

N° d'ordre :

Mémoire de fin d'études déposé à l'



جامعة محمد بوضياف - المسيلة
University of Mohamed Boudiaf - Msila
UNIVERSITÉ MOHAMED BOUDIAF – MSILA

FACULTÉ DES MATHÉMATIQUES ET DE L'INFORMATIQUE
DÉPARTEMENT DE L'INFORMATIQUE

Pour la satisfaction partielle des exigences du diplôme de

Master en Informatique

Par

BOUTERAA YUCEF

Titre du mémoire

Améliorer la classification de CNN à l'aide des filtres pour la reconnaissance des images

Sous la direction de

KHADIDJA DERDOUR

Composition du jury

RAHIMA BENTERCIA

Université de M'sila

Président

AHLEM BOUZAROURA

Université de M'sila

Examineur

Juin.2024

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Remerciements

Je veux exprimer par ces quelques lignes de remerciement notre gratitude envers tous ceux en qui par leur présence, leur soutien, leur disponibilité et leurs conseils, nous avons eu courage D'accomplir ce travail Avant tout, il apparait opportun de rendre grâce à DIEU de nous avoir donné le courage et La Volonté et la patience et le savoir-faire afin d'accomplir ce travail. Je tiens tout d'abord à remercier mon encadreur Dr. DERDOUR KHADIDJA, enseignante à l'Université de M'sila, pour tout le soutien, l'aide, l'orientation, la guidance, ainsi que pour ses précieux

Conseils et ses encouragements lors de la réalisation de mon mémoire. J'ai beaucoup appris à son Contact et ce fut un grand plaisir de travailler avec elle. Je voudrai lui adresser mes vifs Remerciements et de lui témoigner mon sincère reconnaissance.

Je remercierai les membres du jury de j'avoir fait l'honneur de faire partie de mon jury de ce travail.

Je remercierai également toutes les personnes qui, de près ou de loin, m'aident à l'élaboration de ce mémoire.

Je tiens ensuite à remercier mes parents...Ma chère mère, mon soutien et la lumière de mon chemin, celle qui m'a appris la détermination et la persévérance, source d'espoir et d'ambition, mon cher père, comme je suis heureux de te voir fier et heureux de ma réussite.

Merci à tous les amies et les membres de la famille pour leur aides leur soutien et leurs encouragements dans les moments difficiles.

Résumé

La classification des images a fait des progrès significatifs grâce au développement des réseaux de neurones convolutifs. Dans cette note, nous présentons un modèle de réseau neuronal convolutif simple et efficace sur l'ensemble de données MNIST, l'un des ensembles de données les plus populaires utilisés dans ce domaine, composé de 70 000 images en niveaux de gris. De nombreux manuscrits de 0 à 9, répartis en 60 000 images pour l'entraînement et 10 000 images pour les tests. Dans le but d'essayer d'améliorer le modèle en termes de performances en utilisant les filtres suggérés et en comparant ses performances avec un autre modèle qui utilise des filtres par défaut, il a montré des résultats de classification supérieurs à 96 % et inférieurs au modèle qui utilisait des filtres par défaut, et les performances obtenues étaient acceptables en termes de précision et de faible erreur de prédiction.

Mots clés : traitement des images, CNN, filtres

ملخص

حقق تصنيف الصور تقدماً كبيراً بفضل تطوير الشبكات العصبية التلافيفية. في هذه المذكرة، نقدم نموذج شبكة عصبية تلافيفية بسيط وفعال على مجموعة بيانات MNIST، وهي واحدة من مجموعات البيانات الأكثر شيوعاً المستخدمة في هذا المجال، وتتكون من 70000 صورة ذات تدرج رمادي. من الأرقام المكتوبة بخط اليد من 0 إلى 9، مقسمة إلى 60.000 صورة للتدريب و10.000 صورة للاختبار. بهدف محاولة تحسين النموذج من حيث الأداء باستخدام مرشحات مقترحة ومقارنة ادائه مع نموذج آخر يستخدم الفلاتر الافتراضية. أظهر نتائج تصنيفية أعلى من 96% وكان أقل من النموذج الذي يستخدم الفلاتر بشكل افتراضي، وكان الأداء الذي تم الحصول عليه مقبولاً من حيث الدقة وانخفاض خطأ التنبؤ.

الكلمات المفتاحية : معالجة الصور, الشبكة العصبية التلافيفية, المرشحات

Abstract

Image classification has made great progress thanks to the development of convolutional neural networks. In this work, we present a simple and efficient convolutional neural network model on the MNIST dataset, one of the most popular datasets used in this field, consisting of 70,000 grayscale images of handwritten numbers from 0 to 9, divided into 60,000 images for training and 10,000 images for testing. In order to try to improve the model in terms of performance using the proposed filters and compare its performance with another model that uses the default filters. It showed classification results above 96% and was lower than the model that uses the default filters, and the obtained performance was acceptable in terms of accuracy and low prediction error.

Key words : preprocessing, CNN, Filters

Liste des abréviations

IA : Intelligence artificiel

CNN: Convolution neural network

ML: machine learning

DL: deep learning

ReLU: Rectified Linear Unite

ConvNet : **convolution** neural network

LOSS : couche de perte

Pixels : picture Elément

Res Net : Residual Neural Network

Tanh : Tangente Hyperbolique

FC : Full Connexion

GPU : Graphical Processing Unit

Table des matières

Introduction Generale	1
Chapiter01 : Reconnaissance De Formes	2
1 Introduction	3
2 Pretraitement Des Donnees.....	3
2.1 Définition	3
2.2 Importance Du Prétraitement Des Données.....	3
2.3 Caractéristiques Du Prétraitement Des Données.....	3
2.3.1 Validation Des Données	3
2.3.2 Imputation Des Données.....	3
2.4 Etapes Clés Du Prétraitement Des Données	4
3 Apprentissage Automatique	5
3.1 Définition	5
3.2 Applications De L'apprentissage Automatique	6
3.3 Avantages De L'apprentissage Automatique	6
3.4 Inconvénients De L'apprentissage Automatique	6
3.5 Types Des Apprentissage Automatique.....	6
3.5.1 Apprentissage Supervise.....	7
3.5.2 Apprentissage Non Supervisé.....	8
3.5.3 Apprentissage Par Renforcement	9
4 Classifieurs	9
4.1 Régression Logistique	9
4.2 Arbres De Décision	9
4.3 Machines Vectorielles De Support	10
4.4 K Plus Prochesvoisins :	11
4.5 Clustering K-Means	10
4.6 Réseaux Neuronaