principaux, ses dernières dépendent du type de problèmes que l'on voudrait traiter, on retrouve :

- Apprentissage supervisé (Supervised Learning)
- Apprentissage non supervisé (Unsupervised Learning)
- Apprentissage par renforcement (Reinforcement Learning)
- Apprentissage semi supervisé (Semi-supervised Learning)

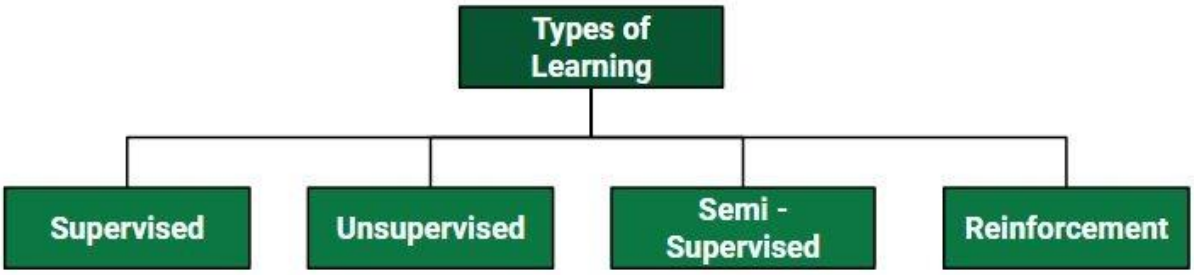


Figure (I.1): Type d'apprentissage en ML

I.2.1. Apprentissage supervisé

L'apprentissage supervisé consiste en la conception d'un modèle reliant des données d'apprentissage à un ensemble de valeurs de sortie. C'est-à-dire que les données d'entrainement qu'on fournit à l'algorithme comportent les solutions désirées, appelées étiquettes (en anglais, labels). Cette méthode permet donc à l'algorithme d'apprendre en comparant sa sortie réelle avec les sorties enseignées, afin de trouver les erreurs et modifier le modèle en conséquent. L'apprentissage supervisé confère au modèle la possibilité de prédire des valeurs d'étiquette sur des données non étiquetées supplémentaires.

Soit d'un ensemble de données, décrit par un ensemble de caractéristiques X, un algorithme d'apprentissage supervisé va trouver une fonction de mapping entre les variables prédictives en entrée X et la variable à prédire Y. La fonction de mapping décrivant la relation entre X et Y s'appelle un modèle de prédiction. Les caractéristiques X peuvent être des valeurs numériques, alphanumériques ou des images.

Un exemple d'utilisation de l'apprentissage supervisé est le filtre anti-spam, l'apprentissage s'effectue à l'aide de nombreux exemples d'e-mails qu'on a étiqueté spam ou normal. A partir de cela, le filtre doit alors être capable de classer de nouveaux e-mails.

Un autre exemple consiste à prédire le prix d'une voiture à partir des valeurs d'un certain nombre d'attributs ou variables qu'on appelle caractéristiques d'une observation ou features en anglais. Ces variables peuvent être le kilométrage, l'âge, la marque, etc. Ils sont également appelés variables explicatives ou prédictives. L'entrainement se fait alors à partir de ces variables et des étiquettes. La catégorie de la variable prédite Y fait décliner l'apprentissage supervisé en deux sous catégories : la classification et la régression, ces deux concepts seront abordés plus tard. [2] [3] [4]

- ✓ **Différence entre classification et régression**

Il y a une grande différence entre les problèmes de classification et de régression. La classification consiste à prédire une étiquette et la régression, à prédire une quantité. Avant de présenter ces deux concepts, il sera utile de comprendre la notion d'approximation de fonction.

- ✓ **Approximation de fonction**

Le terme « modélisation prédictive » (en anglais : Prédictive Modeling) désigne un ensemble de méthodes qui permettent d'analyser et d'interpréter des données définies afin d'effectuer une prédiction sur de futurs données. La modélisation prédictive peut être décrite comme un problème mathématique consistant à approximer une fonction de mappage f entre les variables prédictives en entrée X et la variable à prédire Y. C'est ce qu'on appelle un problème d'approximation de fonction. En règle générale, toutes les tâches d'approximation de fonctions peuvent être divisées en tâches de classification et en tâches de régression. [5]

- ✓ **Classification**

Les algorithmes de classification sont utilisés lorsque la variable à prédire Y est discrète. Exemple : la classification d'email à l'aide de filtre anti spam présenté précédemment. Une classification peut avoir des variables d'entrée à valeurs réelles ou discrètes. Il est courant que les modèles de classification prédisent des valeurs continues comme les probabilités d'appartenance à chaque classe de sortie. Une probabilité prédite peut être convertie en une valeur de classe en sélectionnant l'étiquette de la classe qui présente la probabilité la plus élevée. [5]

- ✓ **Régression**

Les algorithmes de régression sont quant à eux utilisés lorsque la variable à prédire Y est continue. Comme le cas de la prédiction du prix d'une voiture. Un problème de régression nécessite la prédiction d'une quantité. Il peut avoir des variables d'entrée à valeurs réelles ou discrètes.

Remarquons que certains algorithmes ont le mot « régression » dans leur nom, tels que régression linéaire et régression logistique, ce qui peut prêter à confusion, car la régression linéaire est un algorithme de régression alors que la régression logistique est un algorithme de classification. Certains algorithmes peuvent être utilisés à la fois pour la classification et

la régression avec de petites modifications, telles que les arbres de décision et les réseaux de neurones artificiels. [5]

I.2.2. Apprentissage non supervisé

L'apprentissage non supervisé consiste en la conception d'un modèle structurant l'information, c'est-à-dire les données d'apprentissage ne sont pas étiquetées. Cette méthode permet donc à l'algorithme de trouver tout seul des points communs parmi les données d'entrée, le système apprend alors sans professeur. Comme l'étiquetage de données requiert beaucoup de temps, les méthodes d'apprentissage utilisant l'apprentissage non supervisé sont particulièrement utiles. L'apprentissage non supervisé peut être utilisé pour la réduction de dimension ou l'extraction de variable. Cette tâche consiste à simplifier les données sans perdre trop d'informations, il pourra ensuite être fourni à un autre algorithme d'apprentissage automatique (tel qu'un algorithme d'apprentissage supervisé). Le kilométrage d'une voiture, par exemple, peut être fortement corrélé à son âge, de sorte que l'algorithme de réduction de dimension les combinera en une seule variable représentant la vétusté de la voiture.

A première vue, on pourrait penser que l'apprentissage non supervisé a peu d'utilité dans les applications de la vraie vie, mais les applications de cette technique sont nombreuses. Les sites comme Amazon, Netflix ou encore YouTube utilisent les algorithmes de partitionnement ou Clustering en anglais pour faire des recommandations de produits ou de films. Il peut être également utilisé pour explorer de larges ensembles de données et de découvrir d'intéressantes relations entre les variables : pour un supermarché par exemple, exécuter une règle d'association sur les journaux de vente permettrait peut-être de découvrir que les personnes achetant de la sauce barbecue et des chips ont aussi tendance à acheter des grillades. Cela permettrait de réorganiser les rayons afin de présenter ces articles à proximité les uns des autres. [2] [4] [6]

- ✓ **Clustering**

Le clustering permet de séparer les données entrées en un ensemble ou groupe de données qui ont des traits similaires et de les affecter à un cluster. Contrairement à la classification dans l'apprentissage supervisé ces différents clusters ou groupe ne sont pas connus à l'avance, c'est l'algorithme lui-même qui va séparer les données aux nombres de clusters qu'il faut.

Comme l'on peut le voir pouvez le voir dans l'exemple Figure II.3, les points de jeu de données donnés ont été divisés en groupes identifiables par les couleurs rouge, vert et bleu

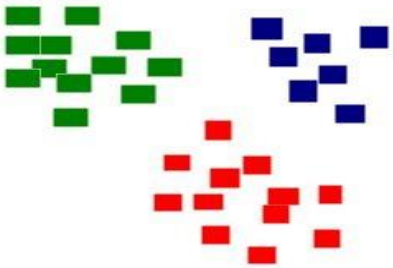


Figure (I.2): Clustering

I.2.3. Différence entre apprentissage supervisé et non supervisé

La majeure différence qu'on peut trouver entre ces 2 types d'apprentissages est la disposition des données d'entrée, mais il existe tout de même d'autres différences qu'on peut trouver dans le tableau ci-dessous.[1]

Ce tableau résume la différence entre apprentissage supervisé et non supervisé :

Tableau (I.1): Supervising vs unsupervised

	Apprentissage supervisé	Apprentissage non supervisé
Les données d'entrée	Utilise des données connues et étiquetées en tant qu'entrée	Utilise des données inconnues en tant qu'entrée Non étiquetées
Nombre Classes de	Connues	Inconnues
Précision du résultat	Résultats précis et fiables	Précision et fiabilité Modéré

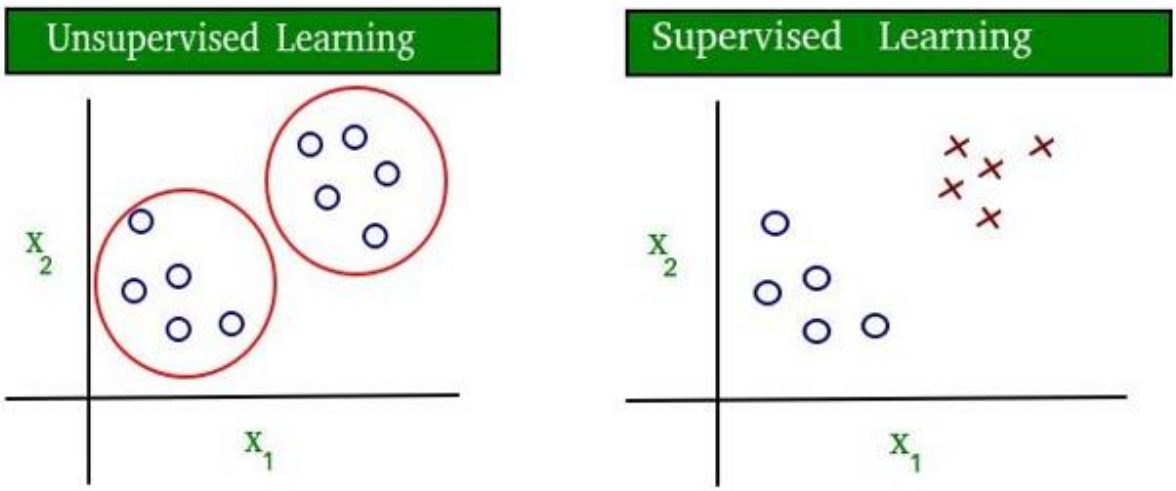


Figure (I.3): Apprentissage supervisé vs apprentissage non supervise

I.2.4. Apprentissage semi supervisé

L'apprentissage semi-supervisé vise à résoudre les problèmes avec relativement peu de données étiquetées et une grande quantité de données non étiquetées. L'apprentissage semi-supervisé réduit également le temps d'étiquetage de grande quantité de données par rapport à un apprentissage supervisé. Il a été démontré que l'utilisation de données non-étiquetées, en combinaison avec des données étiquetées, permet d'améliorer significativement la qualité de l'apprentissage. Ce type d'apprentissage a pour objectif de classer certaines des données non étiquetées à l'aide de l'ensemble d'informations étiquetées.

Un exemple illustrant l'utilisation d'un apprentissage semi-supervisé est le service d'hébergement d'image : Google Photos. Une fois avoir téléchargé des photos de famille sur ce service, le système arrive à reconnaitre qu'une personne A apparaît sur telle ou telle photos et qu'une personne B sur telle autres. Cela est dû à la partie non supervisée de l'algorithme. Une fois que vous aviez identifié ces personnes, juste une étiquette par personne, le système sera capable de nommer les personnes figurant sur chaque photo, ce qui est utile pour des recherches ultérieures. [2]

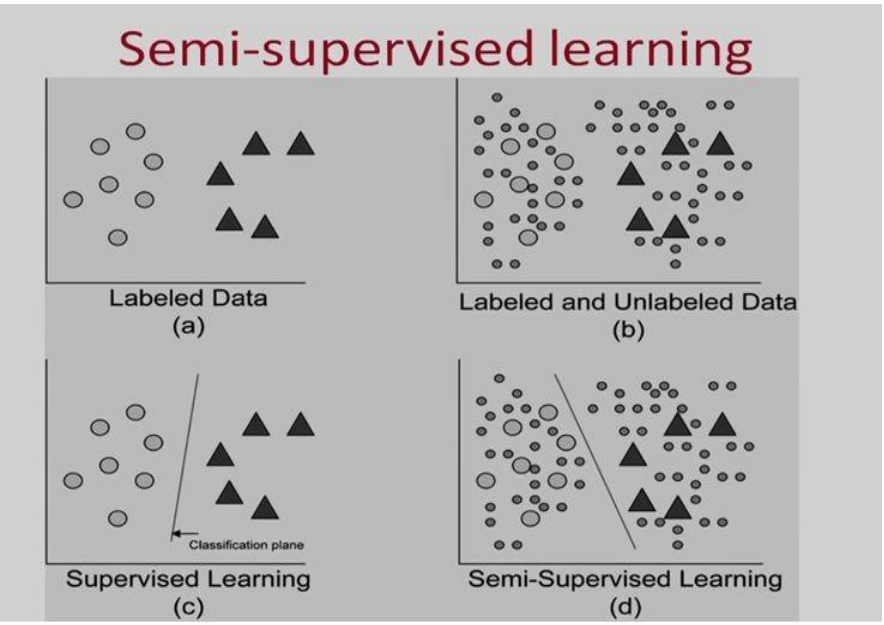


Figure (I.4): Apprentissage semi-supervisé

I.2.5. Apprentissage par renforcement

L'apprentissage par renforcement est très différent des types d'apprentissage vus jusqu'ici. Il consiste à apprendre, à partir d'expériences successives, ce qu'il convient de faire de façon à trouver la meilleure solution. Le système d'apprentissage, que l'on appelle ici « agent », interagit avec l'environnement, en sélectionnant et accomplissant des actions afin de trouver la solution optimale et obtenir en retour des récompenses. L'agent essaie plusieurs solutions, on parle d'" « exploration », observe la réaction de l'environnement et adapte son comportement, c'est-à-dire les variables pour trouver la meilleure stratégie. Pour ce type d'apprentissage, les données d'entrainement proviennent directement de l'environnement.

L'apprentissage par renforcement peut être utilisé pour apprendre à un robot à marcher, ou à un programme à jouer, comme Alpha Go de DeepMind qui a vaincu l'un des meilleurs joueurs de go auniveau mondial. En 2018, une startup issue des travaux de l'université de Cambridge a formé une intelligence artificielle à la conduite automobile. Au bout de seulement vingt minutes, l'IA est parvenue à savoir comment maintenir la voiture sur sa voie de circulation. Durant cette expérience un chauffeur été présent à bord de la voiture pour corriger les écarts de trajectoire causés par le logiciel, en arrêtant la voiture. Le programme a rapidement progressé en suivant une logique pénalité-récompense, en l'occurrence, respectivement, une intervention humaine et la distance maximale parcourue sans correction par le conducteur. [2] [7] [8]

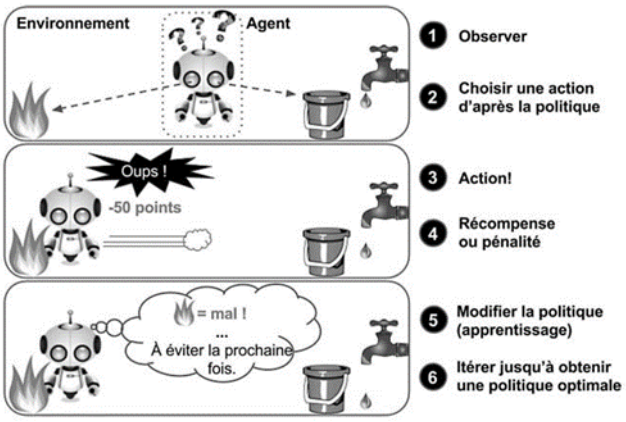


Figure (I.5): Apprentissage par renforcement

I.2.3. Deep Learning

Deep Learning ou apprentissage en profondeur ou DL est une branche du Machine Learning entièrement basée sur des réseaux de neurones artificiels. [9]
Le concept d'apprentissage en profondeur existe depuis plusieurs années, mais il a été laissé à l'abandon faute de moyens nécessaires.

Dans le milieu des années 2000 le Machine Learning fait rage dans les compétitions de reconnaissance visuelle, en 2012 deep mind une startup dans le domaine de l'IA arrive dans la compétition avec un algorithme de deep learning qui bat largement tous les autres compétiteurs, l'année suivante tous les compétiteurs se sont tournés vers le deep learning au vu des résultats obtenus.

L'avancée du DL est dû à l'augmentation en exponentiel qu'ont connu les machines en capacités de calculs, et de stockages, ainsi que la disponibilité de données de masses (big data), ses 3 ingrédients étaient nécessaires pour exploiter le potentiel du DL qui fût chose impossible dans les années 90.

Les pionniers qui ont soutenus le DL tel que Geoffrey Hinton, ou alors Yoshua Bengio qui a développé les réseaux GAN (générative adversal networks), Yann leCun qui est au coeur d'une avancée fulgurante dans le domaine de reconnaissance d'images avec les réseaux Convolution els CNN voire section (II.5), et son architecture LetNet.
Geoffrey Hinton a prouvé que l'apprentissage profond pouvait résoudre des problèmes insolubles par d'autres approches. [5]

I.2.4. Machine Learning vs Deep Learning

La majeure différence qu'on note entre ses 2 concepts provient de la manière dont les données sont présentées au système (modèle).

- Les algorithmes de ML nécessitent presque toujours des données structurées, alors que les réseaux d'apprentissage approfondis reposent sur des couches de réseaux de neurones artificiels (RNA).
- On voit aussi une différence au sein de l'architecture des modèles qui les composent, on note que les modèles type DL sont plus profond que les modèles type ML.
- Deep learning n'utilise que les réseaux de neurones, alors que pour le ML les réseaux de neurones sont qu'une approche de conception des modèles parmis tant d'autres.

En considérant le fait que le DL est la prochaine étape de l'évolution du ML inculquant aux machines la manière de prendre leurs décisions de façon précise sans l'intervention de l'expert humain.

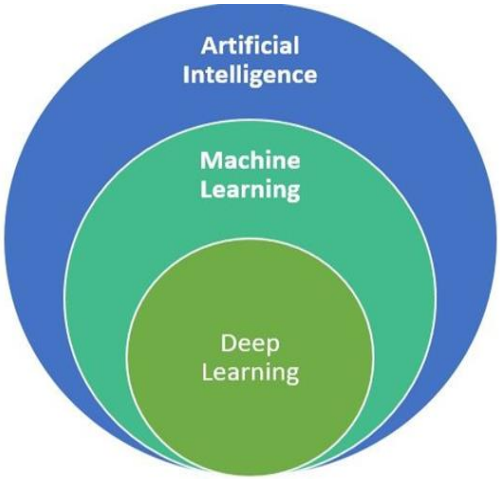


Figure (I.6): IA vs ML vs DL

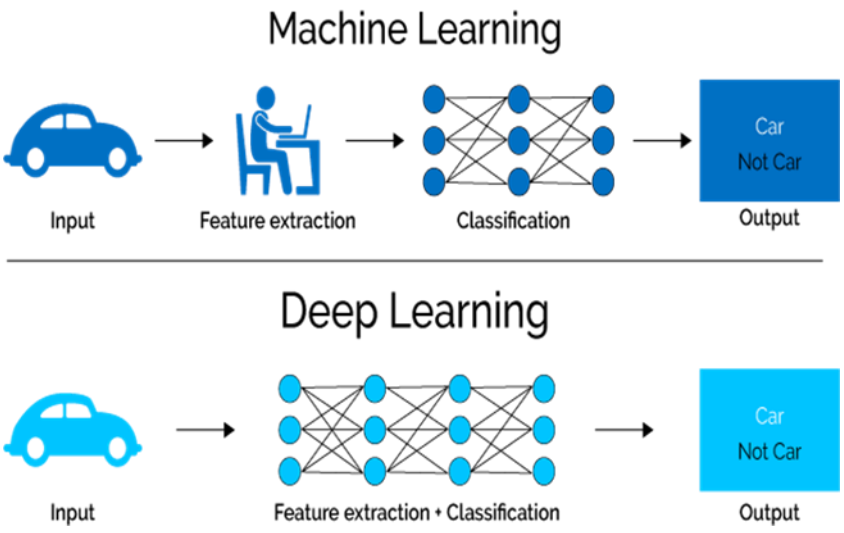


Figure (I.7) : ML vs DL

I.2.5. Pourquoi Deep Learning

Il s'agit d'une combinaison de facteurs dont :

- L'omniprésence des données : Nous somme dans l'ère de l'informatisation (Internet Of Things), et le propre du Deep-Learning est de tirer parti d'une grande quantité de données pour en estimer une représentation abstraite et en tirer parti.
- La puissance de calcul : La théorie des réseaux de neurones existe depuis quelques décennies, mais c'est grâce à la puissance de calcul accessible aujourd'hui qui se démocratise, notamment depuis que les GPUs sont devenus la plateforme de choix pour le Deep-Learning.
- Des besoins croissants dans le domaine de l'IA : vision par ordinateur, reconnaissance vocale, traitement du langage, …etc.
- Un effet de mode. On a tendance à vouloir appliquer le Deep-Learning partout alors que ça reste un moyen et non une fin. Certains problèmes sont tout à fait solubles par d'autres méthodes d'apprentissage statistique. Cela dit, si beaucoup de gens sont prêts à investir dans le Deep-Learning, il est normal qui deviennent si populaire.
- Capacité de Stockage : qui sont devenu beaucoup plus accessible à prix raisonnable.
- Apparition de plateformes et communautés forte encouragent l'évolution de ce domaine, ainsi que sa démocratisation.

I.3. Quelques méthodes de classification

I.3.1. K plus proches voisins (K-NN)

L'algorithme des k plus proches voisins ou K-NN, de son nom anglais K Hearst Neighbors, consisteà supposer qu'une observation est similaire à celle de ses voisins. Son fonctionnement peut être assimilé à l'analogie suivante : « dis-moi qui sont tes voisins, je te dirais qui tu es ». C'est un algorithme d'apprentissage supervisé de types classification ou régression. L'algorithme ne nécessite qu'une notion de distance et l'hypothèse que les points proches les uns des autres sont similaires.

I.3.1.1. Méthode de prédiction

La spécificité de l'algorithme K-NN est le fait qu'il ne nécessite pas d'apprentissage mais simplement le stockage des données. En effet, pour une observation qui ne fait pas

parti du jeu de données qu'on souhaite prédire, l'algorithme va chercher les K instances du jeu de données les plus proches de notre observation. Ensuite, en utilisant les variables de sortie de ses voisins, l'algorithmecalcul la valeur de la variable de sortie de l'observation qu'on souhaite prédire. Si K-NN est utilisé pour la régression, alors on calcule la moyenne ou la médiane des variables de sortie des K plus proches observations. Par ailleurs, si K-NN est utilisé pour une classification, alors c'est le mode des variables de sortie des K plus proches observations qui servira pour la prédiction.

Notons que la médiane désigne une valeur réelle m telle qu'il y ait autant d'observations x_i inférieures ou égales à m que supérieure ou égales à m. Le mode, quant à lui, est la valeur l'a plus représentée d'une variable quelconque dans une population donnée.

I.3.1.2. Notion de distance

L'algorithme K-NN a également besoin d'une fonction de calcul de distance entre deux observations. Plus deux points sont proches l'un de l'autre, plus ils sont similaires et vice versa. Il existe plusieurs fonctions de calcule de distance qui sont choisie en fonction des types de données qu'on manipule. Pour les données quantitatives (exemple : poids, salaires, taille, montant de panier électronique), on a tendance à utiliser la distance euclidienne. Alors que pour des données qui ne sont pas de mêmes types, quantitatives et qualitatives (exemple : âge, sexe, longueur), la distance de Manhattan est plus appropriée.

Dans un hyperespace, espace à n dimensions, la distance euclidienne entre les points X $(x_1, x_2, ..., x_n)$ et Y $(y_1, y_2, ..., y_n)$ est donnée par :

$$D_e(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

La distance de Manhattan entre les points X $(x_1, x_2, ..., x_n)$ et Y $(y_1, y_2, ..., y_n)$ est quant à elledonnée par :

$$D_m(X,Y) = \sum_{i=1}^{n}|x_i - y_i|$$

La figure (I.1) montre une classification avec l'algorithme des k plus proches voisins, ici en prend deux cas : k = 5 ou k = 18. Le nouvel individu sera affecté à la classe « cercle bleu » dans ces deux cas. [4] [10] [11] [12]

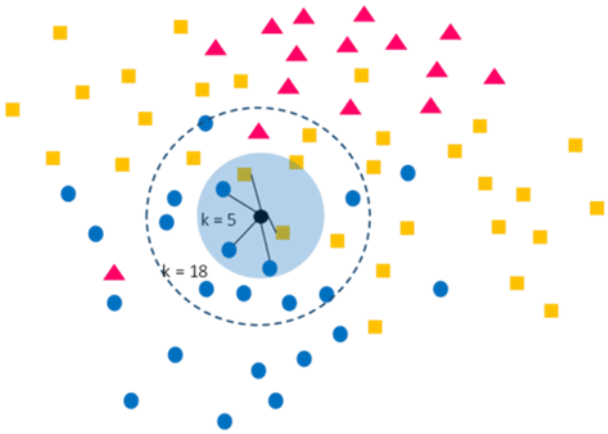


Figure (I.8): Sélection de k plus proches voisins d'une observation

I.3.1.3. Critiques de la méthode

- ✓ **Avantage de la méthodes k-NN**
  - Apprentissage rapide.
  - Méthode facile à comprendre.
  - Adapté aux domaines où chaque classe est représentée par plusieurs prototypes et où les frontières sont irrégulières (Exemple : reconnaissance de chiffre manuscrit ou d'images satellites).
- ✓ **Inconvénients de la méthodes k-NN**
  - Prédiction lente car il faut revoir tous les exemples à chaque fois.
  - Méthode gourmande en place mémoire.
  - Sensible aux attributs non pertinents et corrélés.

I.3.2. Machines à Vecteurs Support (SVM)

Les machines à support de vecteurs (SVM) sont à l'origine des nouvelles méthodes de catégorisations, bien que les premières publications sur le sujet datent des années 60.

Avant d'aborder le principe de fonctionnement général des *SVM* voici quelques notions de base :

- **Hyperplan :** est un séparateur d'objets des classes. De cette notion, nous pouvons dire qu'il est évident de trouver une mainte d'hyperplans mais la propriété délicate des SVM est d'avoir l'hyperplan dont la distance minimale aux exemples d'apprentissage est maximale, cet hyperplan est appelé L'hyperplan optimal, et la distance appelée marge.
- **Vecteurs Support :** ce sont les points qui déterminent l'hyperplan tels qu'ils soient les plus proches de ce dernier.

Voici un schéma représentatif de ces notions :

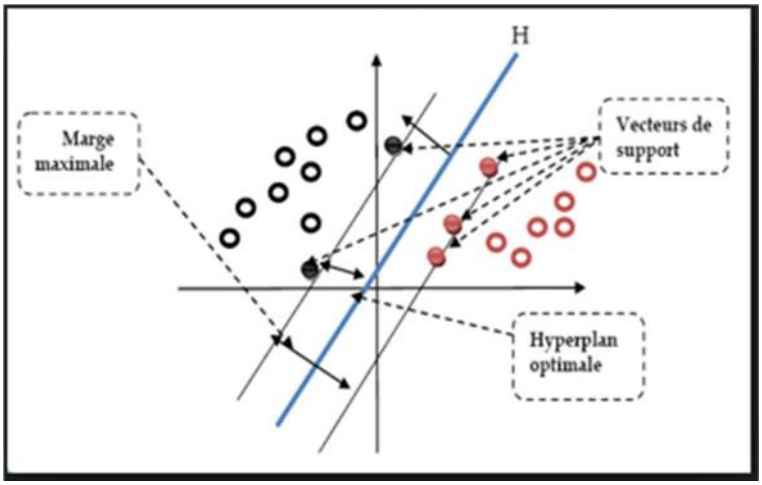


Figure (I.9): Les vecteurs à support

I.3.2.1. Principe de la technique SVM

Cette technique est une méthode de classification à deux classes qui tente de séparer les exemples positifs des exemples négatifs dans l'ensemble des exemples. La méthode cherche alors l'hyperplan qui sépare les exemples positifs des exemples négatifs, en garantissant quela marge entre le plus proche des positifs et des négatifs soit maximale. Cela garantit une généralisation du principe car de nouveaux exemples pourront ne pas être trop similaires à ceux utilisés pour trouver l'hyperplan mais être situés d'un côté ou l'autre de la frontière.

L'intérêt de cette méthode est la sélection de vecteurs supports qui représentent les vecteurs discriminant grâce auxquels est déterminé l'hyperplan. Les exemples utilisés lors de la recherche de l'hyperplan ne sont alors plus utiles et seuls ces

vecteurs supports sont utilisés pour classer un nouveau cas, ce qui peut être considéré comme un avantage pour cette méthode [13].

I.3.2.2. Exemple d'application [13]

1. Classification des données biologiques/ physiques.
2. Classification des documents numériques.
3. Classification expression faciale.

I.3.2.3. Critiques de la méthode [13]

- ✓ **Avantage de la méthodes SVM**

Les avantages théoriques et pratiques des SVM en ont fait un outil très prisé dansnombreux problèmes de classification. On cite parmi ces avantages :

- o Minimisation de l'erreur empirique et structurelle.
- o Algorithms optimisés.
- o Simple, peu de paramètres à régler.
- o Les SVM possèdent des fondements mathématiques solides.

- ✓ **Inconvénients de la méthodes SVM**

Les données dans des espaces de grande dimension sont souvent non linéairementséparables.

- o Classification binaire, d'où la nécessité d'utiliser l'approche un-contre-un.
- o Grande quantité exemples en entrées implique un calcul matriciel important.

I.3.3. Les arbres de décision

Un arbre de décision est, comme son nom l'indique, un outil d'aide à la décision qui permet de classifier une population d'individus selon les valeurs de leurs attributs. C'est une représentation graphique de la procédure de classification où :

- Une feuille indique une classe.
- Un nœud spécifie un test que doit subir un certain attribut.
- Chaque branche sortant de ce nœud correspond à une valeur possible

de l'attribut enquestion (Figure II.3).

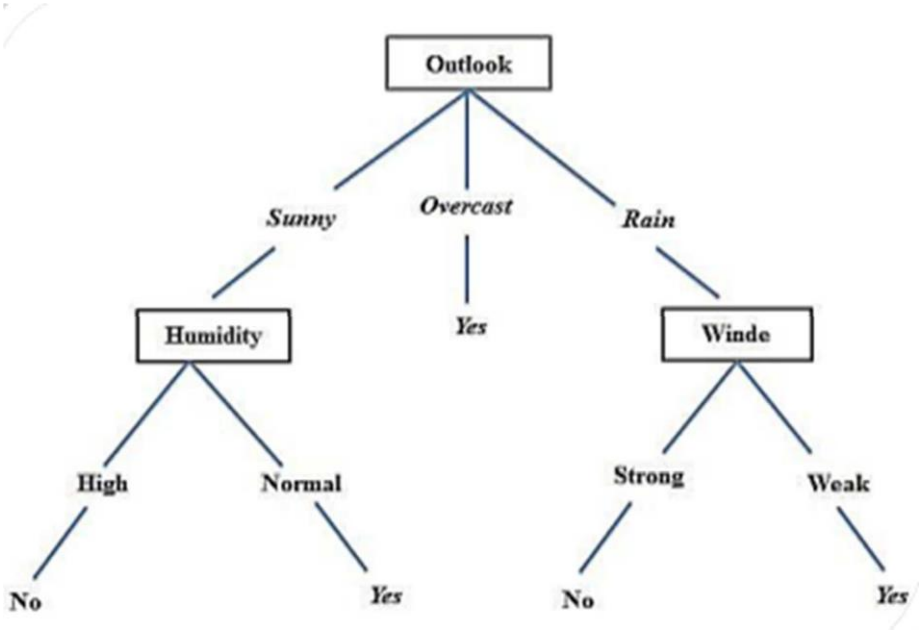


Figure (I.10): l'arbre de décision

Pour classifier un nouvel objet, on suit le chemin partant de la racine (nœud initial) à une feuille en effectuant les différents tests d'attributs à chaque nœud. L'arbre permet d'émettre des prédictions sur les données par réduction, niveau par niveau, du domaine de solutions.

La démarche générale de construction de l'arbre de décision consiste en deux étapes :

- Construction de l'arbre à partir des données (apprentissage).
- Elagage de l'arbre dans le but d'alléger l'arbre résultant souvent volumineux.

I.3.3.1. Construction de l'arbre

Il existe une grande variété d'algorithmes pour construire des arbres de décision ; quelques-uns des plus répandus portent les noms de ID3 (*Inductive Decision-tree*) introduitpar Quinlan et amélioré pour devenir C4.5 [14], CART (*Classification And Régression Trees)*, introduit par Breiman et al [15] et CHAID (*Chi-squared Automatic Interaction Détection*) [16].

Le principe général démarre d'un arbre vide et procède à la construction de l'arbre

de manière inductive (à partir des données) et récursive en commençant par l'ensemble des objets tout entier. Si tous les objets sont de même classes, une feuille est créée avec le nom de la classe. Sinon, l'ensemble d'objets est partagé en sous-ensembles selon la valeur d'uncertain attribut qui subissent le même traitement.

Afin d'avoir un arbre de décision concis et suffisant, il ne suffit pas de traiter les attributs séquentiellement. Au contraire, toute la richesse des arbres de décision consiste à choisir judicieusement les attributs d'éclatement pour aboutir, par le chemin le plus court et nécessaire, au plus grand nombre d'objets de la même classe.

Le choix des attributs peut se faire par plusieurs techniques, entre autres :

- Entropie (ID3, C4.5) [22].
- Indice de Gini (CART) [22].
- Table de Khi-2 (CHAID) [20].

I.3.3.2. Elagage de l'arbre

L'opération d'élagage de l'arbre se fait en deux phases : Le pré-élagage et le post-élagage.

Le pré-élagage consiste à fixer un critère d'arrêt qui permet de stopper la construction del'arbre lors de la phase de construction.

Le post-élagage est un traitement qui intervient après la construction entière de l'arbre. Il consiste à supprimer les sous-arbres qui n'améliorent pas l'erreur de classification.

I.3.3.3. Les domaines d'application

Cette méthode peut être utilisée dans plusieurs domains tels que:

Les études (pour comprendre les critères prépondérants dans l'achat d'un produit, l'impact des dépenses publicitaires), les ventes (pour analyser les performances par région, par enseigne, par vendeur), l'analyse de risques (pour détecter les facteurs prédictifs d'un comportement de non-paiement), Le domaine médical (pour étudier les rapports existants entrecertaines maladies et des particularités physiologiques ou sociologiques).[17]

I.3.3.4. Critiques de la méthode

✓ Avantage de la méthode AD

Les arbres de décision constituent un moyen très efficace de classification, et ce pour les avantages qu'elles présentent. Parmi ses avantages, on peut citer [15] [18] :

- Facilité à manipuler des données catégoriques.
- Traitement facile des variables d'amplitudes très différentes.
- La classe associée à chaque individu peut être justifiée.
- Les attributs apparaissant dans l'arbre sont des attributs pertinents.
- Pour le problème de classification considéré.

✓ Inconvénients de la méthode AD

Ces méthodes présentent tout de même des inconvénients dont les plus importants sont :

- La sensibilité au bruit et aux points aberrants.
- La sensibilité au nombre de classes (plus le nombre de classes est grand plus lesperformances diminuent).
- Le besoin de refaire l'apprentissage si les données évoluent dans le temps.

I.3.4. Réseaux de neurones artificiels (NN)

Les réseaux de neurones sont au cœur des progrès récents de l'apprentissage automatique. On ne peut donc parler de machine learning sans parlé des réseaux de neurones, un ensemble d'algorithmes dont le fonctionnement est inspiré des neurones biologiques.

I.3.4. 1. Neurone biologique

Le cerveau est un organe fascinant, depuis longtemps, on a compris que notre capacité à réfléchir sefaisait grâce au cerveau. Les cellules les plus importantes du cortex cérébral sont les neurones, dont le nombre atteint la centaine de milliards chez l'être humain. Elles sont constituées de trois parties essentielles : le corps cellulaire, les dendrites et l'axone.

Le corps cellulaire contient le noyau du neurone, son rôle est d'effectuer les

transformationsbiochimiques nécessaires à la synthèse des enzymes qui assurent la vie du neurone. Chaque neuronepossède des dendrites, qui sont les récepteurs principaux du neurone et servent à capter les signaux qui lui parviennent. L'axone quant à lui, sert de moyen de transport pour les signaux émis par le neurone. Les neurones sont connectés les uns aux autres suivant des répartitions spatiales complexes. Le lien physique entre deux neurones se fait grâce aux synapses. [19] [20]

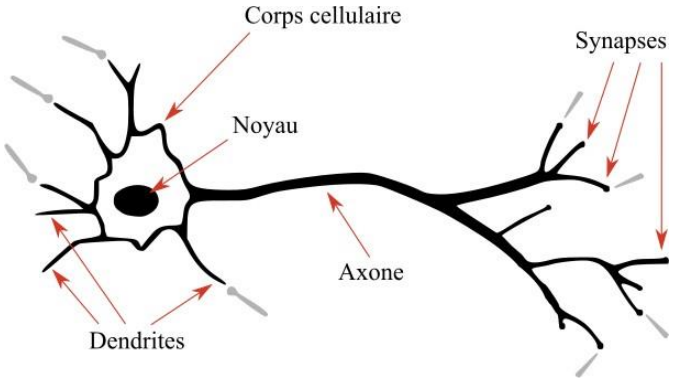


Figure (I.11): Neurone biologique

I.3.4. 2. Neurone formel

Le neurone formel ou artificiel vise à reprendre le fonctionnement du neurone biologique. Le neurone reçoit des entrées et fournit une sortie, grâce à différentes caractéristiques. Dans un neurone formel, les poids permettent de modifier l'importance de certaines entrées par rapport à d'autres. La fonction d'agrégation permet quant à lui d'obtenir une unique valeur à partir des entrées et des poids correspondants. Le seuil permet au neurone de savoir quand il doit agir ou non. Enfin, une fonction d'activation, qui associe à chaque valeur agrégée une unique valeur de sortie dépendant du seuil.

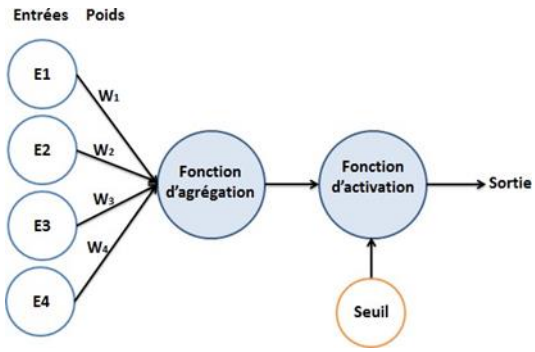


Figure (I.12): Neurone formel

I.3.4. 3. Fonction d'agrégation

Il existe plusieurs types de fonctions d'agrégation, mais les plus courantes sont : la somme pondéréeet le calcul de distance. Le but étant d'associer une seule valeur à l'ensemble des entrées et des poids.

La somme pondérée consiste à calculer la somme de toutes les entrées multipliées par leur poids c'est-à-dire :

$$\sum_{i=1}^{n} E_i * w_i$$

Le calcul des distances consiste plutôt à comparer les entrées aux poids (qui sont les entrées attendues par le neurone), et calculer la distance entre les deux. Dans ce cas la distance est donnée par :

$$\sum_{i=1}^{n} (E_i - w_i)^2$$

I.3.4.4. Fonctions d'activation

Il existe plusieurs fonctions d'activation, après avoir obtenu la valeur donnée par la fonction d'agrégation, le neurone compare cette dernière avec le seuil et décide de la sortie en utilisant la fonction d'activation.

Ci-après les fonctions d'activation les plus utilisées :

La fonction de Heaviside : c'est une fonction qui permet l'obtention de sorties binaires, 1 si la valeur agrégée calculée est plus grande que le seuil, 0 sinon.

$$H(x) = \begin{cases} 0 \; si \; x < 0 \\ 1 \; si \; x \geq 1 \end{cases}$$

Elle est principalement utilisée pour les problèmes de classifications en indiquant qu'un objet appartient ou non à une classe donnée. L'inconvénient, c'est qu'elle n'indique pas à quel point unevaleur est forte.

La fonction sigmoïde est définie par la relation mathématique suivante :

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

Elle est dérivable et donc contrairement à la fonction de Heaviside, permet de savoir vers quelle direction aller pour améliorer les résultats.

La fonction softmax : elle est utilisée pour de la classification multi-classe. Elle prend en entrée un vecteur $z = (z_1, ..., z_K)$ de K nombres réels et en sort un vecteur $\sigma(z)$ de K nombres réels strictement positifs de somme 1. Elle est définie par la relation mathématique suivante : pour tout j ∈ *1, ...,K+. [20] [21]

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$$

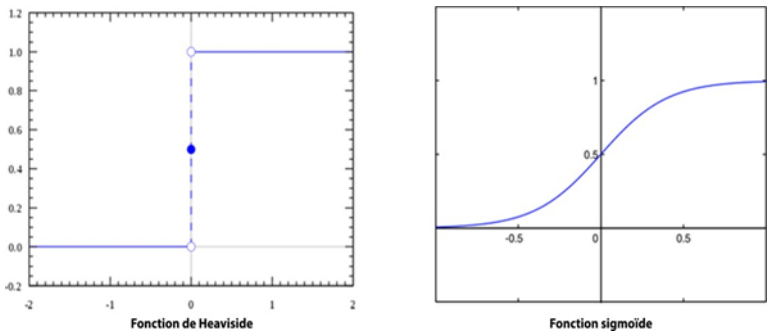


Figure (I.13): fonctions d'activation

I.3.4.5. Poids et seuils

Les neurones se différencient par leurs seuils ainsi que les poids les liant à leurs entrées. Les poids sont accordés à chacune des entrées, permettent de modifier l'importance de certaines par rapport aux autres. Les seuils ou biais quant à eux, permettent d'indiquer quand le neurone doit agir. Il est très difficile de déterminer la valeur des seuils et des poids pour des fonctions complexes. L'apprentissage consiste alors à trouver leurs valeurs optimales afin d'obtenir la sortie voulue. [20]

I.3.4.7. Perceptron

Le perceptron a été inventé par Frank Rosenblatt en 1955, il est constitué d'une première couche d'unité permettant de lire les données. On peut rajouter une unité de biais qui est toujours activée. Ces unités sont reliées à une seule unité de sortie. Mais pour une classification multi-classe, il peut avoir plusieurs sorties.

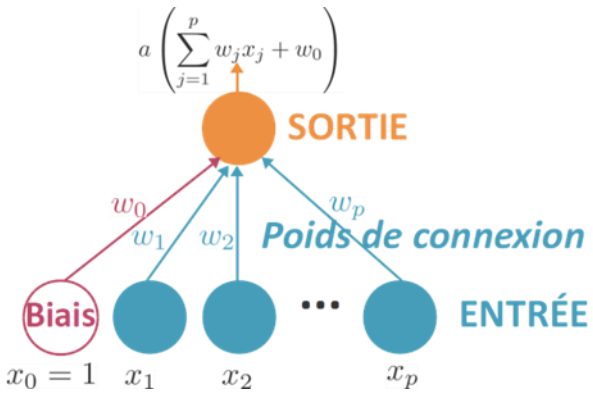


Figure (I.14): Perceptron à une unité de sortie

Ici x_0 représente le biais, $(x_1, x_2, ..., x_p)$ et $(w_1, w_2, ..., w_p)$ représentent respectivement les entrées

et les poids correspondants à chaque entrée $a(\sum_{j=1}^{p} w_j x_j + w_0)$ la fonction d'activation.

Pour les problèmes de classification multi-classe, le réseau va avoir autant de neurones de sortie quede classes. Ainsi, chacune de ces unités sera connectée à toutes les unités d'entrée. Dans ce cas, c'estla sortie ayant la plus forte valeur qui est prise en compte.

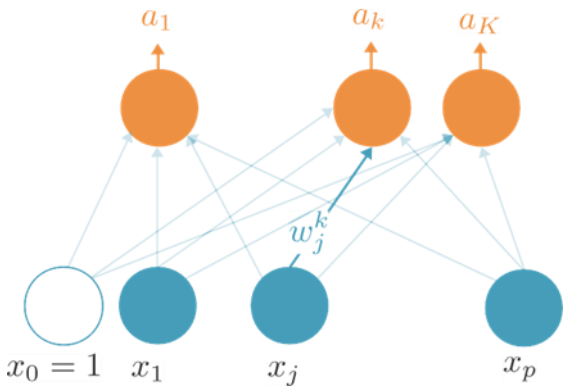


Figure (I.15): Perceptron à plusieurs unités de sortie

Dans cet exemple K représente le nombre de classe.

Le perceptron est simple à mettre en œuvre, son plus grand défaut est qu'il ne permet de

résoudre que les problèmes linéairement séparables. La fonction d'activation utilisée (sigmoïde ou Heaviside) présente un seuil, séparant deux zones de l'espace. [20] [21]

I.3.4.7. Critiques de la méthode

- ✓ **Avantage de la méthode NN**

- Classification très bien précise (si bien paramétré).
- Resistance aux pannes (si un neurone ne fonctionne plus, le réseau ne se perturbe pas).

- ✓ **Inconvénients de la méthodes NN**

- La détermination de l'architecture du réseau est complexe.
- Paramètres difficiles à interpréter (boite noire).
- Difficulté de paramétrage surtout pour le nombre de neurone dans la couche cachée.

I.3.5. Généralités sur les (CNN)

Les réseaux de neurones convolution els sont spécialement conçus pour traiter des images en entrée. Ils sont les plus performants en terme classifications d'images. Ces réseaux comportent deux parties bien distinctes.

La première partie est la partie convolutive, elle fait la particularité de ce type de réseau. Cette partie fonctionne comme un extracteur de caractéristiques. Une image passe à travers une succession de filtres de convolution. Habituellement, une couche de convolution est suivie d'une fonction d'activation. Certains filtres intermédiaires sont là pour réduire la résolution de l'image par une opération de maximum local. Ce procédé peut être réitéré plusieurs fois. Finalement, les cartes de convolutions sont concaténées dans un vecteur, appelé code CNN. Ce vecteur représente la sortiedu premier bloc, et l'entrée du second.

La deuxième partie n'est pas caractéristique d'un CNN, elle est constituée d'une couche entièrement connectée. Cette partie a pour rôle de combiner les caractéristiques du code CNN pour classer l'image. La sortie est une dernière couche qui contient autant d'éléments qu'il y a de classes. Les valeurs numériques obtenues sont généralement normalisées entre 0 et 1, de somme 1, pour produire une distribution de probabilité sur les catégories. Comme pour les réseaux de neurones ordinaires, les paramètres des couches sont déterminés par rétropropagation du gradient. Mais dans le cas des CNN, ces paramètres désignent en particulier les features des images. [22]

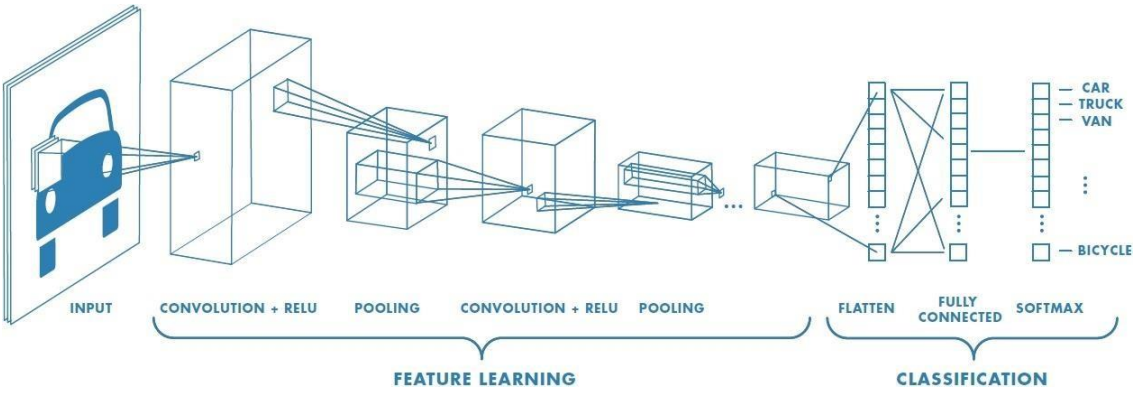


Figure (I.16): Architecture standard d'un réseau de neurones convolutionels

I.3.5. 1. Différentes couches d'un CNN

Les différentes couches constituant un CNN présentées dans le chapitre 1 serons détaillées dans ceparagraphe.

- ✓ **Couche de convolution**

Principe de fonctionnement

La couche de convolution est la composante clé des CNN, elle est présente au moins dans la première couche. Le but de cette couche est de trouver un ensemble de features dans les images reçues en entrée. C'est là qu'intervient l'opération de convolution : le principe est de faire "glisser"une fenêtre représentant la feature sur l'image. Le produit de convolution entre la feature et chaque portion de l'image balayée est alors calculé. Dans ce cas, la feature est considérée comme un filtre. Les filtres correspondent exactement aux features que l'on souhaite retrouver dans les images. En sortie, on obtient une carte d'activation ou feature map, indiquant l'emplacement des features dans

L'image. Dans un CNN, les features ne sont pas prédéfinies selon un formalisme particulier comme pour les méthodes traditionnelles (Viola Jones ou HOG par exemple), elles sont apprises lors de la phase d'entrainement du réseau. Les CNN sont donc capables de déterminer tout seuls les éléments discriminants d'une image.

Calcul des dimensions de la sortie

Le calcul de la dimension de la sortie associée à une couche de convolution nécessite la connaissance de trois hyperparamètres.

- Le pas : dans le contexte d'une opération de convolution, le pas ou « stride » désigne le pas de déplacement du noyau après chaque opération à travers l'image.
- La profondeur : c'est le nombre de noyaux de convolution ou filtre.
- La marge à zéro ou zéro padding : il est commun d'utiliser une épaisseur de marge de zéros autour de l'entrée, le zéro-padding est une technique consistant à ajouter P zéros à chaque côté des frontières de l'entrée.

Soit I le côté associé à l'entrant, F la dimension du noyau, P l'épaisseur de la marge à zéro et s l'amplitude du pas utilisée ; la dimension O de la sortie est alors donnée par :

$$O = \frac{I - F + 2P}{s} + 1$$

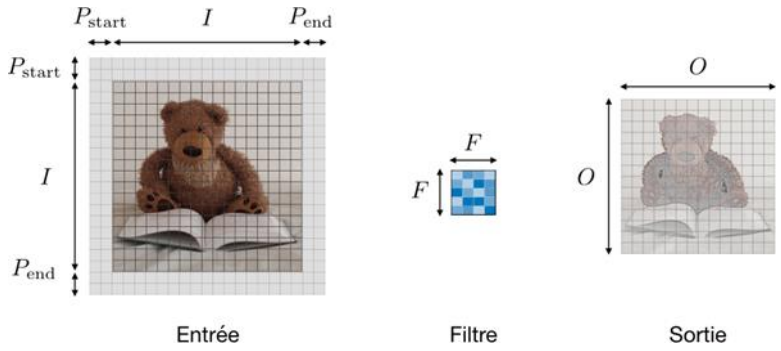


Figure (I.17): Illustration de la taille d'une sortie de couch de convolution

Remarques : si P_{start} est différent de P_{end} alors

$$O = \frac{I - F + P_{start} + P_{end}}{s} + 1$$

[23] [24] [25]

✓ **Couche de pooling**

Principe de fonctionnement

La couche de pooling est souvent placée entre deux couches de convolution. Elle applique une opération de pooling qui est une opération de sous-échantillonnage. Elle

consiste à réduire la taille des images, tout en préservant leurs caractéristiques importantes. Pour se faire, l'image est découpée en cellules régulières de tailles 2 × 2 pixels qui ne se chevauchent pas, ou de taille 3 × 3 pixels, distantes les unes des autres d'un pas de 2 pixels (qui se chevauchent donc). Ensuite, on applique l'opération de pooling. Les sorties obtenues sont de même nombre que les entrées mais de taille plus petite. La couche de pooling permet de réduire le nombre de paramètres et de calculs dans le réseau. On améliore ainsi l'efficacité du réseau et on évite le sur apprentissage. [23]

L'opération pooling

L'idée derrière l'opération pooling est de retirer une certaine information dans un même voisinage pour l'entrant. L'opération est effectuée sur chacune des épaisseurs liées à la profondeur de l'entrant.Généralement, on utilise le max ou l average pooling, où les valeurs maximales et moyennes sont prises respectivement.

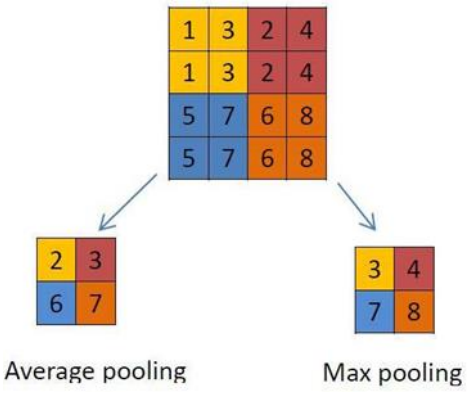


Figure (I.18): Exemple de ça Pour l average pooling

Dans la figure 3.03, on est en présence de cellules de taille 2x2.
Pour l average pooling : chaque case correspond à la moyenne du carré d'entrée de la même couleur, exemple pour le cas de la case jaune, on a :

$$\frac{1+3+3+1}{4}=2$$

Pour le max pooling : chaque case correspond à la valeur maximum du carré d'entrée

de la même couleur, exemple pour la case bleu :

$$\max(5,7,5,7) = 7$$

[24]

- ✓ **Couche Rectified Linear Units**

La fonction ReLU ou Rectified Linear Units est appliquée après la fonction de convolution et laisse les dimensions inchangées. Elle désigne une fonction réelle non linéaire définie par :

$$ReLU(x) = \max(0, x)$$

La couche de correction ReLU remplace donc toutes les valeurs négatives reçues en entrées par des zéros. Elle joue le rôle de fonction d'activation et a pour but d'introduire des complexités non- linéaires au réseau. [24] [25]

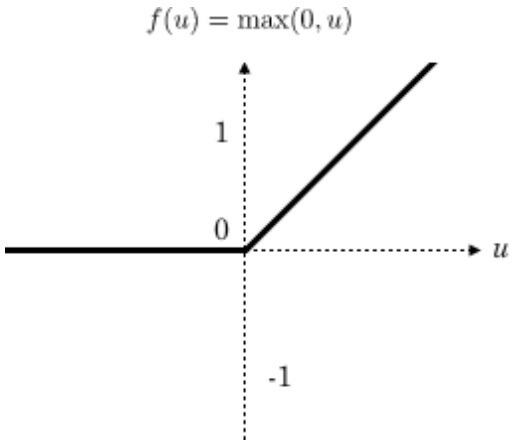


Figure (I.19): fonction Rectified Linear Units

- ✓ **Couche entièrement connectée**

La couche entièrement connectée ou fully-connected layer en anglais constitue toujours la dernière couche d'un réseau de neurones. Cette couche n'est donc pas caractéristique d'un CNN. Elle s'applique sur une entrée préalablement aplatie où chaque entrée est connectée à tous les neurones etproduit un nouveau vecteur en sortie. La dernière couche classifie l'image d'entrée en plusieurs classes et renvoie un vecteur dont la taille est égale au nombre de classe du problème. Chaque élément du vecteur indique la probabilité pour l'image en entrée d'appartenir à une certaine classe.

Pour calculer les probabilités, la couche entièrement connectée multiplie donc chaque élément en entrée par un poids, fait la somme, puis applique une fonction d'activation (ici on utilise la fonction softmax si le nombre de classe $N > 2$ ou sigmoïde logistique si $N = 2$). L'apprentissage des valeurs des poids se fait lors de phase d'entraînement, par rétropropagation du gradient.

La couche entièrement connectée cherche les liens entre la position des features dans l'image et une classe. En effet, le tableau en entrée correspond à une carte d'activation pour une feature donnée : les valeurs élevées indiquent la localisation de cette feature dans l'image. Si la localisation d'une feature à un certain endroit de l'image est caractéristique d'une certaine classe, alors on accorde un poids important à la valeur correspondante dans le tableau. [23] [25]

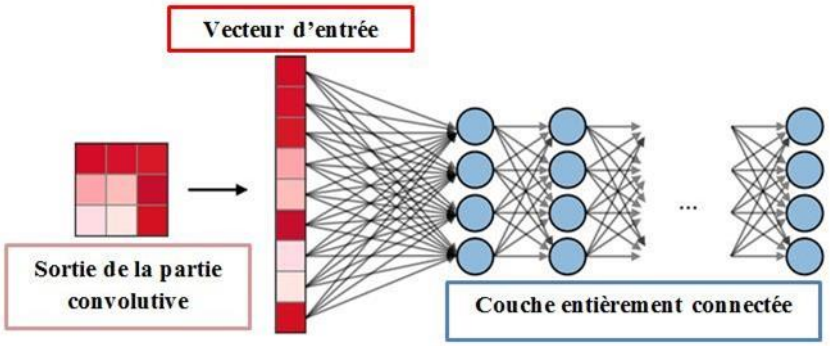


Figure (I.20): Mis en évidence d'une couche entièrement connectée

I.3.5. Réseaux bayésiens

Les Réseaux Bayésiens sont des modèles graphiques qui représentent les relations probabilisées entre un ensemble des variables. Ils deviennent un outil populaire pour représenter et manipuler des connaissances dans un système expert. Ils sont souvent utilisés à cause de leurs avantages.

Les réseaux bayésiens permettent aussi l'utilisation des connaissances puisqu'ils sont polyvalents donc on peut se servir du même modèle pour évaluer, prévoir, prédire, diagnostiquer, ou optimiser des décisions, ce qui contribue à rentabiliser l'effort de construction du réseau bayésien. [26]

La théorie des réseaux bayésiens résulte d'une fusion entre la théorie des probabilités et la théorie des graphes. On définit classiquement un réseau bayésien comme un graphe acyclique dirigé. Il est formé d'un ensemble de variables et d'un ensemble d'arcs entre les variables. Chaque variable correspond à un nœud du réseau. [27].

Les RB sont des modèles qui permettent de représenter des situations de raisonnement probabilistes basé sur le théorème de Bayés.

I.3.5.1. Théorème de bayes

Thomas bayes (1702-1761) est né à Londres en Angleterre a développé un théorème qui repose sur la propagation de l'information au sein du réseau, c'est-à-dire les calculs de probabilités a posteriori de certaines variables à partir d'un certain nombre d'observations sur d'autres variables.

$$P(B|A) = \frac{P(B \cap A)}{P(A)}$$

Par sa symétrie, permet de faire un raisonnement dans les deux sens, le calcul de la probabilité de B sachant A mais aussi de A sachant B. Dans un sens nous cherchons à expliquer une cause dans l'autre nous quantifions une conséquence [27].

Le théorème de Bayes est basé sur les probabilités conditionnelles qui dit qu'un événement B se produise sachant que l'événement A s'est déjà produit. On la note P(B|A) ou PA(B) et on la lit « probabilité que B se réalise sachant que A s'est produit ».

La probabilité conditionnelle revient donc à retrouver la probabilité d'un second événementalors que l'on sait qu'un premier événement s'est déjà produit auparavant.

La formule pour calculer une probabilité conditionnelle est :

$$P(B|A) = \frac{P(B \cap A)}{P(A)}$$

Où P(B∩A) représente la probabilité de l'intersection des deux événements. De plus, il est nécessaire que P(A) soit entre 0 et 1.

I.3.5.2. Graphe de causalité

La représentation graphique la plus intuitive de l'influence d'un événement, d'un fait, ou d'une variable sur une autre, est probablement de relier la cause à l'effet par une flèche orientée.

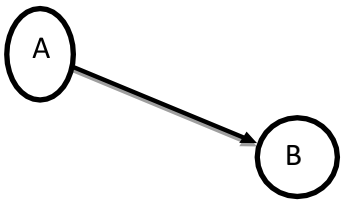


Figure (I.21) : Graphe de causalité

Supposons que A et B soient des événements, qui peuvent être observés ou non, vrais ou faux. Du point de vue du sens commun, le graphe ci-dessus peut se lire comme ceci. « La connaissance que j'ai de A détermine la connaissance que j'ai de B».

La relation causale soit l'implication logique A→B. Cette relation signifie que si A est vrai,B l'est également. Si A est faux, B peut être vrai ou faux.

La tableau1.1 représente les configurations possibles de A et B dans le cas où la relation causale A→B est vraie. Du point de vue de la logique, il s'agit simplement de la contraposée de A→B. Du point de vue de la causalité, cela montre qu'une relation

Causale, donc orientée, est réversible de l'effet vers la cause, même si elle ne l'est que partiellement.

Tableau (I.2): L'implication logique

A	B
Vrai	Vrai
Faux	Vrai
Faux	Faux

S'il existe une relation causale de A vers B, toute information sur A peut modifier la connaissance que j'ai de B, et réciproquement, toute information sur B peut modifier la connaissance que j'ai de A.

Avec la représentation graphique de la causalité on peut connaître la direction de circulation de connaissances dans le graphe mais on ne peut pas connaître la quantité de

cette circulation de connaissances. Alors, il faut une représentation probabiliste associé avec le graphe.

Avec une relation causale : A→B on peut représenter la quantité de cette relation par la probabilité conditionnelle : p(B|A). [28]

I.3.5.3. Structure d'un réseau bayésien

La structure d'un réseau bayésien est un graphe dans lequel les nœuds représentent des variables aléatoires, et les arcs relient ces nœuds qui sont rattachées à des probabilités conditionnelles.

Le graphe est acyclique donc il ne contient pas de boucle. Les arcs représentent des relations entre variables qui sont soit déterministes, soit probabilistes. Ainsi, l'observation d'une ou plusieurs causes n'entraîne pas systématiquement l'effet ou les effets qui endépendent, mais modifie seulement la probabilité de les observer.

La structure est définie par des experts et les tables de probabilités calculées à partir de données expérimentales. Il est possible d'utiliser des algorithmes, le recuit simulé ouencore certains algorithmes génétiques pour construire le réseau. [29]

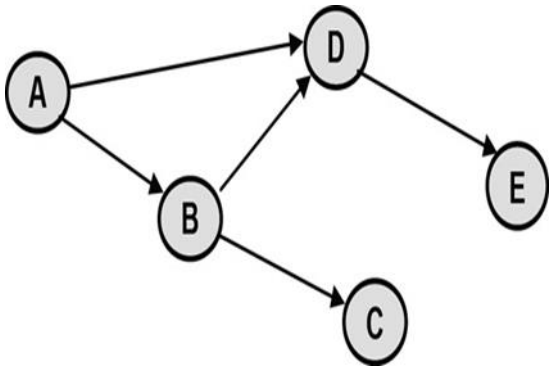


Figure (I.22): Structure d'un réseau bayésien de cinq variables

I.3.6. Forêts aléatoires [30]

La méthode des forêts aléatoires est une technique d'apprentissage statistique introduite par Breiman (2001) (voir aussi Biau et al. (2008), Biau (2012)) basée sur l'agrégation d'arbres de classification CART (Classification And Régression Tree). Une forêt étant construite à partir des arbres CART, nous rappelons d'abord comment on construit un arbre CART.

CART est une méthode non paramétrique d'apprentissage qui construit un arbre de décision aussi bien en régression qu'en classification (Breiman et al. (1984)). Dans cette méthode, l'arbre est construit de la façon suivante : partant de la racine (les données complètes), on choisit la variable qui produit la meilleure coupure en deux des données. La coupure des données portant sur une variable zj

Se fait en partitionnant les observations en deux groupes (zj a et zj > a) qui prédisent le mieux la variable réponse Y. Les nœuds de l'arbre sont associés aux

Éléments de la partition. La même procédure est appliquée à chaque nœud "fils". On arrête la procédure lorsqu'il n'y a plus assez d'observations dans un nœud pour être partitionné en deux. L'arbre final est ensuite élagué pour éviter le surappren- tissage. Les nœuds terminaux, encore appelés feuilles, sont associés aux partitions les plus fines de l'arbre. Ils sont utilisés comme prédictions. Dans le cas des don- nées Actu-Palu par exemple, pour prédire le statut d'un nouveau ménage, on lui associera la réponse (foyer à risque vs foyer non à risque) majoritairement présentée dans le nœud terminal.

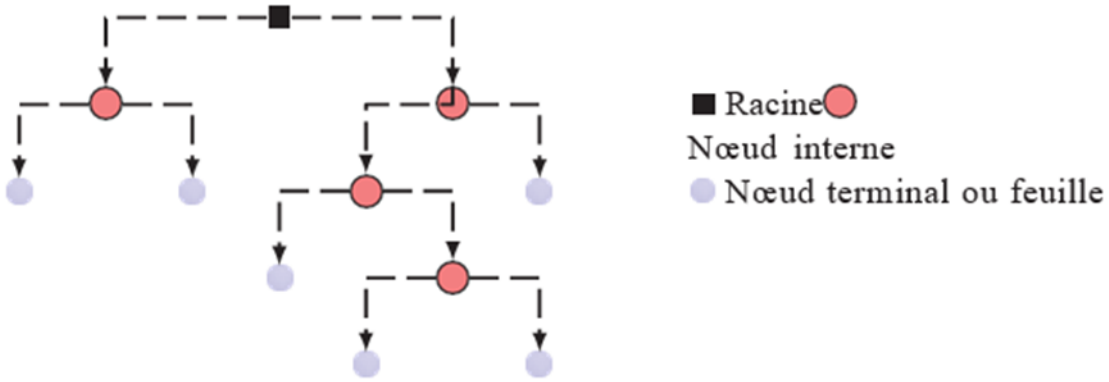


Figure (I.23): forme forêt aléatoire

Une forêt aléatoire est construite en agrégeant les informations fournies par m arbres de classification. Notons n = (z_1, Y_1), . . ., (Z_n, Y_n). Chaque arbre notér$_k$ (., θ_k, $_n$), k = 1, . . ., *m est* construit en introduisant de l'aléatoire représenté par θ_k, d'où le nom forêt aléatoire. L'aléatoire est dû au fait que chaque arbre est construit sur un échantillon Bootstrap ^{l}, l = 1, . . ., *m*, et à chaque nœud on tire *mtry* < *d* variables de façon aléatoire et c'est dans cet ensemble de variables quel'on cherche celle qui réalise la coupure optimale. Le choix d'un petit nombre de variables à chaque nœud permet de réduire la complexité de l'algorithme

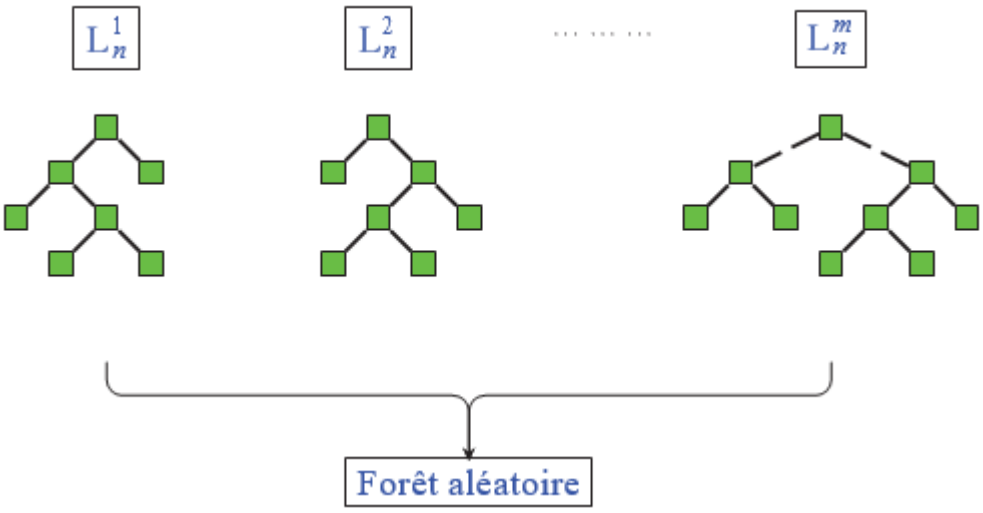


Figure (I.24): forêt aléatoire complexe

Pour une nouvelle variable explicative *z*, la prédiction par une forêt aléatoire sefait en prenant la majorité des votes de chacun des arbres

$$RF(z) = \begin{cases} 1 & \text{si } \frac{1}{m}\sum_{k=1}^{m} r_k(z, \theta_k, \mathcal{L}_n) \geq \frac{1}{2} \\ 0 & \text{si non} \end{cases}$$

Les arbres de la forêt ne sont pas élagués, ils ont donc une grande variance etun petit biais. L'agrégation des arbres permet d'avoir une forêt

aléatoire avec une petite variance (Breiman (2001)). Les forêts aléatoires sont utilisables en grande dimension et permettent de prendre en compte la corrélation et les interactions entre les variables explicatives (voir Chen et Ishwaran (2012)). La construction d'une forêt aléatoire fait intervenir deux paramètres impor- tants :
-Le nombre *m* d'arbres de la forêt. Il doit être choisi de façon à assurer la stabilité de la forêt.

-Le nombre *mtry* des variables choisies à chaque nœud de l'arbre. Il est com-pris entre 1 et *d*, c'est le paramètre le plus important. Une petite valeurde *mtry* réduit la probabilité de choisir les variables importantes à chaque nœud, ce qui peut dégrader les performances de la forêt aléatoire. Une grande valeur de mtry *augmente* la complexité de l'algorithme. Breiman a suggéré de prendre $mtry = \sqrt{d}$ pour des problèmes de classification. Ce choix a ensuite été confirmé par plusieurs travaux, voir par exemple Liaw etWiener (2002), Díaz-Uriarte et De Andres (2006).

I.4. Conclusion

Après avoir abordé les concepts théoriques concernant l'apprentissage automatique et l'apprentissage en profondeur, ainsi que la relation entre l'apprentissage automatique et l'apprentissage en profondeur, et nous avons abordé un groupe de méthodes de classification qui aident à comprendre et à compléter la partie appliquée, et dans le suivant partie, nous allons essayer d'éclairer la partie appliquée, qui se compose de deux chapitres. Dans le deuxième chapitre, nous verrons d'abord la structure L'année de notre projet, dans laquelle nous donnerons les différentes phases. Le troisième chapitre construira modèles de classification et analyser et interpréter les résultats obtenus.

Chapitre II

Méthodologie du Travail

II.1. Introduction

Dans ce chapitre nous verrons d'abord la structure générale de notre projet dans laquelle nous donnerons les différentes phases. Nous détaillerons chaque étape en indiquant son objectif ainsi que toutesp les justifications.
Nous discuterons brièvement du logiciel et de la plate-forme d'apprentissage automatique les plus importants utilisés dans la recherche.

II.2. Maladie de pneumonie

La pneumonie est une affection inflammatoire des poumons affectant principalement les petits sacs aériens appelés alvéoles. Les symptômes comprennent généralement une combinaison de toux productive ou sèche, de douleurs thoraciques, de fièvre et de difficultés respiratoires. La gravité de la condition est variable. La pneumonie est généralement causée par une infection par des virus ou des bactéries et moins fréquemment par d'autres micro-organismes, certains médicaments ou des conditions telles que les maladies auto-immunes. Les facteurs de risque comprennent la fibrose kystique, la maladie pulmonaire obstructive chronique (MPOC), l'asthme, le diabète, l'insuffisance cardiaque, des antécédents de tabagisme, une mauvaise capacité à tousser, comme après un accident vasculaire cérébral et un système immunitaire affaibli. Le diagnostic repose souvent sur les symptômes et l'examen physique. Une radiographie pulmonaire, des tests sanguins et une culture des expectorations peuvent aider à confirmer le diagnostic. La maladie peut être classée selon le lieu où elle a été contractée, comme une pneumonie communautaire ou nosocomiale ou nosocomiale.

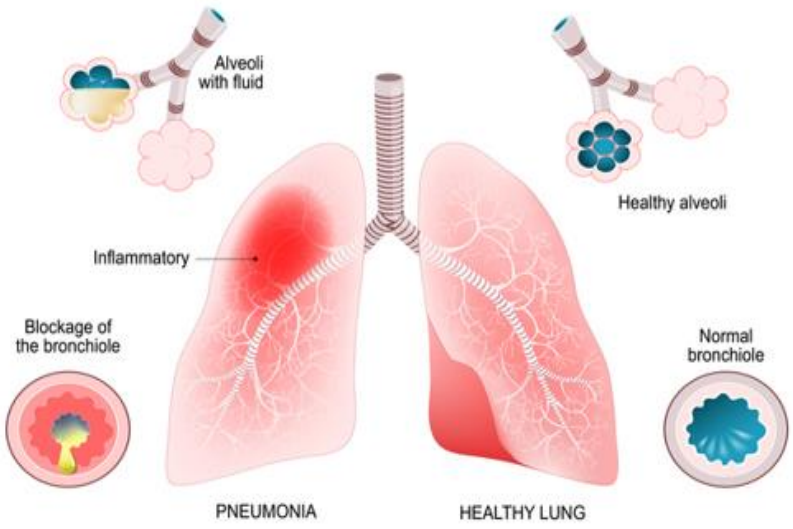


Figure (II.1) : Les poumons sont infectés par une pneumonie

II.3. Les étapes de base pour créer un modèle de la classification

Il y a 9 étapes de base pour créer un modèle de classification, qui sont les suivantes :

- **Etape 1:** La collecte des données
- **Etape 2:** Exploration des données
- **Etape 3:** Preparation des données
- **Etape 4:** Extraction des paramètres importants pour la prédiction
- **Etape 5:** Choisir et construire le modèle
- **Etape 6:** Validation du modèle
- **Etape 7:** Test du modèle
- **Etape 8:** Evaluation du modéle
- **Etape 9:** La décision (Acceptation ou Rejeter le modèle)

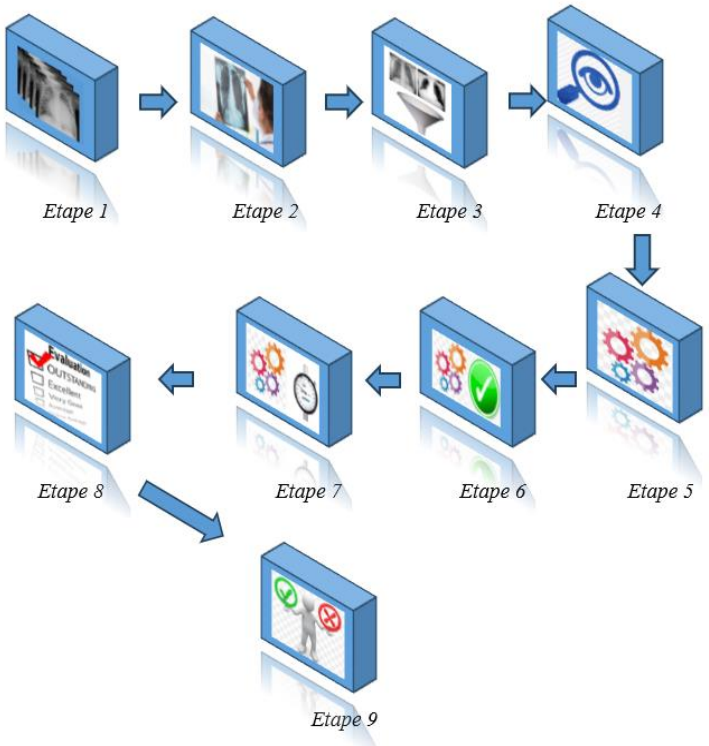


Figure (II.2) : Les étapes de base pour créer un modèle de la classification

II.3.1. La collecte des données

II.3.1.1. Description de l'ensemble de données sur la pneumonie

Le jeu de données est organisé en 3 dossiers (train, test, val) et contient des sous-dossiers pour chaque catégorie d'image (Pneumonia/Normal). Il y a 5 856 images radiographiques (JPEG) et 2 catégories (Pneumonie/Normal). Des images radiographiques thoraciques (antéro-postérieures) ont été sélectionnées à partir de cohortes rétrospectives de patients pédiatriques âgés de un à cinq ans du Guangzhou Women and Children's Medical Center, Guangzhou. Toutes les radiographies pulmonaires ont été réalisées dans le cadre des soins cliniques de routine des patients.

✓ **Présentation statistique**

Tableau (II.1) : Description de l'ensemble de données sur la pneumonie

Etat	Fréquence	Pourcentage %
Normale	1583	27.03
Pneumonia	4273	72.97
Total	5856	100

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

✓ **Présentation graphique**

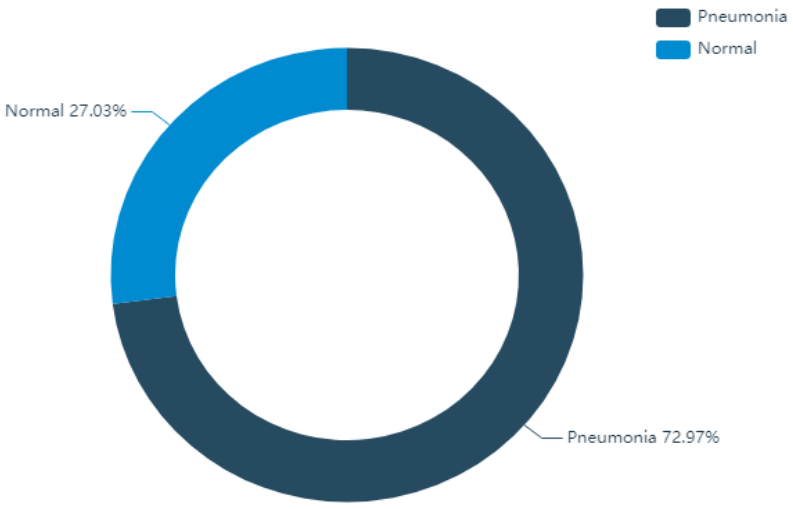
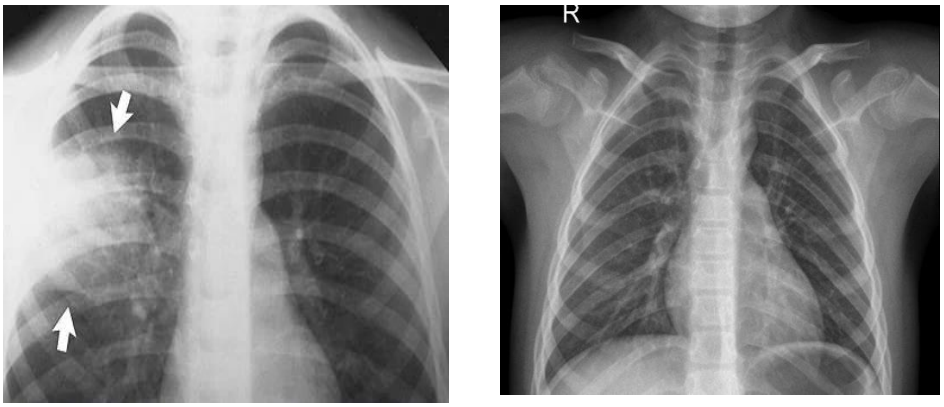


Figure (II.3) : Description de l'ensemble de données sur la pneumonie

II.3.2. Exploration des données

Les données sont explorées par un médecin spécialisé dans les maladies respiratoires afin que nous puissions connaître l'emplacement de la blessure dans l'image et la manière dont le médecin identifie l'emplacement de la blessure à travers l'image, ce qui aide à construire un modèle transparent.

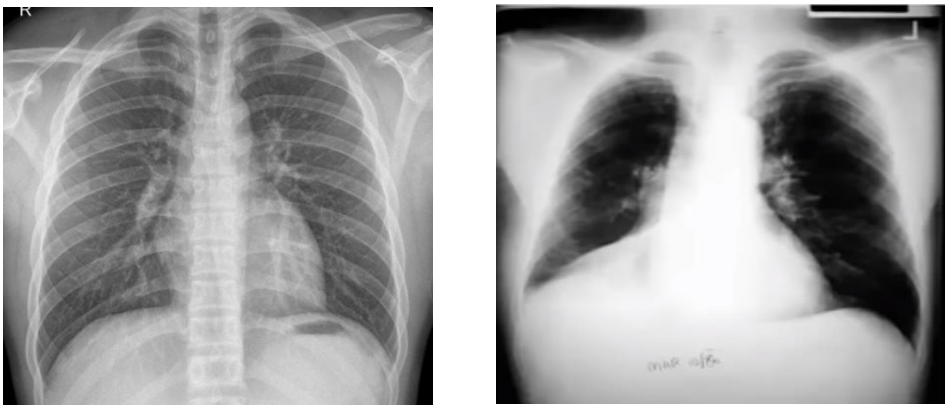


(A) (B)

Figure (II.4) : image (A)les poumons infectés et image (B)les poumons ne sont pas infectés

II.3.3. Préparation des données

Pour l'analyse des images radiographiques pulmonaires, toutes les radiographies pulmonaires ont été initialement examinées pour le contrôle de la qualité en supprimant tous les scans de mauvaise qualité ou illisibles. Les diagnostics des images ont ensuite été évalués par deux médecins experts avant d'être autorisés à former le système d'IA. Afin de tenir compte d'éventuelles erreurs d'enregistrement, le groupe d'évaluation a également été examiné par un troisième expert et l'image est un exemple dans la figure (B) image floue :



(A) (B)

Figure (II.5) : image (A) est clair et image (B) est floue

II.3.4. Extraction des paramètres importants pour la prédiction

L'extraction des paramètres de l'image n'est pas moins difficile que la construction du modèle approprié lui-même, et ce diagramme montre comment extraire les paramètres **:**

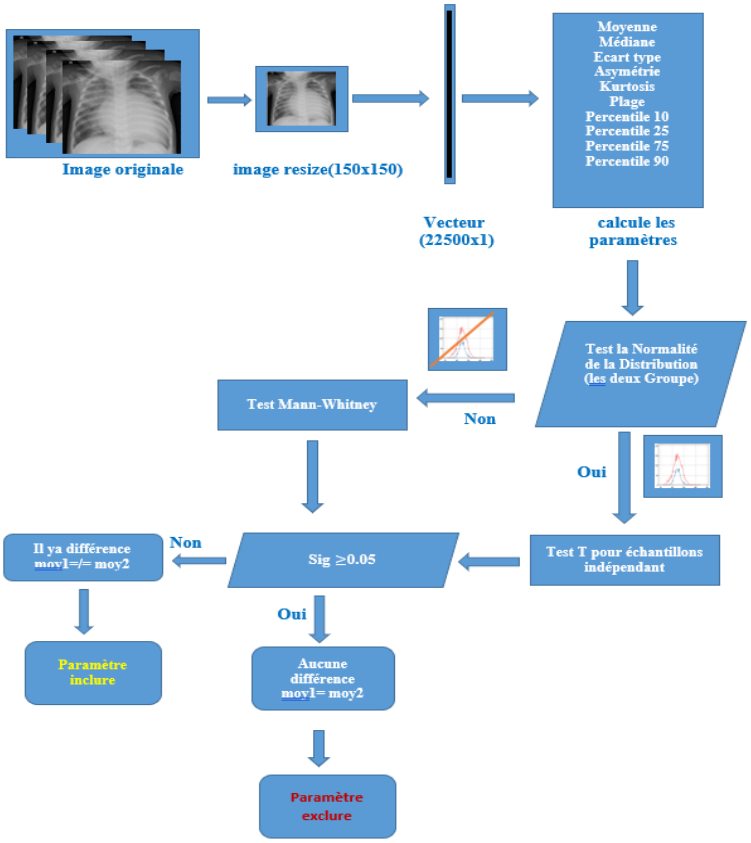


Figure (II.6) : Extraction des paramètres importants pour la prédiction

Et les paramètres impliqués sont :

La Moyenne, Médian, Ecart type, Asymétrie, Kurtosis, Plage, Percentile 10, Percentile 25, Percentile 75, Percentile 90.

II.3.4.1. Définition d'une distribution normale

La distribution normale, également connue sous le nom de distribution gaussienne, est une distribution de probabilité symétrique par rapport à la moyenne, montrant que les données proches de la moyenne sont plus fréquentes que les données éloignées de la moyenne. Sous forme de graphique, la distribution normale apparaîtra sous la forme d'une courbe en cloche.

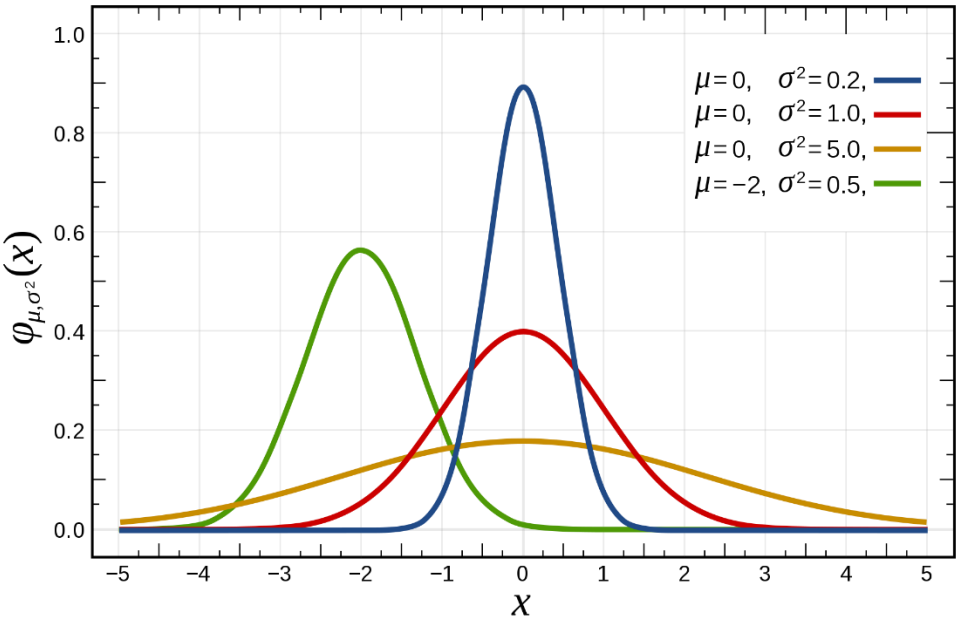


Figure (II.7) : La forme de la distribution normale

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}}\, e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

II.3.4.2. Test T pour deux échantillons indépendants

- **Équation de test T**

$$t = \frac{\bar{X}_1 - \bar{X}_2}{s_p \cdot \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

avec :

$$s_p = \sqrt{\frac{(n_1 - 1)s_{X_1}^2 + (n_2 - 1)s_{X_2}^2}{n_1 + n_2 - 2}}$$

t *:* la valeur du test T

$\overline{\boldsymbol{X1}}$: Moyenne du premier échantillon

X2 *:* Moyenne du deuxième échantillon

Sx1 *:* écart type du premier groupe

Sx2 *:* écart type de la deuxième variable

n1 *:* La taille du première échantillon

n2 *:* La taille du deuxième échantillon

- **Conditions d'utilisation du test T**

✓ Il doit être une variable quantitative
✓ données aléatoires
✓ Distribution normale pour les deux groupes

- **Utilisation de test T**

✓ **Hypothèses et le niveau de confiance (1-α)**

Hypothèse nulle : est que l'hypothèse selon laquelle il n'y a pas de différence.
Hypothèse alternative : est que l'hypothèse selon laquelle il existe une différence.
Le texte de l'hypothèse nulle et alternative
H0 : Il n'y a pas de différence statistiquement significative entre les deux échantillon**s.**
H1 : Il existe des différences statistiquement significatives entre les deux échantillons.

α : niveau d'erreur (La plus grande erreur que nous pouvons accepter par exemple α =5% =0.05 01c'est à dire niveau de confiance =1-α = 95% ou α=1% = 0.01c'est à dir niveau de confiance =1-α = 99%)

✓ **L'acceptation et le rejet de l'hypothèse**

L'acceptation de l'hypothèse (si T calculé inférieur à T tabulé[1]) ou (la signification de test supérieur ou equale **α**)
Rejeté l'hypothèse (si T calculé supérieur à T tabulé) ou (la signification de test inférieur ou equale **α**)
La figure montre la région pro de l'acceptation et le rejet de l'hypothèse H0

[1] Tableau à l'annexe H

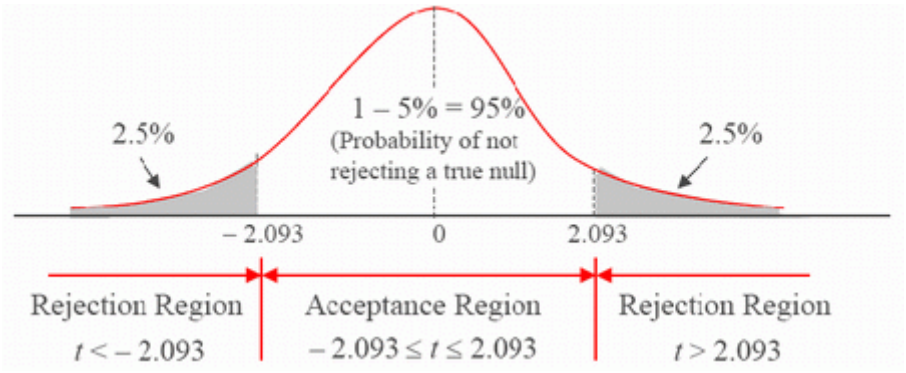


Figure (II.8) : l'acceptation et le rejet de l'hypothèse nulle H0 au niveau de confiance à 95%

II.3.4.3. Test de Mann et Whitney

On dispose des mesures des valeurs de X dans deux échantillons indépendants E1 et E2, de tailles respectives n1 et n2. On souhaite comparer les deux moyennes expérimentales, c'est-à-dir tester l'hypothèse nulle (H0) : $\mu_1 = \mu_2$

On commence par trier les valeurs obtenues dans la réunion des deux échantillons par ordre croissant. Pour chaque valeur x_ii issue de E1, on compte le nombre de valeurs issues de E2 situées après lui dans la liste ordonnée (celles qui sont égales à xi ne comptent que pour 1/2).

On note μ_1 la somme des nombres ainsi associés aux différentes valeurs issues de E1. On fait de même en échangeant les rôles des deux échantillons, ce qui donne la somme μ_2. Soit u la plus petite des deux sommes obtenues :

$$u = \min\{u_1 \ ; \ u_2\}$$

On note u la variable aléatoire associée.

- Pour n1 et n2 quelconques, on lit dans les tables du test de Mann et Whitney le nombre m_ tel que, sous (H0), P (U ≤ m_α_) = α . On rejette (H0) au risque d'erreur α si U ≤ m_α.

Autrement on accepte (H0).

- Si n1 et n2 sont assez grands (≥ 20 en général), sous (*H0*), U suit approximativement la loi normale N(u ,σ) avec

$$\mu = \frac{n_1 n_2}{2} \quad \text{et} \quad \sigma = \sqrt{\frac{n_1 n_2 (n_1 + n_2 + 1)}{12}}$$

On lit u_ dans la table de l'écart réduit de la loi normale tel que P (|N| ≥ u_α) on calcule

$$\varepsilon = \frac{u - \mu}{\sigma}$$

et on rejette (***H0***) au risque d'erreur α si $\varepsilon = \frac{u-\mu}{\sigma} \in]-u_\alpha ; u_\alpha[$. Autrement on accepte (***H0***).

- **Utilisation de test Mann et Whitney**

✓ **Hypothèses et le niveau de confiance (1-α)**

Hypothèse nulle : est que l'hypothèse selon laquelle il n'y a pas de différence.
Hypothèse alternative : est que l'hypothèse selon laquelle il existe une différence.
Le texte de l'hypothèse nulle et alternative
H0 : Il n'y a pas de différence statistiquement significative entre les deux échantillon**s.**
H1 : Il existe des différences statistiquement significatives entre les deux échantillons.

α : niveau d'erreur (La plus grande erreur que nous pouvons accepter par exemple α =5% =0.05 01c'est à dire niveau de confiance =1-α = 95% ou α=1% = 0.01c'est à dir niveau de confiance =1-α = 99%)

✓ **L'acceptation et le rejet de l'hypothèse**

L'acceptation de l'hypothèse (si **U** calculé inférieur à **U** tabulé) ou (la signification de test supérieur ou equale **α**)
Rejeté l'hypothèse (si **U** calculé supérieur à **U** tabulé) ou (la signification de test inférieur ou equale **α**)

II.3.5. Choisir et construire le modèle

Le choix du modèle approprié dépend du phénomène étudié, s'il est classé ou régression et notre étude consiste à classer les images médicales en deux catégories (0,1) (classification supervisé) et A été choisi les méthodes de classification suivante :

k plus proches voisins (KNN), machine à vecteurs de support (SVM), Réseau Bayésien (RB), Arbre de décision (AD), Régression de Logistique (LR), réseau de neurones (NN), Arbres aléatoire (AA).

La figure suivante montre différents classificateurs superviser

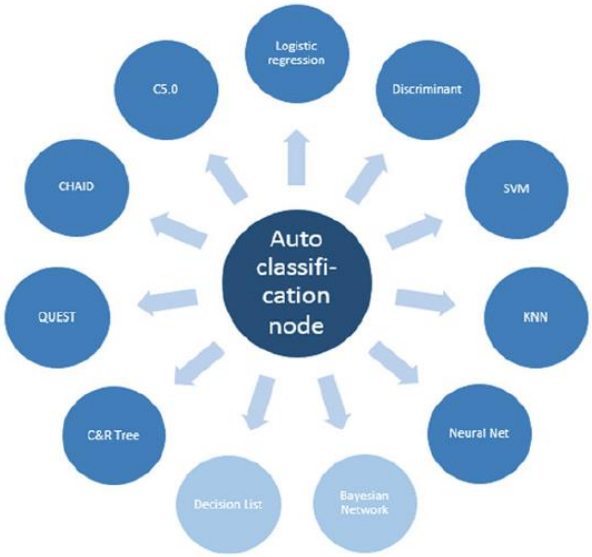


Figure (II.9) : les différents classificateurs superviser

Et la figure suivante montre comment choisir l'algorithme approprié

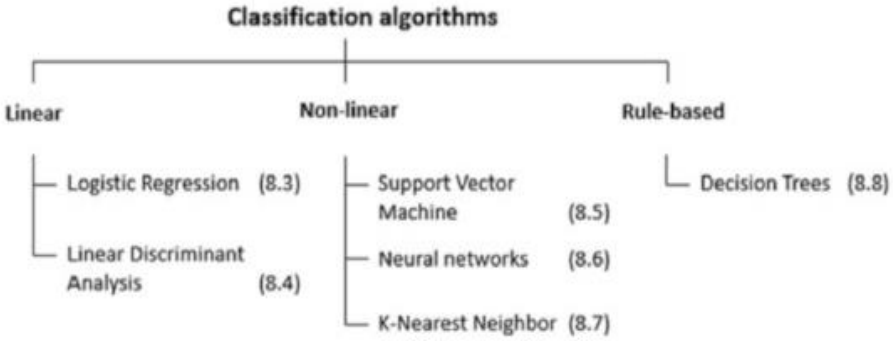


Figure (II.10) : choisir l'algorithme de classification

✓ **Partition les données**

Les données ont été divisées en trois sections : données d'apprentissages est 89.9% , données du validation est 9.1% et données du test est 1% et le tableau montre que :

Tableau (II.2) : Partition les données (Apprentissage, Validation, Test) pour les modèles KNN, SVM, RB,*AD, LR, NN, AA*

Etat	**Partition**				**Total**
	Effectif	**Apprentissage**	**Test**	**Validation**	
Normal	**Fréquence**	1426	140	17	1583
	Pourcentage %	90.1	8.8	1.1	100
Pneumonia	**Fréquence**	3841	391	41	4273
	Pourcentage %	89.9	9.2	1	100
Total	**Fréquence**	5267	531	58	5856
	Pourcentage %	89.9	9.1	1	100

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

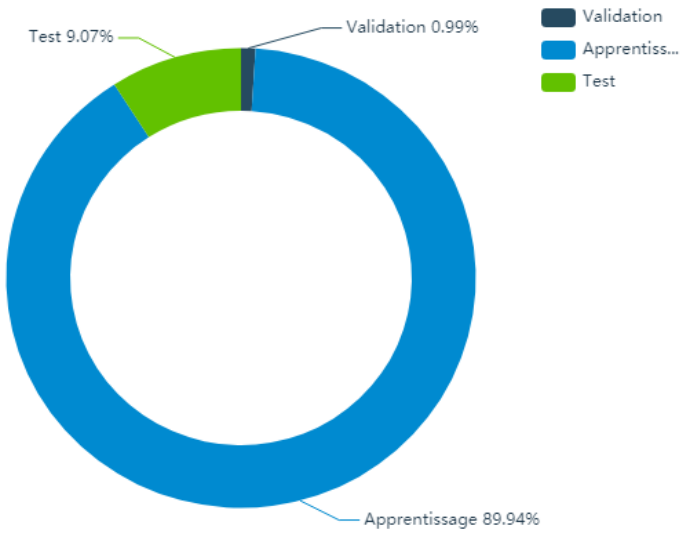


Figure (II.11) : Partition les données (Apprentissage, Validation, Test) pour les modèles KNN, SVM, RB,*AD, LR, NN, AA*

Mais les Partition des données **Convolution Neural Network (CNN**) comme suivante :

Les données ont été divisées en trois sections : données d'apprentissages est 89.07% , données du validation est 10.66 % et données du test est 0.27 % et le tableau montre que :

Tableau (II.3) : Partition les données (Apprentissage, Validation, Test) pour le modèle CNN

Etat	Partition				Total
	Effectif	Apprentissage	Validation	Test	
Normal	Fréquence	1341	234	08	1583
	Pourcentage %	84.7	14.8	0.5	100
Pneumonia	Fréquence	3875	390	08	4273
	Pourcentage %	90.7	9.1	0.2	100
Total	Fréquence	5216	624	16	5856
	Pourcentage %	89.07	10.66	0.27	100

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

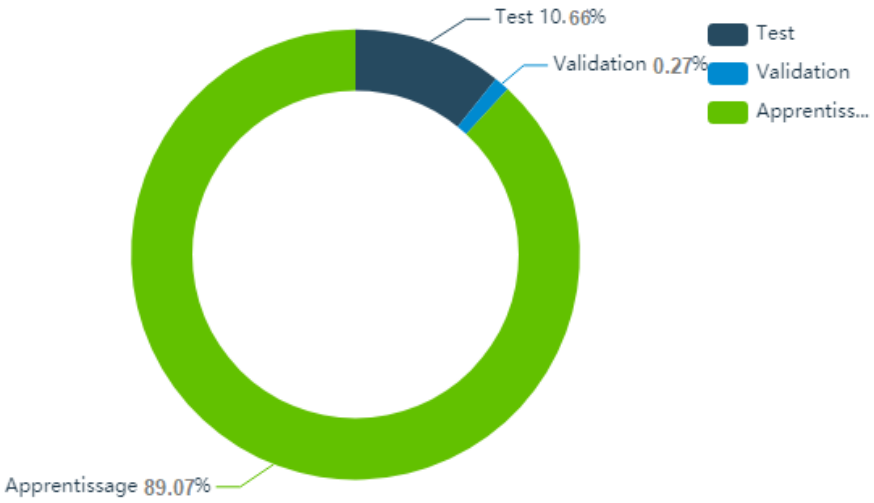


Figure (II.12) : Partition les données (Apprentissage, Validation, Test) pour CNN

II.3.6. Validation du modèle

La figure *(II.13)* suivante montre comment validation du modelé

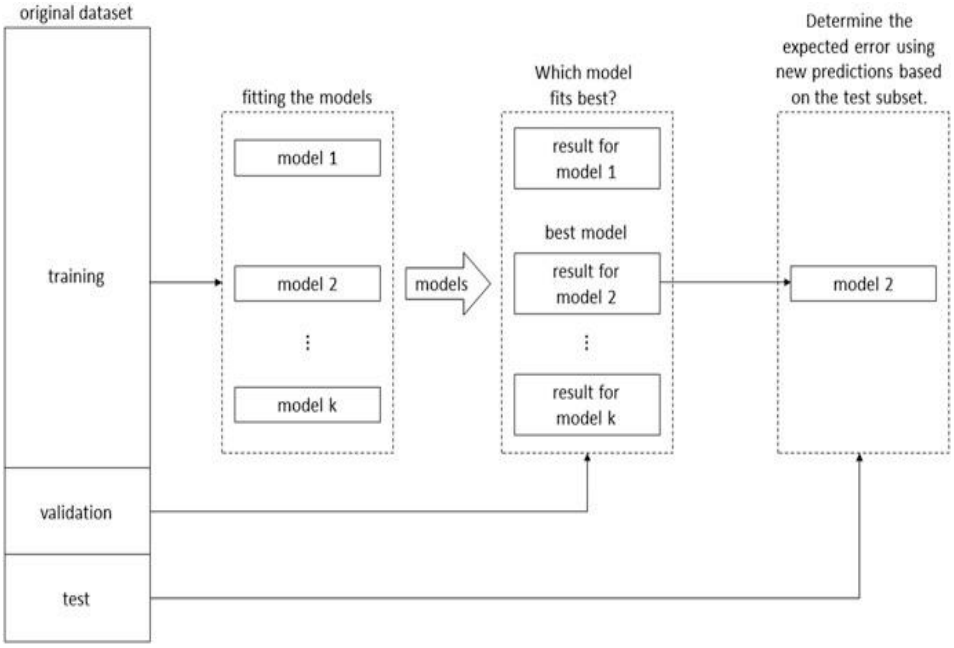


Figure (II.13) : Validation du modèle

II.3.7. Test du modèle

Nous testons le modèle pour voir s'il peut être utilisé en pratique avec une matrice de confusion, puis extrayons les critères de précision

II.3.8. Evaluation du modelé

Dans cette partie, on va s'intéresser à comment les modèles de prédictions en Machine Learning sont évalués, et plus précisément dans le cas de la classification.
L'étape d'évaluation s'avère être une étape importante dans la démarche de choix du Modèle, d'où la nécessité d'avoir un outil (matrice de confusion) de précision qui nous permet de mener cette étape en bonne et due forme.

II.3.8. 1. Matrice de Confusion

La Matrice de confusion est généralement utilisée dans le domaine du Machine Learning et plus précisément dans les problèmes de type classification statistique, également appelée matrice d'erreur**.**
Cette matrice permet de vérifier la performance d'un classifieur, ainsi que la

confusion qui règne entre les différentes classes, sur un ensemble de données de test dont les vraies valeurs sont connues .

Elle résume et englobe les prédictions effectuées par le modèle de classification, les Informations qu'elle nous transmet, nous permettent de voir la confusion entre les classes, et de distinguer les erreurs commises par ses dernières, ainsi que leurs natures.

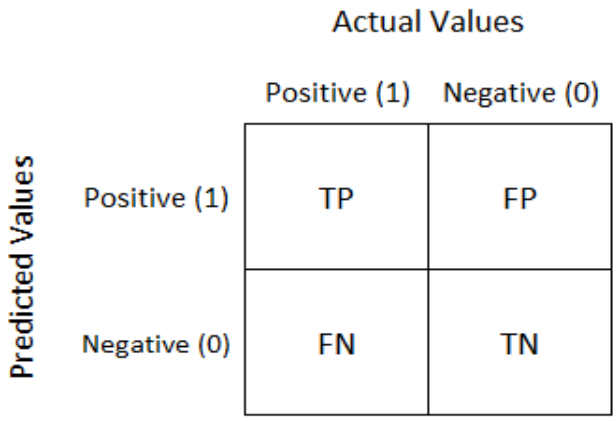


Figure (II.14) : Matrice de Confusion

II.3.8. 2. Critères d'évaluation

La matrice de confusion contient des valeurs liées aux observations obtenues par le Modèle sur le dataset, et à partir de ses dernières on est en mesure de calculer des métriques d'évaluation.

- ✓ **Accuracy**

La accuracy correspond au nombre de prédictions correctes faites par le modèle. Elle représente le ratio entre le nombre de prédictions correctes et le nombre total de prédiction. Ceci peut être calculé en utilisant les valeurs de la matrice de confusion et en utilisant la formule suivante :

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

✓ **Précision**

La précision correspond au nombre d'éléments corrects rendus par le modèle. En d'autres termes, cela correspond au ratio entre le nombre de classifications positives correctes et le nombre total de prédiction positives. Elle peut être calculée avec la formule suivante :

$$Precision = \frac{TP}{TP + FP}$$

✓ **Sensibilité**

La Sensibilité est la mesure de la proportion de cas positifs réels prédits Positifs (vrais positifs), et elle peut être définie comme le rapport entre le nombre total d'exemples positifs correctement classés et le nombre total d'exemples positifs.
La sensibilité est donnée par la relation :

$$Sensitivity = \frac{TP}{TP + FN}$$

✓ **Specificity**

La spécificité est définie comme la proportion de négatifs réels, ce qui a été prédit commenégatif (ou vrai négatif). Cette proportion pourrait également être appelée un taux de faux positif. La somme de la spécificité et du taux de faux positifs serait toujours égale à 1. La mesure de la spécificité se fait grâce à la formule :

$$Specificity = \frac{TN}{TN + FP}$$

✓ **Neg-predictive:**

C'est le rapport des cas vrais négatifs sur les cas sains négatifs et les cas faux positifs

$$\boldsymbol{Neg - predective = \frac{TN}{(TN + FN)}}$$

✓ **F1-Score:**

Le F1score, également appelé score F1, est une mesure de la précision d'un modèle sur un ensemble de données. Il est utilisé pour évaluer les systèmes de classification binaires, qui classent les exemples en "positifs" ou "négatifs". Le score F est un moyen de combiner la précision et le rappel du modèle, et il est défini comme la moyenne harmonique de la précision du modèle**.**

$$\text{F1 Score} = \frac{TP}{TP + \frac{1}{2}(FP + FN)}$$

Ou sous la forme suivante :

$$F1score = 2 * \frac{Sensitivity * Precision}{Sensitivity + Precision}$$

✓ **AUC et Gini index**

Une autre mesure courante de la qualité de l'ajustement pour le classificateur est l'indice de Gini, qui est juste une transformation de l'AUC

$$Gini = 2 * AUC - 1$$

II.3.9. La décision (Acceptation ou Rejeter le modèle)

L'acceptation et le rejet dépendent de la précision du modèle, et cela dépend de la matrice de confusion, ainsi que la nature du phénomène joue un rôle important dans la décision d'accepter ou de rejeter le modèle.

II.4. Les Outils utilisés

II.4.1. Plateforme de l'apprentissage automatique (ML)

Il existe un grand nombre d'outils pour effectuer du ML. Parmi un grand nombre de plates-formes possibles, le SPSS Modeler v18.0 a été choisi pour diverses raisons. Prend en charge plusieurs sources de données l'utilisateur peut manipuler des données à partir de feuilles de calcul, de fichiers non indexés et de bases de données principales. Préparation automatique des données : elle fournit à l'utilisateur une conversion automatique des données, après analyse, dans les formats de modèles les plus précis et les plus prédictifs et avec le moins d'erreurs.

Ces avantages font de SPSS Modeler v18.0 la quatrième place des logiciels d'analyse de données utilisés dans la recherche académique (Muenchen, 2015), après Python, R et Knime, cependant, il est important de noter que SBS possède des nœuds dans lesquels du code Python ou R peut être intégré.

Étant donné que SPSS Modeler v18.0 facilite le travail de l'analyseur de données grâce à une interface graphique plus intuitive que le code logiciel pur

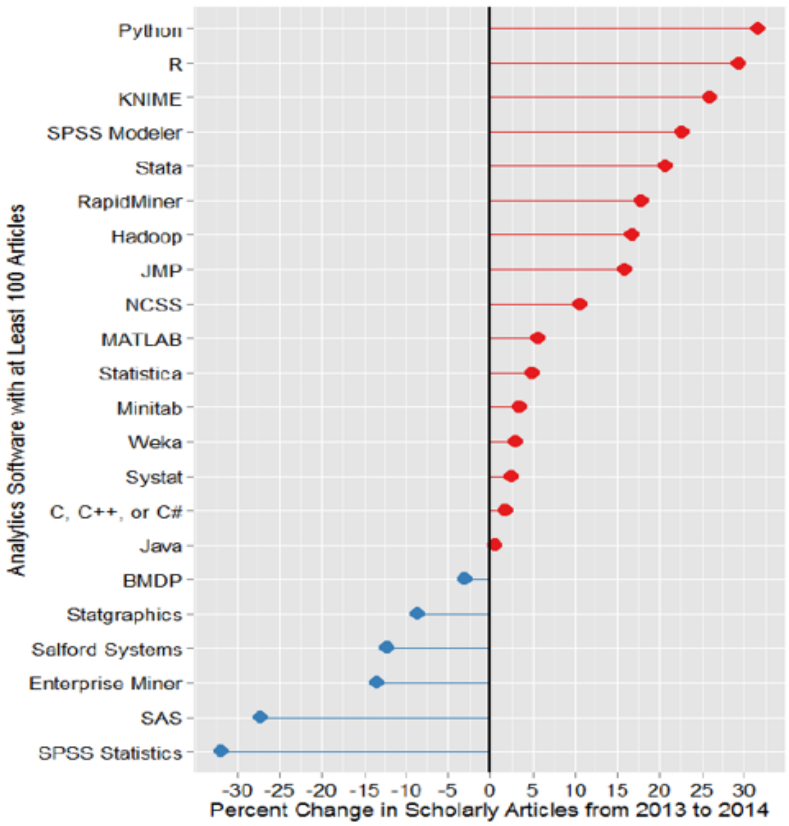


Figure (II.15) : Progression des articles académiques par outil d'analyse des données

II.4.2. IBM SPSS Statistique

SPSS Statistics est une suite logicielle statistique développée par IBM pour la gestion des données, l'analyse avancée, l'analyse multivariée, l'informatique décisionnelle et les enquêtes criminelles. Longtemps produit par SPSS Inc., il a été racheté par IBM en 2009.

Système d'exploitation :

Windows (x86-64), macOS (x86-64), Linux (x86-64, ppc64le, IBM Z)

Versions : 28.0

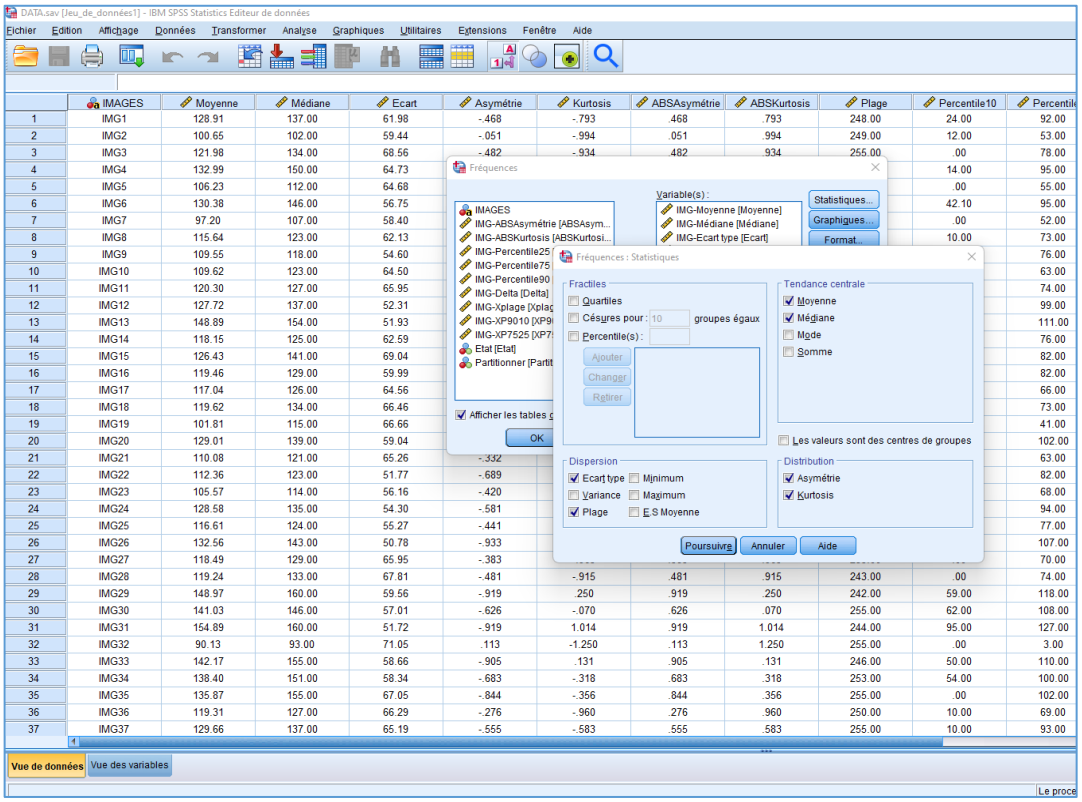


Figure (II.16) : interface principale du spss v 28

II.4.3. IBM SPSS Modeler

Traduit de l'anglais-IBM SPSS Modeler est une application logicielle d'exploration de données et d'analyse de texte d'IBM. Il est utilisé pour créer des modèles prédictifs et effectuer d'autres tâches analytiques.

Systèmes d'exploitation : Microsoft Windows, Linux, Unix, Mac OS X

Versions : 18.0

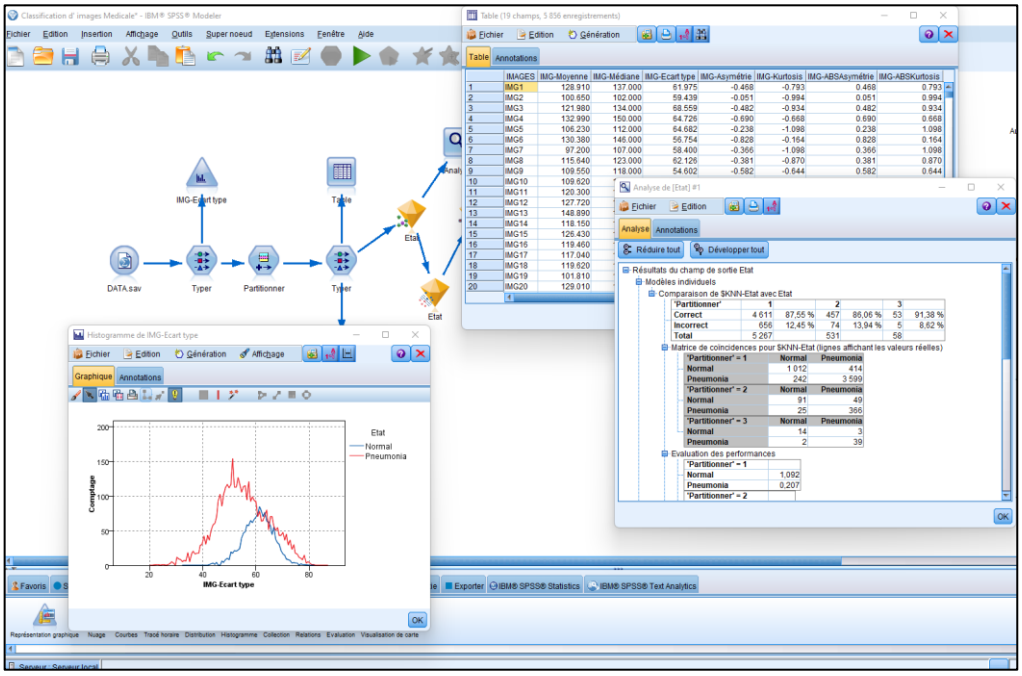


Figure (II.17): interface principale spss Modeler v 18

II.4.4. Langage Matlab

MATLAB est un langage de script émulé par un environnement de développement du même nom ; il est utilisé à des fins de calcul numérique**.**

Systèmes d'exploitation : Linux, Unix, Mac OS, Windows

Versions : 2020

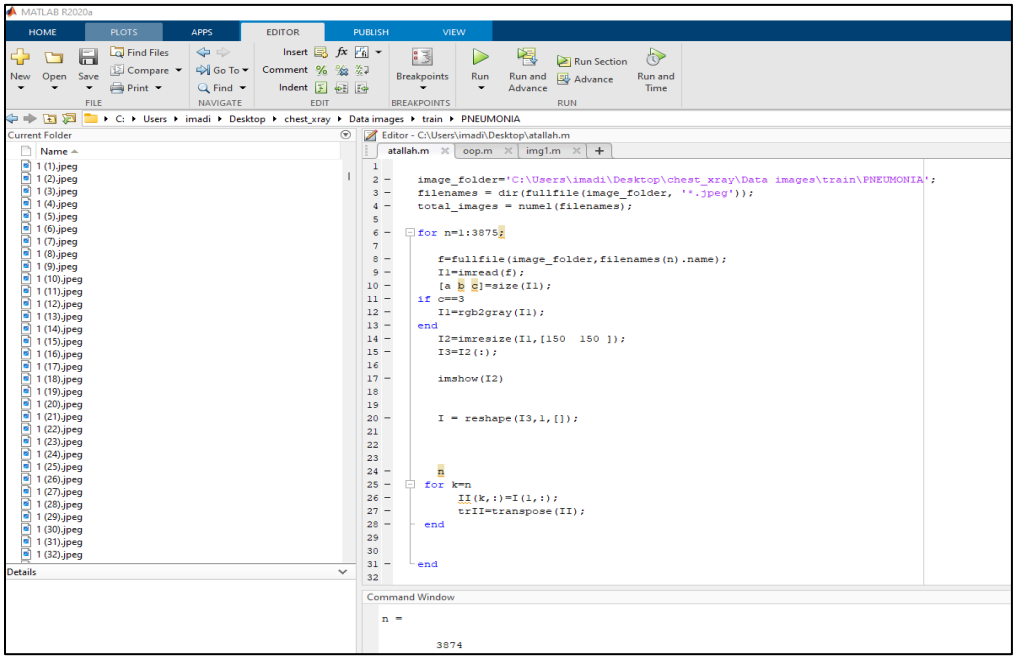


Figure (II.18) : interface principale Matlab 2020

II.4.5. Langage Python

Python est un langage de programmation interprété, multiparadigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet.

Systèmes d'exploitation : Linux, Unix, Mac OS, Windows

Versions 3.11.3

Nous avons utilisé Python dans plateforme kaggle

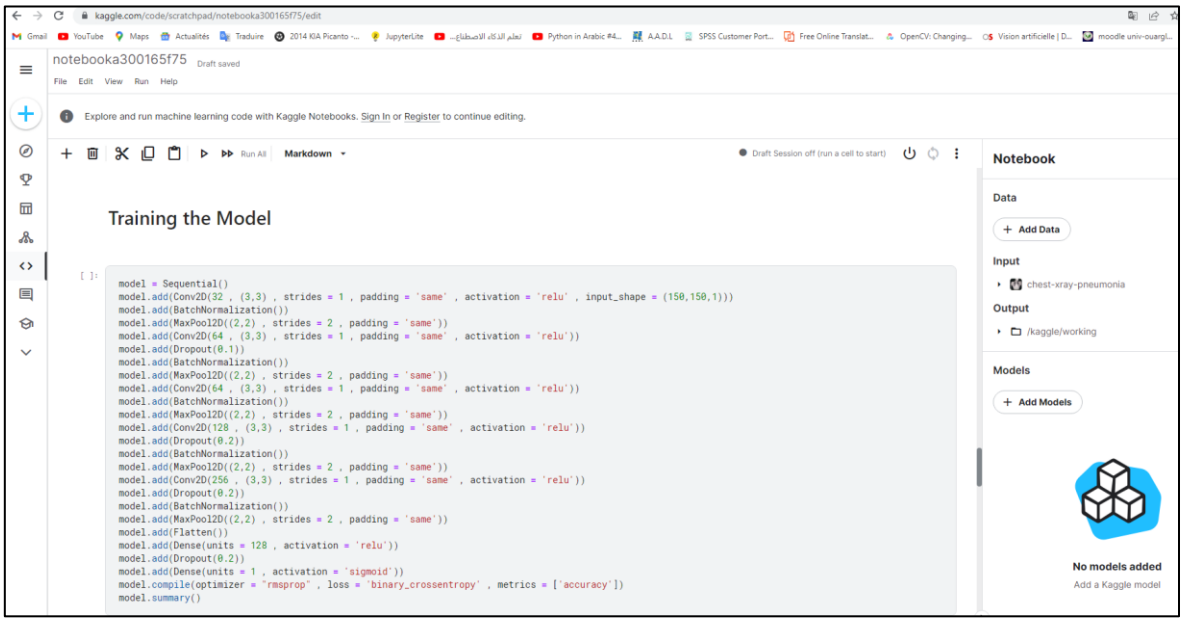


Figure (II.19) : interface principale plateforme Kaggle

II.5. Conclusion

Après nous avons traité de la structure générale de notre projet, dans laquelle nous avons détaillé les différentes étapes. Nous avons détaillé chaque étape en montrant son objectif ainsi que ses algorithmes avec toutes les justifications et dans le chapitre suivant nous essaierons de mettre en évidence la partie expérimentale de notre projet. Ensuite, les résultats obtenus seront analysés et interprétés.

Chapitre III

Réalisation des modèles
de classification

III.1. Introduction

Dans ce chapitre, nous présenterons la solution que nous proposons pour répondre au problème de détection précoce de la pneumonie.

Le chapitre se compose de trois sections principales. Premièrement, nous extrayons les paramètres des images pour les deux cas (Normal et Pneumonia) et les comparons dans les deux cas à l'aide des tests statistiques en utilisant IBM SPSS Statistiques v28. Deuxièmement, nous construisons différents modèles de classification tels que k plus proches voisins, machine à vecteurs de support, réseau bayésien, arbre de décision, régression de logistique, réseau de neurones, arbres aléatoire et convolution neural network, en utilisant IBM SPSS Modeler v18.0. Enfin, nous les comparons en utilisant la précision Accuracy, Précision, Sensibilité, Spécificité, Nég-Prédictif, F1-Score, AUC et Gini, nous adoptons les meilleurs modèles pour une utilisation dans la vie pratique.

III.2. Les Paramètres d'images

III.2.1. Moyenne d'image (IMG-Moyenne)

✓ **Description statistique**

Tableau (III.1) : Description statistique pour la " Moyenne d'image" pour les deux cas

Etat	Moyenne	Ecart type	Maximum	Minimum	Plage
Normal	122.64	13.53	169.67	73.30	96.37
Pneumonia	122.84	19.89	221.53	58.72	162.81

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la moyenne arithmétique de la "moyenne d'image" pour l'état " **Normal** " est égale à 122,64, avec un écart-type égal à 13,53, tandis que la plage est égale à 96.37, mais la moyenne arithmétique de la "moyenne d'image" pour l'état " **Pneumonia** " est égale à 122,84, avec un écart-type égal à 19,89, tandis que la plage est égale à 162.81.

✓ **Description graphique**

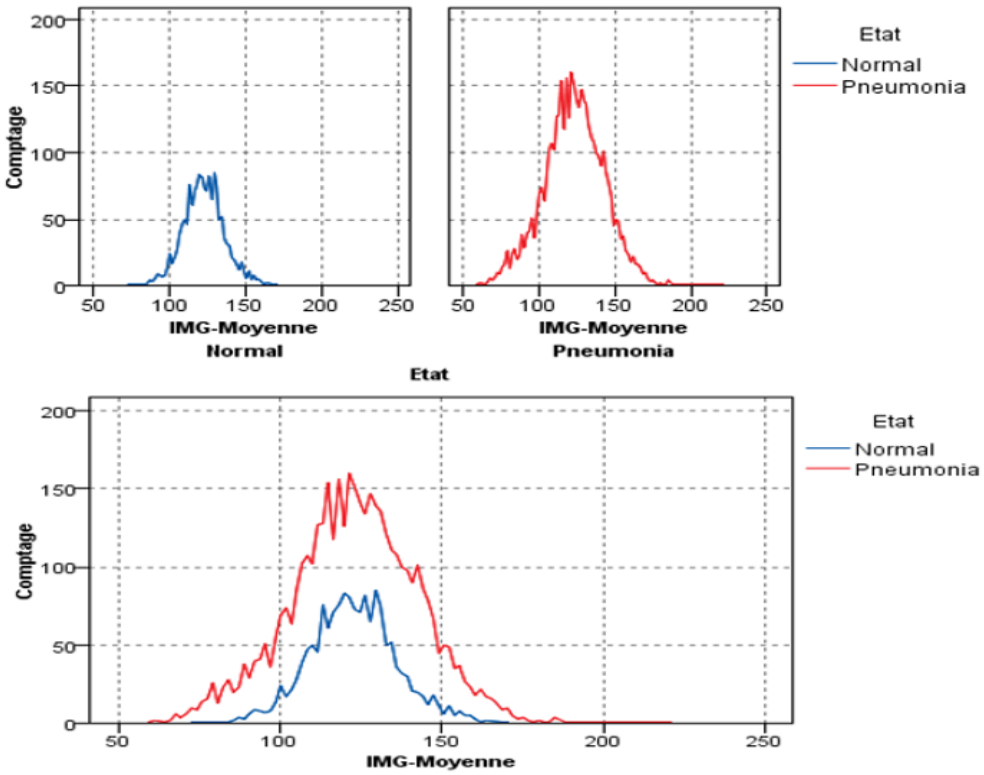


Figure (III.1) : la distribution " Moyenne d'image"
Pour les deux cas (Normal, Pneumonia) avant l'ajustement

✓ **Ajustement de simulation pour détection de type de distribution**

Pour détection de type de distribution pour les paramètres d'image nous avons deux critères de la qualité d'ajustement soit critère **Anderson-Darling** ou **Kolmogorov-Smirnov.** Choisissons et appliquons critère **Kolmogorov-Smirnov** pour l'ajustement.

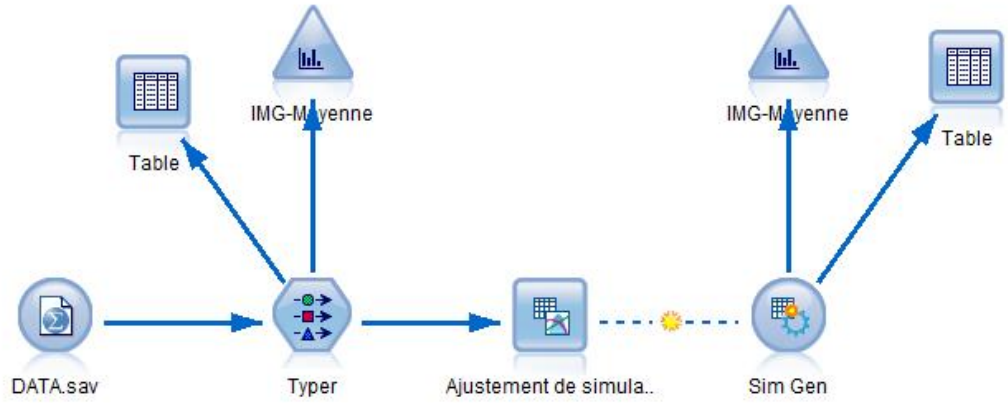


Figure (III.2) : Fleux d'ajustement pour les paramètres d'images Dans le deux cas (Normal, Pneumonia)

✓ **Distribution (IMG-Moyenne)**

Tableau (III.2) : Distribution " Moyenne d'image" pour les deux cas

Critère de la qualité d'ajustement Kolmogrov-Simirnov			
Etat	**Statistique d'ajustement**	**Paramétré**	**Distribution**
Normal	A=1.25 P=0.0 (K=0.03 P=0.01)	Moy=122.64 Ecart=13.53	Normale
Pneumonia	A=1.73 P=0.0 (K=0.03 P=0.01)	Moy=122.85 Ecart=19.89	Normale

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

D'après le tableau, nous remarquons que la distribution " Moyenne d'image" pour le cas" **Normal** " est Normale et la distribution " Moyenne d'image" pour le cas" **Pneumonia** " est aussi Normale.

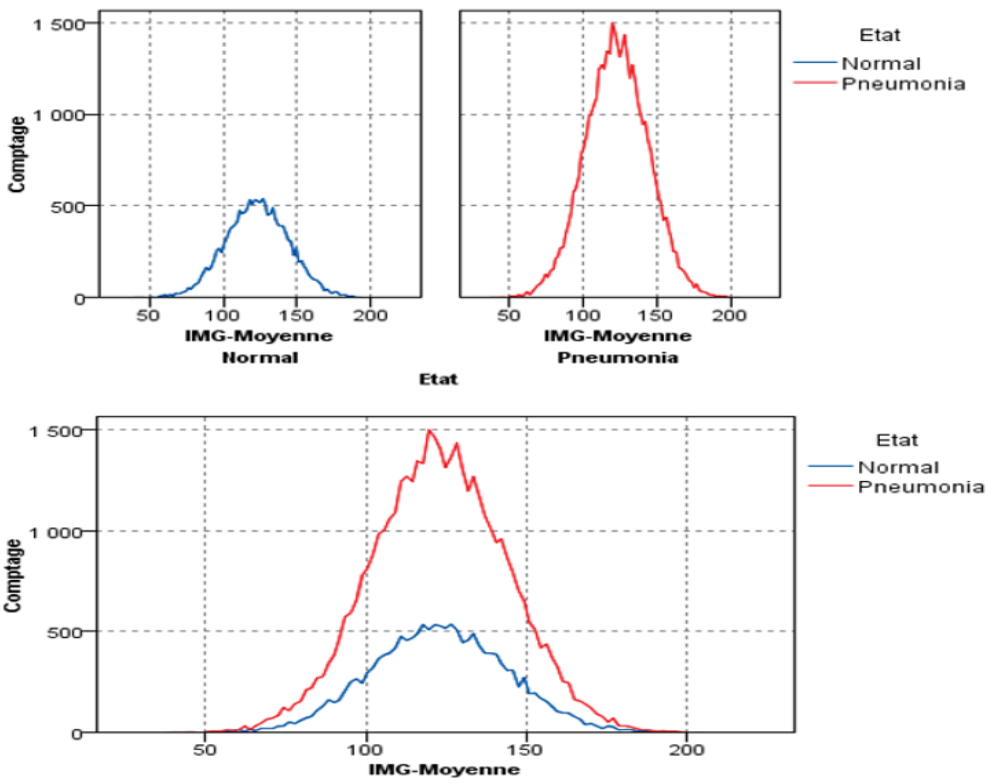


Figure (III.3) : la distribution " Moyenne d'image"
Pour les deux cas (Normal, Pneumonia) après l'ajustement

✓ **Comparaison entre les deux cas (Normal, Pneumonia)**

Tableau (III.3) : Résultat de test T pour " Moyenne d'image" dans les deux cas

Test T des échantillons indépendants					
Etat	**Moyenne**	**Ecart type**	**T**	**Sig**	**Résultat**
Normal	122.64	13.53	-0.448	0.654	NS
Pneumonia	122.84	19.89			

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la valeur de test T égale à T= - 0,448 avec niveau de signification égale 0.654 il est supérieur à 0.05 donc il n'y a pas de différences statistiquement significatives dans la Moyenne d'image entre les deux cas (Normal, Pneumonia).

Décision : Puisque la moyenne arithmétique de Moyenne d'image n'est pas différente dans les deux cas, nous ne l'utiliserons pas pour construire le modèle.

III.2.2. Médian d'image (IMG-Médian)

✓ **Description statistique**

Tableau (III.4) : Description statistique pour la " Médian d'image" pour les deux cas

Etat	Moyenne	Ecart type	Maximum	Minimum	Plage
Normal	131.88	15.67	191	63	128
Pneumonia	133.35	25.65	227	56	171

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la moyenne arithmétique de la "Médian d'image" pour l'état " **Normal** " est égale à 131,88**,** avec un écart-type égal à 15,67, tandis que la plage est égale à 128, mais la moyenne arithmétique de la " Médian d'image" pour l'état " **Pneumonia** " est égale à 133,35, avec un écart-type égal à 25,65, tandis que la plage est égale à 171.

✓ **Description graphique**

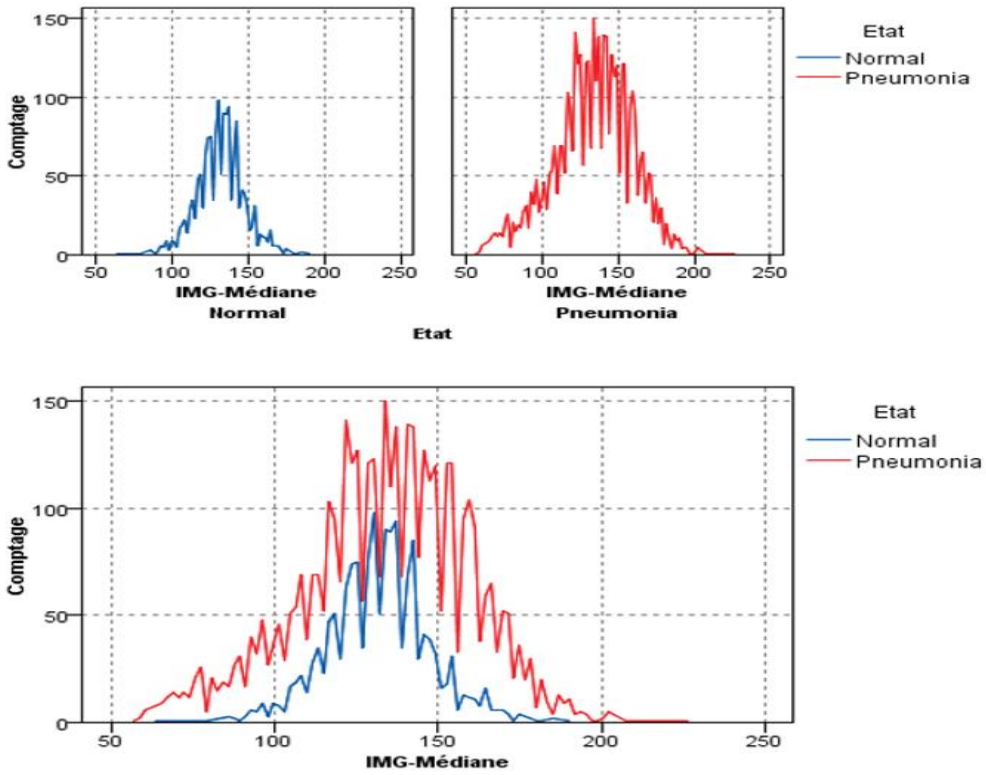


Figure (III.4) : la distribution " Médian d'image"
Pour les deux cas (Normal, Pneumonia) avant l'ajustement

✓ Distribution (IMG-Médian)

Tableau (III.5) : Distribution " Médian d'image" pour les deux cas

Critère de la qualité d'ajustement Kolmogrov-Simirnov			
Etat	**Statistique d'ajustement**	**Paramétré**	**Distribution**
Normal	A=3.02 P=0.0 (K=0.05 P=0.01)	Moy=131.89 Ecart=15.67	Normale
Pneumonia	A=1.55 P=0.01 (K=0.02 P=0.06)	a=143.71 b=5.97 c=0.0	Weibull

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

D'après le tableau, nous remarquons que la distribution " Médian d'image" pour le cas" **Normal** " est Normale et la distribution " Médian d'image" pour le cas" **Pneumonia** " est Weibull.

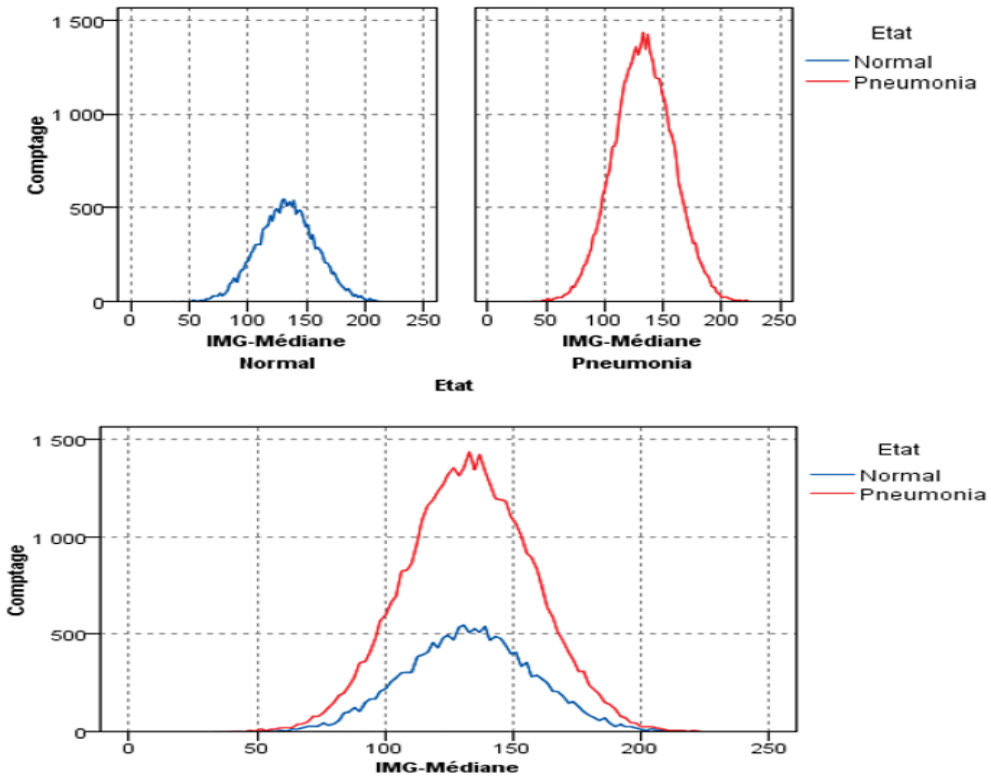


Figure (III.5) : la distribution " Médian d'image"
Pour les deux cas (Normal, Pneumonia) après l'ajustement

- ✓ **Comparaison entre les deux cas (Normal, Pneumonia)**

Tableau (III.6) : Résultat de test Mann-Whitney pour " Médian d'image" dans les deux cas

Test de Mann-Whitney					
Etat	**Rang moyen**	**Somme des rangs**	**U**	**Sig**	**Résulta**
Normal	2771.97	4388034.50	3134298.50	< 0.001	S
Pneumonia	2986.49	12761261.50			

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la valeur du test *Mann-Whitney* est égale à U=3134298.50 avec un niveau de signification < 0.001, qui est inférieur à 0.05, il y a donc des différences statistiquement significatives dans la Médian d'image entre les deux cas (normal, pneumonie).

Décision : Puisque la moyenne arithmétique de Médian d'image est différente dans les deux cas, nous allons l'utiliser pour construire le modèle.

III.2.3. Ecart type d'image (IMG- Ecart type)

- ✓ **Description statistique**

Tableau (III.7) : Description statistique pour la " Ecart type d'image" pour les deux cas

Etat	**Moyenne**	**Ecart type**	**Maximum**	**Minimum**	**Plage**
Normal	60.85	5.81	82.68	32.39	50.29
Pneumonia	55.03	9.91	87.28	19.90	67.38

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la moyenne arithmétique de la " Ecart type d'image" pour l'état " **Normal** " est égale à 60.85, avec un écart-type égal à 5.81, tandis que la plage est égale à 50.29, mais la moyenne arithmétique de la " Ecart type d'image" pour l'état " **Pneumonia** " est égale à 55.03, avec un écart-type égal à 9.91, tandis que la plage est égale à 67.38.

✓ **Description graphique**

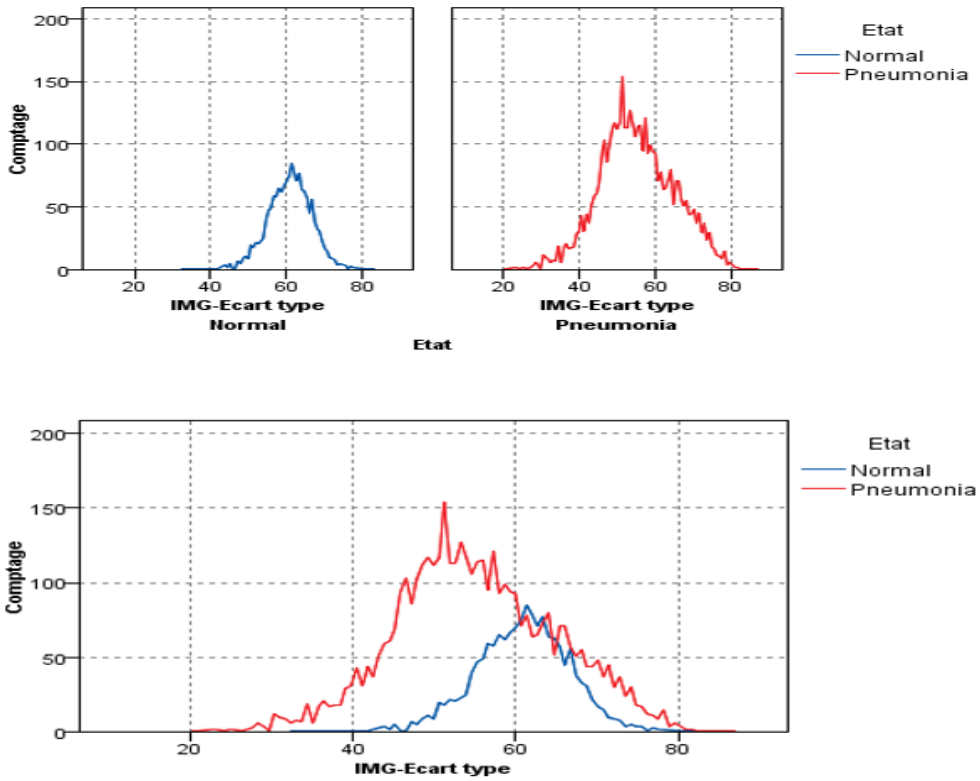


Figure (III.6) : la distribution " Ecart type d'image"
Pour les deux cas (Normal, Pneumonia) avant l'ajustement

✓ **Distribution (IMG-Ecart type)**

Tableau (III.8) : Distribution " Ecart type d'image" pour les deux cas

Critère de la qualité d'ajustement Kolmogrov-Simirnov			
Etat	**Statistique d'ajustement**	**Paramétré**	**Distribution**
Normal	A=1.16 P=0.0 (K=0.02 P=0.04)	Moy=60.85 Ecart=5.82	Normal
Pneumonia	A=4.85 P=0.01 (K=0.02 P=0.01)	Scale=0.54 Shape=29.52	Gamma

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

D'après le tableau, nous remarquons que la distribution " Ecart type d'image" pour le cas" **Normal** " est Normale et la distribution " Ecart type d'image" pour le cas" **Pneumonia** " est Gamma.

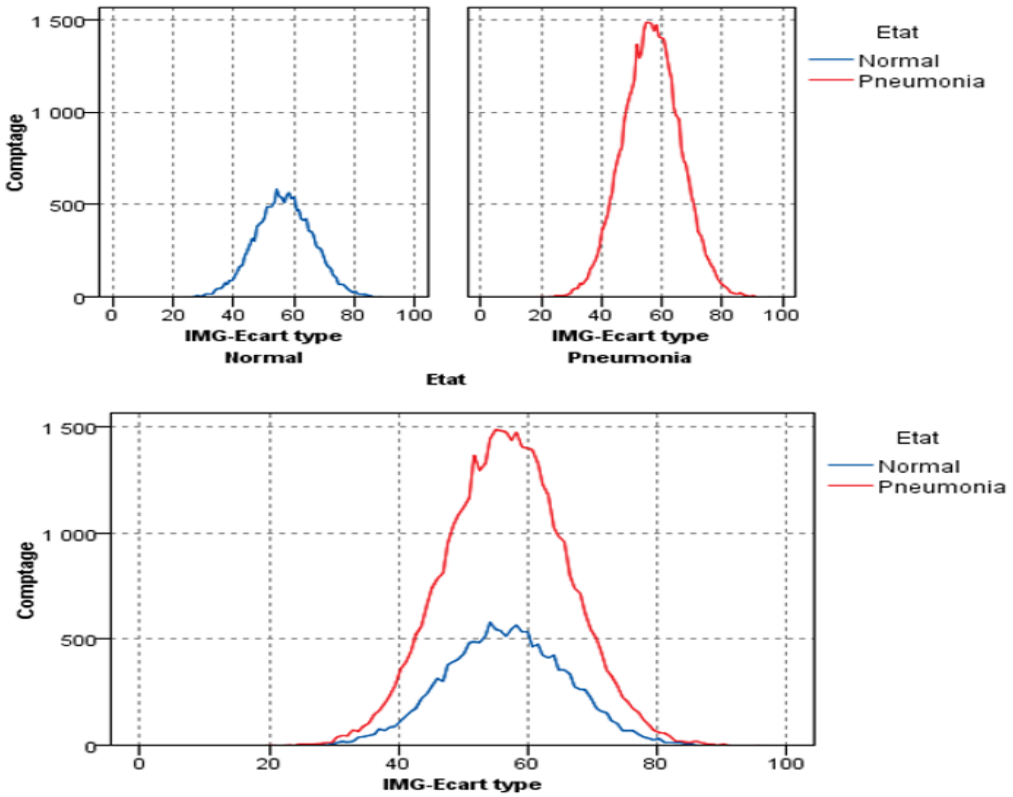


Figure (III.7) : la distribution " Ecart type d'image"
Pour les deux cas (Normal, Pneumonia) après l'ajustement

✓ **Comparaison entre les deux cas (Normal, Pneumonia)**

Tableau (III.9) : Résultat de test Mann-Whitney pour " Ecart type d'image" dans les deux cas

Test de Mann-Whitney					
Etat	**Rang moyen**	**Somme des rangs**	**U**	**Sig**	**Résulta**
Normal	3772.48	5971831	2046064	< 0.001	S
Pneumonia	2615.84	1117745			

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la valeur du test *Mann-Whitney* est égale à U=2046064 avec un niveau de signification < 0.001, qui est inférieur à 0.05, il y a donc des différences statistiquement significatives dans la Ecart type d'image entre les deux cas (normal, pneumonie).

Décision : Puisque la moyenne arithmétique de Ecart type d'image est différente dans les deux cas, nous allons l'utiliser pour construire le modèle.

III.2.4. Asymétrie d'image (IMG- Asymétrie)

✓ **Description statistique**

Tableau (III.10) : Description statistique pour la "Asymétrie d'image" pour les deux cas

Etat	Moyenne	Ecart type	Maximum	Minimum	Plage
Normal	-0.4942	0.2276	0.784	-1.468	2.252
Pneumonia	-0.6076	0.4419	0.911	-2.764	3.675

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la moyenne arithmétique de la "moyenne d'image" pour l'état " **Normal** " est égale à - 0.4942, avec un écart-type égal à 0.2276, tandis que la plage est égale à 2.252, mais la moyenne arithmétique de la " Asymétrie d'image" pour l'état " **Pneumonia** " est égale à – 0.6076, avec un écart-type égal à 0.4419, tandis que la plage est égale à 3.675.

✓ **Description graphique**

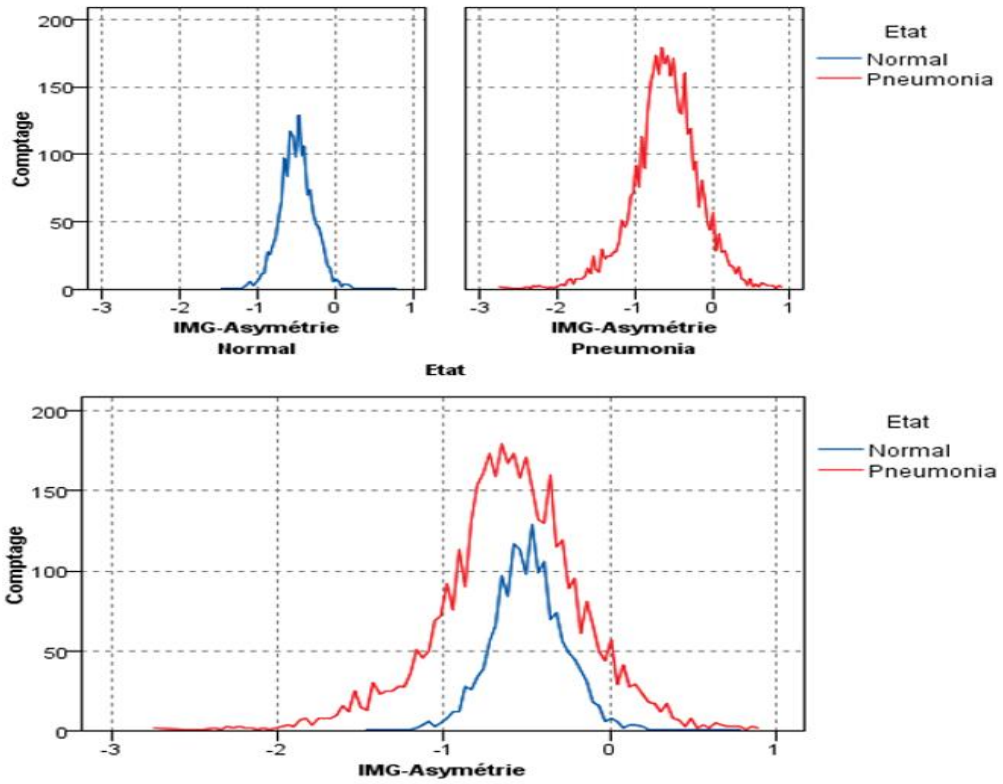


Figure (III.8) : la distribution " Asymétrie d'image"
Pour les deux cas (Normal, Pneumonia) avant l'ajustement

✓ **Distribution (IMG- Asymétrie)**

Tableau (III.11) : Distribution " Asymétrie d'image" pour les deux cas

Critère de la qualité d'ajustement Kolmogrov-Simirnov			
Etat	**Statistique d'ajustement**	**Paramétré**	**Distribution**
Normal	A=2.22 P=0.0 (K=0.03 P=0.01)	Moy=0.49 Ecart=0.23	Normale
Pneumonia	A=16.67 P=0.0 (K=0.05 P=0.01)	Moy=-0.61 Ecart=0.44	Normale

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

D'après le tableau, nous remarquons que la distribution " Asymétrie d'image" pour le cas" **Normal** " est Normale et la distribution " Asymétrie d'image" pour le cas" **Pneumonia** " est aussi Normale.

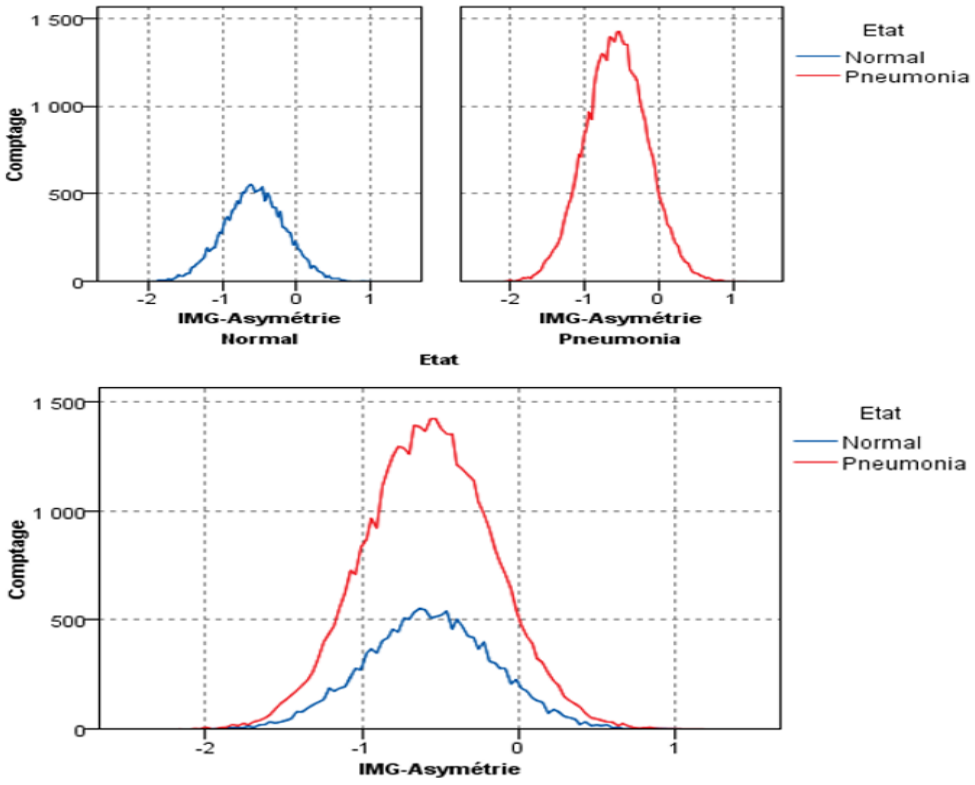


Figure (III.9) : la distribution " Asymétrie d'image"
Pour les deux cas (Normal, Pneumonia) après l'ajustement

- ✓ **Comparaison entre les deux cas (Normal, Pneumonia)**

Tableau (III.12) : Résultat de test T pour " Asymétrie d'image" dans les deux cas

Test T des échantillons indépendants					
Etat	**Moyenne**	**Ecart type**	**T**	**Sig**	**Résultat**
Normal	-0.4942	0.2276	12.804	< 0.001	S
Pneumonia	-0.6076	0.4419			

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la valeur de test T égale à T= 12.804 avec niveau de signification égale < 0.001 il est inférieur à 0.05, donc il y a donc des différences statistiquement significatives dans l'Asymétrie d'image entre les deux cas (Normal, Pneumonia)**.**

Décision : Puisque la moyenne arithmétique de l'Asymétrie d'image est différente dans les deux cas, nous allons l'utiliser pour construire le modèle.

III.2.5. Kurtosis d'image (IMG- Kurtosis)

- ✓ **Description statistique**

Tableau (III.13) : Description statistique pour la " Kurtosis d'image" pour les deux cas

Etat	Moyenne	Ecart type	Maximum	Minimum	Plage
Normal	-0.6334	0.3963	2.073	-1.498	3.571
Pneumonia	-0.4521	0.9078	9.241	-1.610	10.851

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la moyenne arithmétique de la " Kurtosis d'image" pour l'état " **Normal** " est égale à – 0.6334, avec un écart-type égal à 0.3963, tandis que la plage est égale à 3.571, mais la moyenne arithmétique de la " Kurtosis d'image" pour l'état " **Pneumonia** " est égale à – 0.4521, avec un écart-type égal à 0.9078, tandis que la plage est égale à 10.851.

✓ **Description graphique**

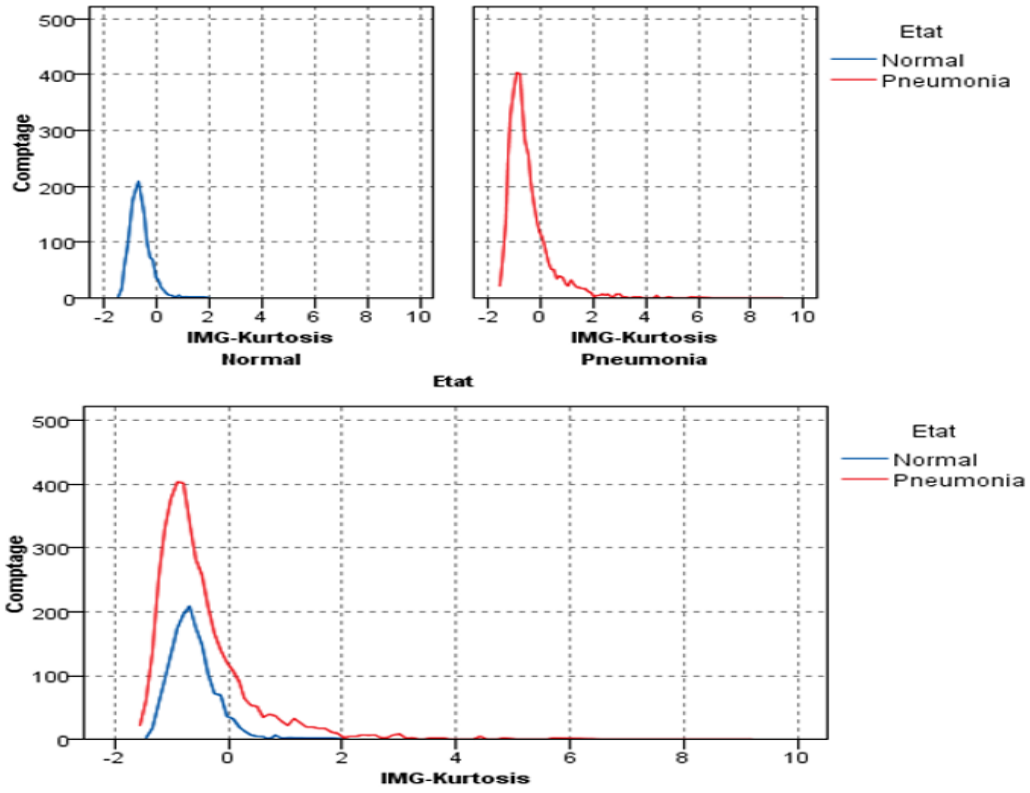


Figure (III.10) : la distribution " Kurtosis d'image"
Pour les deux cas (Normal, Pneumonia) avant l'ajustement

✓ **Distribution (IMG- Kurtosis)**

Tableau (III.14) : Distribution " Kurtosis d'image" pour les deux cas

Critère de la qualité d'ajustement Kolmogrov-Simirnov			
Etat	**Statistique d'ajustement**	**Paramétré**	**Distribution**
Normal	A=15.32 P=0.0 (K=0.07 P=0.01)	Moy=-0.63 Ecart=0.4	Normale
Pneumonia	A=241.02 P=0.0 (K=0.16 P=0.01)	Moy=-0.45 Ecart=0.91	Normale

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

D'après le tableau, nous remarquons que la distribution " Kurtosis d'image" pour le cas" **Normal** " est Normale et la distribution " Kurtosis d'image" pour le cas" **Pneumonia** " est aussi Normale.

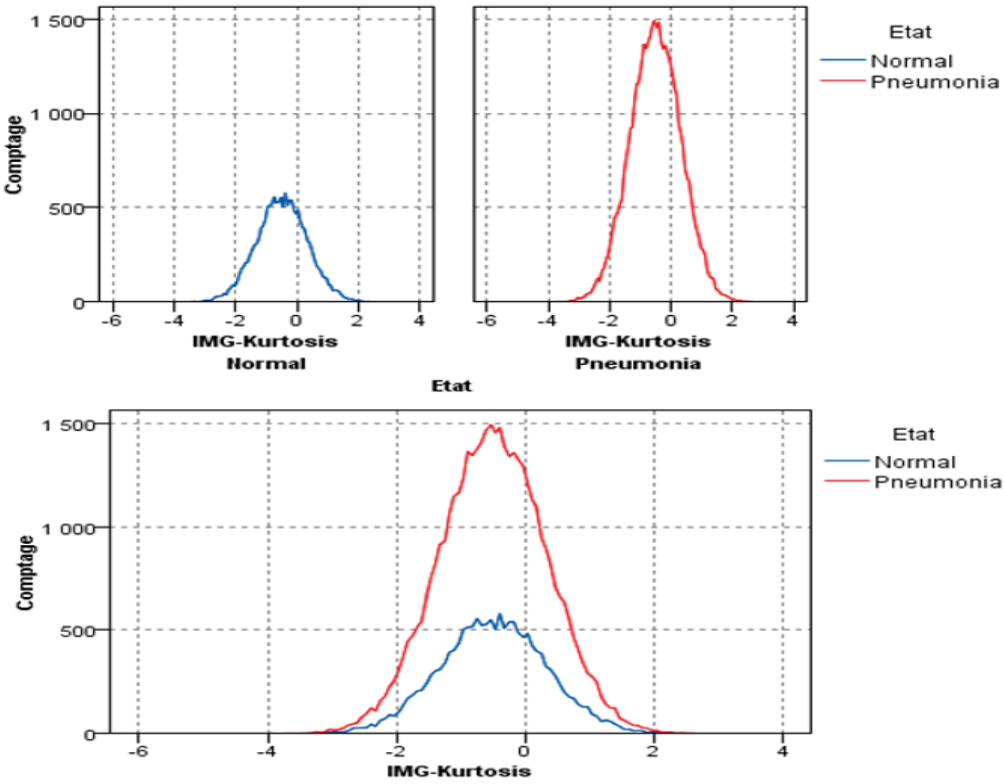


Figure (III.11) : la distribution " Kurtosis d'image"
Pour les deux cas (Normal, Pneumonia) après l'ajustement

✓ **Comparaison entre les deux cas (Normal, Pneumonia)**

Tableau (III.15) : Résultat de test T pour " Kurtosis d'image" dans les deux cas

Test T des échantillons indépendants					
Etat	**Moyenne**	**Ecart type**	**T**	**Sig**	**Résultat**
Normal	-0.6334	0.3963	-10.609	< 0.001	S
Pneumonia	-0.4521	0.9078			

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la valeur de test T égale à T= - 10.609 avec niveau de signification égale < 0.001 il est inférieur à 0.05, donc il y a donc des différences statistiquement significatives dans la Kurtosis d'image entre les deux cas (Normal, Pneumonia).

Décision : Puisque la moyenne arithmétique de Kurtosis d'image est différente dans les deux cas, nous allons l'utiliser pour construire le modèle.

III.2.6. La Plage d'image (IMG- Plage)

✓ **Description statistique**

Tableau (III.16) : Description statistique pour la " Moyenne d'image" pour les deux cas

Etat	Moyenne	Ecart type	Maximum	Minimum	Plage
Normal	243.95	12.23	255	175	80
Pneumonia	218.01	22.37	255	109	146

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la moyenne arithmétique de la " Plage d'image" pour l'état " **Normal** " est égale à 243.95, avec un écart-type égal à 12.23, tandis que la plage est égale à 80 , mais la moyenne arithmétique de la " Plage d'image" pour l'état " **Pneumonia** " est égale à 218.01, avec un écart-type égal à 22.37, tandis que la plage est égale à 146.

✓ **Description graphique**

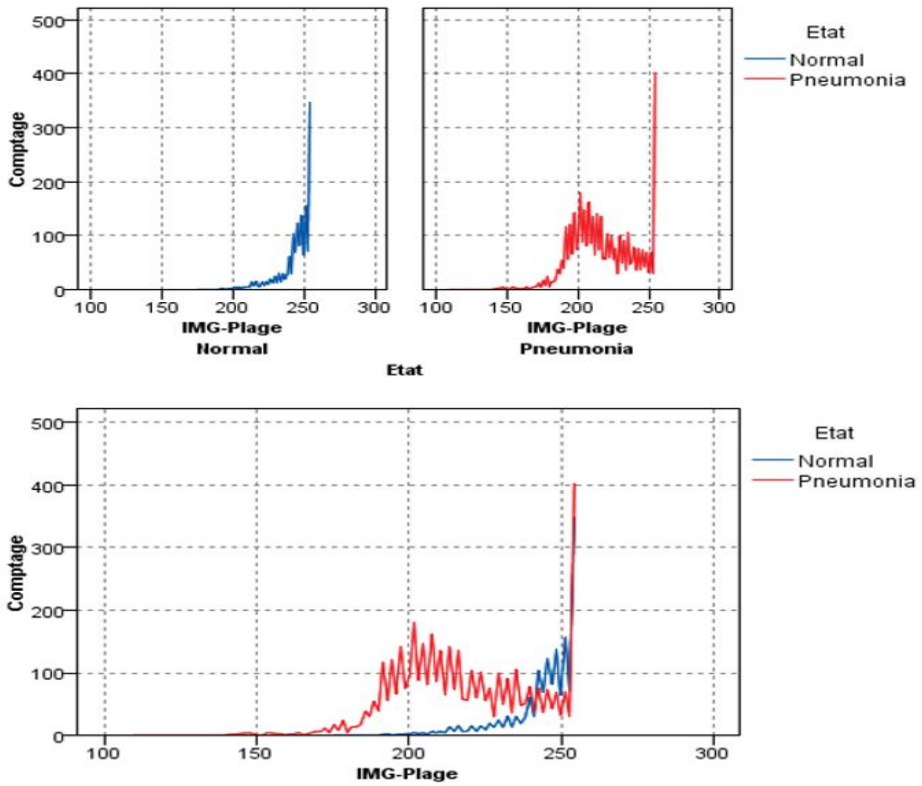


Figure (III.12) : la distribution " Plage d'image"
Pour les deux cas (Normal, Pneumonia) avant l'ajustement

✓ **Distribution (IMG- Plage)**

Tableau (III.17) : Distribution " Plage d'image" pour les deux cas

Critère de la qualité d'ajustement Kolmogrov-Simirnov			
Etat	**Statistique d'ajustement**	**Paramétré**	**Distribution**
Normal	A=39.85 P=0.01 (K=0.11 P=0.01)	a=248.79 b=32 c=0.0	Weibull
Pneumonia	A=30.12 P=0.0 (K=0.06 P=0.01)	a=216.85 b=0.1	Lognormale

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

D'après le tableau, nous remarquons que la distribution " Plage d'image" pour le cas" **Normal** " est Weibull et la distribution " Plage d'image" pour le cas" **Pneumonia** " est Lognormale.

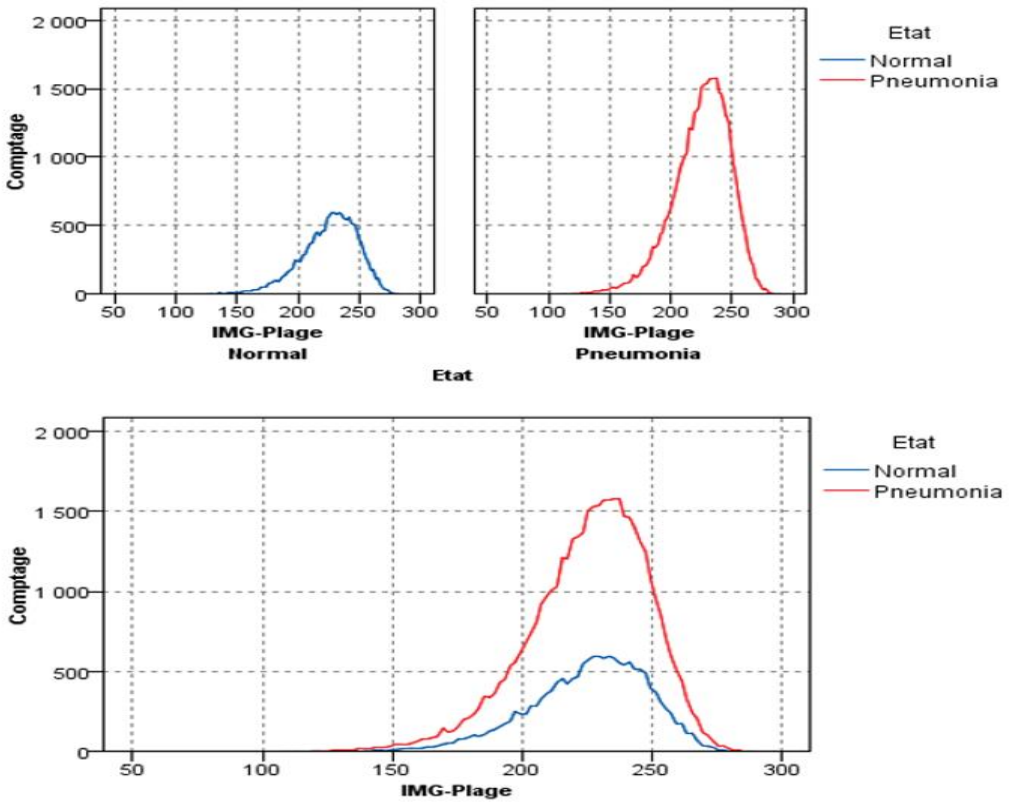


Figure (III.13) : la distribution " Plage d'image"
Pour les deux cas (Normal, Pneumonia) après l'ajustement

✓ **Comparaison entre les deux cas (Normal, Pneumonia)**

Tableau (III.18) : Résultat de test Mann-Whitney pour " Plage d'image" dans les deux cas

Test de Mann-Whitney					
Etat	**Rang moyen**	**Somme des rangs**	**U**	**Sig**	**Résulta**
Normal	4303.02	6811687.50	1206207.50	< 0.001	S
Pneumonia	2419.29	10337608.50			

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la valeur du test *Mann-Whitney* est égale à U=1206207.50 avec un niveau de signification < 0.001, qui est inférieur à 0.05, il y a donc des différences statistiquement significatives dans la Plage d'image entre les deux cas (normal, pneumonie).

Décision : Puisque la moyenne arithmétique de la Plage moyenne est différente dans les deux cas, nous allons l'utiliser pour construire le modèle.

III.2.7. Percentile 10 d'image (IMG- Percentile10)

✓ **Description statistique**

Tableau (III.19) : Description statistique pour la " Percentile 10 d'image" pour les deux cas

Etat	**Moyenne**	**Ecart type**	**Maximum**	**Minimum**	**Plage**
Normal	23.88	23.03	113	00	113
Pneumonia	34.33	27.69	194	00	194

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la moyenne arithmétique de la " Percentile 10 d'image" pour l'état " **Normal** " est égale à 23.88, avec un écart-type égal à 23.03, tandis que la plage est égale à 113, mais la moyenne arithmétique de la " Percentile 10 d'image" pour l'état " **Pneumonia** " est égale à 34.33, avec un écart-type égal à 27.69, tandis que la plage est égale à 194.

✓ **Description graphique**

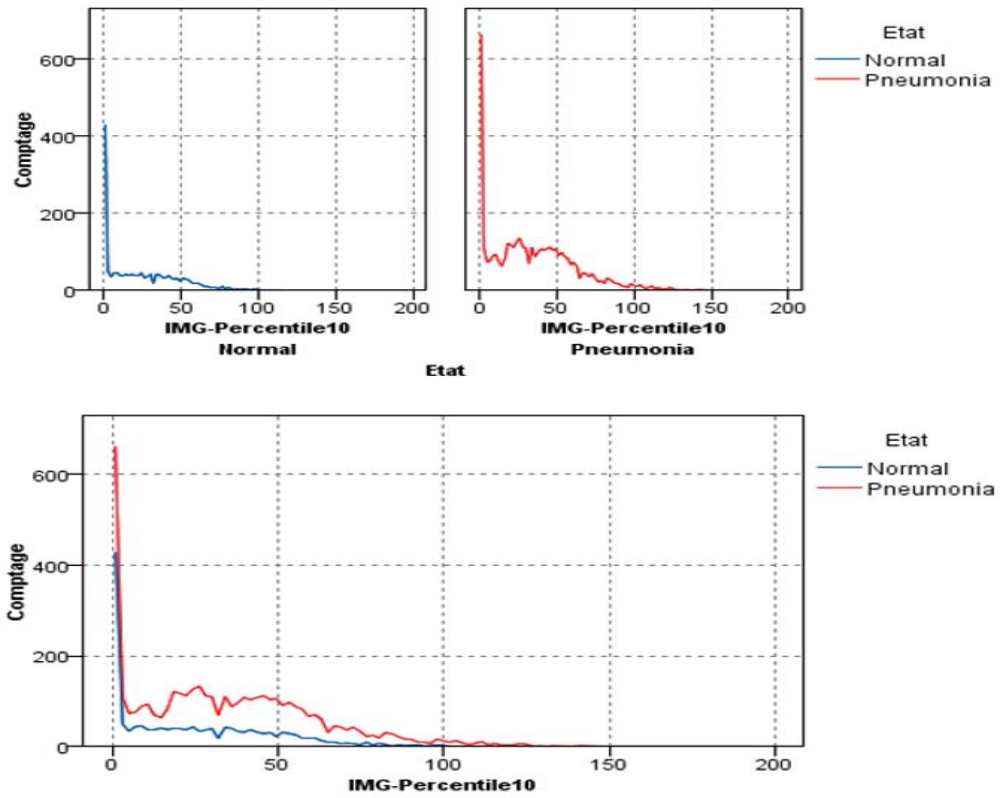


Figure (III.14) : la distribution " Percentile 10 d'image"
Pour les deux cas (Normal, Pneumonia) avant l'ajustement

✓ **Distribution (IMG- Percentile10)**

Tableau (III.20) : Distribution " Percentile 10 d'image" pour les deux cas

Critère de la qualité d'ajustement Kolmogrov-Simirnov			
Etat	**Statistique d'ajustement**	**Paramétré**	**Distribution**
Normal	A=-235.12 P=1.0 (K=0.26 P=0.00)	Scale=0.04	Exponentielle
Pneumonia	A=-177.98 P=0.0 (K=0.14 P=0.00)	Scale=0.03	Exponentielle

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

D'après le tableau, nous remarquons que la distribution " Percentile 10 d'image" pour le cas" **Normal** " est Exponentielle et la distribution " Percentile 10 d'image" pour le cas" **Pneumonia** " est aussi Exponentielle.

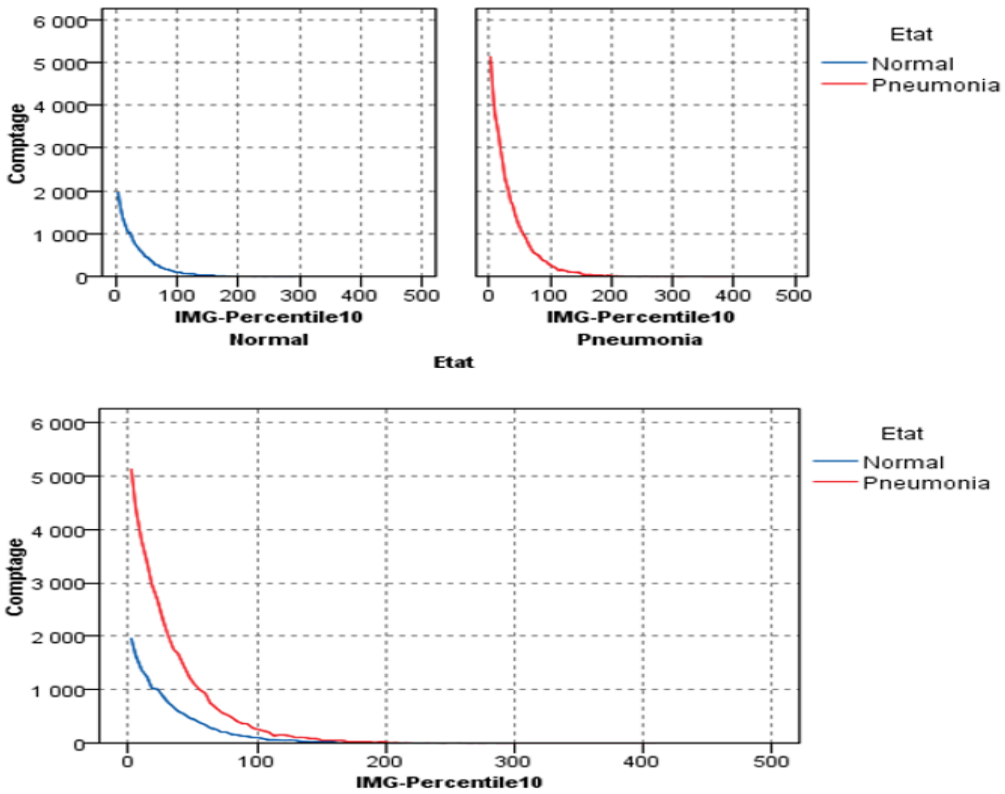


Figure (III.15) : la distribution " Percentile 10 d'image"
Pour les deux cas (Normal, Pneumonia) après l'ajustement

✓ **Comparaison entre les deux cas (Normal, Pneumonia)**

Tableau (III.21) : Résultat de test Mann-Whitney pour " Percentile 10 d'image" dans les deux cas

Test de Mann-Whitney					
Etat	**Rang moyen**	**Somme des rangs**	**U**	**Sig**	**Résulta**
Normal	2450.84	3879677.50	2625941.50	< 0.001	S
Pneumonia	3105.46	13269618.50			

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la valeur du test *Mann-Whitney* est égale à U=2625941.50 avec un niveau de signification < 0.001, qui est inférieur à 0.05, il y a donc des différences statistiquement significatives dans la Percentile 10 d'image entre les deux cas (normal, pneumonie).

Décision : Puisque la moyenne arithmétique de Percentile 10 d'image est différente dans les deux cas, nous allons l'utiliser pour construire le modèle.

III.2.8. Percentile 25 d'image (IMG- Percentile25)

✓ **Description statistique**

Tableau (III.22) : Description statistique pour la " Percentile 25 d'image" pour les deux cas

Etat	Moyenne	Ecart type	Maximum	Minimum	Plage
Normal	81.82	18.87	142	00	142
Pneumonia	86.16	28.34	212	00	212

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la moyenne arithmétique de la " Percentile 25 d'image" pour l'état " **Normal** " est égale à 81.82, avec un écart-type égal à 18.87, tandis que la plage est égale à 142, mais la moyenne arithmétique de la " Percentile 25 d'image" pour l'état " **Pneumonia** " est égale à 86.16, avec un écart-type égal à 28.34, tandis que la plage est égale à 212.

✓ **Description graphique**

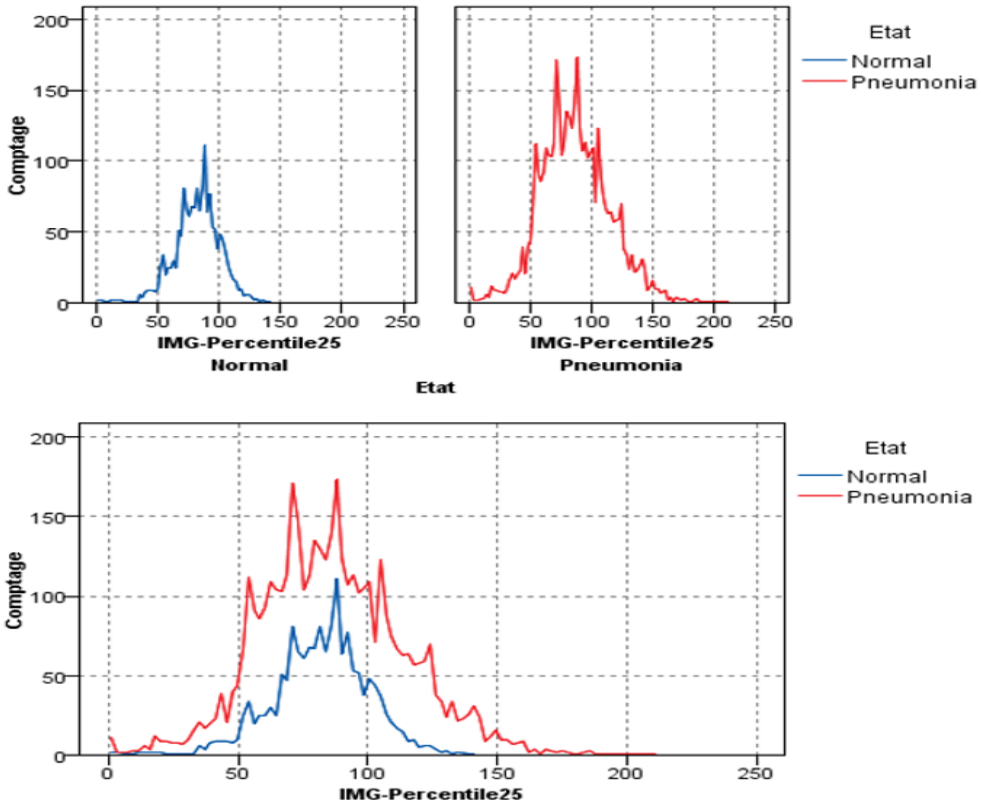


Figure (III.16) : la distribution " Percentile 25 d'image"
Pour les deux cas (Normal, Pneumonia) avant l'ajustement

- ✓ **Distribution (IMG- Percentile 25)**

Tableau (III.23) : Distribution " Percentile 25 d'image" pour les deux cas

Critère de la qualité d'ajustement Kolmogrov-Simirnov			
Etat	**Statistique d'ajustement**	**Paramétré**	**Distribution**
Normal	A=3.85 P=0.0 (K=0.04 P=0.01)	Moy=81.83 Ecart=18.86	Normale
Pneumonia	A=7.63 P=0.0 (K=0.04 P=0.01)	Moy=86.16 Ecart=28.34	Normale

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

D'après le tableau, nous remarquons que la distribution " Percentile 25 d'image" pour le cas" **Normal** " est Normale et la distribution " Percentile 25 d'image" pour le cas" **Pneumonia** " est aussi Normale.

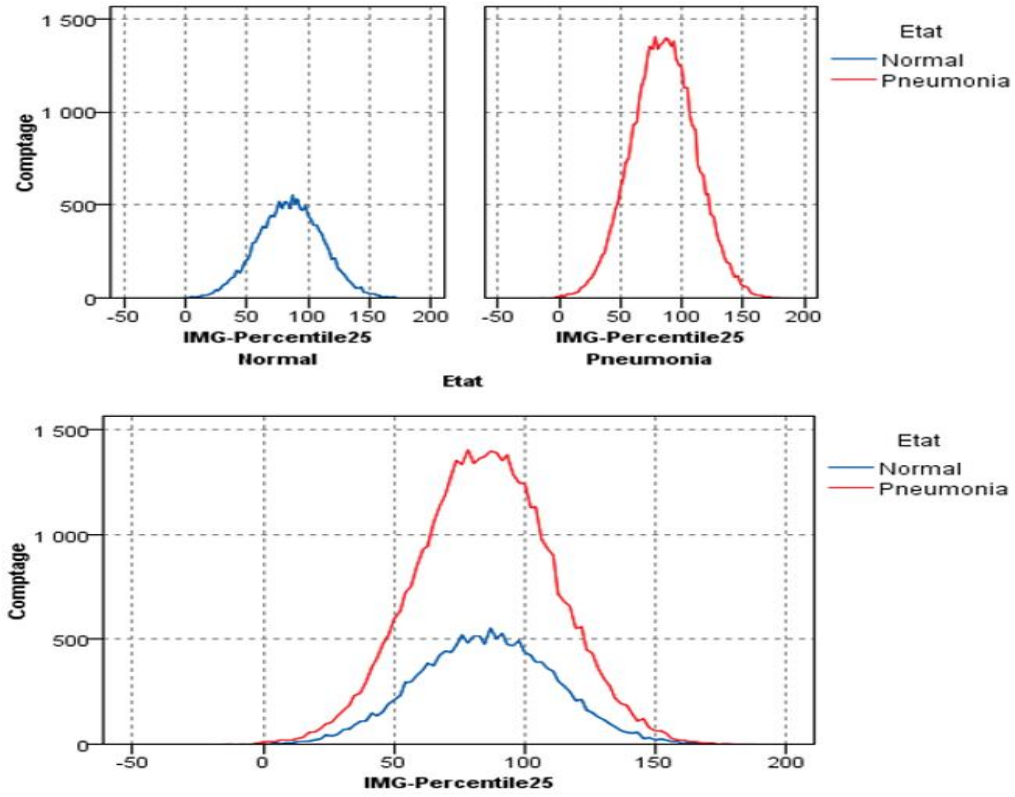


Figure (III.17) : la distribution " Percentile 25 d'image"
Pour les deux cas (Normal, Pneumonia) après l'ajustement

- ✓ **Comparaison entre les deux cas (Normal, Pneumonia)**

Tableau (III.24) : Résultat de test T pour " Percentile 25 d'image" dans les deux cas

Test T des échantillons indépendants					
Etat	**Moyenne**	**Ecart type**	**T**	**Sig**	**Résultat**
Normal	81.82	18.87	-6.743	< 0.001	S
Pneumonia	86.16	28.34			

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la valeur de test T égale à T= - 6.743 avec niveau de signification égale < 0.001 il est inférieur à 0.05, donc il y a donc des différences statistiquement significatives dans la Percentile 25 d'image entre les deux cas (Normal, Pneumonia).

Décision : Puisque la moyenne arithmétique de Percentile 25 d'image est différente dans les deux cas, nous allons l'utiliser pour construire le modèle.

III.2.9. Percentile 75 d'image (IMG- Percentile75)

- ✓ **Description statistique**

Tableau (III.25) : Description statistique pour la " Percentile 75 d'image" pour les deux cas

Etat	**Moyenne**	**Ecart type**	**Maximum**	**Minimum**	**Plage**
Normal	172.10	13.95	225	97	128
Pneumonia	168.71	21.52	235	70	165

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la moyenne arithmétique de la " Percentile 75 d'image" pour l'état " **Normal** " est égale à 172.10, avec un écart-type égal à 13.95, tandis que la plage est égale à 128, mais la moyenne arithmétique de la " Percentile 75 d'image" pour l'état " **Pneumonia** " est égale à 168.71, avec un écart-type égal à 21.52, tandis que la plage est égale à 165.

✓ **Description graphique**

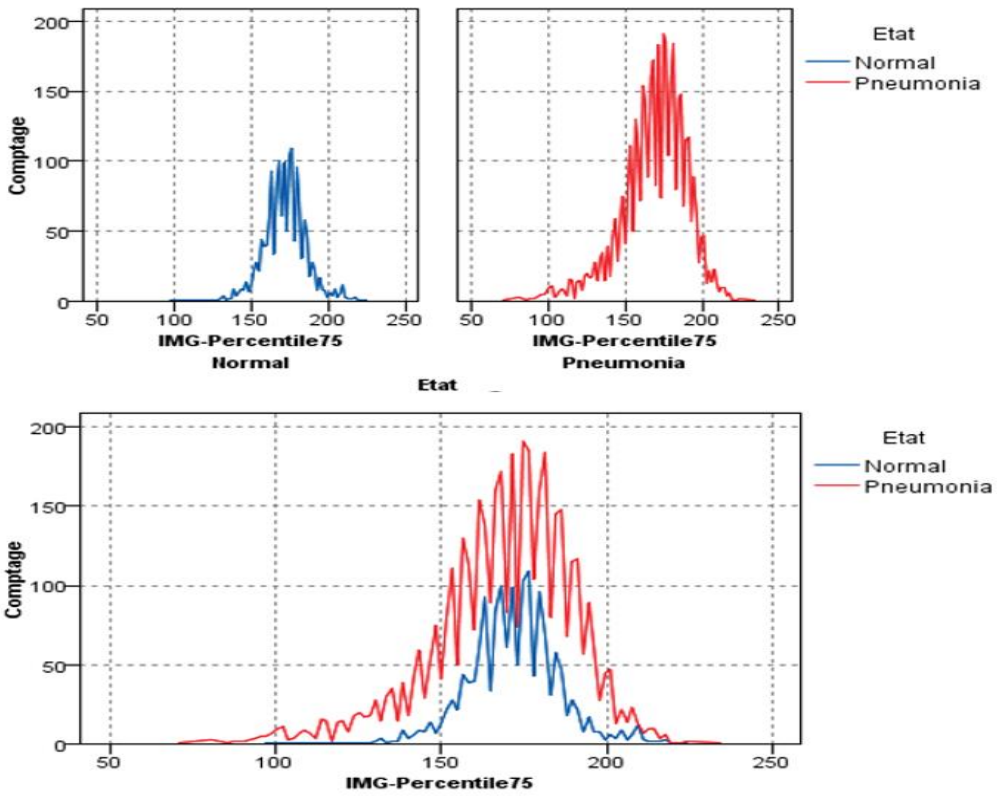


Figure (III.18) : la distribution " Percentile 75 d'image"
Pour les deux cas (Normal, Pneumonia) avant l'ajustement

✓ **Distribution (IMG- Percentile75)**

Tableau (III.26) : Distribution " Percentile 75 d'image" pour les deux cas

Critère de la qualité d'ajustement Kolmogrov-Simirnov			
Etat	**Statistique d'ajustement**	**Paramétré**	**Distribution**
Normal	A=6.0 P=0.0 (K=0.05 P=0.01)	Moy=172.1 Ecart=13.95	Normale
Pneumonia	A=4.51 P=0.01 (K=0.03 P=0.01)	a=177.67 b=9.53 c=0.0	Weibull

Source :préparation de l'étudiant à l'aide IBM SPSS V 28.0

D'après le tableau, nous remarquons que la distribution " Percentile 75 d'image" pour le cas" **Normal** " est Normale et la distribution " Percentile 75 d'image" pour le cas" **Pneumonia** " est Weibull.

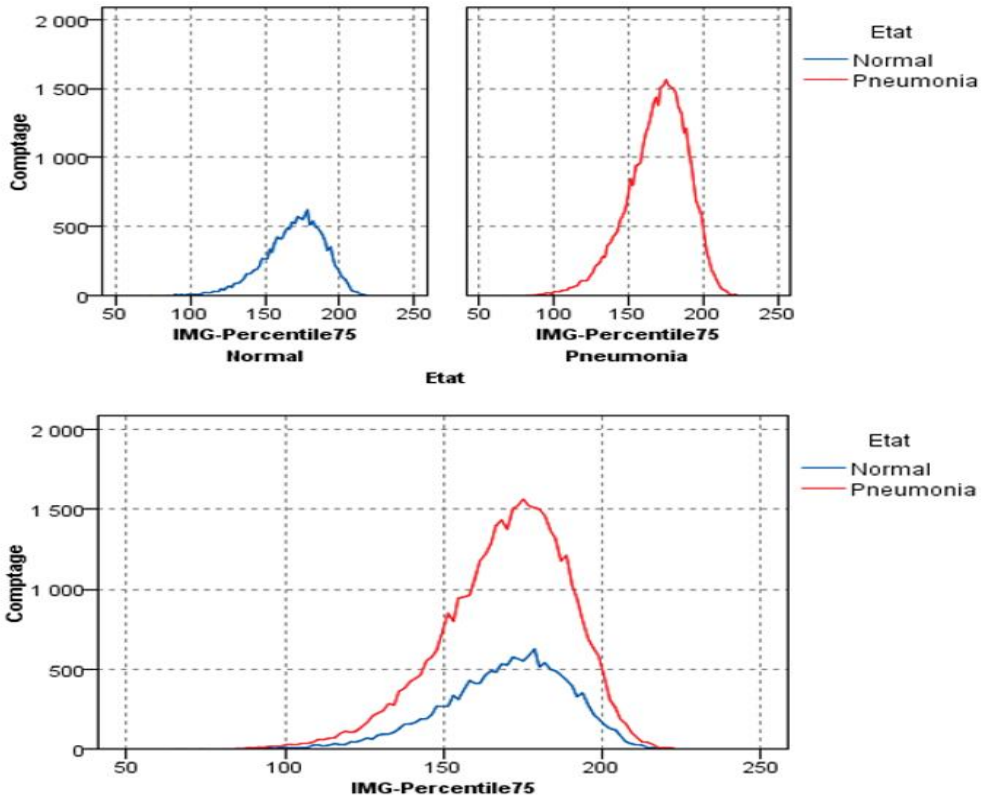


Figure (III.19) : la distribution " Percentile 75 d'image"
Pour les deux cas (Normal, Pneumonia) après l'ajustement

✓ **Comparaison entre les deux cas (Normal, Pneumonia)**

Tableau (III.27) : Résultat de test Mann-Whitney pour " Percentile 75 d'image" dans les deux cas

Test de Mann-Whitney					
Etat	**Rang moyen**	**Somme des rangs**	**U**	**Sig**	**Résulta**
Normal	3028.82	4794614.50	3223280.50	0.006	S
Pneumonia	2891.34	12354681.50			

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la valeur du test *Mann-Whitney* est égale à U=3223280.50 avec un niveau de signification 0.006, qui est inférieur à 0.05, il y a donc des différences statistiquement significatives dans la Percentile 75 d'image entre les deux cas (normal, pneumonie).

Décision : Puisque la moyenne arithmétique de Percentile 75 d'image est différente dans les deux cas, nous allons l'utiliser pour construire le modèle.

III.2.10. Percentile 90 d'image (IMG- Percentile90)

✓ **Description statistique**

Tableau (III.28) : Description statistique pour la " Percentile 90 d'image" pour les deux cas

Etat	Moyenne	Ecart type	Maximum	Minimum	Plage
Normal	195.35	13.62	247	121	126
Pneumonia	185.42	18.79	241	92	149

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la moyenne arithmétique de la " Percentile 90 d'image" pour l'état " **Normal** " est égale à 195.35, avec un écart-type égal à 13,62, tandis que la plage est égale à 126, mais la moyenne arithmétique de la " Percentile 90 d'image" pour l'état " **Pneumonia** " est égale à 185.42, avec un écart-type égal à 18.79, tandis que la plage est égale à 149.

✓ **Description graphique**

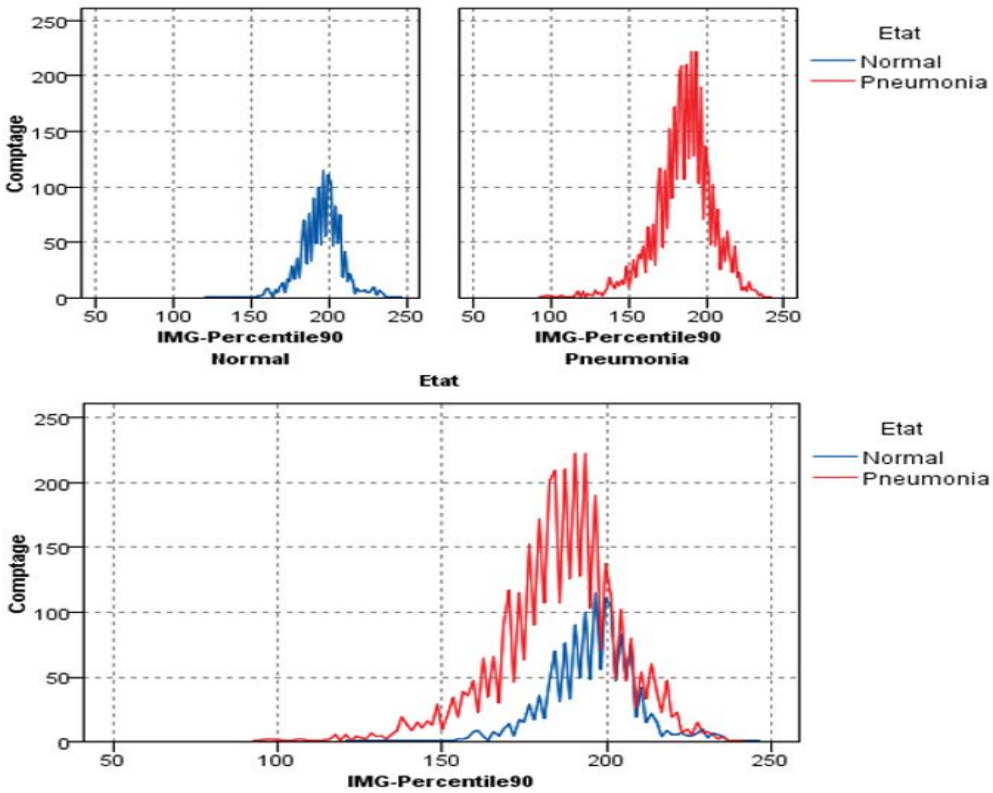


Figure (III.20) : la distribution " Percentile 90 d'image"
Pour les deux cas (Normal, Pneumonia) avant l'ajustement

✓ **Distribution (IMG- Percentile 90)**

Tableau (III.29) : Distribution " Percentile 90 d'image" pour les deux cas

Critère de la qualité d'ajustement Kolmogrov-Simirnov			
Etat	**Statistique d'ajustement**	**Paramétré**	**Distribution**
Normal	A=7.7 P=0.0 (K=0.06 P=0.01)	Moy=195.36 Ecart=13.62	Normale
Pneumonia	A=21.95 P=0.01 (K=0.06 P=0.01)	a=193.58 b=11.28 c=0.0	Weibull

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

D'après le tableau, nous remarquons que la distribution " Percentile 90 d'image" pour le cas" **Normal** " est Normale et la distribution " Percentile 90 d'image" pour le cas" **Pneumonia** " est Weibull.

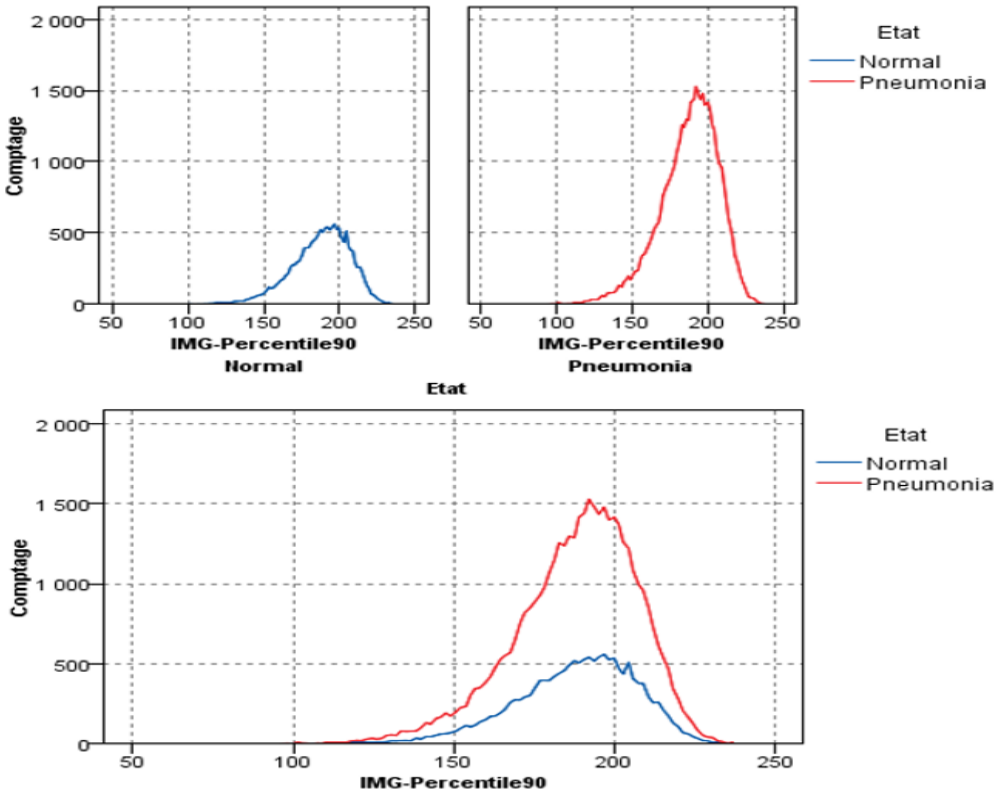


Figure (III.21) : la distribution " Percentile 90 d'image"
Pour les deux cas (Normal, Pneumonia) après l'ajustement

- ✓ **Comparaison entre les deux cas (Normal, Pneumonia)**

Tableau (III.30) : Résultat de test Mann-Whitney pour " Percentile 90 d'image" dans les deux cas

Test de Mann-Whitney					
Etat	**Rang moyen**	**Somme des rangs**	**U**	**Sig**	**Résulta**
Normal	3661.09	5795504.50	2222390.50	< 0.001	S
Pneumonia	2657.10	11353791.50			

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

A travers le tableau, on remarque que la valeur du test *Mann-Whitney* est égale à U=2222390.50 avec un niveau de signification < 0.001, qui est inférieur à 0.05, il y a donc des différences statistiquement significatives dans la Percentile 90 d'image entre les deux cas (normal, pneumonie).

Décision : Puisque la moyenne arithmétique de Percentile 90 d'image est différente dans les deux cas, nous allons l'utiliser pour construire le modèle.

Enfin, nous pouvons dire que les paramètres qui seront adoptés dans la construction du modèle sont : Médian, Ecart type, Asymétrie, Kurtosis, Plage, Percentile 10, Percentile 25, Percentile 75 et Percentile 90.

III.3. Construire des modèles

Dans cette partie, nous allons construire différents modèles, afin que nous puissions choisir le meilleur modèle à utiliser pour prédire l'état (normal, pneumonie).

III.3.1. Modèle k plus proches voisins (KNN)

III.3.1.1. Création du classificateur

En utilisant ce fleux dans SPSS Modeler v18.0, nous générons le classificateur KNN pour *(K=1,2,3,4,5,6,7,8,9,10)*, afin de choisir le meilleur classificateur.

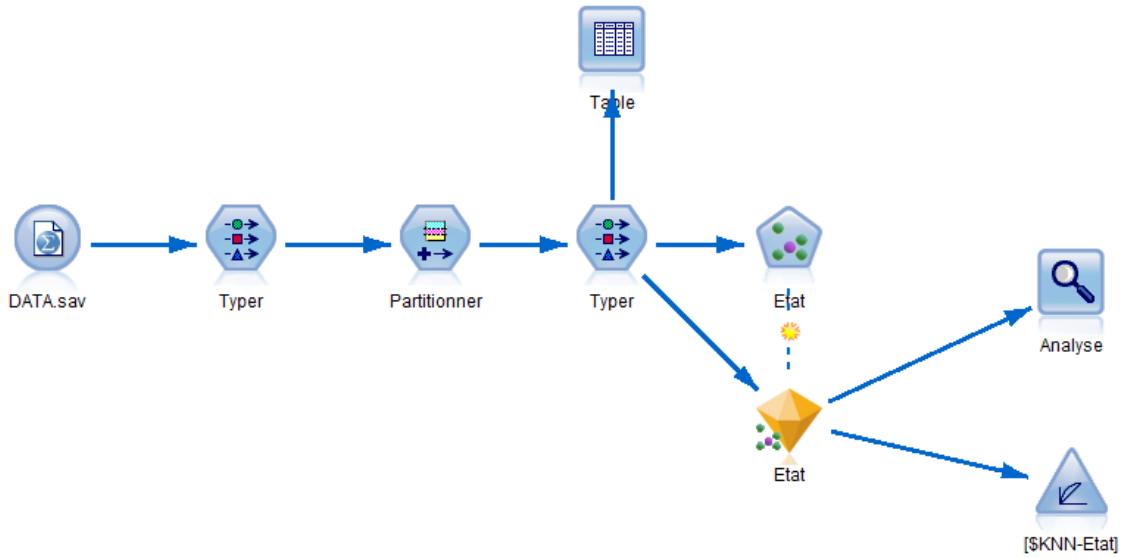


Figure (III.22) : Fleux du classificateur KNN

La figure (III.23) suivante représente la fenêtre Propriétés du classificateur pour sélectionner le meilleur k.

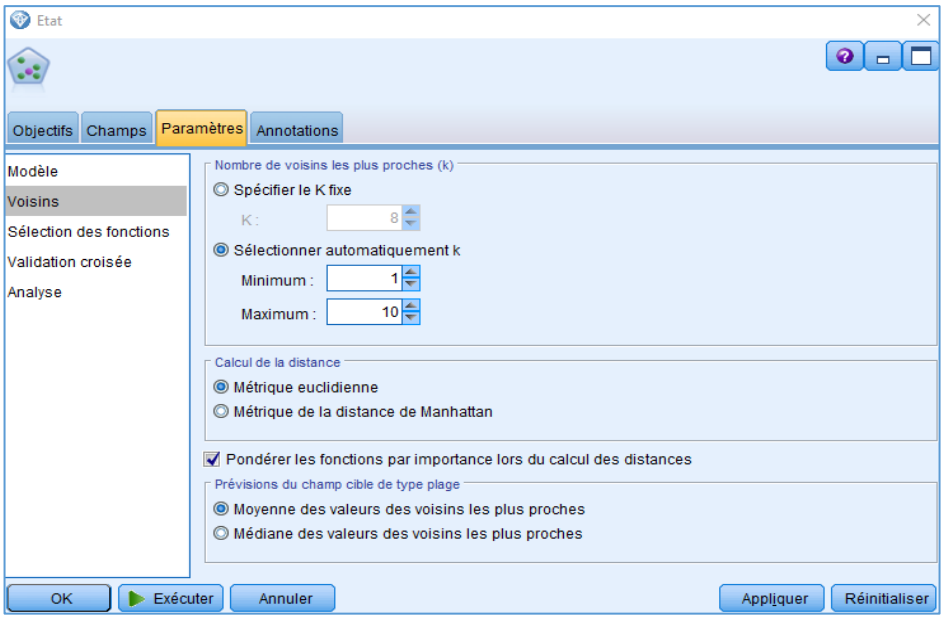


Figure (III.23) : Fenêtre de propriétés du classificateur KNN

Après traitement, nous avons trouvé que le meilleur classifieur qui donne la plus petite valeur d'erreur est le classifieur KNN pour K=8, et le tableau correspondant montre la valeur d'erreur a fonction nombre k.

Tableau (III.31) : Taux d'erreur a fonction nombre k pour la classificateur KNN

Taux d'erreur a fonction nombre k										
Nombre de K	1	2	3	4	5	6	7	8	9	10
Taux d'erreur	0.194	0.188	0.172	0.169	0.158	0.163	0.157	0.155	0.156	0.158

La figure (III.24) correspondante le montre

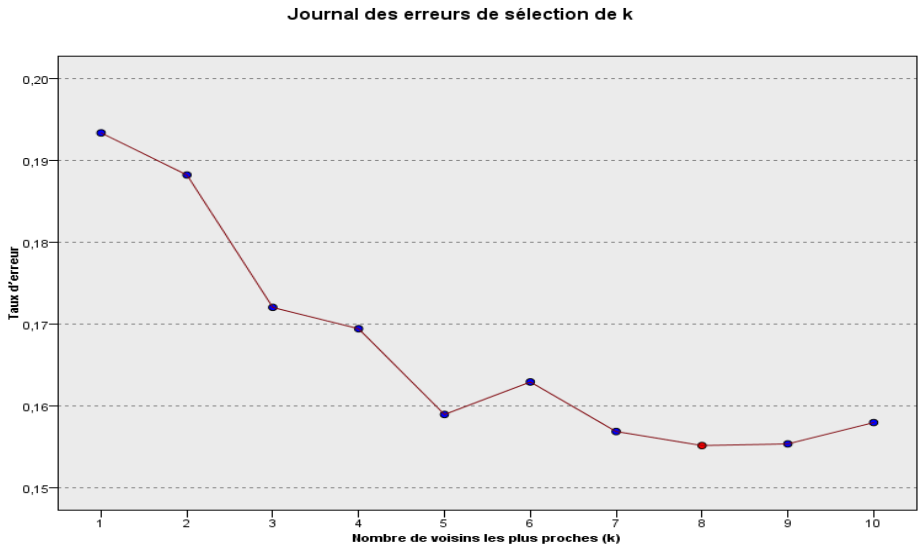


Figure (III.24) : Journal des erreurs pour (K=1,2,3,4,5,6,7,8,9,10)

III.3.1.2. Résultats du classificateur

- ✓ **Récapitulatif du classificateur KNN**

Tableau (III.32) : Récapitulatif du classificateur KNN

Les Caractéristiques modèle KNN	
Nombre de K	08
Calcule de distance	Métrique euclidienne

La figure (III.25) correspondante indique espace du prédicteur dans la classificateur KNN

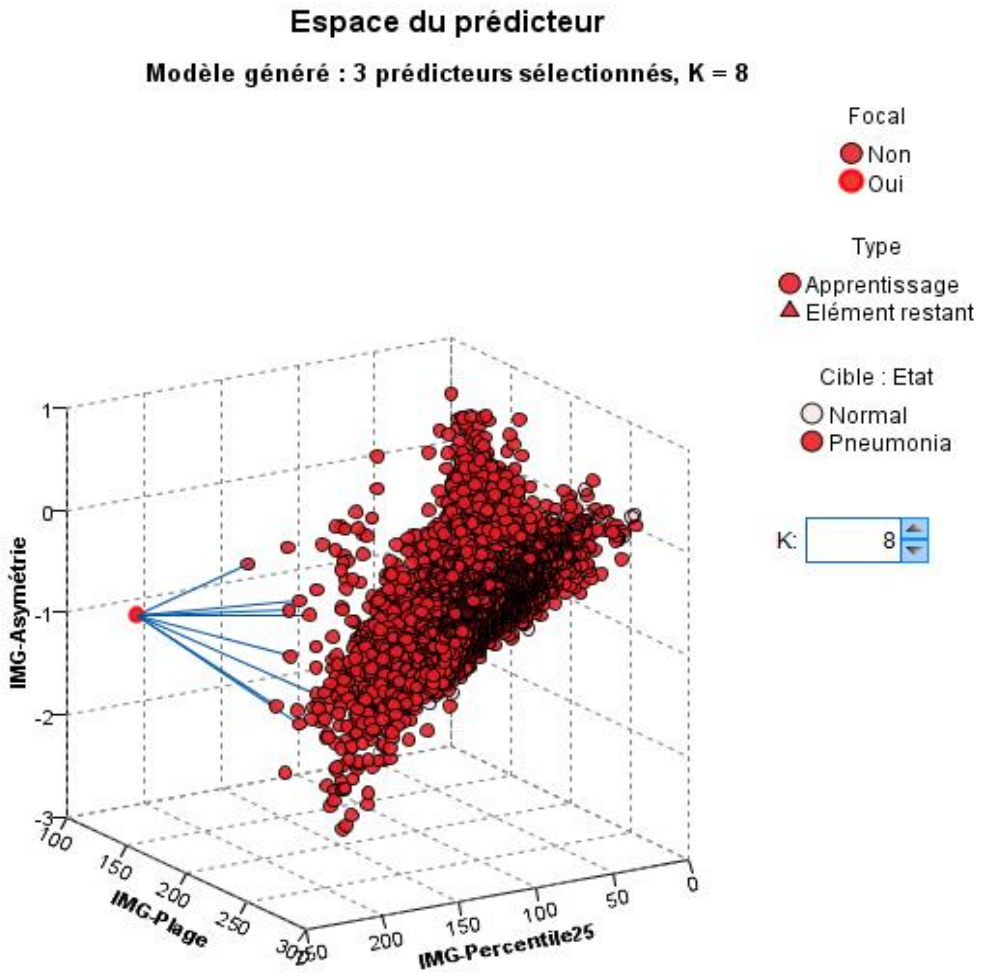


Figure (III.25) : Espace du prédicteur dans la classificateur KNN

La figure (III.26) correspondante indique graphique des homologues dans la classificateur KNN

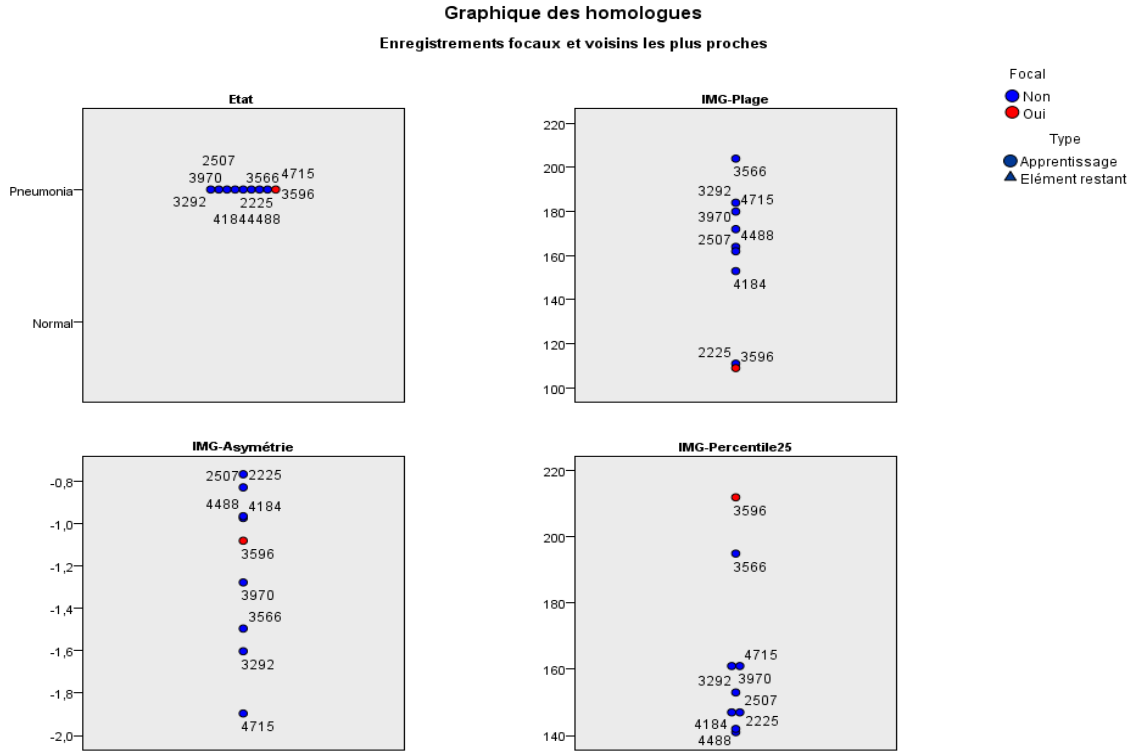


Figure (III.26) : graphique des homologues dans la classificateur KNN

- ✓ **L'importance des prédicteurs du classificateur KNN**

Tableau (III.33) : l'importance des prédicteurs du classificateur KNN

Les Paramètres	**Importance %**
IMG-Median	10,94
IMG-Ecart type	11,01
IMG-Asymétrie	11,27
IMG-Kurtosis	11,04
IMG-Plage	11,65
IMG-Percentile10	10,77
IMG-Percentile25	11,22
IMG-Percentile75	11,16
IMG-Percentile90	10,94

Source : préparation de l'étudiant à l'aide IBM SPSS V 28.0

Dans le tableau suivant qui représente là l'importance des prédicteurs du classificateur KNN on remarque que tous les paramètres entrés dans la construction du modèle ont presque la même importance de sorte que :

- L'importance **IMG-Médian** est (10,94 %)
- L'importance **IMG-Ecart type** est (11,01 %)
- L'importance **IMG-Asymétrie** est (11,27 %)
- L'importance **IMG-Kurtosis** est (11,04 %)
- L'importance **IMG-Plage** est (11,65 %)
- L'importance **IMG-Percentile10** est (10,77 %)
- L'importance **IMG-Percentile25** est (11,22 %)
- L'importance **IMG-Percentile75** est (11,16 %)
- L'importance **IMG-Percentile90** est (10,94 %)

Et la figure(III.27) ci-contre montre que

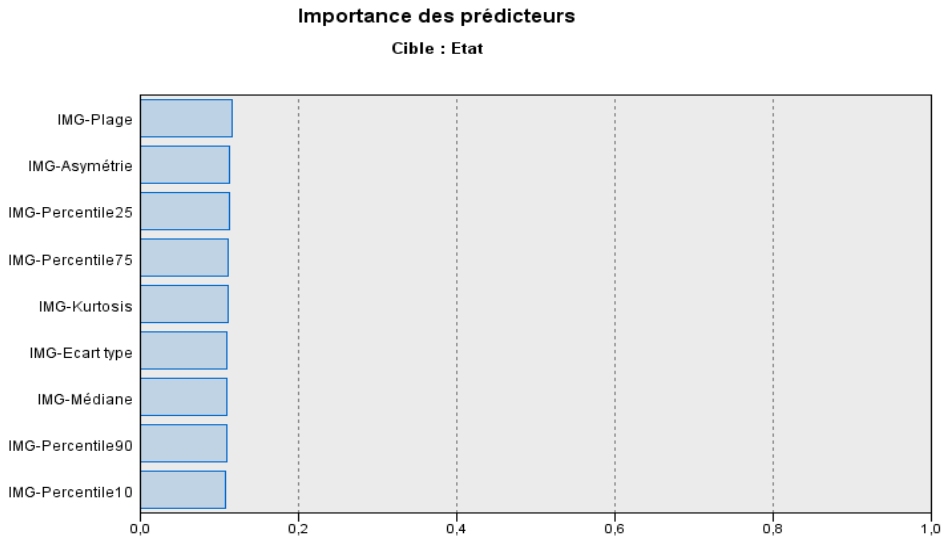


Figure (III.27) : l'importance des prédicteurs du classificateur KNN

- ✓ **Matrice de confusion du classificateur KNN**

Tableau (III.34) : Matrice de confusion du classificateur KNN

	Prédite		
Actual	**Etat**	**Pneumonia**	**Normal**
	Pneumonia	366	25
	Normal	49	91

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

D'après le tableau, nous remarquons que la classification correcte est **457** répartis en deux cas **TP = 366** et **TN = 91**, mais la classification incorrecte est **74** répartis en deux cas **FP = 25** et **FN = 49**.

- ✓ **Evaluation du classificateur KNN**

- o **Présentation numérique**

Tableau (III.35) : La précision du classificateur KNN

Précision du Modèle								
Accuracy d'Apprentissage : 87.5%				**Accuracy de validation : 91.3%**				
Test								
Accuracy	**Précision**	**Sensibilité**	**Spécificité**	**Nég-Prédictif**	**F-Score**	**AUC**	**Gini**	**Heure de création**
86.0%	88.1%	93.6%	65.0%	78.4%	90.6%	91.3%	82.6%	12s

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Du tableau, nous remarquons ce qui suit :

- **Accuracy d'apprentissage** est 87.5% et **Accuracy de validation** est 91.3%
- **Accuracy** de test est 86.0%
- **Précision** de test est 88.1%
- **Sensibilité** de test est 93.6%
- **Spécificité** de test est 65.0%
- **Nég-Prédictif** de test est 78.4%
- **F-Score** de test est 90.6%
- **AUC** de test est 91.3%
- **Gini** de test est 82.6%
- **Heure de création** est 12s

- **Présentation graphique**

La figure (III.28) suivante qui représente : roc du classificateur KNN (Apprentissage, Validation et Test).

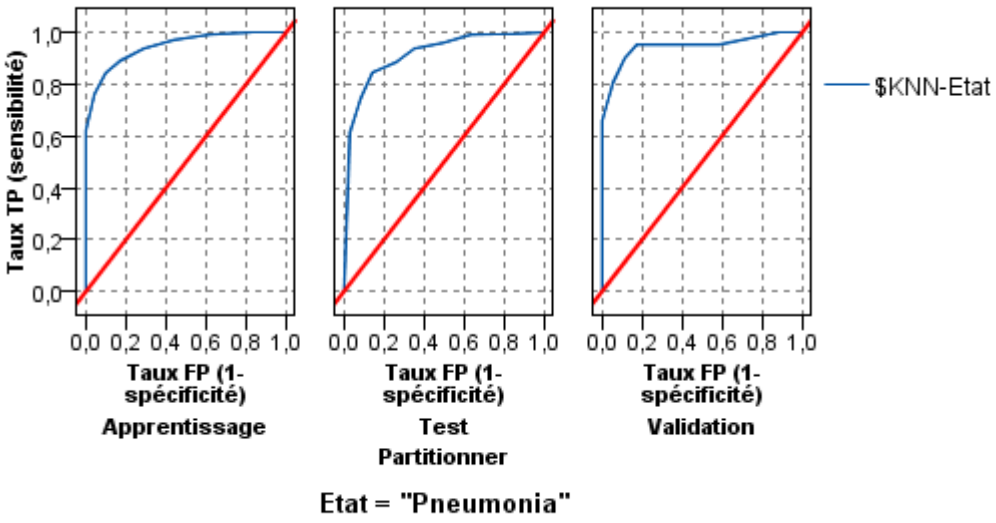


Figure (III.28) : ROC du classificateur KNN

La figure (III.29) suivante qui représente : correspondance résultats actual et prédite du classificateur KNN

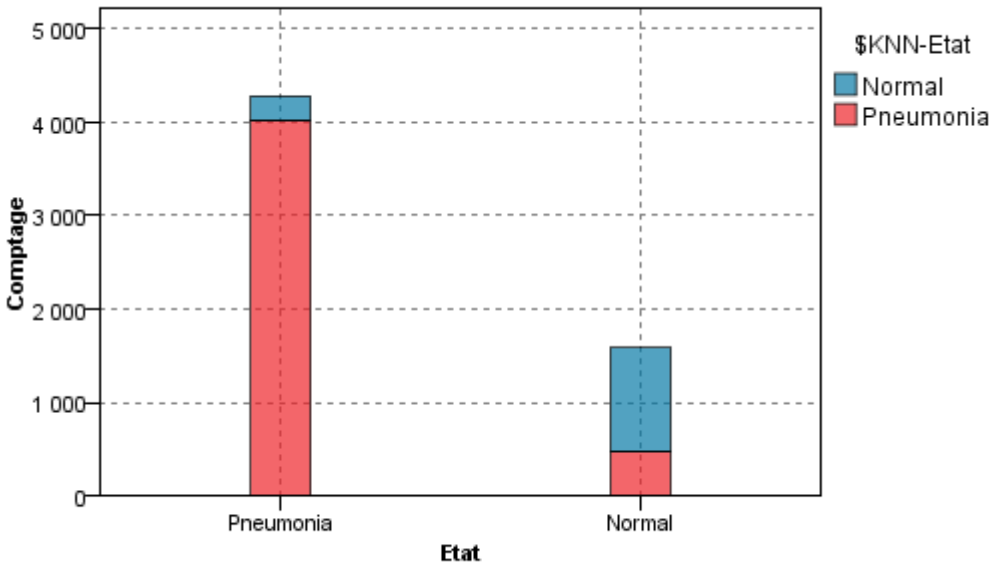


Figure (III.29) Correspondance résultats actual et prédite du classificateur KNN

III.3.2. Modèle Machine à vecteurs de support (SVM)

III.3.2.1. Création du classificateur

En utilisant ce fleux dans SPSS Modeler v18.0, nous générons le classificateur SVM

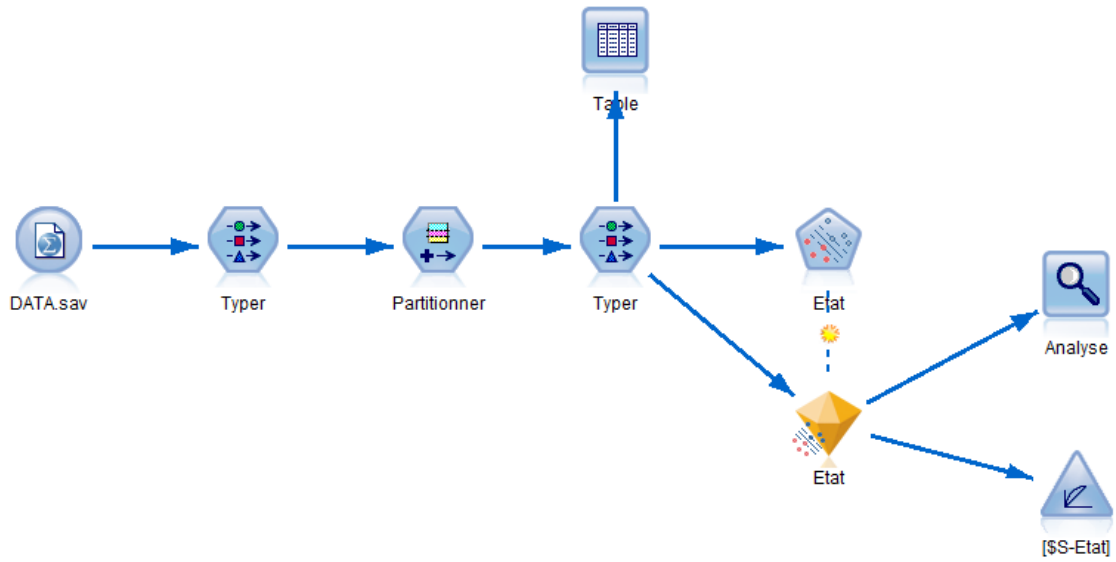


Figure (III.30) : Fleux du classificateur SVM

III.3.2.2. Résultats du classificateur

- ✓ **L'importance des prédicteurs du classificateur SVM**

Tableau (III.36) : l'importance des prédicteurs du classificateur SVM

Les Paramètres	Importance %
IMG-Median	03,90
IMG-Ecart type	07,60
IMG-Asymétrie	02,44
IMG-Kurtosis	03,50
IMG-Plage	46,28
IMG-Percentile10	02,93
IMG-Percentile25	09,08
IMG-Percentile75	01,34
IMG-Percentile90	22,94

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Dans le tableau suivant qui représente là l'importance des prédicteurs du classificateur KNN On remarque que tous les paramètres entrés dans la construction du modèle ont une importance différente de sorte que :

- L'importance **IMG-Médian** est (03,90 %)
- L'importance **IMG-Ecart type** est (07,60 %)
- L'importance **IMG-Asymétrie** est (02,44 %)
- L'importance **IMG-Kurtosis** est (03,50 %)
- L'importance **IMG-Plage** est (46,28 %)
- L'importance **IMG-Percentile10** est (02,93 %)
- L'importance **IMG-Percentile25** est (09,08 %)
- L'importance **IMG-Percentile75** est (01,34 %)
- L'importance **IMG-Percentile90** est (22,94 %)

Et la figure (III.31) ci-contre montre que

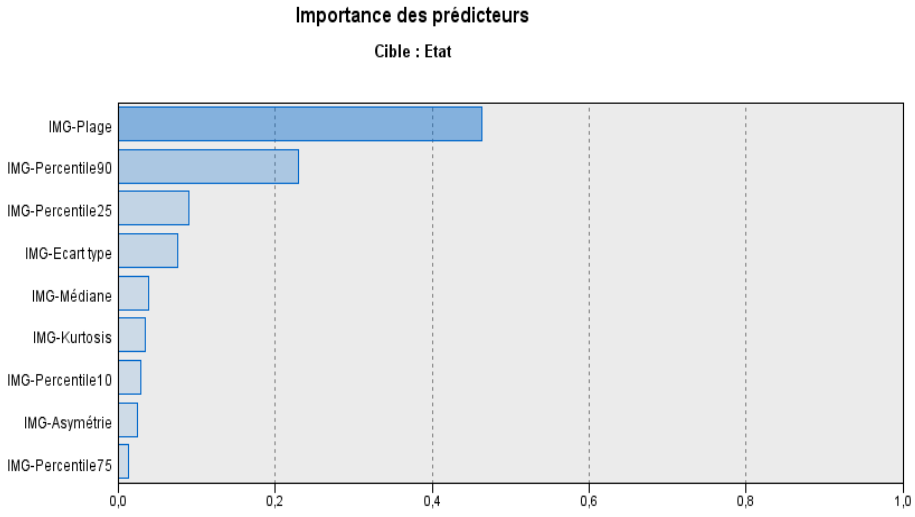


Figure (III.31) : l'importance des prédicteurs du classificateur SVM

- ✓ **Matrice de confusion du classificateur SVM**

Tableau (III.37) : Matrice de confusion du classificateur SVM

	Prédite		
Actual	**Etat**	**Pneumonia**	**Normal**
	Pneumonia	361	30
	Normal	44	96

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

D'après le tableau, nous remarquons que la classification correcte est **457** répartis en deux cas **TP = 361** et **TN = 96**, mais la classification incorrecte est **74** répartis en deux cas **FP = 30** et **FN = 44**.

- ✓ **Evaluation du classificateur SVM**
  - o **Présentation numérique**

Tableau (III.38) : La précision du classificateur SVM

Précision du Modèle								
Accuracy d'Apprentissage : 84.6%				**Accuracy de validation : 89.6%**				
Test								
Accuracy	**Précision**	**Sensibilité**	**Spécificité**	**Nég-Prédictif**	**F-Score**	**AUC**	**Gini**	**Heure de création**
86.0%	89.1%	92.3%	68.5%	76.1%	90.6%	91.3%	82.6%	12s

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Du tableau, nous remarquons ce qui suit :

- **Accuracy d'apprentissage** est 84.6% et **Accuracy de validation** est **89.6%**
- **Accuracy** de test est 86.0%
- **Précision** de test est 89.1%
- **Sensibilité** de test est 92.3%
- **Spécificité** de test est 68.5%
- **Nég-Prédictif** de test est 76.1%
- **F-Score** de test est 90.6%
- **AUC** de test est 91.3%
- **Gini** de test est 82.6%
- **Heure de création** est 12s

- **Présentation graphique**

La figure (III.32) suivante qui représente : roc du classificateur SVM (Apprentissage, Validation et Test).

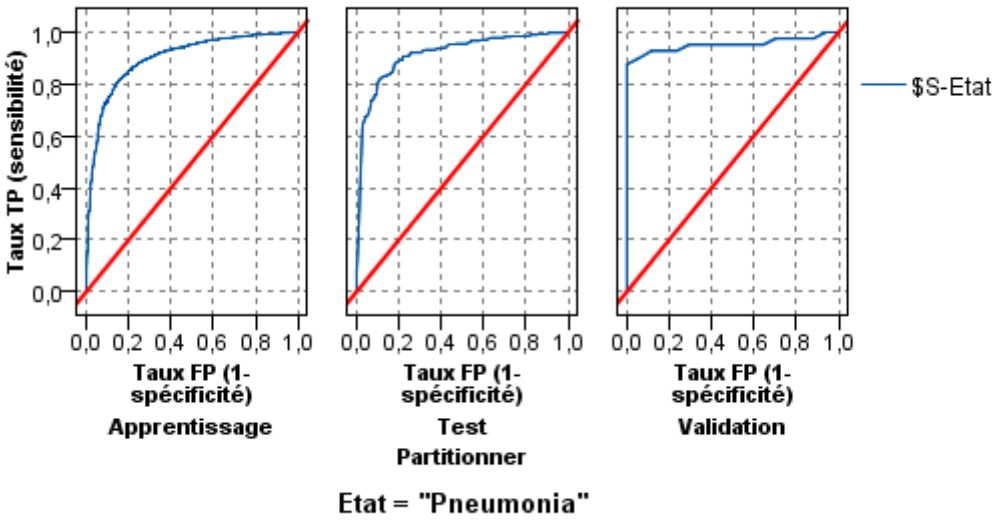


Figure (III.32) : ROC du classificateur SVM

La figure (III.33) suivante qui représente : correspondance résultats actual et prédite du classificateur SVM

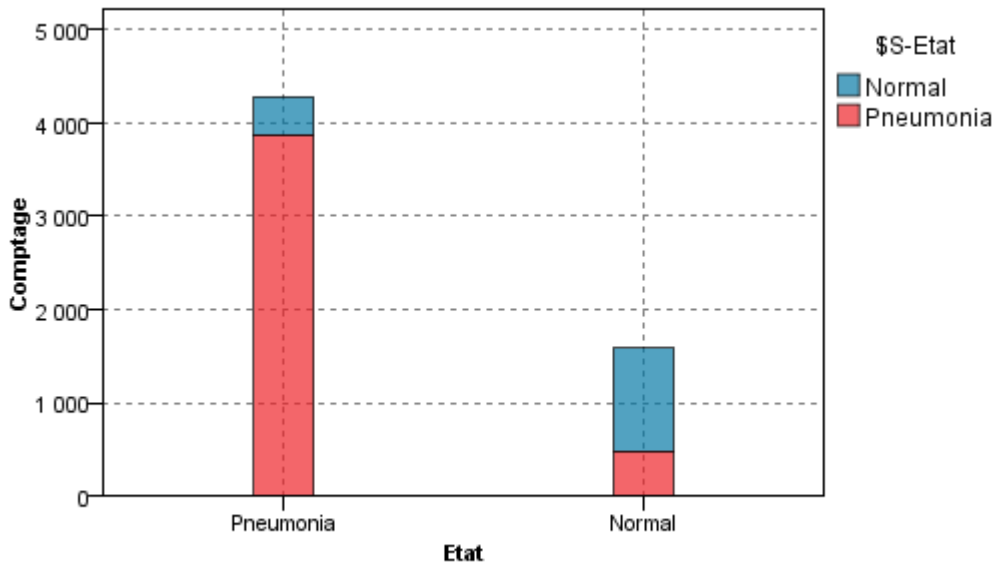


Figure (III.33) Correspondance résultats actual et prédite du classificateur SVM

III.3.3. Modèle Réseau Bayésien (RB)

III.3.3.1. Création du classificateur

En utilisant ce fleux dans SPSS Modeler v18.0, nous générons le classificateur RB

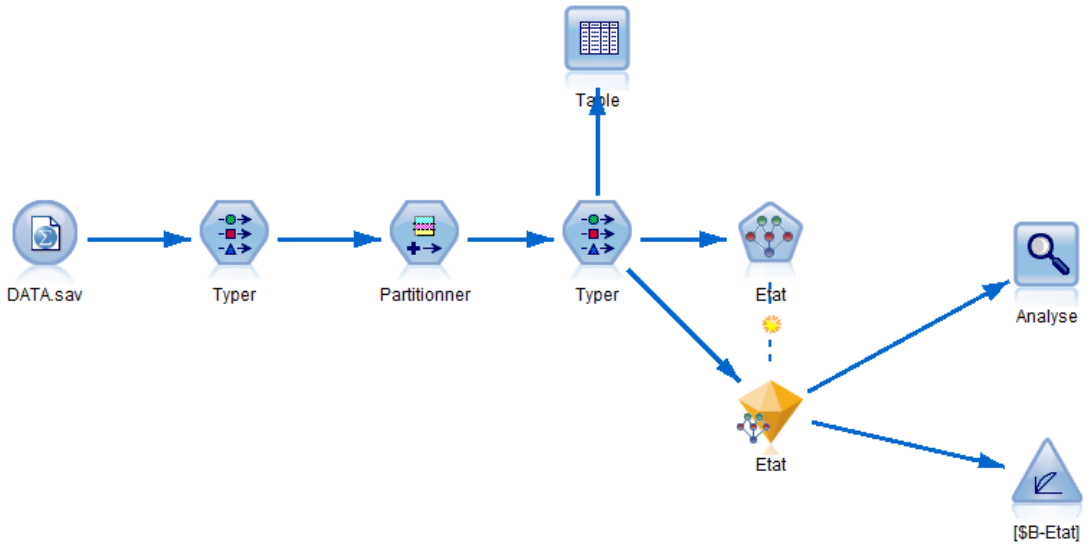


Figure (III.34) : Fleux du classificateur RB

III.3.3.2 Résultats du classificateur

- ✓ **Récapitulatif du classificateur RB**

La figure (III.35) correspondante indique Réseau bayésien RB

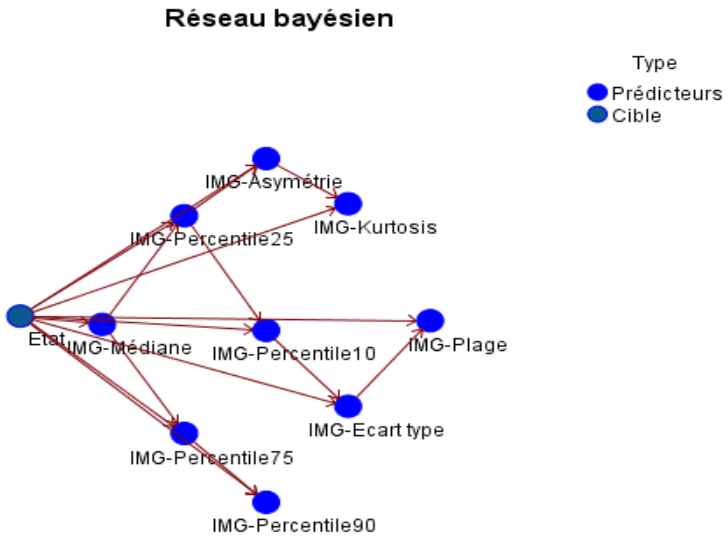


Figure (III.35) : Réseau bayésien

- ✓ **Matrice de confusion du classificateur RB**

Tableau (III.39) : Matrice de confusion du classificateur RB

	Prédite		
Actual	**Etat**	**Pneumonia**	**Normal**
	Pneumonia	331	58
	Normal	45	95

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

D'après le tableau, nous remarquons que la classification correcte est **426** répartis en deux cas **TP = 331** et **TN = 95**, mais la classification incorrecte est **103** répartis en deux cas **FP = 58** et **FN = 45**.

- ✓ **Evaluation du classificateur RB**
  - o **Présentation numérique**

Tableau (III.40) : La précision du classificateur RB

Précision du Modèle								
Accuracy d'Apprentissage : 78.7%				**Accuracy de validation : 79.3%**				
Test								
Accuracy	**Précision**	**Sensibilité**	**Spécificité**	**Nég-Prédictif**	**F-Score**	**AUC**	**Gini**	**Heure de création**
80.2%	88.0%	85.0%	67.8%	62.0%	86.4%	87.1%	74.2%	9s

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Du tableau, nous remarquons ce qui suit :

- **Accuracy d'apprentissage** est 78.7% et **Accuracy de validation** est 79.3%
- **Accuracy** de test est 80.2%
- **Précision** de test est 88.0%
- **Sensibilité** de test est 85.0%
- **Spécificité** de test est 67.8%
- **Nég-Prédictif** de test est 62.0%
- **F-Score** de test est 86.4%
- **AUC** de test est 87.1%
- **Gini** de test est 74.2%
- **Heure de création** est 9s

- **Présentation graphique**

La figure (III.36) suivante qui représente : roc du classificateur RB (Apprentissage, Validation et Test).

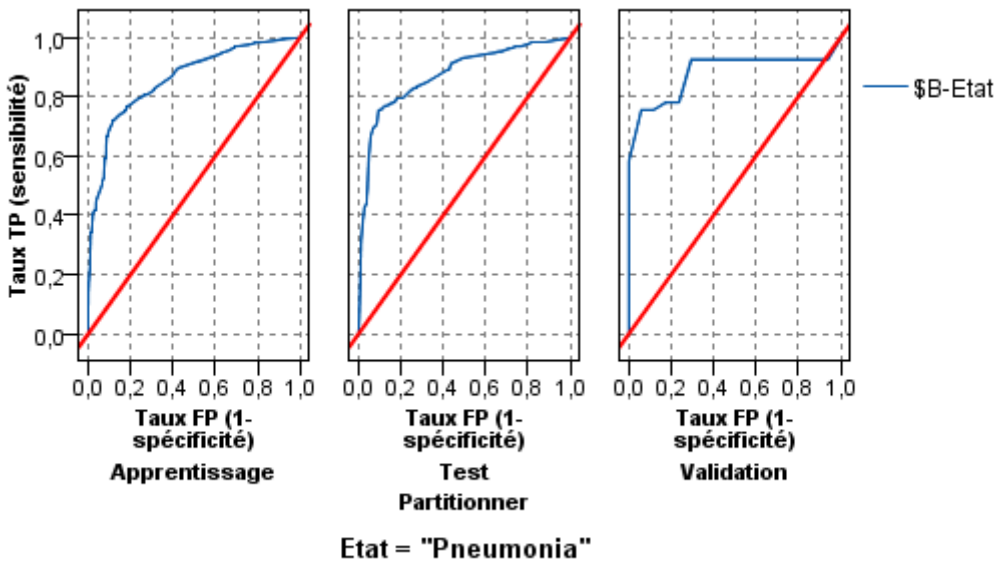


Figure (III.36) : ROC du classificateur RB

La figure (III.37) suivante qui représente : correspondance résultats actual et prédite du classificateur RB

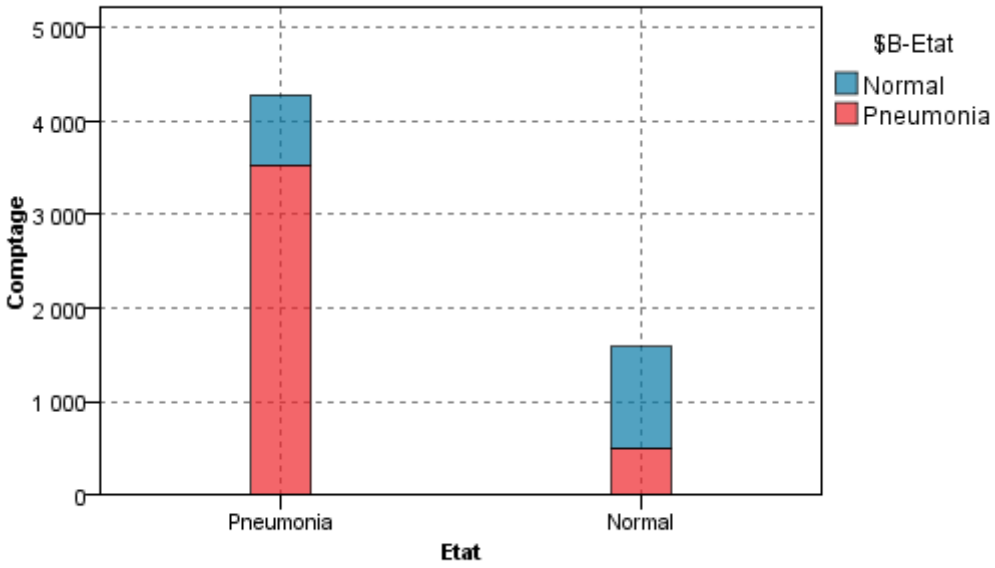


Figure (III.37) Correspondance résultats actual et prédite du classificateur RB

III.3.4. Modèle Arbre de Décision (AD)

III.3.4.1. Création du classificateur

En utilisant ce fleux dans SPSS Modeler v18.0, nous générons le classificateur AD

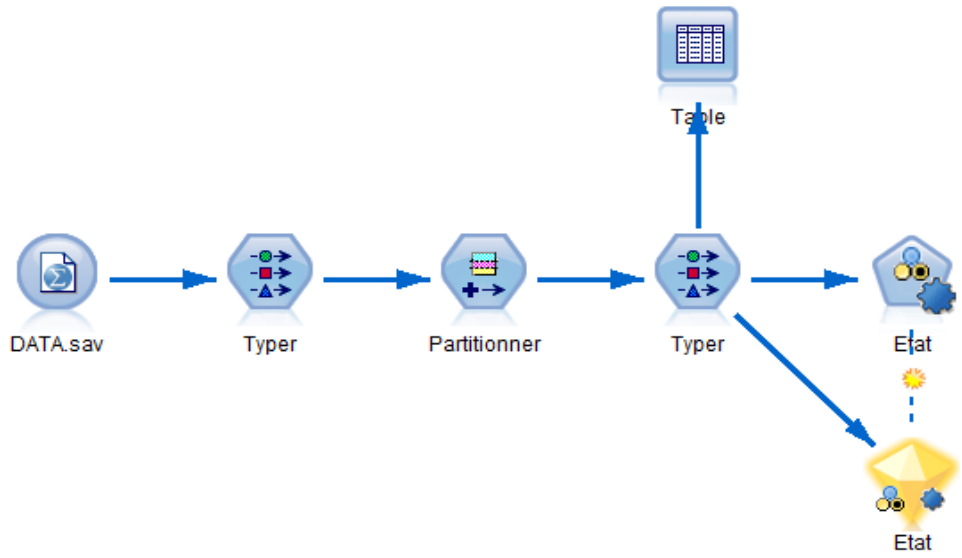


Figure (III.38) : Fleux de choisir de classificateur AD

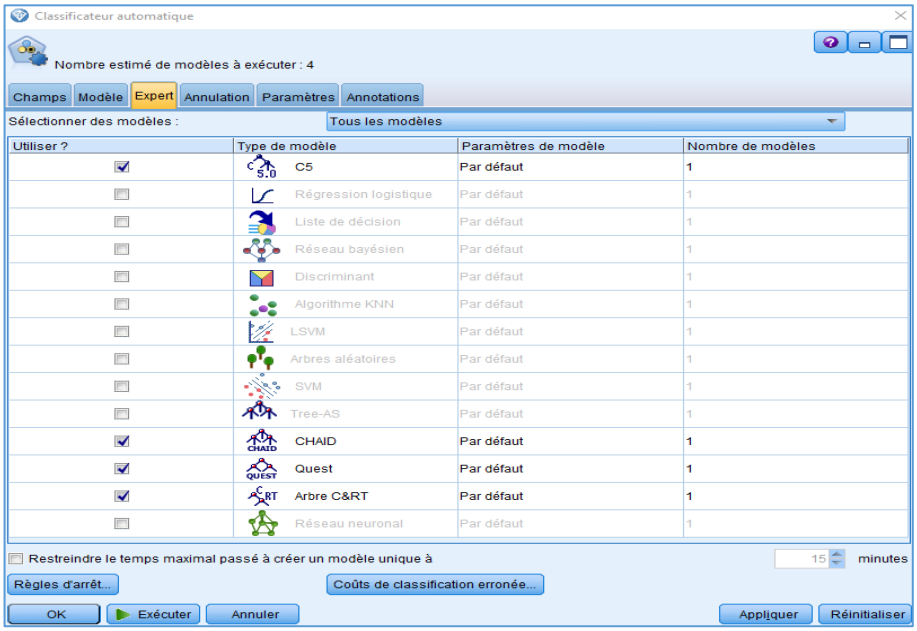


Figure (III.39) : Fenêtre de choisir les déférents des classificateurs AD

III.3.4.2. Résultats du classificateur

✓ **Choisissez le meilleur classificateur**

De la figure (III.40) correspondante, on remarque que le meilleur arbre de décision est un arbre de décision de type **C5.0**

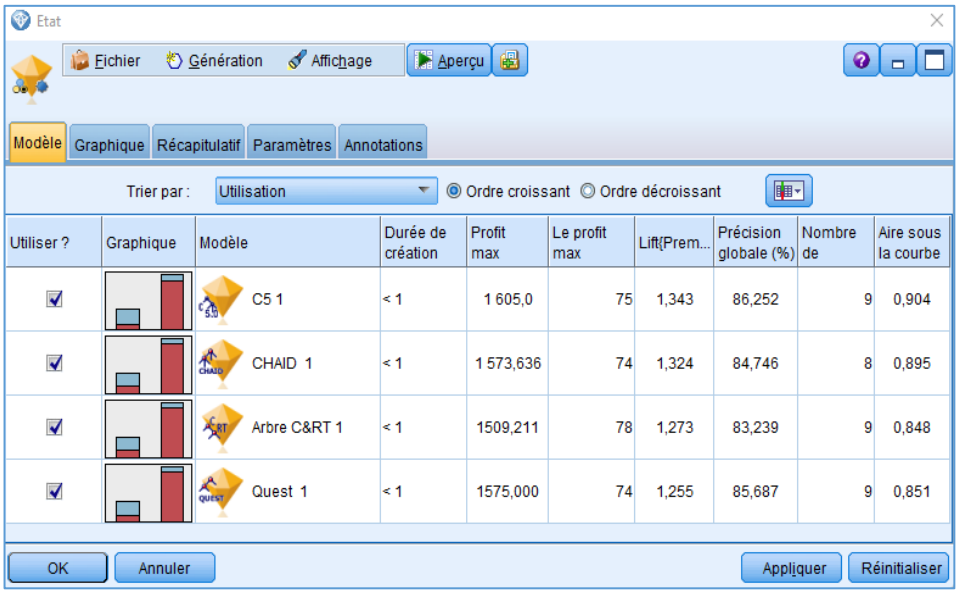


Figure (III.40) : Résultat de choisir les déférents des classificateurs AD

On va maintenant utiliser le type d'arbre de décision C5.0 En utilisant ce fleux dans SPSS Modeler v18.0 .

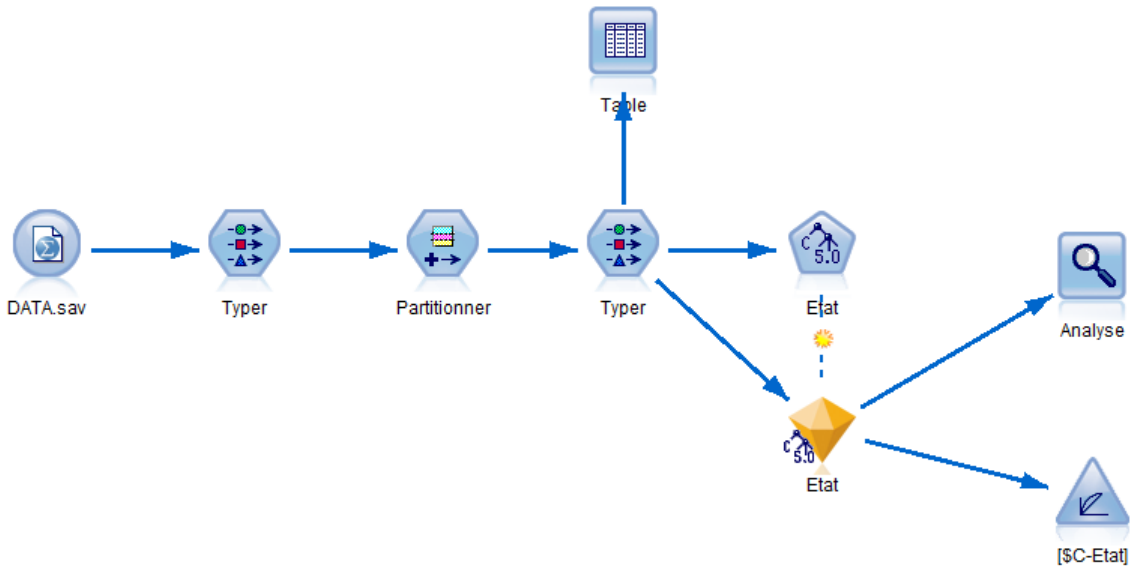


Figure (III.41) : Fleux du classificateur AD (C 5.0)

✓ **Récapitulatif du classificateur AD**

Cette figure (III.42) représente un arbre de décision type **C5.0** en forme bar

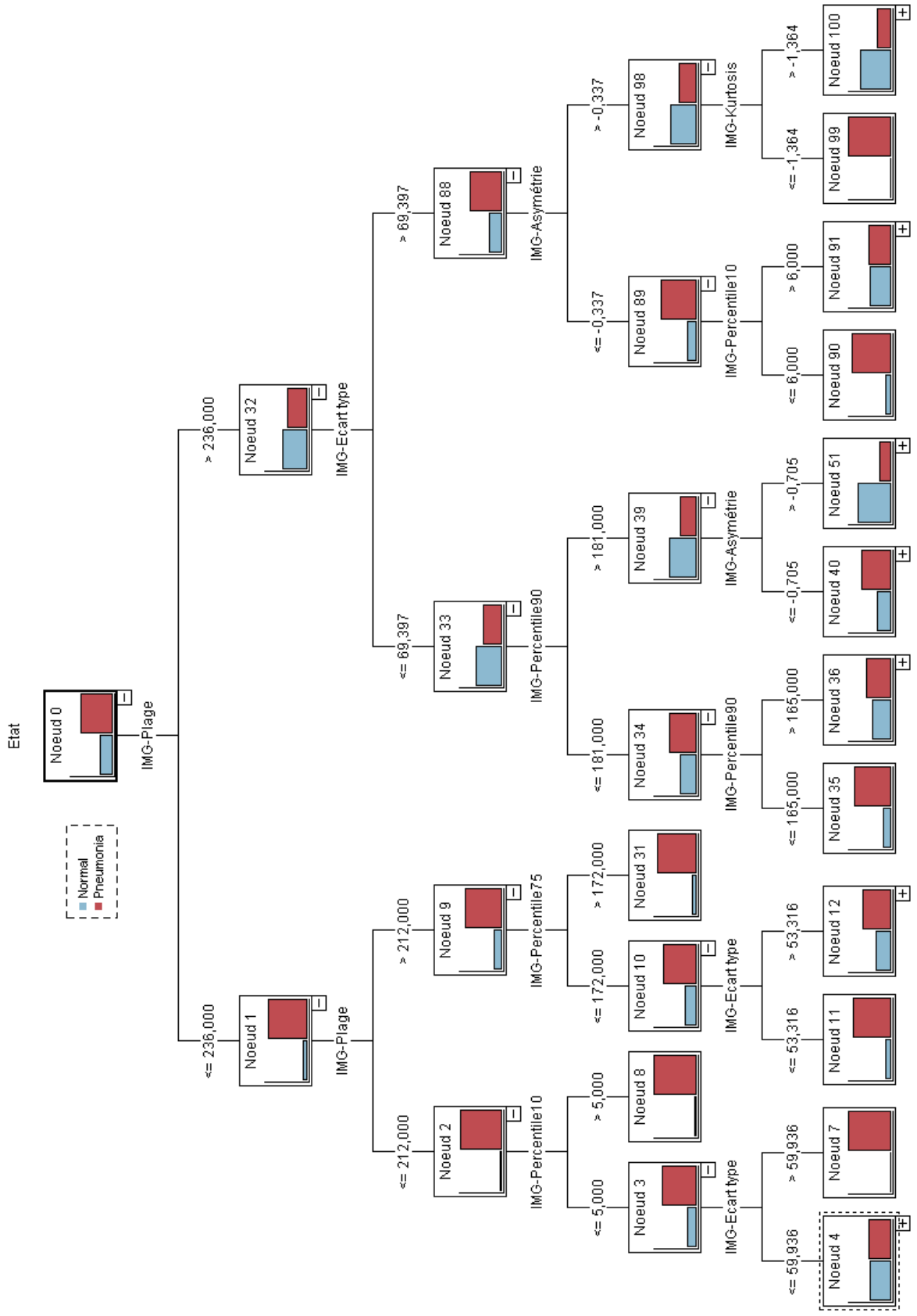


Figure (III.42) : Arbre de décision (C 5.0) forme Bar

Ou cette figure (III.43) représente un arbre de décision type **C5.0** en forme numérique

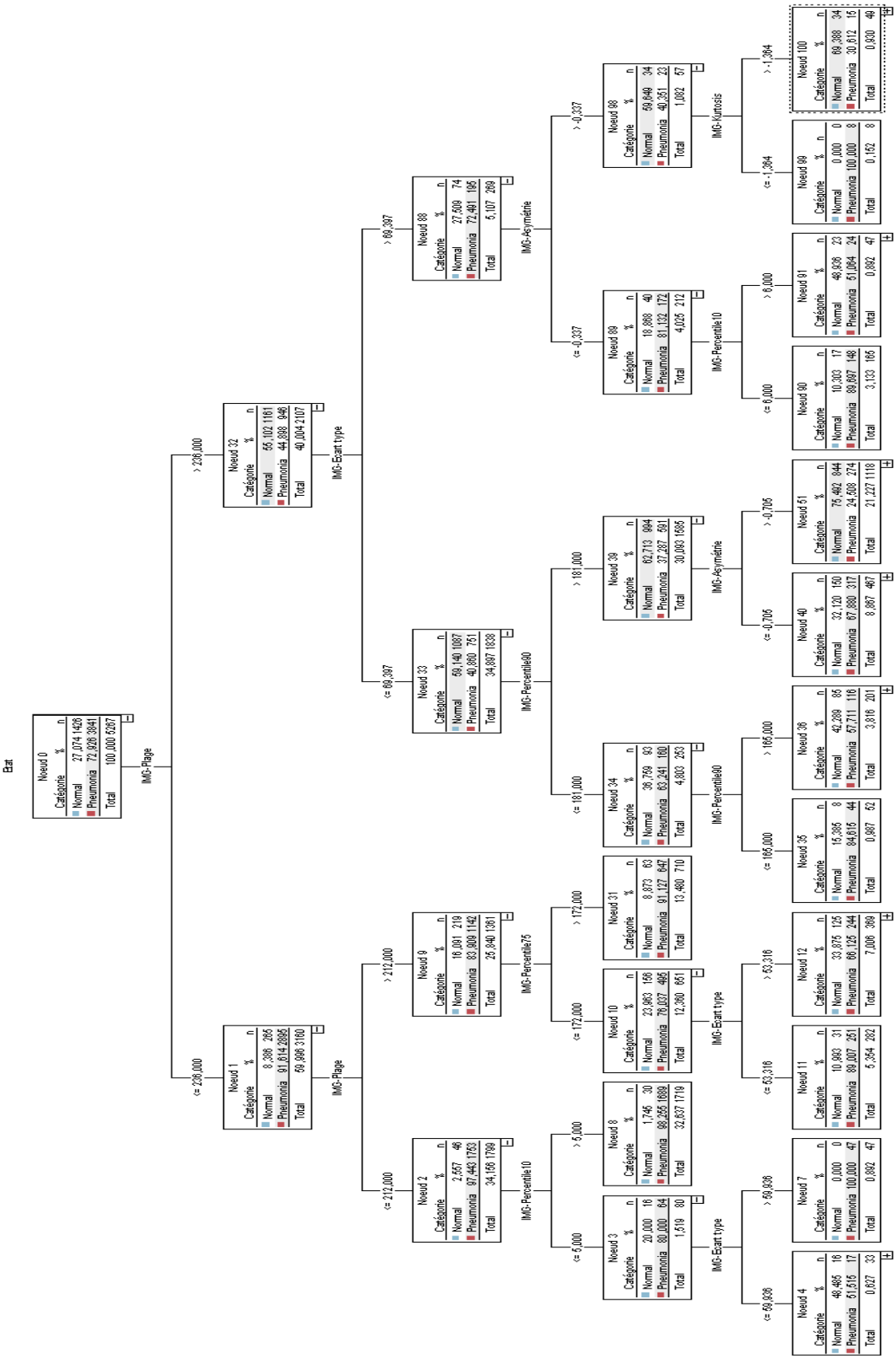


Figure (III.43) : Arbre de décision (C 5.0) forme numérique

- ✓ **L'importance des prédicteurs du classificateur AD**

Tableau (III.41) : l'importance des prédicteurs du classificateur AD

Les Paramètres	**Importance %**
IMG-Median	03,64
IMG-Ecart type	00,10
IMG-Asymétrie	18,04
IMG-Kurtosis	00,68
IMG-Plage	40,23
IMG-Percentile10	11,75
IMG-Percentile25	04,15
IMG-Percentile75	15,23
IMG-Percentile90	06,28

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Dans le tableau suivant qui représente là l'importance des prédicteurs du classificateur KNN On remarque que tous les paramètres entrés dans la construction du modèle ont une importance différente de sorte que :

- L'importance **IMG-Médian** est (03,64 %)
- L'importance **IMG-Ecart type** est (00,10 %)
- L'importance **IMG-Asymétrie** est (18,04 %)
- L'importance **IMG-Kurtosis** est (00,68 %)
- L'importance **IMG-Plage** est (40,23 %)
- L'importance **IMG-Percentile10** est (11,75%)
- L'importance **IMG-Percentile25** est (04,15 %)
- L'importance **IMG-Percentile75** est (15,23 %)
- L'importance **IMG-Percentile90** est (06,28 %)

Et la figure(III.44) ci-contre montre que

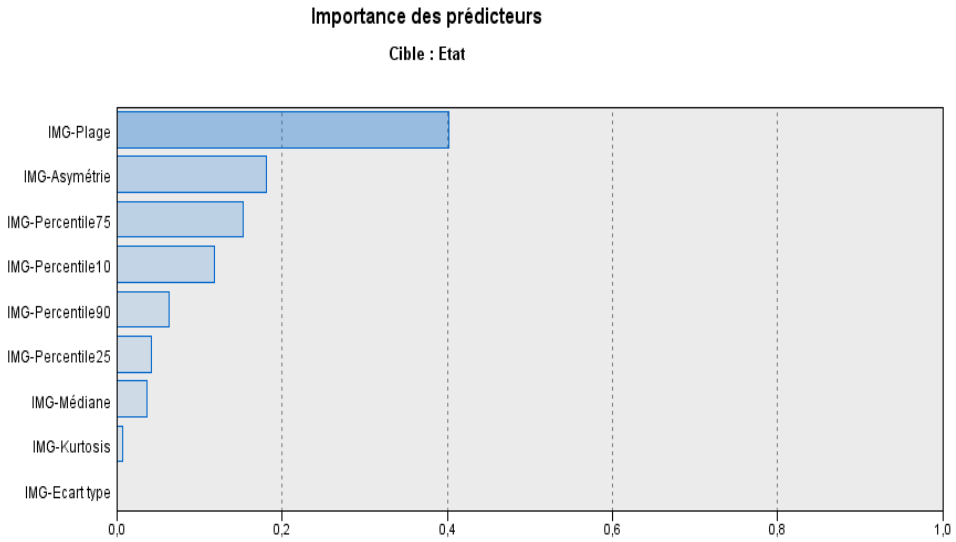


Figure (III.44) : l'importance des prédicteurs du classificateur AD

- ✓ **Matrice de confusion du classificateur AD**

Tableau (III.42) : Matrice de confusion du classificateur AD

	Prédite		
Actual	**Etat**	**Pneumonia**	**Normal**
	Pneumonia	356	35
	Normal	38	102

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

D'après le tableau, nous remarquons que la classification correcte est **458** répartis en deux cas **TP = 356** et **TN = 102**, mais la classification incorrecte est **73** répartis en deux cas **FP = 35** et **FN = 38**.

- ✓ **Evaluation du classificateur AD**
  - ○ **Présentation numérique**

Tableau (III.43) : La précision du classificateur AD

Précision du Modèle								
Accuracy d'Apprentissage : 87.9%				**Accuracy de validation : 84.4%**				
Test								
Accuracy	**Précision**	**Sensibilité**	**Spécificité**	**Nég-Prédictif**	**F-Score**	**AUC**	**Gini**	**Heure de création**
86.2%	90.3%	91.0%	72.8%	74.4%	90.5%	90.4%	80.7%	9s

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Du tableau, nous remarquons ce qui suit :

- **Accuracy d'apprentissage** est 87.9%et **Accuracy de validation** est 84.4%
- **Accuracy** de test est 86.2%
- **Précision** de test est 90.3%
- **Sensibilité** de test est 91.0%
- **Spécificité** de test est 72.8%
- **Nég-Prédictif** de test est 74.4%
- **F-Score** de test est 90.5%
- **AUC** de test est 90.4%
- **Gini** de test est 80.7%
- **Heure de création** est 9s

- **Présentation graphique**

La figure (III.45) suivante qui représente : roc du classificateur AD (Apprentissage, Validation et Test).

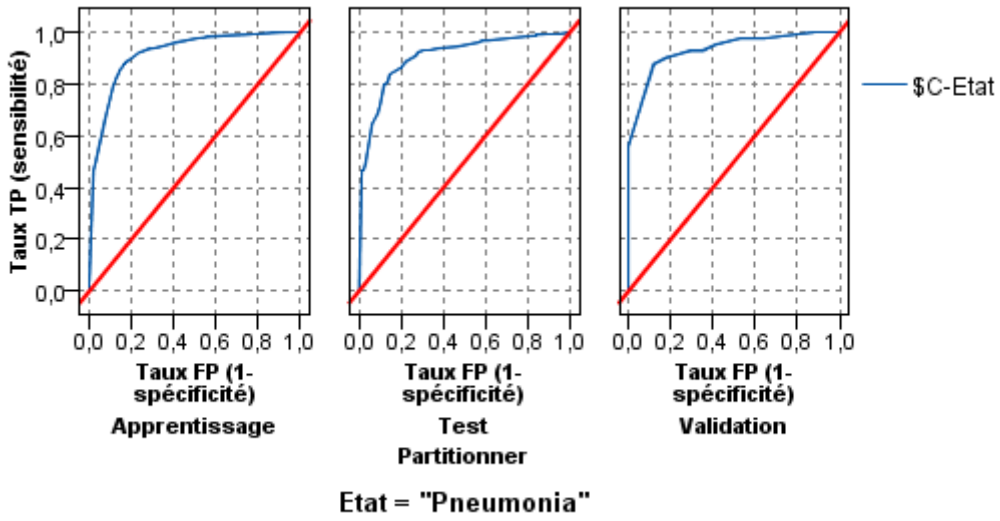


Figure (III.45) : ROC du classificateur AD

La figure (III.46) suivante qui représente : correspondance résultats actual et prédite du classificateur AD

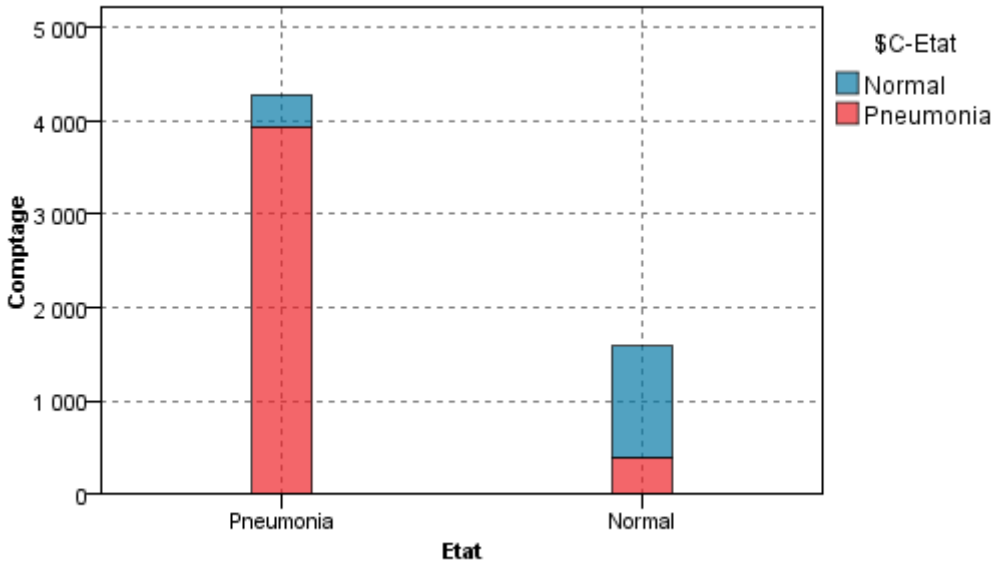


Figure (III.46) Correspondance résultats actual et prédite du classificateur AD

III.3.5. Modèle Régression de Logistique (LR)

III.3.5.1. Création du classificateur

En utilisant ce fleux dans SPSS Modeler v18.0, nous générons le classificateur LR

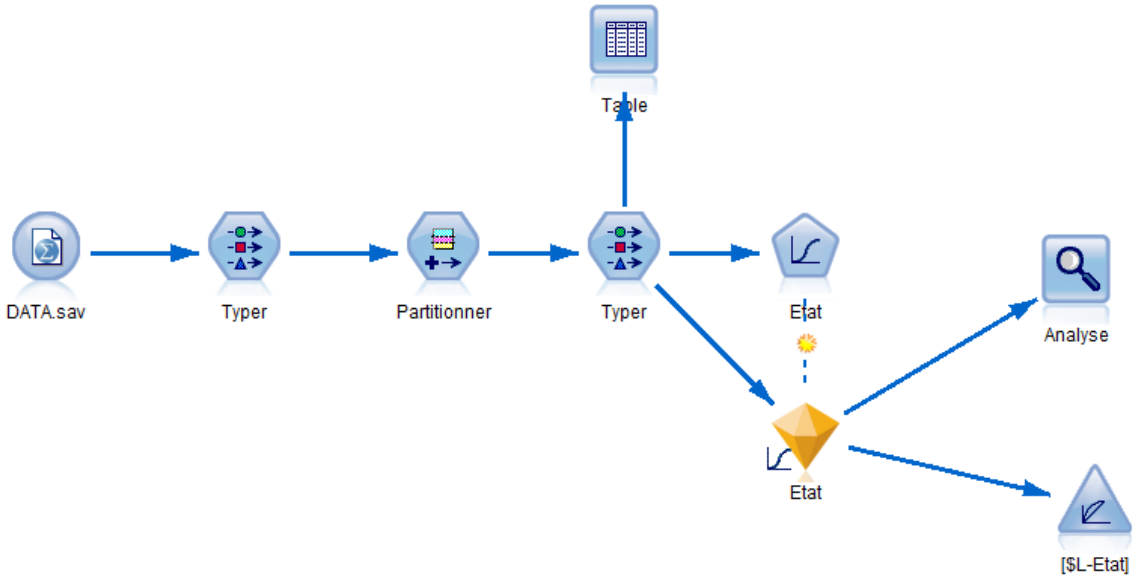


Figure (III.47) : Fleux du classificateur LR

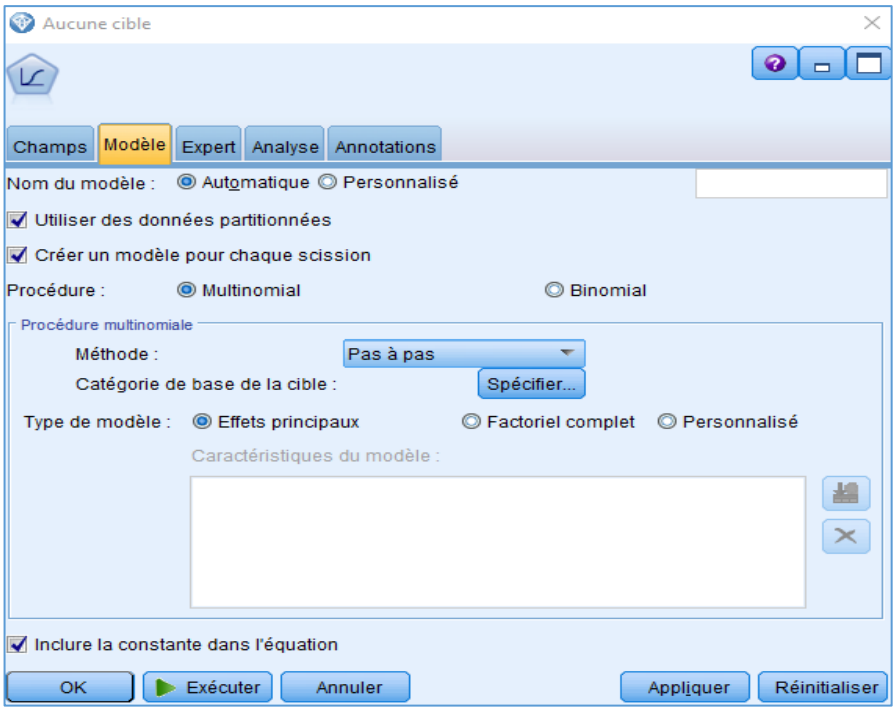


Figure (III.48) : Fenêtre de propriétés du classificateur LR

III.3.5.2. Résultats du classificateur

- ✓ **Récapitulatif du classificateur LR**

Tableau (III.44) : Récapitulatif du classificateur LR

Modèle	Critères d'ajustement du Modèle			Test du rapport de vraisemblance		Pseudo R-deux	
	AIC	BIC	Log de vraisemblance	Khi-deux	Sig	Cox et Snell	Negelkerke
Constante uniquement	6836.86	6843.53	6834.86	2685.02	0.000	0.368	0.534
Final	4167.84	4227.91	4149.84				

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Du tableau correspondant, on remarque qu'il y a une diminution dès l'indicateurs statistique **AIC** de la valeur 6836.86 à la valeur 4167.84, BIC de la valeur 6843.53à la valeur 4227.91 et **Log de vraisemblance** de la valeur 6834.86à la valeur 4149.84, cela signifie que les paramètres sont améliorés dans le modèle et valeur de **Khi-deux** =2685.02 avec signification de test est 0.000 il est inférieur à 0,05, ce qui signifie que le **modèle est acceptable**.

Le tableau ci-dessous explique Les paramètres de l'équation du classificateur LR

Tableau (III.45) : Les paramètres de l'équation du classificateur LR

Etat	Les Paramètres	B	Wald	Sig	Exp(B)	Intervalle de confiance a 95 % pour Exp(B)	
						B inf	B sup
Pneumonia	**Constante**	23.262	917.52	0.000	-	-	-
	IMG-Médiane	-0.036	9.54	0.002	0.964	0.943	0.987
	IMG-Asymétrie	4.584	59.98	0.000	97.92	30.69	312.38
	IMG-Kurtosis	2.701	98.99	0.000	14.89	8.751	25.367
	IMG-Plage	-0.067	559.40	0.000	0.935	0.935	0.940
	IMG-percentile 10	-0.025	62.55	0.000	0.975	0.969	0.981
	IMG-percentile 25	-0.022	15.38	0.000	0.979	0.968	0.989
	IMG-percentile 75	0.377	378.57	0.000	1.458	1.404	1.515
	IMG-percentile 90	-0.312	426.97	0.000	0.732	0.710	0.754

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Du tableau correspondant, on remarque la signification de test Wald il est inférieur à 0,05 Cela signifie que tous les paramètres sont significatifs Ainsi, l'équation du modèle est :

$$Y = -0.036\mathbf{X1} + 4.584\mathbf{X2} + 2.701\mathbf{X3} - 0.067\mathbf{X4} - 0.025\mathbf{X5} - 0.022\mathbf{X6} + 0.377\mathbf{X7} - 0.312\mathbf{X8} + 23.262$$

X1 : IMG-Médiane
X2 : IMG-Asymétrie
X3 : IMG-Kurtosis
X4 : IMG-Plage
X5 : IMG-percentile 10
X6 : IMG-percentile 25
X7 : IMG-percentile 75
X8 : IMG-percentile 90

Nous utilisons l'équation suivante pour trouver la probabilité du cas

$$P = \frac{e^{y}}{1 + e^{y}}$$

- ✓ **L'importance des prédicteurs du classificateur LR**

Tableau (III.46) : l'importance des prédicteurs du classificateur LR

Les Paramètres	**Importance %**
IMG-Median	01,80
IMG-Ecart type	00.00
IMG-Asymétrie	04,56
IMG-Kurtosis	03,44
IMG-Plage	47,56
IMG-Percentile10	01,7
IMG-Percentile25	02,05
IMG-Percentile75	15,46
IMG-Percentile90	23,43

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Dans le tableau suivant qui représente là l'importance des prédicteurs du classificateur LR On remarque que tous les paramètres entrés dans la construction du modèle ont une importance différente de sorte que :

- L'importance **IMG-Médian** est (01,80 %)
- L'importance **IMG-Ecart type** est (00.00 %)
- L'importance **IMG-Asymétrie** est (04,56 %)
- L'importance **IMG-Kurtosis** est (03,44 %)
- L'importance **IMG-Plage** est (47,56 %)
- L'importance **IMG-Percentile10** est (01,7 %)
- L'importance **IMG-Percentile25** est (02,05 %)
- L'importance **IMG-Percentile75** est (15,46 %)
- L'importance **IMG-Percentile90** est (23,43%)

Et la figure(III.1) ci-contre montre que

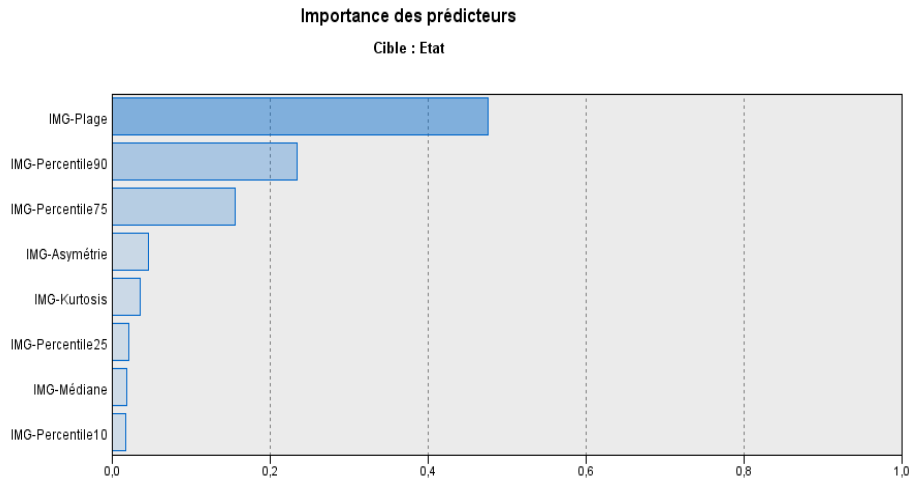


Figure (III.49) : l'importance des prédicteurs du classificateur LR

✓ **Matrice de confusion du classificateur LR**

Tableau (III.47) : Matrice de confusion du classificateur LR

	Prédite		
Actual	**Etat**	**Pneumonia**	**Normal**
	Pneumonia	359	32
	Normal	46	94

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

D'après le tableau, nous remarquons que la classification correcte est **453** répartis en deux cas **TP = 359** et **TN = 94**, mais la classification incorrecte est **78** répartis en deux cas **FP = 32** et **FN = 46**.

✓ **Evaluation du classificateur LR**

○ **Présentation numérique**

Tableau (III.48) : La précision du classificateur LR

Précision du Modèle								
Accuracy d'Apprentissage : 83.5%				**Accuracy de validation : 82.7%**				
Test								
Accuracy	**Précision**	**Sensibilité**	**Spécificité**	**Nég-Prédictif**	**F-Score**	**AUC**	**Gini**	**Heure de création**
85.3%	88.6%	91.8%	67.1%	74.6%	90.1%	91.5%	83.0%	9s

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Du tableau, nous remarquons ce qui suit :

- **Accuracy d'apprentissage** est 83.5%et **Accuracy de validation** est 82.7%
- **Accuracy** de test est 85.3%
- **Précision** de test est 88.6%
- **Sensibilité** de test est 91.8%
- **Spécificité** de test est 67.1%
- **Nég-Prédictif** de test est 74.6%
- **F-Score** de test est 90.1%
- **AUC** de test est 91.5%
- **Gini** de test est 83.0%
- **Heure de création** est 9s

- **Présentation graphique**

La figure (III.50) suivante qui représente : roc du classificateur LR (Apprentissage, Validation et Test).

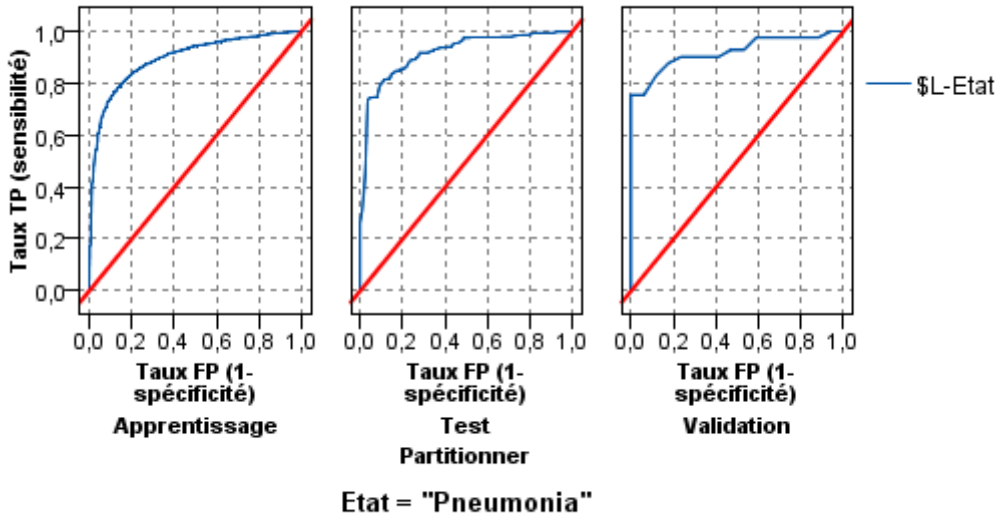


Figure (III.50) : ROC du classificateur LR

La figure (III.51) suivante qui représente : correspondance résultats actual et prédite du classificateur LR

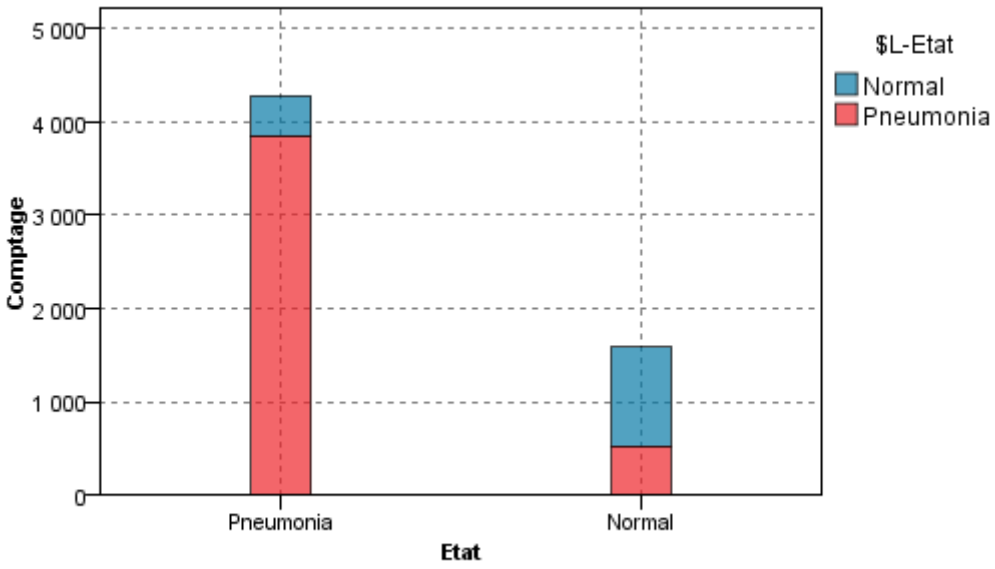


Figure (III.51) Correspondance résultats actual et prédite du classificateur LR

III.3.6. Modèle réseau de neurones (NN)

III.3.6.1. Création du classificateur

En utilisant ce fleux dans SPSS Modeler v18.0, nous générons le classificateur NN

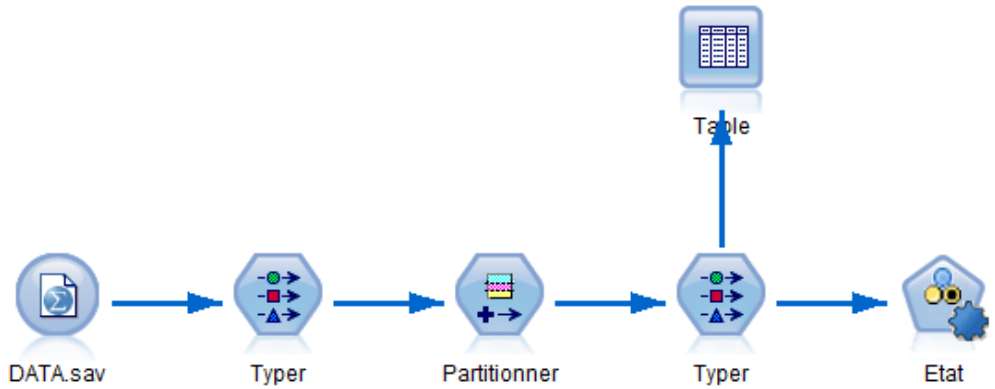


Figure (III.52) : Fleux du classificateur NN

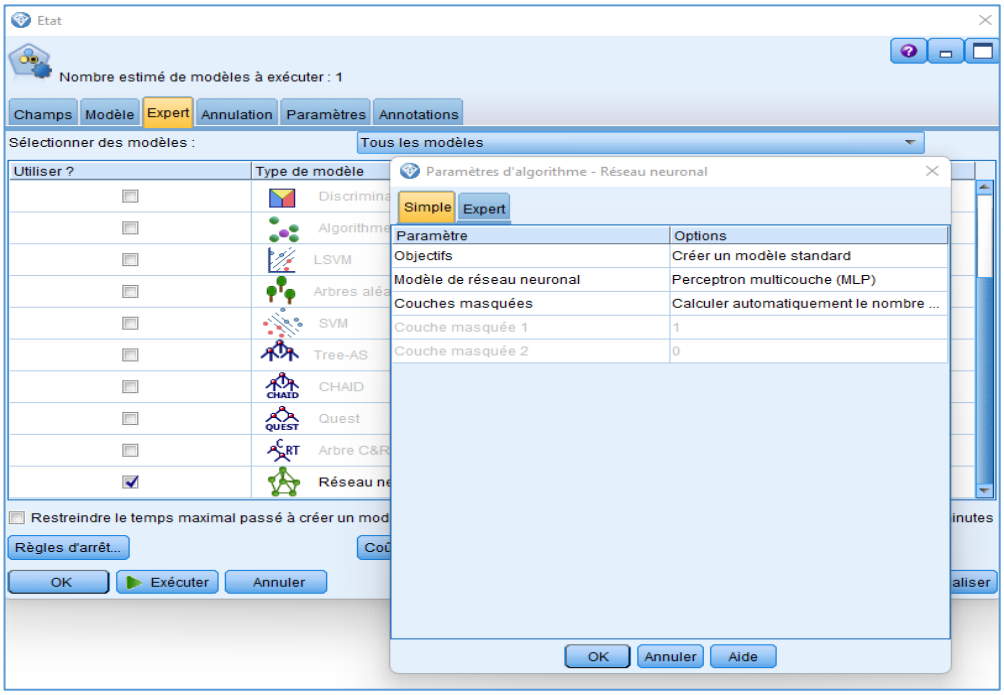


Figure (III.53) : Fenêtre de choisir les déférents des classificateurs NN

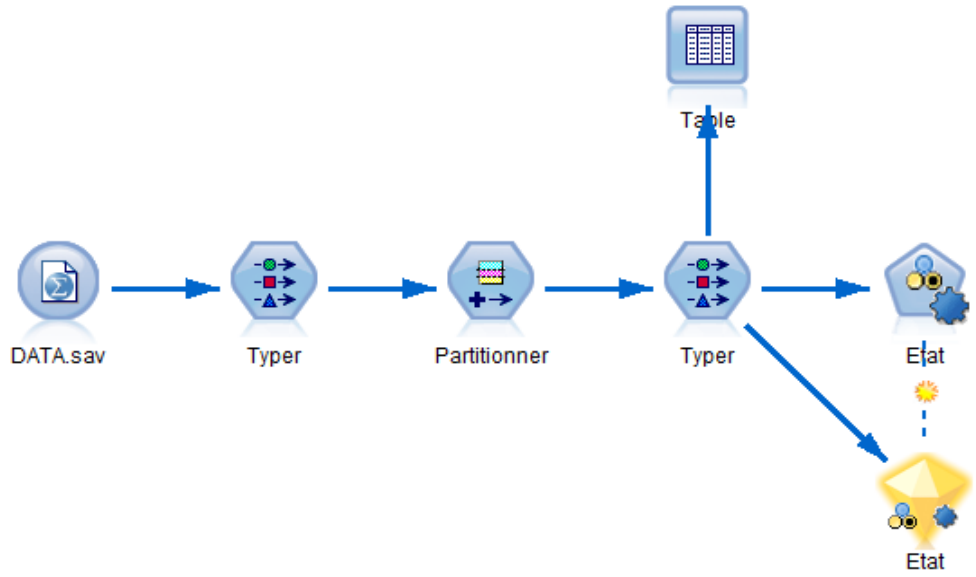


Figure (III.54) : Fleux de choisir de classificateur NN

De la figure (III.1) correspondante, on remarque que le meilleur réseau de neurone est un réseau de neurone de type **(Perceptron multicouche).**

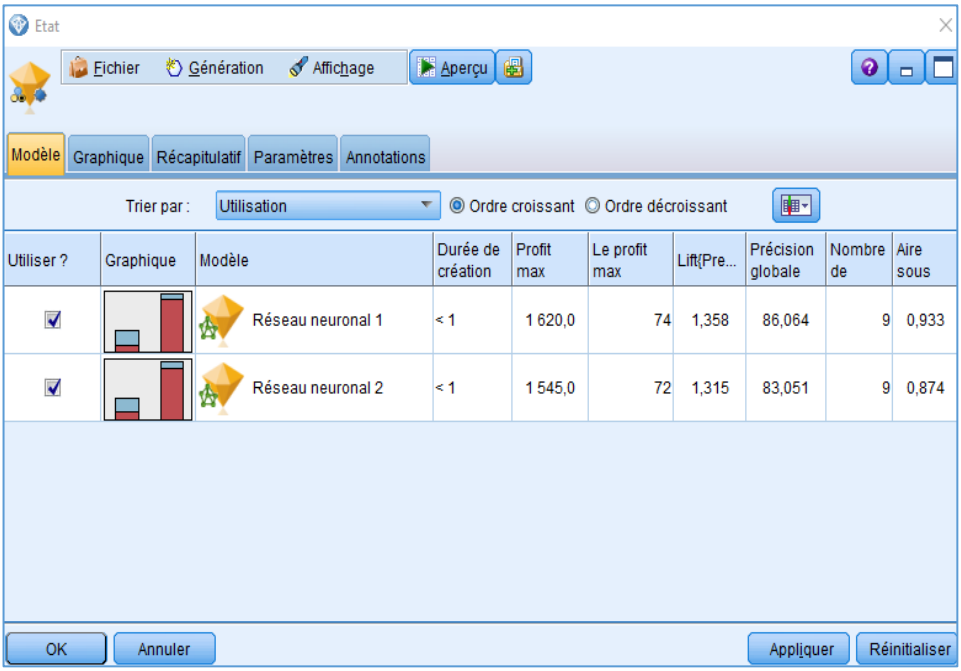


Figure (III.55) : Résultat de choisir les déférents des classificateurs NN

En utilisant ce fleux dans SPSS Modeler v18.0, nous générons le classificateur NN

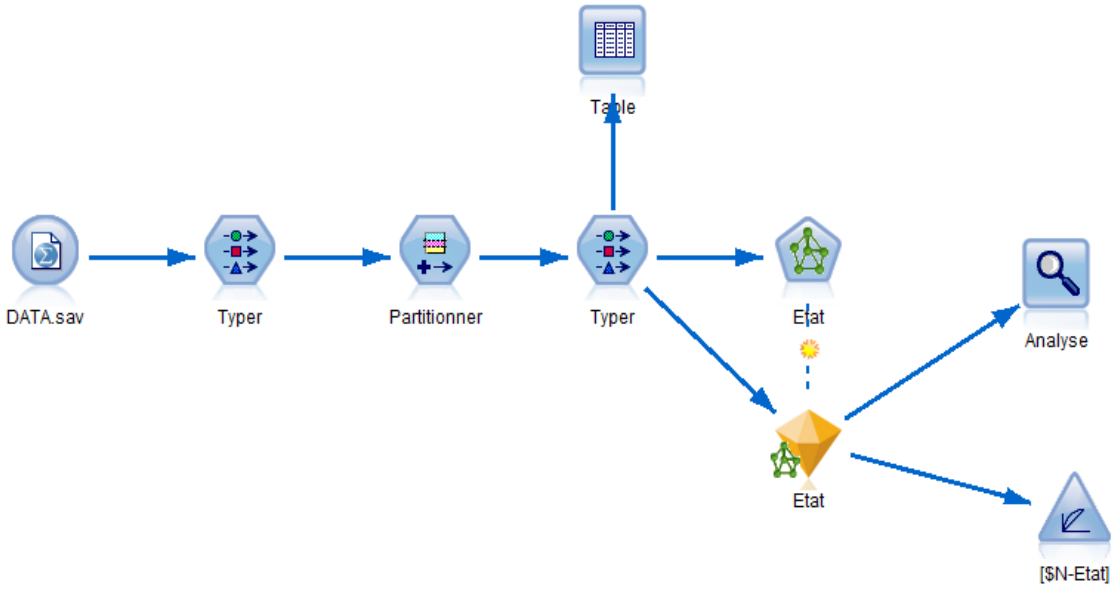


Figure (III.56) : Fleux de classificateur NN (Perceptron multicouche)

III.3.6.2. Résultats du classificateur

✓ **Récapitulatif du classificateur NN**

Tableau (III.49) : Récapitulatif du classificateur NN

Caractéristiques réseau de Neurones	
Cible	Etat (Normal -Pneumonia)
Type de réseau	Perceptron multicouche
Nombres des neurones de primeur couche masquée	06
Activation de la couche masquée	Tangente hyperbolique
Activation de la couche de sortie	softmax

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Du tableau, nous remarquons ce qui suit :

- **Type de réseau** est Perceptron multicouche
- **Nombres des neurones de primeur couche masquée** est 06
- **Activation de la couche masquée** est Tangente hyperbolique
- **Activation de la couche de sortie** est softmax

La figure (III.7) correspondante indique réseau neurone (Perceptron multicouche)

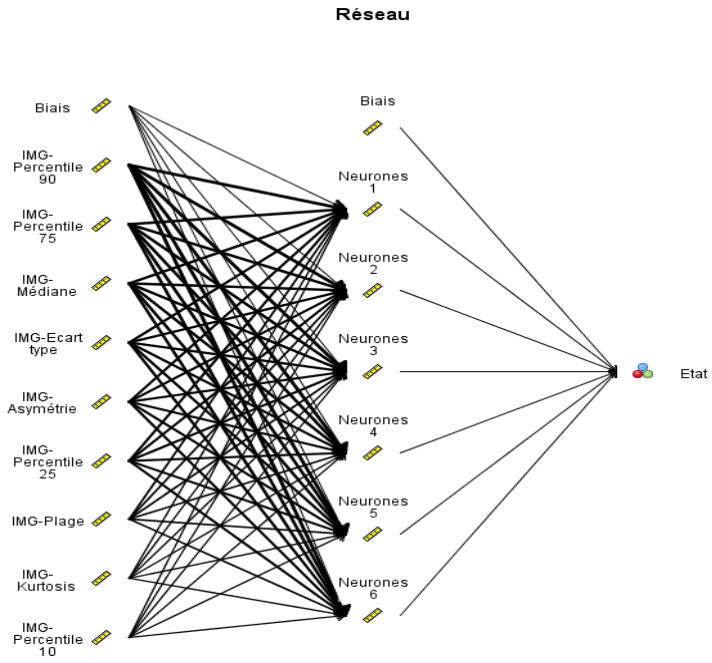


Figure (III.57) : Visualisation du NN (Perceptron multicouche)

Et la figure (III.8) correspondante indique des coefficients réseau neurone (Perceptron multicouche)

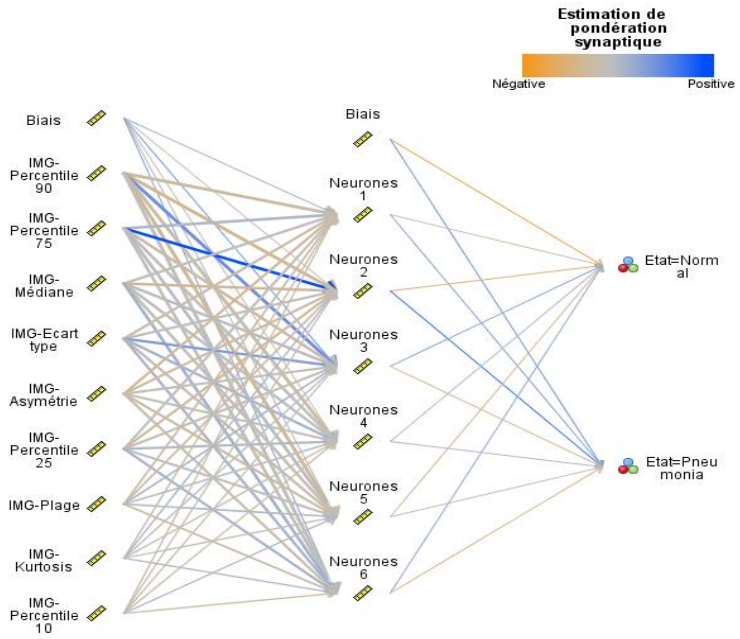


Figure (III.58) : Visualisation des coefficients du NN (Perceptron multicouche)

- ✓ **L'importance des prédicteurs du classificateur KNN**

Figure (III.50) : l'importance des prédicteurs du classificateur NN

Les Paramètres	**Importance %**
IMG-Median	15,00
IMG-Ecart type	13,00
IMG-Asymétrie	12,00
IMG-Kurtosis	04,00
IMG-Plage	07,00
IMG-Percentile10	02,00
IMG-Percentile25	12,00
IMG-Percentile75	16,00
IMG-Percentile90	19,00

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Dans le tableau suivant qui représente là l'importance des prédicteurs du classificateur KNN On remarque que tous les paramètres entrés dans la construction du modèle ont une importance différente de sorte que :

- L'importance **IMG-Médian** est (15,00%)
- L'importance **IMG-Ecart type** est (13,00 %)
- L'importance **IMG-Asymétrie** est (12,00 %)

- L'importance **IMG-Kurtosis** est (04,00 %)
- L'importance **IMG-Plage** est (07,00 %)
- L'importance **IMG-Percentile10** est (02,00 %)
- L'importance **IMG-Percentile25** est (12,00 %)
- L'importance **IMG-Percentile75** est (16,00 %)
- L'importance **IMG-Percentile90** est (19,00 %)

Et la figure(III.59) ci-contre montre que

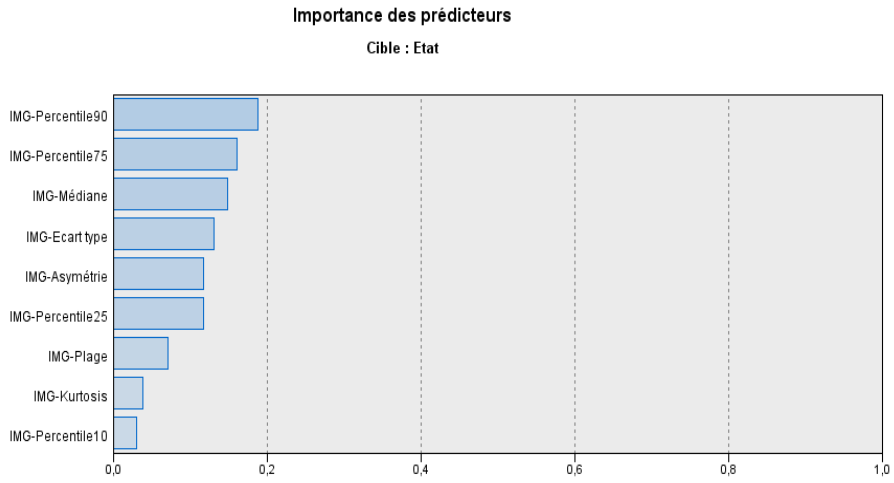


Figure (III.59) : l'importance des prédicteurs du classificateur NN

✓ **Matrice de confusion du classificateur NN**

Tableau (III.51) : Matrice de confusion du classificateur NN

	Prédite		
Actual	**Etat**	**Pneumonia**	**Normal**
	Pneumonia	366	25
	Normal	49	91

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

D'après le tableau, nous remarquons que la classification correcte est **457** répartis en deux cas **TP = 366** et **TN = 91**, mais la classification incorrecte est **74** répartis en deux cas **FP = 25** et **FN = 49**.

- ✓ **Evaluation du classificateur NN**

- o **Présentation numérique**

Tableau (III.52) : La précision du classificateur NN

Précision du Modèle								
Accuracy d'Apprentissage : 86.2%				**Accuracy de validation : 87.9%**				
Test								
Accuracy	**Précision**	**Sensibilité**	**Spécificité**	**Nég-Prédictif**	**F-Score**	**AUC**	**Gini**	**Heure de création**
86.0%	88.1%	93.6%	65.0%	78.4%	90.6%	93.3%	86.5%	12s

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Du tableau, nous remarquons ce qui suit :

- **Accuracy d'apprentissage** est 86.2%et **Accuracy de validation** est 87.9%
- **Accuracy** de test est 86.0%
- **Précision** de test est 88.1%
- **Sensibilité** de test est 93.6%
- **Spécificité** de test est 65.0%
- **Nég-Prédictif** de test est 78.4%
- **F-Score** de test est 90.6%
- **AUC** de test est 93.3%
- **Gini** de test est 86.5%
- **Heure de création** est 12s

- o **Présentation graphique**

La figure (III.60) suivante qui représente : roc du classificateur NN (Apprentissage, Validation et Test).

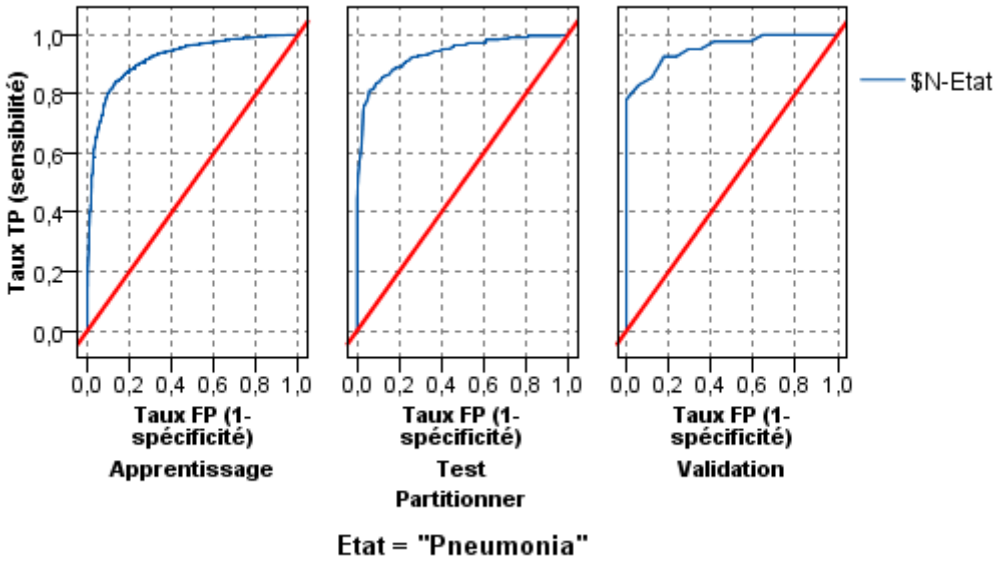


Figure (III.60) : ROC du classificateur NN

La figure (III.61) suivante qui représente : correspondance résultats actual et prédite du classificateur NN

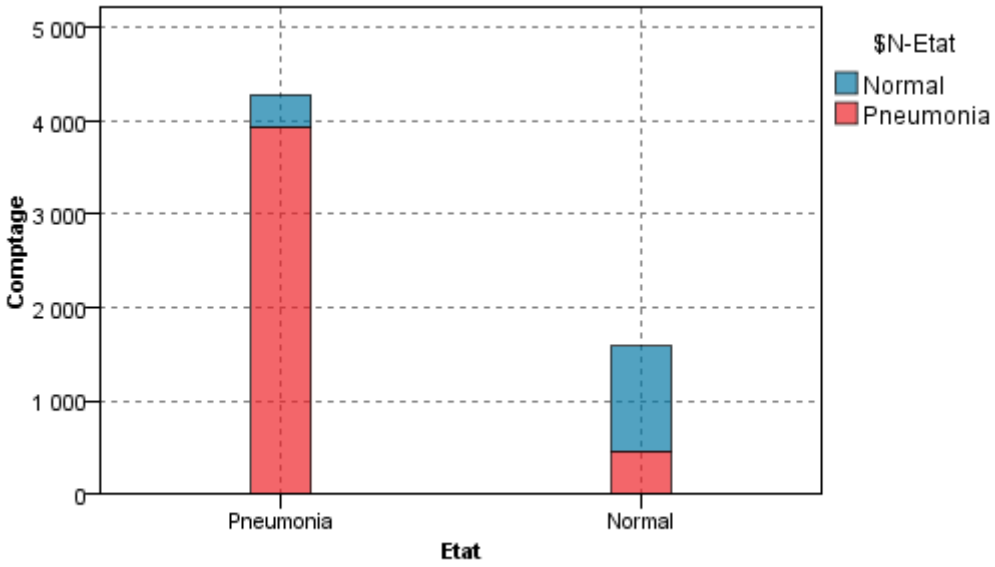


Figure (III.61) Correspondance résultats actual et prédite du classificateur NN

III.3.7. Modèle Arbres aléatoire (AA)

III.3.7.1. Création du classificateur

En utilisant ce fleux dans SPSS Modeler v18.0, nous générons le classificateur AA

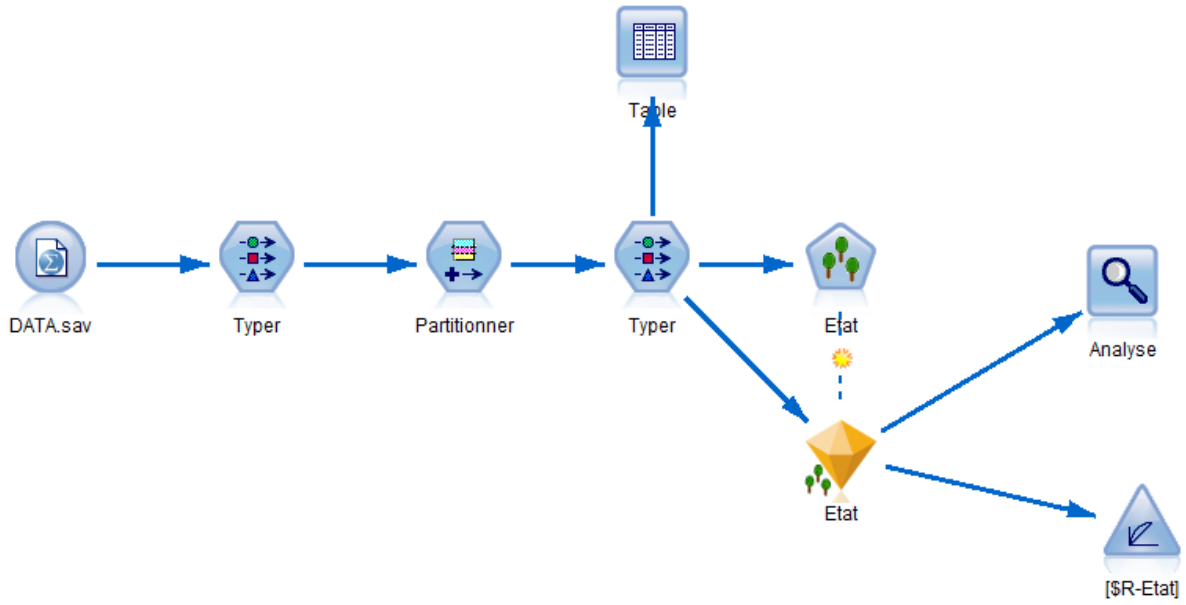


Figure (III.62) : Fleux de classificateur AA

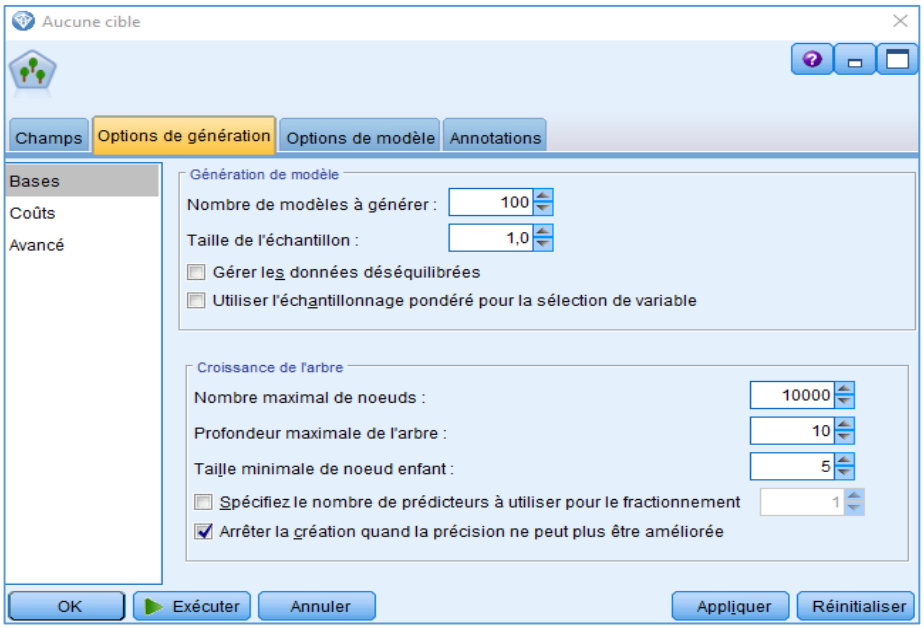


Figure (III.63) : Fenêtre de propriétés du classificateur AA

III.3.7.2. Résultats du classificateur

- ✓ **Matrice de confusion du classificateur AA**

Tableau (III.53) : Matrice de confusion du classificateur AA

	Prédite		
Actual	**Etat**	**Pneumonia**	**Normal**
	Pneumonia	321	70
	Normal	23	117

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

D'après le tableau, nous remarquons que la classification correcte est **438** répartis en deux cas **TP = 321** et **TN = 117**, mais la classification incorrecte est **93** répartis en deux cas **FP = 23** et **FN = 49**.

- ✓ **Evaluation du classificateur AA**
  - o **Présentation numérique**

Tableau (III.54) : La précision du classificateur AA

Précision du Modèle								
Accuracy d'Apprentissage : 86.8%				**Accuracy de validation : 89.6%**				
Test								
Accuracy	**Précision**	**Sensibilité**	**Spécificité**	**Nég-Prédictif**	**F-Score**	**AUC**	**Gini**	**Heure de création**
82.4%	93.3%	82.0%	83.5%	62.5%	87.2%	90.7%	81.3%	9s

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Du tableau, nous remarquons ce qui suit :

- **Accuracy d'apprentissage** est 86.8%et **Accuracy de validation** est 89.6%
- **Accuracy** de test est 82.4%
- **Précision** de test est 93.3%
- **Sensibilité** de test est 82.0%
- **Spécificité** de test est 83.5%
- **Nég-Prédictif** de test est 62.5%
- **F-Score** de test est 87.2%
- **AUC** de test est 90.7%
- **Gini** de test est 81.3%
- **Heure de création** est 9s

- **Présentation graphique**

La figure (III.64) suivante qui représente : roc du classificateur AA (Apprentissage, Validation et Test).

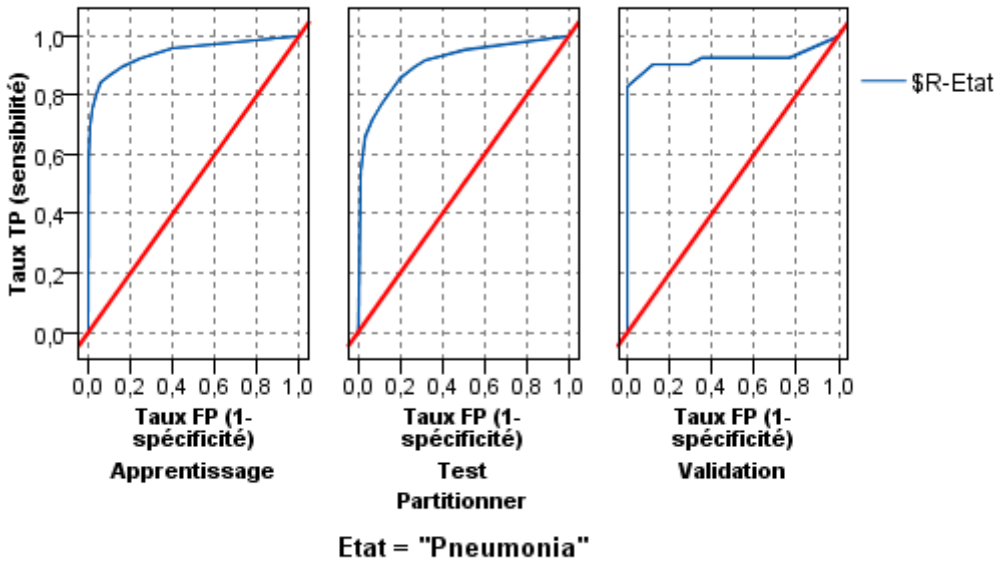


Figure (III.64) : ROC du classificateur AA

La figure (III.65) suivante qui représente : correspondance résultats actual et prédite du classificateur AA

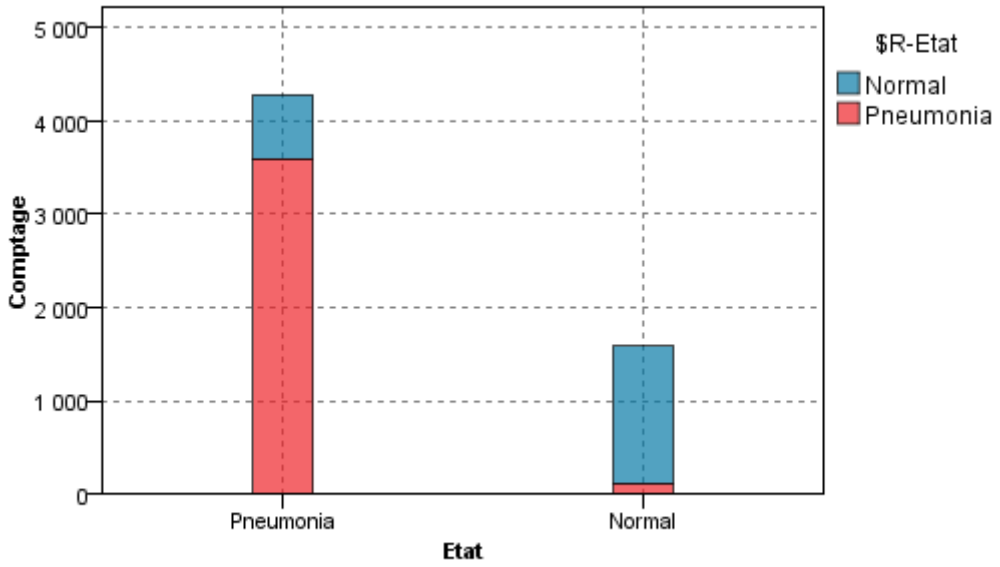


Figure (III.65) Correspondance résultats actual et prédite du classificateur AA

III.3.8. Comparaison entre les modèles

En utilisant ce fleux dans SPSS Modeler v18.0, pour la comparaison entre les déférents classificateurs.

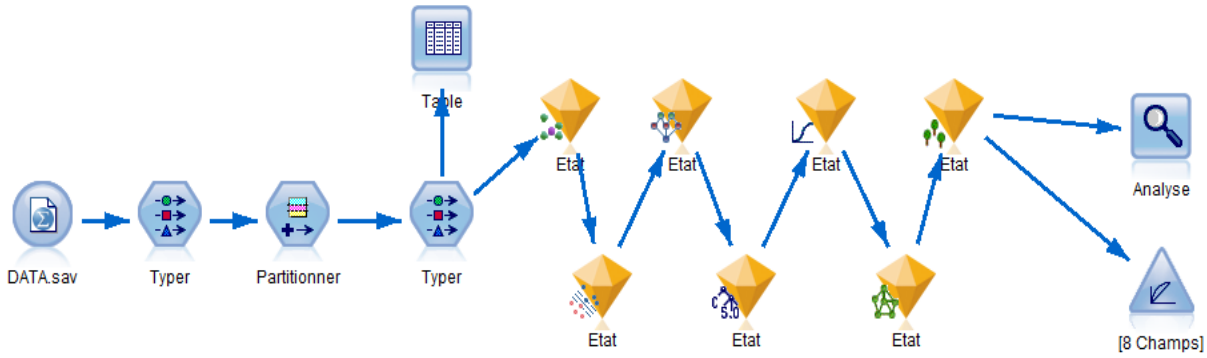


Figure (III.66) : Fleux de comparaison entre les déférents classificateurs

III.3.8.1. Précision d'apprentissage et validation

Tableau (III.55) : Comparaison la précision d'apprentissage et validation pour les déférents classificateurs

	Précision du Modèle	
Modèle	**Accuracy d'Apprentissage**	**Accuracy de validation**
KNN	87.5 %	91.3 %
SVM	84.6 %	89.6 %
RB	78.7 %	79.3 %
AD	87.9 %	84.4 %
LR	83.5 %	82.7 %
NN	86.2 %	87.9 %
AA	86.6 %	89.6 %

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Dans le tableau, nous remarquons que la plus grande valeur est pour **Accuracy d'Apprentissage** est 87.9 %pour le modèle **AD** , et la plus petite valeur est 78.7 % pour le modèle **RB** ,mais la plus grande valeur est pour **Accuracy de validation** est 91.3 % pour le modèle **KNN** ,et la plus petite valeur est 79.3 % pour le modèle **RB**

III.3.8.2. Précision de test

Tableau (III.56) : Comparaison la précision de test pour les déférents classificateurs

Précision du Modèle									
Test									
Modèle	**Accuracy**	**Précision**	**Sensibilité**	**Spécificité**	**Nég-Prédictif**	**F-Score**	**AUC**	**Gini**	**Heure de création**
KNN	86.0%	88.1%	93.6%	65.0%	78.4%	90.6%	91.3%	82.6%	12s
SVM	86.0%	89.1%	92.3%	68.5%	76.1%	90.6%	91.3%	82.6%	12s
RB	80.2%	88.0%	85.0%	67.8%	62.0%	86.4%	87.1%	74.2%	9s
AD	86.2%	90.3%	91.0%	72.8%	74.4%	90.5%	90.4%	80.7%	9s
LR	85.3%	88.6%	91.8%	67.1%	74.6%	90.1%	91.5%	83.0%	9s
NN	86.0%	88.1%	93.6%	65.0%	78.4%	90.6%	93.3%	86.5%	12s
AA	82.4%	93.3%	82.0%	83.5%	62.5%	87.2%	90.7%	81.3%	9s

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Nous utiliserons le tableau correspondant pour pondérer et comparer les modèles

Tableau (III.57) : les coefficients de préférence

L'intervalle	Coefficients
60.0 % ----------65.0 %	1
65.1 % ----------70.1 %	2
70.2 % ----------75.2 %	3
75.3 % ----------80.3 %	4
80.4 % ----------85.4 %	5
85.5 % ----------90.5 %	6
90.6 % ----------95.6 %	7
95.7 % ----------100 %	8

Tableau (III.58) : Préférence les déférents classificateurs

Les Modèles	KNN	SVM	RB	AD	LR	NN	AA
Accuracy	6	6	4	6	5	6	5
Précision	6	6	6	6	6	6	7
Sensibilité	7	7	5	7	7	7	5
Spécificité	1	2	2	3	2	1	5
Nég-Prédictif	4	4	1	3	3	4	1
F-Score	7	7	6	6	6	7	6
AUC	7	7	6	6	7	7	7
Gini	5	5	3	5	5	6	5
Total	43	44	33	42	41	44	41
Preference	**Rang 2**	**Rang 1**	**Rang 5**	**Rang 3**	**Rang 4**	**Rang 1**	**Rang 4**

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0

Du tableau, on remarque que les meilleurs modèles sont les modèles **NN** et **SVM**, avec une valeur pondérée de 44 pour chaque modèle, et la figure ci-dessous montre la précision pour chaque modèle.

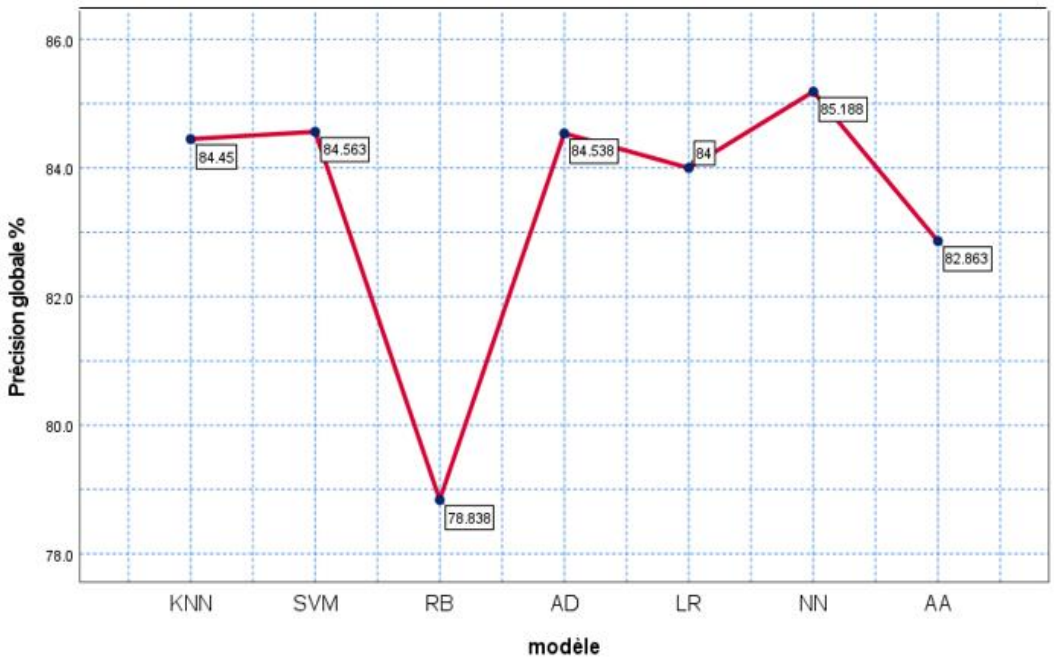


Figure (III.67) : précision globale pour les déférent modèle

La figure (III.68) suivante qui représente : roc pour les déférents classificateurs (KNN, SVM, RB, AD, LR, NN et AA) d'Apprentissage, Validation et Test.

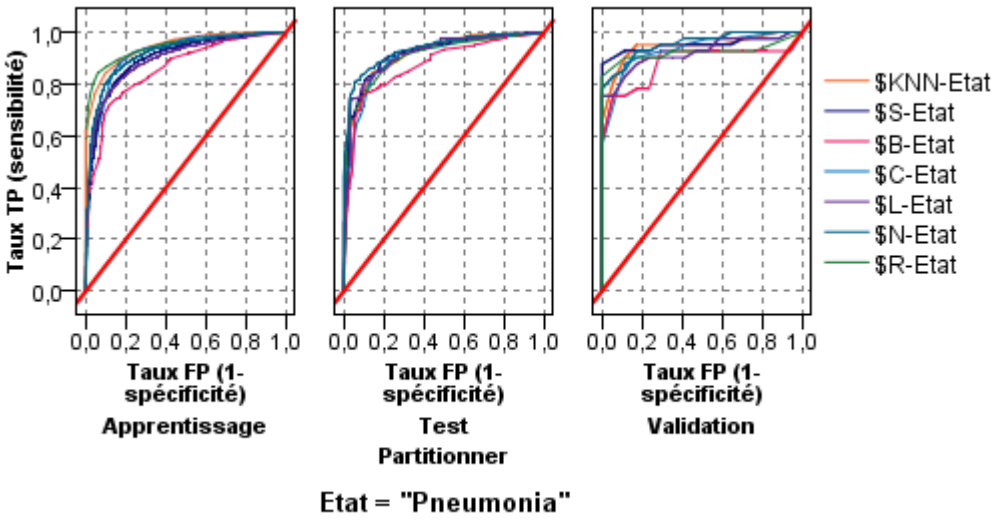


Figure (III.68) : ROC les déférents classificateurs (KNN, SVM, RB, AD, LR, NN, AA)

III.3.9. Modèle Convolution neural network (CNN)

III.3.9.1. Création du classificateur

- ✓ **Importation des bibliothèques nécessaires**

Ce programme est pour importation des bibliothèques nécessaires :

```
import matplotlib.pyplot as plt
import seaborn as sns
import keras
from keras.models import Sequential
from keras.layers import Dense, Conv2D , MaxPool2D , Flatten , Dropout , BatchNormalization
from keras.preprocessing.image import ImageDataGenerator
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report,confusion_matrix
from keras.callbacks import ReduceLROnPlateau
import cv2
import os
```

Figure (III.69) Programme pour importation des bibliothèques nécessaires

✓ **Redimensionner les données**

Ce programme est pour redimensionner les données :

```
labels = ['PNEUMONIA', 'NORMAL']
img_size = 150
def get_training_data(data_dir):
    data = []
    for label in labels:
        path = os.path.join(data_dir, label)
        class_num = labels.index(label)
        for img in os.listdir(path):
            try:
                img_arr = cv2.imread(os.path.join(path, img), cv2
.IMREAD_GRAYSCALE)
                resized_arr = cv2.resize(img_arr, (img_size, img_
size)) # Reshaping images to preferred size
                data.append([resized_arr, class_num])
            except Exception as e:
                print(e)
    return np.array(data)
```

Figure (III.70) Programme pour redimensionner les données

✓ **Visualisation des données**

Ce programme est pour Visualisation des données :

```
plt.figure(figsize = (5,5))
plt.imshow(train[0][0], cmap='gray')
plt.title(labels[train[0][1]])

plt.figure(figsize = (5,5))
plt.imshow(train[-1][0], cmap='gray')
plt.title(labels[train[-1][1]])
```

Figure (III.71) Programme pour Visualisation des données

Aperçu des images des deux classes

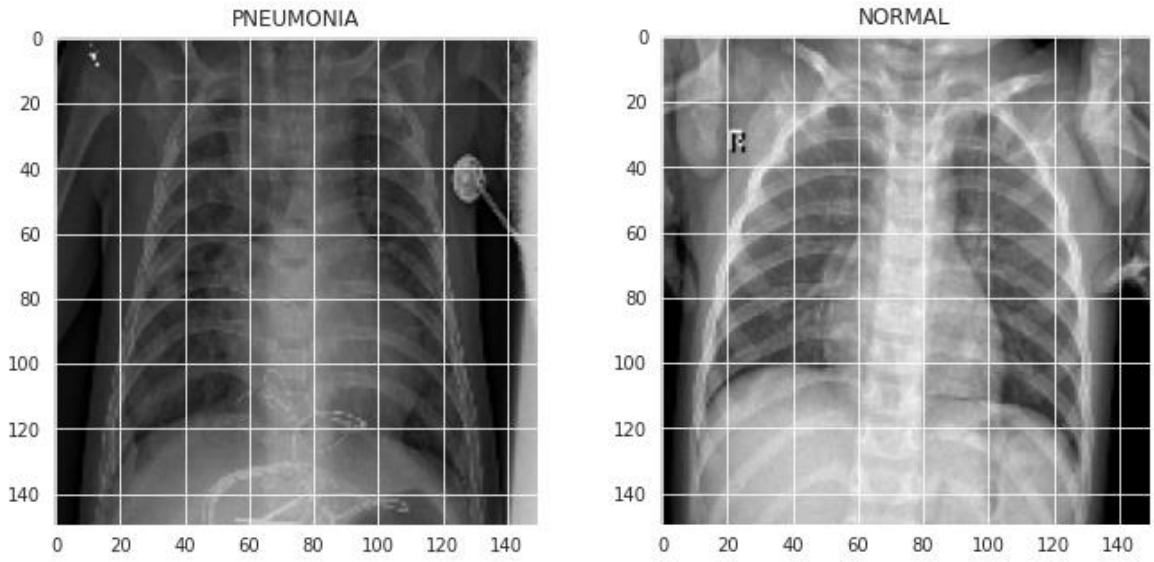


Figure (III.72) : images des deux classes (Normal, Pneumonia)

- ✓ **Augmentation des données**

1. Faites pivoter au hasard certaines images d'entraînement de 30 degrés
2. Zoom aléatoire de 20 % sur certaines images d'entraînement
3. Décalez aléatoirement les images horizontalement de 10 % de la largeur
4. Décalez aléatoirement les images verticalement de 10 % de la hauteur
5. Retournez aléatoirement les images horizontalement. Une fois que notre modèle est prêt, nous adaptons l'ensemble de données d'entraînement.

Ce programme est pour augmentation des données **:**

```
datagen = ImageDataGenerator(
        featurewise_center=False,
        samplewise_center=False,
        featurewise_std_normalization=False,
        samplewise_std_normalization=False,
        zca_whitening=False,
        rotation_range = 30,
        zoom_range = 0.2,
        width_shift_range=0.1,
        height_shift_range=0.1,
        horizontal_flip = True,
        vertical_flip=False)


datagen.fit(x_train)
```

Figure (III.73) Programme pour augmentation des données

- ✓ **Construction du modèle CNN**

Ce programme est la structure du modèle :

```
model = Sequential()
model.add(Conv2D(32 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu' , input_shape = (150,150,1)))
model.add(BatchNormalization())
model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))
model.add(Conv2D(64 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))
model.add(Dropout(0.1))
model.add(BatchNormalization())
model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))
model.add(Conv2D(64 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))
model.add(BatchNormalization())
model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))
model.add(Conv2D(128 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))
model.add(Dropout(0.2))
model.add(BatchNormalization())
model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))
model.add(Conv2D(256 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))
model.add(Dropout(0.2))
model.add(BatchNormalization())
model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))
model.add(Flatten())
model.add(Dense(units = 128 , activation = 'relu'))
model.add(Dropout(0.2))
model.add(Dense(units = 1 , activation = 'sigmoid'))
model.compile(optimizer = "rmsprop" , loss = 'binary_crossentropy' , metrics = ['accuracy'])
model.summary()
```

Figure (III.74) : Construction du modèle CNN

III.3.9.2. Résultats du classificateur

- ✓ **Récapitulatif du classificateur CNN**

Maintenant, on va visualiser notre modèle en utilisant l'instruction suivante :
" Model.summary() "

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 150, 150, 32)      320
_________________________________________________________________
batch_normalization_1 (Batch (None, 150, 150, 32)      128
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 75, 75, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 75, 75, 64)        18496
_________________________________________________________________
dropout_1 (Dropout)          (None, 75, 75, 64)        0
_________________________________________________________________
batch_normalization_2 (Batch (None, 75, 75, 64)        256
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 38, 38, 64)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 38, 38, 64)        36928
_________________________________________________________________
batch_normalization_3 (Batch (None, 38, 38, 64)        256
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 19, 19, 64)        0
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 19, 19, 128)       73856
_________________________________________________________________
dropout_2 (Dropout)          (None, 19, 19, 128)       0
_________________________________________________________________
batch_normalization_4 (Batch (None, 19, 19, 128)       512
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 10, 10, 128)       0
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 10, 10, 256)       295168
_________________________________________________________________
dropout_3 (Dropout)          (None, 10, 10, 256)       0
_________________________________________________________________
batch_normalization_5 (Batch (None, 10, 10, 256)       1024
_________________________________________________________________
max_pooling2d_5 (MaxPooling2 (None, 5, 5, 256)         0
_________________________________________________________________
flatten_1 (Flatten)          (None, 6400)              0
_________________________________________________________________
dense_1 (Dense)              (None, 128)               819328
_________________________________________________________________
dropout_4 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 129
=================================================================
Total params: 1,246,401
Trainable params: 1,245,313
Non-trainable params: 1,088
_________________________________________________________________
```

Figure (III.75) les différentes couches de notre architecture de CNN

✓ **Matrice de confusion du classificateur CNN**

Tableau (III.59) : Matrice de confusion du classificateur CNN

	Prédite		
Actual	**Etat**	**Pneumonia**	**Normal**
	Pneumonia	366	24
	Normal	28	206

Source : préparation de l'étudiant à l'aide Python

D'après le tableau, nous remarquons que la classification correcte est **572** répartis en deux cas **TP = 366** et **TN = 206**, mais la classification incorrecte est **52** répartis en deux cas **FP = 24** et **FN = 28**.

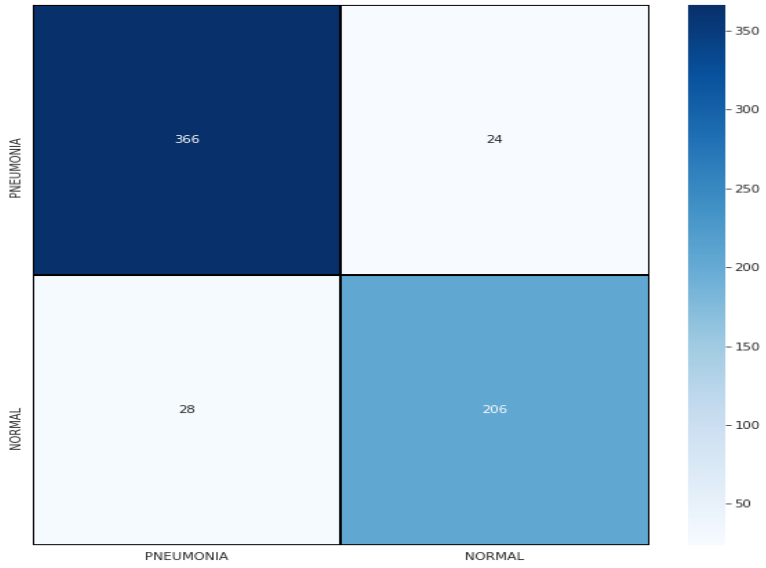


Figure (III.76) : Matrice de confusion du classificateur CNN

(Zone de chaleur)

✓ **Evaluation du classificateur CNN**

○ **Présentation numérique**

Tableau (III.60) : Précision du Modèle CNN

Précision du Modèle								
Accuracy d'Apprentissage : 91.9%				**Accuracy de validation : 75.0%**				
Test								
Accuracy	**Précision**	**Sensibilité**	**Spécificité**	**Nég-Prédictif**	**F-Score**	**AUC**	**Gini**	**Heure de création**
91.6%	92.8%	93.8%	88.0%	89.5%	93.4%	95.7%	91.4%	Supérieur 1min

Source : préparation de l'étudiant à l'aide Python

Du tableau, nous remarquons ce qui suit :

- **Accuracy d'apprentissage** est 91.9%et **Accuracy de validation** est 75.0%
- **Accuracy** de test est 91.6%
- **Précision** de test est 92.8%
- **Sensibilité** de test est 93.8%
- **Spécificité** de test est 88.0%
- **Nég-Prédictif** de test est 89.5%
- **F-Score** de test est 93.4%
- **AUC** de test est 95.7%
- **Gini** de test est 91.4%
- **Heure de création** est Supérieur 1min

○ **Présentation graphique**

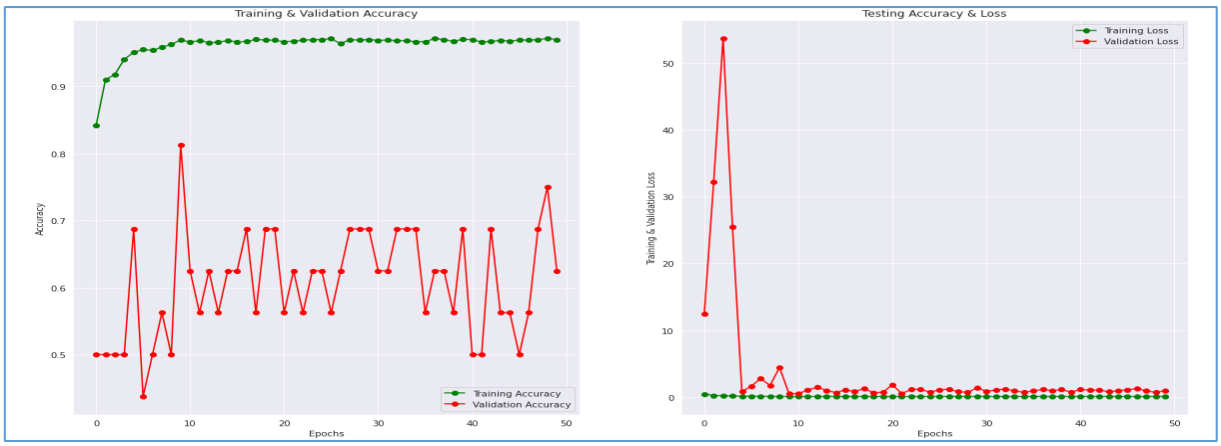


Figure (III.77) : Accuracy et loss d'apprentissage et validation (Epochs =50)

III.3.10. Comparaison entre les deux modèles SVM, NN et CNN

III.3.10.1. Précision d'apprentissage et validation

Tableau (III.61) : Comparaison la précision d'apprentissage et validation pour les deux classificateurs

	Précision du Modèle	
Modèle	**Accuracy d'Apprentissage**	**Accuracy de validation**
SVM	84.6 %	89.6 %
NN	86.2 %	87.9 %
CNN	91.9 %	75.0 %

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0 et Python

Dans le tableau, nous remarquons que la plus grande valeur est pour **Accuracy d'Apprentissage** est 91.9 % pour le modèle **CNN**, et la plus petite valeur est 84.6 % pour le modèle **SVM**, mais la plus grande valeur est pour **Accuracy de validation** 89.6 % pour le modèle **SVM**, et la plus petite valeur est 75.0 % pour le modèle **CNN**.

III.3.10.2. Précision de test

Tableau (III.62) : Comparaison la précision d'apprentissage et validation pour les deux classificateurs

Précision du Modèle									
Test									
Modèle	**Accuracy**	**Précision**	**Sensibilité**	**Spécificité**	**Nég-Prédictif**	**F-Score**	**AUC**	**Gini**	**Heure de création**
SVM	86.0%	89.1%	92.3%	68.5%	76.1%	90.6%	91.3%	82.6%	12s
NN	86.1%	88.1%	93.6%	65.0%	78.4%	90.6%	93.3%	86.5%	12s
CNN	91.6%	92.8%	93.8%	88.0%	89.5%	93.4%	95.7%	91.4%	Supérieur 1min

Source : préparation de l'étudiant à l'aide IBM SPSS Modeler V 18.0 et Python

Nous utiliserons le tableau correspondant pour pondérer et comparer les modèles

Tableau (III.63) : les coefficients de préférence

L'intervalle	Coefficients
60.0 % ----------65.0 %	1
65.1 % ----------70.1 %	2
70.2 % ----------75.2 %	3
75.3 % ----------80.3 %	4
80.4 % ----------85.4 %	5
85.5 % ----------90.5 %	6
90.6 % ----------95.6 %	7
95.7 % ----------100 %	8

Tableau (III.64) : Préférence les deux classificateurs

Les Modèles	**SVM**	**NN**	**CNN**
Accuracy	6	6	**7**
Précision	6	6	**7**
Sensibilité	7	7	**7**
Spécificité	2	1	6
Nég-Prédictif	4	4	6
F-Score	7	7	**7**
AUC	7	7	**8**
Gini	5	6	**7**
Total	44	44	55
Préférence	**Rang 2**	**Rang 2**	**Rang 1**

Source : préparation de l'étudiant à l'aide
IBM SPSS Modeler V 18.0 et Python

Du tableau, on remarque que les meilleurs modèles sont les modèles C**NN** avec une valeur pondérée de 55, mais le pire des modèles est le modèle **NN** et SVM, et la figure ci-dessous montre la précision pour chaque modèle.

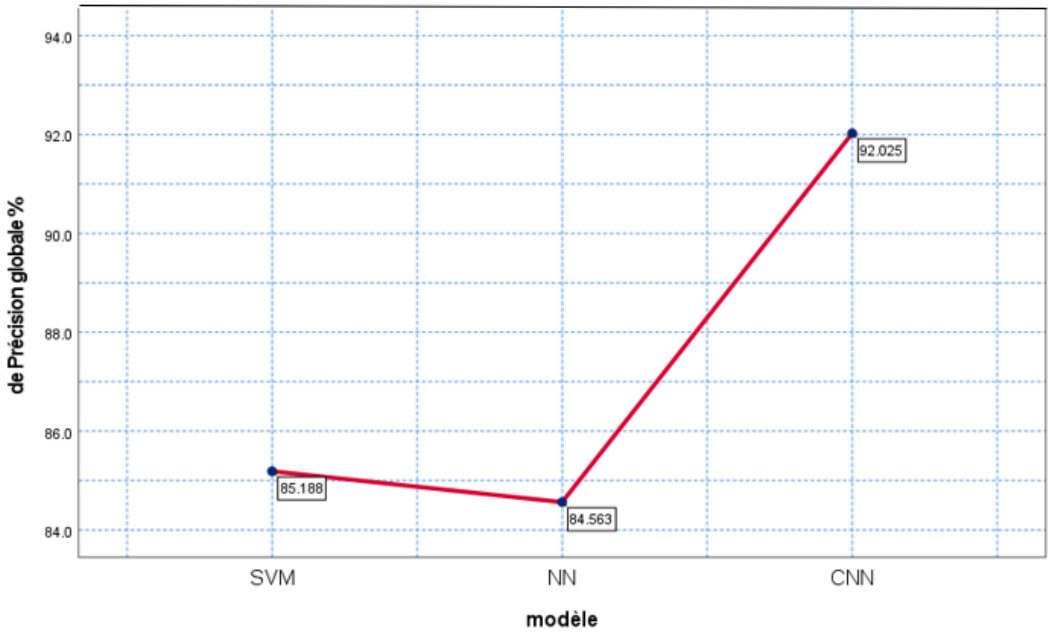


Figure (III.78) : précision globale pour les modèles (SVM, NN, CNN)

III.3.10.3. Décision finale concernant le problème de recherche

A partir des résultats précédents, nous concluons que le meilleur modèle pour prédire l'état est le modèle de **CNN**, suivi des modèles **NN** et **SVM**.

III.4. Conclusion

Dans ce chapitre, un ensemble de modèles différents comme k plus proches voisins, machine à vecteurs de support, réseau bayésien, arbre de décision, régression de logistique, réseau de neurones arbres aléatoire et convolution neural network, a été construit, et en utilisant un ensemble de critères de précision Accuracy, Précision, Sensibilité, Spécificité, Nég-Prédictif, F-Score, AUC et Gini, le modèle CNN a été choisi a été choisi pour être adopté dans la vie pratique.

Conclusion générale

Conclusion générale

La protection des enfants contre la pneumonie est une composante essentielle de la stratégie visant à réduire les taux de mortalité infantile. La vaccination contre Haemophilus influenzae de type B, les pneumocoques, la rougeole et la coqueluche (coqueluche) est le moyen le plus efficace de prévenir la pneumonie.

Il est bien connu qu'une bonne nutrition est essentielle pour améliorer les défenses naturelles des enfants, à commencer par l'allaitement maternel exclusif pendant les six premiers mois de la vie. Outre l'efficacité de cette méthode dans la prévention de la pneumonie, elle contribue également à réduire la période de maladie chez l'enfant.

La lutte contre les facteurs environnementaux tels que la pollution de l'air intérieur (par exemple, en fournissant des poêles propres et peu coûteux pour une utilisation en intérieur) et la promotion d'une bonne hygiène personnelle dans les maisons surpeuplées contribuent également à réduire le nombre d'enfants développant une pneumonie, ainsi qu'à un diagnostic et un traitement efficace. de la pneumonie est essentielle. L'œdème pulmonaire est essentiel pour améliorer les chances de survie de l'enfant.

Le travail que nous avons présenté dans cette thèse s'inscrit dans ce cadre et constitue une modeste contribution à la lutte contre cette épidémie par un diagnostic précis de l'épidémie, le travail consiste à construire et concevoir un ensemble de modèles d'intelligence artificielle pour classer les images radiographiques afin de détecter si une personne a une pneumonie ou non Facilite le processus de diagnostic pour les médecins.

En conséquence, nous avons utilisé une base de données composée d'images radiographiques d'enfants avec et sans cette maladie, la base de données radiographiques était composée de 5856 radiographies thoraciques (antéropostérieures) sélectionnées à partir de cohortes rétrospectives de patients pédiatriques âgés de 1 à 5 ans du centre. Centre médical pour femmes et enfants, Guangzhou. Toutes les radiographies pulmonaires ont été prises dans le cadre des soins cliniques de routine des patients.

Les résultats obtenus ont montré que tous les modèles conçus avaient une précision comprise entre 80.2 % et 91.6 % Le modèle de réseau neuronal convolutif a été choisi.

Références

Références

[1] Senani SAMY, « *Réseaux de neurones convolutionnels pour la détection précoce de la rétinopathie diabétique »* Mémoire de Master, Université Mouloud Mammeri de Tizi-Ouzou, Département Informatique, juillet 2019.

[2] A. Géron, « *Machine Learning avec Scikit-Learn* », Dunod: Paris, 2017.

[3] M. Taffar, « *Initiation à l'apprentissage automatique* », Cours Master, Ment Informatique-Faculté des Sciences Exactes et de l'informatique.

[4] Y. Benzaki, « *Machine Learning made easy* », https://mrmint.fr/, Mai 2023.

[5] J. Brownlee, « *Difference Between Classification and Regression in Machine Learning* », https://machinelearningmastery.com/classification-versus-regression-in-machine-learning/,Mai 2023.

[6] J. B. Metomo, « *Machine Learning : Introduction à l'apprentissage automatique* » https://www.supinfo.com/articles/single/6041-machine-learning-introduction-apprentissage-automatique, Mai 2023.

[7] I. Bellin, N. Vayatis, J. Audiffren, S. Peignier A. Nicolai « *Apprentissage par renforcement* », https://dataanalyticspost.com/Lexique/apprentissage-par-renforcement, Mai 2023.

[8] M. Zaffagni, « *Cette IA a appris à conduire une voiture autonome en 20 minutes* », https://www.futura-sciences.com/tech/actualites/intelligence-artificielle-cette-ia-appris-conduire-voiture-autonome-20-minutes-71942/, Mai 2023.

[9] M. Parizeau, « *Le perceptron multicouche et son algorithme de rétropropagation des erreurs* », Université Laval, Département de génie électrique et de génie informatique, Septembre 2004.

[10] J. Y Baudot, « *Les distances* », http://www.jybaudot.fr/Analdonnees/distances.html, Mai 2023.

[11] J. Grus, « *Data science par la pratique Eyrolles* », Eyrolles : Paris, 2017.

[12] B. Sarra, K. Ikram, « *Algorithmes de classification pour la prévision des maladies agricoles* », *Mémoire de Master,* Ecole Nationale Supérieure d'informatique, Option Système informatique, 2017.

[13] D. Bensalem, C. Bounouar, Z. Boudia, Classification automatique de documents : de la Classification classique à la classification utilisant une ressource externe, Département Informatique de l'UMMTO, 2014

[14] *Aggarwal C. C., Yu P. S.: On Variable Constraints in Privacy-Preserving Data Mining. SIAM Conference, 2005.*

[15] Aggarwal C. C.: On Randomization, Public Information and the Curse of Dimensionality.
ICDE Conference, 2007.

[16] A-G. Bosser. Réplications Distribuées pour la Définition des Interactions de Jeux Massivement Multi-Joueurs. PhD thesis, Université de Paris 7, France, Novembre 2005.

[17] Les arbres de décision (decisiontrees), Christine Decaestecker, ULB, Marco Saerens, UCL, LINF2275.

[18] I. Foster and C. Kesselmann. The Grid: Blueprint for Future Computing Infrastructure. Morgan Kaufmann, San Francisco, 1999.

[19] « *Les réseaux de Neurones Artificiels* », Cours, Université de M‟SILA, Faculté des Sciences et Sciences de l'Ingénieur, Département d'électrotechnique, 2006.

[20] V. Mathivet, « *L'Intelligence Artificielle pour les développeurs* », Eni : France, Décembre 2014.

[21] A. Azencott, « *Utilisez des modèles supervisés non linéaires* », https://openclassrooms.com/fr/courses/4470406-utilisez-des-modeles-supervises-non-lineaires/, Mai 2023.

[22] A. L. Jacquart, « *Le contraste en photographie : prise de vue, traitement, expressivité* » http://www.questionsphoto.com/le-contraste-en-photographie-prise-de-vue-traitement-expressivite, Mai 2023.

[23] P. Monasse, K. Nadjahi, « *Classez et segmentez des données visuelles* », https://openclassrooms.com/fr/courses/4470531-classez-et-segmentez-des-donnees-visuelles, Mai 2023.

[24] S. Bédard-Venne, « *Réseau de neurones artificiels et application à la classification d'images* », Mémoire de Master, Université du Québec à Montréal, Département Mathématiques, Mai 2018.

[25] A. Amidi, S. Amidi, « *Pense-bête VIP : Réseaux de neurones convolutionnels* », https://stanford.edu/~shervine/l/fr/teaching/cs-230/pense-bete-reseaux-neurones-convolutionnels, Mai 2023.

[26] NAOUAR, Fatiha « Algorithme approximatif pour le diagnostic médical basé sur un réseau bayésien possibiliste guidé par ontologie floue » université Sousse, 2007

[27] Etude comparative entre réseaux bayésien et autre méthode de classification appliqué sur une base cardiologue, Thèse de Master, université Tlemcen 2012.

[28] Patrick Naïm, Pierre-Henri Wuillemin, Philippe Leray, Olivier Pourret, Anna Becker "Réseaux bayésiens », Livre, Groupe Eyrolles, 2008.

[29] Olivier PARENT Julien EUSTACHE, Les Réseaux Bayésiens A la recherche de la verité, **2007.**

[30] Marius Kwémou Djoukoué, « *Reduction de dimension en régression logistique application aux données actu-palu »* these de doctrant, Université Gaston Berger, Laboratoire de Mathématique et Modélisation d' evry, septembre 2014.

Annexes

Annexe A

Résultats obtenus par SPSS Statistique V28.0
pour la description des données(Normal et Pneumonia)
(Apprentissage, validation et test)

Etat

		Fréquence	Pourcentage	Pourcentage valide	Pourcentage cumulé
Valide	Normal	1583	27.0	27.0	27.0
	Pneumonia	4273	73.0	73.0	100.0
	Total	5856	100.0	100.0	

Tableau croisé Etat * Partitionner

Effectif

		Partitionner			Total
		training	Testing	Validation	
Etat	Normal	1426	140	17	1583
	Pneumo	3841	391	41	4273
Total		5267	531	58	5856

Tableau croisé Etat * Partitionner

			Partitionner			Total
			training	Testing	Validation	
Etat	Normal	Effectif	1426	140	17	1583
		% dans Etat	90.1%	8.8%	1.1%	100.0%
	Pneumo	Effectif	3841	391	41	4273
		% dans Etat	89.9%	9.2%	1.0%	100.0%
Total		Effectif	5267	531	58	5856
		% dans Etat	89.9%	9.1%	1.0%	100.0%

Tableau croisé Etat * Partitionner

Effectif

		Partitionner			Total
		training	testing	validation	
Etat	Normal	1341	234	8	1583
	Pneumonia	3875	390	8	4273
Total		5216	624	16	5856

Tableau croisé Etat * Partitionner

			Partitionner			Total
			training	testing	validation	
Etat	Normal	Effectif	1341	234	8	1583
		% dans Etat	84.7%	14.8%	0.5%	100.0%
	Pneumonia	Effectif	3875	390	8	4273
		% dans Etat	90.7%	9.1%	0.2%	100.0%
Total		Effectif	5216	624	16	5856
		% dans Etat	89.1%	10.7%	0.3%	100.0%

Annexe B

Résultats obtenus par SPSS Statistique V28.0
Statistique Descriptive pour les Paramètres d'images

Statistiques

IMG-Moyenne

Normal	**N**	**Valide**	1583
		Manquant	0
	Moyenne		122.6415
	Médiane		122.3900
	Ecart type		13.53719
	Plage		96.37
	Minimum		73.30
	Maximum		169.67
Pneumonia	**N**	**Valide**	4273
		Manquant	0
	Moyenne		122.8460
	Médiane		122.8800
	Ecart type		19.89256
	Plage		162.81
	Minimum		58.72
	Maximum		221.53

Statistiques

IMG-Médiane

Normal	**N**	**Valide**	1583
		Manquant	0
	Moyenne		131.8853
	Médiane		132.0000
	Ecart type		15.67949
	Plage		128.00
	Minimum		63.00
	Maximum		191.00
Pneumonia	**N**	**Valide**	4273
		Manquant	0
	Moyenne		133.3527
	Médiane		135.0000
	Ecart type		25.65615
	Plage		171.00
	Minimum		56.00
	Maximum		227.00

Statistiques

IMG-Ecart type

Normal	**N**	**Valide**	1583
		Manquant	0
	Moyenne		60.8545
	Médiane		61.0880
	Ecart type		5.81723
	Plage		50.29
	Minimum		32.39
	Maximum		82.68
Pneumonia	**N**	**Valide**	4273
		Manquant	0
	Moyenne		55.0377
	Médiane		54.4140
	Ecart type		9.91736
	Plage		67.38
	Minimum		19.90
	Maximum		87.28

Statistiques

IMG-Asymétrie

Normal	**N**	**Valide**	1583
		Manquant	0
	Moyenne		-.49422
	Médiane		-.49200
	Ecart type		.227634
	Plage		2.252
	Minimum		-1.468
	Maximum		.784
Pneumonia	**N**	**Valide**	4273
		Manquant	0
	Moyenne		-.60762
	Médiane		-.59800
	Ecart type		.441944
	Plage		3.675
	Minimum		-2.764
	Maximum		.911

Statistiques

IMG-Kurtosis

Normal	**N**	**Valide**	1583
		Manquant	0
	Moyenne		-.63348
	Médiane		-.68900
	Ecart type		.396301
	Plage		3.571
	Minimum		-1.498
	Maximum		2.073
Pneumonia	**N**	**Valide**	4273
		Manquant	0
	Moyenne		-.45216
	Médiane		-.69300
	Ecart type		.907823
	Plage		10.851
	Minimum		-1.610
	Maximum		9.241

Statistiques

IMG-Plage

Normal	**N**	**Valide**	1583
		Manquant	0
	Moyenne		243.9577
	Médiane		247.0000
	Ecart type		12.23397
	Plage		80.00
	Minimum		175.00
	Maximum		255.00
Pneumonia	**N**	**Valide**	4273
		Manquant	0
	Moyenne		218.0112
	Médiane		215.0000
	Ecart type		22.37966
	Plage		146.00
	Minimum		109.00
	Maximum		255.00

Statistiques

IMG-Percentile10

Normal	**N**	**Valide**	1583
		Manquant	0
	Moyenne		23.8866
	Médiane		19.0000
	Ecart type		23.03250
	Plage		113.00
	Minimum		.00
	Maximum		113.00
Pneumonia	**N**	**Valide**	4273
		Manquant	0
	Moyenne		34.3337
	Médiane		31.0000
	Ecart type		27.69048
	Plage		194.00
	Minimum		.00
	Maximum		194.00

Statistiques

IMG-Percentile25

Normal	**N**	**Valide**	1583
		Manquant	0
	Moyenne		81.8271
	Médiane		83.0000
	Ecart type		18.87057
	Plage		142.00
	Minimum		.00
	Maximum		142.00
Pneumonia	**N**	**Valide**	4273
		Manquant	0
	Moyenne		86.1608
	Médiane		84.0000
	Ecart type		28.34788
	Plage		212.00
	Minimum		.00
	Maximum		212.00

Statistiques

IMG-Percentile75

Normal	**N**	**Valide**	1583
		Manquant	0
	Moyenne		172.1003
	Médiane		172.0000
	Ecart type		13.95113
	Plage		128.00
	Minimum		97.00
	Maximum		225.00
Pneumonia	**N**	**Valide**	4273
		Manquant	0
	Moyenne		168.7105
	Médiane		171.0000
	Ecart type		21.52777
	Plage		165.00
	Minimum		70.00
	Maximum		235.00

Statistiques

IMG-Percentile90

Normal	**N**	**Valide**	1583
		Manquant	0
	Moyenne		195.3579
	Médiane		196.0000
	Ecart type		13.62739
	Plage		126.00
	Minimum		121.00
	Maximum		247.00
Pneumonia	**N**	**Valide**	4273
		Manquant	0
	Moyenne		185.4284
	Médiane		187.0000
	Ecart type		18.79978
	Plage		149.00
	Minimum		92.00
	Maximum		241.00

Annexe C

Résultats obtenus par SPSS Statistique V28.0
Pour tester la distribution les Paramètres d'images
dans les deux cas (Normal et Pneumonia)

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=1.25 P=0.0(K=0.03 P=0.01)	mean=122.64 stddev=13.53	☑
○	Gamma	A=1.39 P=0.01(K=0.02 P=0.08)	scale=0.66 shape=81.29	☑
○	Lognormale	A=2.2 P=0.0(K=0.03 P=0.01)	a=121.89 b=0.11	☑
○	Weibull	A=15.59 P=0.01(K=0.06 P=0.01)	a=128.73 b=9.37 c=0.0	☑
○	Triangulaire	A=64.66 P=.(K=0.12 P=.)	max=169.67 min=73.3 mode=...	☑
○	Uniforme	A=216.71 P=0.0(K=0.24 P=0.0)	max=169.67 min=73.3	☑
○	Exponentielle	A=579.89 P=0.0(K=0.52 P=0.01)	scale=0.01	☑
○	Empirique		min=73.3 max=169.67	☑
○	Bêta	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=1.73 P=0.0(K=0.02 P=0.01)	mean=122.85 stddev=19.89	☑
○	Gamma	A=11.58 P=0.01(K=0.03 P=0.01)	scale=0.3 shape=36.47	☑
○	Weibull	A=13.45 P=0.01(K=0.04 P=0.01)	a=131.23 b=6.69 c=0.0	☑
○	Lognormale	A=20.86 P=0.0(K=0.05 P=0.01)	a=121.17 b=0.17	☑
○	Triangulaire	A=391.07 P=.(K=0.2 P=.)	max=221.53 min=58.72 mode...	☑
○	Uniforme	A=858.08 P=0.0(K=0.36 P=0.0)	max=221.53 min=58.72	☑
○	Exponentielle	A=1389.24 P=0.0(K=0.47 P=0....	scale=0.01	☑
○	Empirique		min=58.72 max=221.53	☑
○	Bêta	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=3.02 P=0.0(K=0.05 P=0.01)	mean=131.89 stddev=15.67	☑
○	Gamma	A=4.45 P=0.01(K=0.05 P=0.01)	scale=0.52 shape=69.06	☑
○	Lognormale	A=6.15 P=0.0(K=0.06 P=0.01)	a=130.93 b=0.12	☑
○	Weibull	A=16.48 P=0.01(K=0.08 P=0.01)	a=138.82 b=8.75 c=0.0	☑
○	Triangulaire	A=112.31 P=.(K=0.17 P=.)	max=191.0 min=63.0 mode=1...	☑
○	Uniforme	A=267.86 P=0.0(K=0.3 P=0.0)	max=191.0 min=63.0	☑
○	Exponentielle	A=570.42 P=0.0(K=0.51 P=0.01)	scale=0.01	☑
○	Empirique		min=63.0 max=191.0	☑
○	Bêta	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Weibull	A=1.55 P=0.01(K=0.02 P=0.06)	a=143.71 b=5.97 c=0.0	☑
○	Normale	A=9.08 P=0.0(K=0.04 P=0.01)	mean=133.35 stddev=25.65	☑
○	Gamma	A=34.24 P=0.01(K=0.06 P=0.01)	scale=0.18 shape=24.54	☑
○	Lognormale	A=53.14 P=0.0(K=0.08 P=0.01)	a=130.64 b=0.21	☑
○	Triangulaire	A=159.85 P=.(K=0.12 P=.)	max=227.0 min=56.0 mode=1...	☑
○	Uniforme	A=551.94 P=0.0(K=0.27 P=0.0)	max=227.0 min=56.0	☑
○	Exponentielle	A=1285.77 P=0.0(K=0.43 P=0....	scale=0.01	☑
○	Empirique		min=56.0 max=227.0	☑
○	Bêta	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=1.16 P=0.0(K=0.02 P=0.04)	mean=60.85 stddev=5.82	☑
○	Gamma	A=3.38 P=0.01(K=0.04 P=0.01)	scale=1.75 shape=106.28	☑
○	Lognormale	A=5.06 P=0.0(K=0.04 P=0.01)	a=60.57 b=0.1	☑
○	Weibull	A=8.09 P=0.01(K=0.05 P=0.01)	a=63.46 b=11.23 c=0.0	☑
○	Triangulaire	A=139.12 P=.(K=0.19 P=.)	max=82.68 min=32.39 mode=...	☑
○	Uniforme	A=298.06 P=0.0(K=0.32 P=0.0)	max=82.68 min=32.39	☑
○	Exponentielle	A=598.35 P=0.0(K=0.53 P=0.01)	scale=0.02	☑
○	Empirique		min=32.39 max=82.68	☑
○	Bêta	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Gamma	A=4.85 P=0.01(K=0.02 P=0.01)	scale=0.54 shape=29.52	☑
○	Normale	A=5.43 P=0.0(K=0.03 P=0.01)	mean=55.04 stddev=9.92	☑
○	Lognormale	A=9.94 P=0.0(K=0.03 P=0.01)	a=54.11 b=0.19	☑
○	Weibull	A=22.78 P=0.01(K=0.05 P=0.01)	a=59.17 b=6.07 c=0.0	☑
○	Triangulaire	A=137.65 P=.(K=0.11 P=.)	max=87.28 min=19.9 mode=5...	☑
○	Uniforme	A=539.64 P=0.0(K=0.25 P=0.0)	max=87.28 min=19.9	☑
○	Exponentielle	A=1328.43 P=0.0(K=0.46 P=0....	scale=0.02	☑
○	Empirique		min=19.9 max=87.28	☑
○	Bêta	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=2.22 P=0.0(K=0.03 P=0.01)	mean=-0.49 stddev=0.23	☑
○	Triangulaire	A=187.58 P=.(K=0.23 P=.)	max=0.78 min=-1.47 mode=-0.8	☑
○	Uniforme	A=338.16 P=0.0(K=0.36 P=0.0)	max=0.78 min=-1.47	☑
○	Empirique		min=-1.47 max=0.78	☑
○	Bêta	Pas d'ajustement		☐
○	Exponentielle	Pas d'ajustement		☐
○	Gamma	Pas d'ajustement		☐
○	Lognormale	Pas d'ajustement		☐
○	Weibull	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=16.67 P=0.0(K=0.05 P=0.01)	mean=-0.61 stddev=0.44	☑
○	Triangulaire	A=411.44 P=.(K=0.21 P=.)	max=0.91 min=-2.76 mode=0....	☑
○	Uniforme	A=844.2 P=0.0(K=0.34 P=0.0)	max=0.91 min=-2.76	☑
○	Empirique		min=-2.76 max=0.91	☑
○	Bêta	Pas d'ajustement		☐
○	Exponentielle	Pas d'ajustement		☐
○	Gamma	Pas d'ajustement		☐
○	Lognormale	Pas d'ajustement		☐
○	Weibull	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=15.32 P=0.0(K=0.07 P=0.01)	mean=-0.63 stddev=0.4	☑
○	Triangulaire	A=229.35 P=.(K=0.29 P=.)	max=2.07 min=-1.5 mode=-1.5	☑
○	Uniforme	A=738.33 P=0.0(K=0.53 P=0.0)	max=2.07 min=-1.5	☑
○	Empirique		min=-1.5 max=2.07	☑
○	Bêta	Pas d'ajustement		☐
○	Exponentielle	Pas d'ajustement		☐
○	Gamma	Pas d'ajustement		☐
○	Lognormale	Pas d'ajustement		☐
○	Weibull	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=241.02 P=0.0(K=0.16 P=0.01)	mean=-0.45 stddev=0.91	☑
○	Triangulaire	A=2646.87 P=.(K=0.55 P=.)	max=9.24 min=-1.61 mode=-1....	☑
○	Uniforme	A=4974.39 P=0.0(K=0.7 P=0.0)	max=9.24 min=-1.61	☑
○	Empirique		min=-1.61 max=9.24	☑
○	Bêta	Pas d'ajustement		☐
○	Exponentielle	Pas d'ajustement		☐
○	Gamma	Pas d'ajustement		☐
○	Lognormale	Pas d'ajustement		☐
○	Weibull	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Weibull	A=39.85 P=0.01(K=0.11 P=0.01)	a=248.79 b=32.0 c=0.0	☑
○	Normale	A=85.83 P=0.0(K=0.18 P=0.01)	mean=243.96 stddev=12.23	☑
○	Gamma	A=92.8 P=0.01(K=0.19 P=0.01)	scale=1.52 shape=372.01	☑
○	Lognormale	A=96.39 P=0.0(K=0.19 P=0.01)	a=243.63 b=0.05	☑
○	Triangulaire	A=543.59 P=.(K=0.43 P=.)	max=255.0 min=175.0 mode=...	☑
○	Exponentielle	A=664.13 P=0.0(K=0.55 P=0.01)	scale=0.0	☑
○	Uniforme	A=1313.72 P=0.0(K=0.59 P=0.0)	max=255.0 min=175.0	☑
○	Empirique		min=175.0 max=255.0	☑
○	Bêta	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Lognormale	A=30.12 P=0.0(K=0.06 P=0.01)	a=216.85 b=0.1	☑
○	Gamma	A=33.08 P=0.01(K=0.06 P=0.01)	scale=0.43 shape=93.41	☑
○	Normale	A=41.32 P=0.0(K=0.07 P=0.01)	mean=218.01 stddev=22.38	☑
○	Weibull	A=76.13 P=0.01(K=0.1 P=0.01)	a=228.15 b=10.76 c=0.0	☑
○	Triangulaire	A=187.09 P=.(K=0.24 P=.)	max=255.0 min=109.0 mode=...	☑
○	Uniforme	A=1487.15 P=0.0(K=0.49 P=0.0)	max=255.0 min=109.0	☑
○	Exponentielle	A=1586.04 P=0.0(K=0.54 P=0....	scale=0.0	☑
○	Empirique		min=109.0 max=255.0	☑
○	Bêta	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Exponentielle	A=-235.12 P=1.0(K=0.26 P=0.0...	scale=0.04	☑
○	Triangulaire	A=-48.17 P=.(K=0.26 P=.)	max=113.0 min=0.0 mode=0.0	☑
○	Normale	A=50.2 P=0.0(K=0.15 P=0.01)	mean=23.89 stddev=23.03	☑
○	Uniforme	A=500.75 P=0.0(K=0.41 P=0.0)	max=113.0 min=0.0	☑
○	Empirique		min=0.0 max=113.0	☑
○	Bêta	Pas d'ajustement		☐
○	Gamma	Pas d'ajustement		☐
○	Lognormale	Pas d'ajustement		☐
○	Weibull	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Exponentielle	A=-177.98 P=1.0(K=0.14 P=0.0...	scale=0.03	☑
○	Normale	A=52.1 P=0.0(K=0.11 P=0.01)	mean=34.33 stddev=27.69	☑
○	Triangulaire	A=712.91 P=.(K=0.32 P=.)	max=194.0 min=0.0 mode=0.0	☑
○	Uniforme	A=2551.33 P=0.0(K=0.54 P=0.0)	max=194.0 min=0.0	☑
○	Empirique		min=0.0 max=194.0	☑
○	Bêta	Pas d'ajustement		☐
○	Gamma	Pas d'ajustement		☐
○	Lognormale	Pas d'ajustement		☐
○	Weibull	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=3.85 P=0.0(K=0.04 P=0.01)	mean=81.83 stddev=18.86	☑
○	Triangulaire	A=109.48 P=.(K=0.18 P=.)	max=142.0 min=0.0 mode=10...	☑
○	Uniforme	A=271.27 P=0.0(K=0.31 P=0.0)	max=142.0 min=0.0	☑
○	Exponentielle	A=436.7 P=0.0(K=0.41 P=0.01)	scale=0.01	☑
○	Empirique		min=0.0 max=142.0	☑
○	Bêta	Pas d'ajustement		☐
○	Gamma	Pas d'ajustement		☐
○	Lognormale	Pas d'ajustement		☐
○	Weibull	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=7.63 P=0.0(K=0.04 P=0.01)	mean=86.16 stddev=28.34	☑
○	Triangulaire	A=305.09 P=.(K=0.19 P=.)	max=212.0 min=0.0 mode=46....	☑
○	Uniforme	A=759.24 P=0.0(K=0.32 P=0.0)	max=212.0 min=0.0	☑
○	Exponentielle	A=892.13 P=0.0(K=0.37 P=0.01)	scale=0.01	☑
○	Empirique		min=0.0 max=212.0	☑
○	Bêta	Pas d'ajustement		☐
○	Gamma	Pas d'ajustement		☐
○	Lognormale	Pas d'ajustement		☐
○	Weibull	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=6.0 P=0.0(K=0.05 P=0.01)	mean=172.1 stddev=13.95	☑
○	Gamma	A=6.42 P=0.01(K=0.05 P=0.01)	scale=0.87 shape=150.32	☑
○	Lognormale	A=7.17 P=0.0(K=0.05 P=0.01)	a=171.53 b=0.08	☑
○	Weibull	A=30.2 P=0.01(K=0.1 P=0.01)	a=178.54 b=12.23 c=0.0	☑
○	Triangulaire	A=184.5 P=.(K=0.25 P=.)	max=225.0 min=97.0 mode=1...	☑
○	Uniforme	A=338.41 P=0.0(K=0.37 P=0.0)	max=225.0 min=97.0	☑
○	Exponentielle	A=619.78 P=0.0(K=0.54 P=0.01)	scale=0.01	☑
○	Empirique		min=97.0 max=225.0	☑
○	Bêta	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Weibull	A=4.51 P=0.01(K=0.03 P=0.01)	a=177.67 b=9.53 c=0.0	☑
○	Normale	A=34.92 P=0.0(K=0.07 P=0.01)	mean=168.71 stddev=21.53	☑
○	Gamma	A=64.46 P=0.01(K=0.09 P=0.01)	scale=0.33 shape=55.3	☑
○	Lognormale	A=82.83 P=0.0(K=0.1 P=0.01)	a=167.19 b=0.14	☑
○	Triangulaire	A=362.15 P=.(K=0.21 P=.)	max=235.0 min=70.0 mode=2...	☑
○	Uniforme	A=823.16 P=0.0(K=0.34 P=0.0)	max=235.0 min=70.0	☑
○	Exponentielle	A=1511.16 P=0.0(K=0.48 P=0....	scale=0.01	☑
○	Empirique		min=70.0 max=235.0	☑
○	Bêta	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Normale	A=7.7 P=0.0(K=0.06 P=0.01)	mean=195.36 stddev=13.62	☑
○	Gamma	A=9.88 P=0.01(K=0.07 P=0.01)	scale=1.03 shape=200.9	☑
○	Lognormale	A=11.35 P=0.0(K=0.07 P=0.01)	a=194.87 b=0.07	☑
○	Weibull	A=23.33 P=0.01(K=0.09 P=0.01)	a=201.6 b=14.74 c=0.0	☑
○	Triangulaire	A=190.06 P=.(K=0.26 P=.)	max=247.0 min=121.0 mode=...	☑
○	Uniforme	A=345.72 P=0.0(K=0.37 P=0.0)	max=247.0 min=121.0	☑
○	Exponentielle	A=634.05 P=0.0(K=0.55 P=0.01)	scale=0.01	☑
○	Empirique		min=121.0 max=247.0	☑
○	Bêta	Pas d'ajustement		☐

Utilisation	Distribution	Statistiques d'ajustement	Paramètres	Réajuster
◉	Weibull	A=21.95 P=0.01(K=0.06 P=0.01)	a=193.58 b=11.28 c=0.0	☑
○	Normale	A=25.06 P=0.0(K=0.07 P=0.01)	mean=185.43 stddev=18.8	☑
○	Gamma	A=42.09 P=0.01(K=0.08 P=0.01)	scale=0.49 shape=91.2	☑
○	Lognormale	A=53.01 P=0.0(K=0.09 P=0.01)	a=184.41 b=0.11	☑
○	Triangulaire	A=438.67 P=.(K=0.23 P=.)	max=241.0 min=92.0 mode=2...	☑
○	Uniforme	A=947.71 P=0.0(K=0.37 P=0.0)	max=241.0 min=92.0	☑
○	Exponentielle	A=1601.35 P=0.0(K=0.51 P=0....	scale=0.01	☑
○	Empirique		min=92.0 max=241.0	☑
○	Bêta	Pas d'ajustement		☐

Annexe D

Résultats obtenus par SPSS Statistique V28.0
Pour tester la déférence entre les deux cas
(Normal et Pneumonia) pour les Paramètres d'images

Statistiques de groupe

	Etat	N	Moyenne	Ecart type	Moyenne d'erreur standard
IMG-Moyenne	Normal	1583	122.6415	13.53719	.34024
	Pneumonia	4273	122.8460	19.89256	.30432

Test des échantillons indépendants

		Test t pour égalité des moyennes			
				Signification	
		t	df	p unilatéral	p bilatéral
IMG-Moyenne	Hypothèse de variances inégales	-.448-	4143.542	.327	.654

Rangs

	Etat	N	Rang moyen :	Somme des rangs
IMG-Médiane	Normal	1583	2771.97	4388034.50
	Pneumonia	4273	2986.49	12761261.50
	Total	5856		

Tests statistiques[a]

	IMG-Médiane
U de Mann-Whitney	3134298.500
W de Wilcoxon	4388034.500
Z	-4.313-
Sig. asymptotique (bilatérale)	<.001

a. Variable de regroupement : Etat

Rangs

	Etat	N	Rang moyen :	Somme des rangs
IMG-Ecart type	Normal	1583	3772.48	5971831.00
	Pneumonia	4273	2615.84	11177465.00
	Total	5856		

Tests statistiques[a]

	IMG-Ecart type
U de Mann-Whitney	2046064.000
W de Wilcoxon	11177465.00
Z	-23.252-
Sig. asymptotique (bilatérale)	<.001

a. Variable de regroupement : Etat

Statistiques de groupe

	Etat	N	Moyenne	Ecart type	Moyenne d'erreur standard
IMG-Asymétrie	Normal	1583	-.49422	.227634	.005721
	Pneumonia	4273	-.60762	.441944	.006761

Test des échantillons indépendants

		Test t pour égalité des moyennes			
				Signification	
		t	df	p unilatéral	p bilatéral
IMG-Asymétrie	Hypothèse de variances inégales	12.804	5275.550	<.001	<.001

Statistiques de groupe

	Etat	N	Moyenne	Ecart type	Moyenne d'erreur standard
IMG-Kurtosis	Normal	1583	-.63348	.396301	.009961
	Pneumonia	4273	-.45216	.907823	.013888

Test des échantillons indépendants

		Test t pour égalité des moyennes			
				Signification	
		t	df	p unilatéral	p bilatéral
IMG-Kurtosis	Hypothèse de variances inégales	-10.609-	5714.325	<.001	<.001

Rangs

	Etat	N	Rang moyen :	Somme des rangs
IMG-Plage	Normal	1583	4303.02	6811687.50
	Pneumonia	4273	2419.29	10337608.50
	Total	5856		

Tests statistiques[a]

	IMG-Plage
U de Mann-Whitney	1206207.500
W de Wilcoxon	10337608.50
Z	-37.896-
Sig. asymptotique (bilatérale)	<.001

a. Variable de regroupement : Etat

Rangs

	Etat	N	Rang moyen :	Somme des rangs
IMG-Percentile10	Normal	1583	2450.84	3879677.50
	Pneumonia	4273	3105.46	13269618.50
	Total	5856		

Tests statistiques[a]

	IMG-Percentile10
U de Mann-Whitney	2625941.500
W de Wilcoxon	3879677.500
Z	-13.193-
Sig. asymptotique (bilatérale)	<.001

a. Variable de regroupement : Etat

Statistiques de groupe

	Etat	N	Moyenne	Ecart type	Moyenne d'erreur standard
IMG-Percentile25	Normal	1583	81.8271	18.87057	.47429
	Pneumonia	4273	86.1608	28.34788	.43366

Test des échantillons indépendants

		Test t pour égalité des moyennes			
				Signification	
		t	df	p unilatéral	p bilatéral
IMG-Percentile25	Hypothèse de variances inégales	-6.743-	4236.398	<.001	<.001

Rangs

	Etat	N	Rang moyen :	Somme des rangs
IMG-Percentile75	Normal	1583	3028.82	4794614.50
	Pneumonia	4273	2891.34	12354681.50
	Total	5856		

Tests statistiques[a]

	IMG-Percentile75
U de Mann-Whitney	3223280.500
W de Wilcoxon	12354681.50
Z	-2.764-
Sig. asymptotique (bilatérale)	.006

a. Variable de regroupement : Etat

Rangs

	Etat	N	Rang moyen :	Somme des rangs
IMG-Percentile90	Normal	1583	3661.09	5795504.50
	Pneumonia	4273	2657.10	11353791.50
	Total	5856		

Tests statistiques[a]

	IMG-Percentile90
U de Mann-Whitney	2222390.500
W de Wilcoxon	11353791.50
Z	-20.187-
Sig. asymptotique (bilatérale)	<.001

a. Variable de regroupement : Etat

Annexe E

Résultats obtenus par SPSS Modeler V18.0
Matrice de confusion (Apprentissage, validation et test)
pour les déférents classificateurs

Résultats du champ de sortie Etat

Modèles individuels

Comparaison de $KNN-Etat avec Etat

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Correct	4 611	87,55 %	457	86,06 %	53	91,38 %
Incorrect	656	12,45 %	74	13,94 %	5	8,62 %
Total	5 267		531		58	

Matrice de coïncidences pour $KNN-Etat (lignes affichant les valeurs réelles)

'Partitionner' = 1_Apprentissage	Normal	Pneumonia
Normal	1 012	414
Pneumonia	242	3 599

'Partitionner' = 2_Test	Normal	Pneumonia
Normal	91	49
Pneumonia	25	366

'Partitionner' = 3_Validation	Normal	Pneumonia
Normal	14	3
Pneumonia	2	39

Métriques d'évaluation

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Modèle	AUC	Gini	AUC	Gini	AUC	Gini
$KNN-Etat	0,944	0,888	0,913	0,826	0,943	0,887

Résultats du champ de sortie Etat

Modèles individuels

Comparaison de $S-Etat avec Etat

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Correct	4 458	84,64 %	457	86,06 %	52	89,66 %
Incorrect	809	15,36 %	74	13,94 %	6	10,34 %
Total	5 267		531		58	

Matrice de coïncidences pour $S-Etat (lignes affichant les valeurs réelles)

'Partitionner' = 1_Apprentissage	Normal	Pneumonia
Normal	991	435
Pneumonia	374	3 467

'Partitionner' = 2_Test	Normal	Pneumonia
Normal	96	44
Pneumonia	30	361

'Partitionner' = 3_Validation	Normal	Pneumonia
Normal	14	3
Pneumonia	3	38

Métriques d'évaluation

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Modèle	AUC	Gini	AUC	Gini	AUC	Gini
$S-Etat	0,9	0,8	0,913	0,826	0,948	0,897

Résultats du champ de sortie Etat

Modèles individuels

Comparaison de $B-Etat avec Etat

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Correct	4 145	78,7 %	426	80,23 %	46	79,31 %
Incorrect	1 122	21,3 %	105	19,77 %	12	20,69 %
Total	5 267		531		58	

Matrice de coïncidences pour $B-Etat (lignes affichant les valeurs réelles)

'Partitionner' = 1_Apprentissage	Normal	Pneumonia
Normal	985	441
Pneumonia	681	3 160

'Partitionner' = 2_Test	Normal	Pneumonia	$null$
Normal	95	45	0
Pneumonia	58	331	2

'Partitionner' = 3_Validation	Normal	Pneumonia
Normal	12	5
Pneumonia	7	34

Métriques d'évaluation

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Modèle	AUC	Gini	AUC	Gini	AUC	Gini
$B-Etat	0,86	0,719	0,871	0,742	0,874	0,747

Résultats du champ de sortie Etat

Modèles individuels

Comparaison de $C-Etat avec Etat

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Correct	4 630	87,91 %	458	86,25 %	49	84,48 %
Incorrect	637	12,09 %	73	13,75 %	9	15,52 %
Total	5 267		531		58	

Matrice de coïncidences pour $C-Etat (lignes affichant les valeurs réelles)

'Partitionner' = 1_Apprentissage	Normal	Pneumonia
Normal	1 088	338
Pneumonia	299	3 542

'Partitionner' = 2_Test	Normal	Pneumonia
Normal	102	38
Pneumonia	35	356

'Partitionner' = 3_Validation	Normal	Pneumonia
Normal	11	6
Pneumonia	3	38

Métriques d'évaluation

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Modèle	AUC	Gini	AUC	Gini	AUC	Gini
$C-Etat	0,917	0,834	0,904	0,807	0,92	0,839

Résultats du champ de sortie Etat

Modèles individuels

Comparaison de $L-Etat avec Etat

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Correct	4 401	83,56 %	453	85,31 %	48	82,76 %
Incorrect	866	16,44 %	78	14,69 %	10	17,24 %
Total	5 267		531		58	

Matrice de coïncidences pour $L-Etat (lignes affichant les valeurs réelles)

'Partitionner' = 1_Apprentissage	Normal	Pneumonia
Normal	949	477
Pneumonia	389	3 452

'Partitionner' = 2_Test	Normal	Pneumonia
Normal	94	46
Pneumonia	32	359

'Partitionner' = 3_Validation	Normal	Pneumonia
Normal	11	6
Pneumonia	4	37

Métriques d'évaluation

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Modèle	AUC	Gini	AUC	Gini	AUC	Gini
$L-Etat	0,895	0,789	0,915	0,83	0,914	0,828

Résultats du champ de sortie Etat

Modèles individuels

Comparaison de $N-Etat avec Etat

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Correct	4 540	86,2 %	457	86,06 %	51	87,93 %
Incorrect	727	13,8 %	74	13,94 %	7	12,07 %
Total	5 267		531		58	

Matrice de coïncidences pour $N-Etat (lignes affichant les valeurs réelles)

'Partitionner' = 1_Apprentissage	Normal	Pneumonia
Normal	1 020	406
Pneumonia	321	3 520

'Partitionner' = 2_Test	Normal	Pneumonia
Normal	91	49
Pneumonia	25	366

'Partitionner' = 3_Validation	Normal	Pneumonia
Normal	12	5
Pneumonia	2	39

Métriques d'évaluation

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Modèle	AUC	Gini	AUC	Gini	AUC	Gini
$N-Etat	0,919	0,838	0,933	0,865	0,948	0,897

Résultats du champ de sortie Etat

Modèles individuels

Comparaison de $R-Etat avec Etat

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Correct	4 576	86,88 %	438	82,49 %	52	89,66 %
Incorrect	691	13,12 %	93	17,51 %	6	10,34 %
Total	5 267		531		58	

Matrice de coïncidences pour $R-Etat (lignes affichant les valeurs réelles)

'Partitionner' = 1_Apprentissage	Normal	Pneumonia
Normal	1 340	86
Pneumonia	605	3 236

'Partitionner' = 2_Test	Normal	Pneumonia
Normal	117	23
Pneumonia	70	321

'Partitionner' = 3_Validation	Normal	Pneumonia
Normal	15	2
Pneumonia	4	37

Métriques d'évaluation

'Partitionner'	1_Apprentissage		2_Test		3_Validation	
Modèle	AUC	Gini	AUC	Gini	AUC	Gini
$R-Etat	0,942	0,884	0,907	0,813	0,923	0,846

Annexe F

Résultats obtenus par SPSS Modeler V18.0
Sortie du classificateurs régression logistique

Informations sur l'ajustement du modèle

Modèle	Critères d'ajustement du modèle			Tests du rapport de vraisemblance		
	AIC	BIC	Log de vraisemblance -2	Khi-deux	ddl	Sig.
Constante uniquement	6836.864	6843.539	6834.864			
Final	4167.841	4227.918	4149.841	2685.023	8	.000

Pseudo R-deux

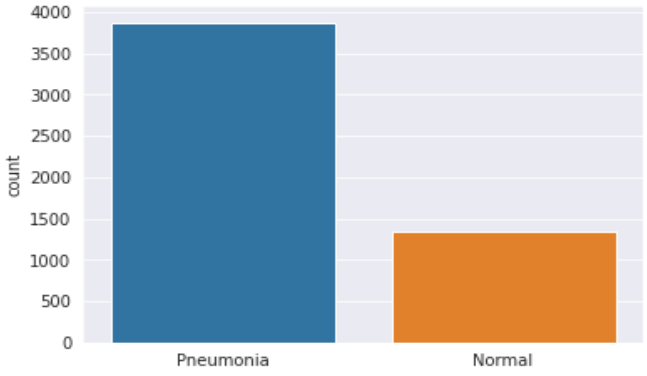
Cox et Snell	.368
Nagelkerke	.534
McFadden	.393

Estimations des paramètres

Etat[a]		B	Erreur standard	Wald	ddl	Sig.	Exp(B)	Intervalle de confiance à 95 % pour Exp(B)	
								Borne inférieure	Borne supérieure
Pneumonia	Constante	23.262	.768	917.529	1	.000			
	IMG-Médiane	-.036-	.012	9.548	1	.002	.964	.943	.987
	IMG-Asymétrie	4.584	.592	59.987	1	.000	97.922	30.695	312.386
	IMG-Kurtosis	2.701	.271	98.996	1	.000	14.899	8.751	25.367
	IMG-Plage	-.067-	.003	559.406	1	.000	.935	.930	.940
	IMG-Percentile10	-.025-	.003	62.550	1	.000	.975	.969	.981
	IMG-Percentile25	-.022-	.006	15.384	1	.000	.979	.968	.989
	IMG-Percentile75	.377	.019	378.570	1	.000	1.458	1.404	1.515
	IMG-Percentile90	-.312-	.015	426.979	1	.000	.732	.710	.754

Annexe G

Résultats obtenus par Python
Sorties du classificateur CNN




```
Epoch 1/50
163/163 [==============================] - 14s 84ms/step - loss: 0.4688 - accuracy: 0.8416 - val_loss: 12.4223 - val_accuracy: 0.5000
Epoch 2/50
163/163 [==============================] - 10s 63ms/step - loss: 0.2494 - accuracy: 0.9099 - val_loss: 32.1245 - val_accuracy: 0.5000
Epoch 3/50
163/163 [==============================] - 10s 64ms/step - loss: 0.2347 - accuracy: 0.9179 - val_loss: 53.6475 - val_accuracy: 0.5000

Epoch 00003: ReduceLROnPlateau reducing learning rate to 0.0003000000142492354.
Epoch 4/50
163/163 [==============================] - 11s 68ms/step - loss: 0.1670 - accuracy: 0.9404 - val_loss: 25.4721 - val_accuracy: 0.5000
Epoch 5/50
163/163 [==============================] - 10s 64ms/step - loss: 0.1424 - accuracy: 0.9509 - val_loss: 0.8120 - val_accuracy: 0.6875
```

..

..

```
Epoch 45/50
163/163 [==============================] - 11s 70ms/step - loss: 0.0952 - accuracy: 0.9674 - val_loss: 0.9513 - val_accuracy: 0.5625
Epoch 46/50
163/163 [==============================] - 12s 74ms/step - loss: 0.0917 - accuracy: 0.9693 - val_loss: 1.1138 - val_accuracy: 0.5000
Epoch 47/50
163/163 [==============================] - 11s 69ms/step - loss: 0.0883 - accuracy: 0.9691 - val_loss: 1.2652 - val_accuracy: 0.5625
Epoch 48/50
163/163 [==============================] - 11s 69ms/step - loss: 0.0824 - accuracy: 0.9697 - val_loss: 0.9599 - val_accuracy: 0.6875
Epoch 49/50
163/163 [==============================] - 12s 73ms/step - loss: 0.0837 - accuracy: 0.9716 - val_loss: 0.6991 - val_accuracy: 0.7500
Epoch 50/50
163/163 [==============================] - 11s 69ms/step - loss: 0.0910 - accuracy: 0.9699 - val_loss: 0.9594 - val_accuracy: 0.6250
```

```
5216/5216 [==============================] - 3s 497us/step
Loss of the model is -  0.2612732592928173
5216/5216 [==============================] - 3s 499us/step
Accuracy of the model (train) is -  91.92867875099182 %
```

```
16/16 [==============================] - 0s 749us/step
Loss of the model is -  0.43926677107810974
16/16 [==============================] - 0s 675us/step
Accuracy of the model (validation) is -  75.0 %
```

Annexe H

Valeurs tabulaires pour le test t

t Table

cum. prob	$t_{.50}$	$t_{.75}$	$t_{.80}$	$t_{.85}$	$t_{.90}$	$t_{.95}$	$t_{.975}$	$t_{.99}$	$t_{.995}$	$t_{.999}$	$t_{.9995}$
one-tail	0.50	0.25	0.20	0.15	0.10	0.05	0.025	0.01	0.005	0.001	0.0005
two-tails	1.00	0.50	0.40	0.30	0.20	0.10	0.05	0.02	0.01	0.002	0.001
df											
1	0.000	1.000	1.376	1.963	3.078	6.314	12.71	31.82	63.66	318.31	636.62
2	0.000	0.816	1.061	1.386	1.886	2.920	4.303	6.965	9.925	22.327	31.599
3	0.000	0.765	0.978	1.250	1.638	2.353	3.182	4.541	5.841	10.215	12.924
4	0.000	0.741	0.941	1.190	1.533	2.132	2.776	3.747	4.604	7.173	8.610
5	0.000	0.727	0.920	1.156	1.476	2.015	2.571	3.365	4.032	5.893	6.869
6	0.000	0.718	0.906	1.134	1.440	1.943	2.447	3.143	3.707	5.208	5.959
7	0.000	0.711	0.896	1.119	1.415	1.895	2.365	2.998	3.499	4.785	5.408
8	0.000	0.706	0.889	1.108	1.397	1.860	2.306	2.896	3.355	4.501	5.041
9	0.000	0.703	0.883	1.100	1.383	1.833	2.262	2.821	3.250	4.297	4.781
10	0.000	0.700	0.879	1.093	1.372	1.812	2.228	2.764	3.169	4.144	4.587
11	0.000	0.697	0.876	1.088	1.363	1.796	2.201	2.718	3.106	4.025	4.437
12	0.000	0.695	0.873	1.083	1.356	1.782	2.179	2.681	3.055	3.930	4.318
13	0.000	0.694	0.870	1.079	1.350	1.771	2.160	2.650	3.012	3.852	4.221
14	0.000	0.692	0.868	1.076	1.345	1.761	2.145	2.624	2.977	3.787	4.140
15	0.000	0.691	0.866	1.074	1.341	1.753	2.131	2.602	2.947	3.733	4.073
16	0.000	0.690	0.865	1.071	1.337	1.746	2.120	2.583	2.921	3.686	4.015
17	0.000	0.689	0.863	1.069	1.333	1.740	2.110	2.567	2.898	3.646	3.965
18	0.000	0.688	0.862	1.067	1.330	1.734	2.101	2.552	2.878	3.610	3.922
19	0.000	0.688	0.861	1.066	1.328	1.729	2.093	2.539	2.861	3.579	3.883
20	0.000	0.687	0.860	1.064	1.325	1.725	2.086	2.528	2.845	3.552	3.850
21	0.000	0.686	0.859	1.063	1.323	1.721	2.080	2.518	2.831	3.527	3.819
22	0.000	0.686	0.858	1.061	1.321	1.717	2.074	2.508	2.819	3.505	3.792
23	0.000	0.685	0.858	1.060	1.319	1.714	2.069	2.500	2.807	3.485	3.768
24	0.000	0.685	0.857	1.059	1.318	1.711	2.064	2.492	2.797	3.467	3.745
25	0.000	0.684	0.856	1.058	1.316	1.708	2.060	2.485	2.787	3.450	3.725
26	0.000	0.684	0.856	1.058	1.315	1.706	2.056	2.479	2.779	3.435	3.707
27	0.000	0.684	0.855	1.057	1.314	1.703	2.052	2.473	2.771	3.421	3.690
28	0.000	0.683	0.855	1.056	1.313	1.701	2.048	2.467	2.763	3.408	3.674
29	0.000	0.683	0.854	1.055	1.311	1.699	2.045	2.462	2.756	3.396	3.659
30	0.000	0.683	0.854	1.055	1.310	1.697	2.042	2.457	2.750	3.385	3.646
40	0.000	0.681	0.851	1.050	1.303	1.684	2.021	2.423	2.704	3.307	3.551
60	0.000	0.679	0.848	1.045	1.296	1.671	2.000	2.390	2.660	3.232	3.460
80	0.000	0.678	0.846	1.043	1.292	1.664	1.990	2.374	2.639	3.195	3.416
100	0.000	0.677	0.845	1.042	1.290	1.660	1.984	2.364	2.626	3.174	3.390
1000	0.000	0.675	0.842	1.037	1.282	1.646	1.962	2.330	2.581	3.098	3.300
z	0.000	0.674	0.842	1.036	1.282	1.645	1.960	2.326	2.576	3.090	3.291
	0%	50%	60%	70%	80%	90%	95%	98%	99%	99.8%	99.9%
	Confidence Level										

ملخص:

في هذا العمل نقدم تصنيف صور الاشعة السينية بهدف الكشف ما إذا كان الشخص مصابا بمرض الالتهاب الرئوي ام لا مما يسهل عملية التشخيص للأطباء ولذلك قمنا بتصميم مجموعة من نماذج الذكاء الاصطناعي وتم المفاضلة بينهم من خلال مقاييس الدقة وكانت دقة النماذج تتراوح بين 80.2 %أ الى 91.6 % وتم اختيار أفضل النماذج الا وهو نموذج الشبكة العصبية الالتفافية **CNN**بدقة تقدر 91.6 % ونشير الى ان النموذج يعتبر نموذج مساعد للطبيب المتخصص وليس بديلا عن الطبيب ولا يحل محله.

الكلمات المفتاحية: تصنيف الصور -التعلم الالي – نماذج التصنيف - الشبكة العصبية الالتفافية – تشخيص مرض الالتهاب الرئوي – SPSS Modeler.

Résumé:

Dans ce travail, nous présentons la classification des images radiographiques afin de détecter si une personne a une pneumonie ou non, ce qui facilite le processus de diagnostic pour les médecins par conséquent, nous avons conçu un ensemble de modèles d'intelligence artificielle, et des comparaisons ont été faites entre eux par des mesures de précision, et la précision des modèles variait de 80.2 % à 91.6 % , les meilleurs modèles ont été choisis, à savoir le modèle de réseau neuronal convolutif, avec une précision de 91.6 % Nous soulignons que le modèle est considéré comme un auxiliaire modèle pour le médecin spécialiste et ne se substitue pas au médecin et ne le remplace pas.

Mots clés : Classification d'images - apprentissage automatique - modèles de classification - réseau de neurones convolutifs - Diagnostic de pneumonie - SPSS Modeler.

Abstract:

In this work, we present the classification of x-ray images in order to detect whether a person has pneumonia or not, which facilitates the diagnosis process for doctors. Therefore, we designed a set of artificial intelligence models, and comparisons were made between them through accuracy measures, and the accuracy of the models ranged from 80.2 % to 91.6 %, the best models were chosen, which is the convolutional neural network model, with an accuracy of 91.6 % we point out that the model is considered an auxiliary model for the specialized doctor and is not a substitute for the doctor and does not replace him.

Keywords: Image classification - machine learning - classification models - convolutional neural network - Diagnosis of pneumonia - SPSS Modeler.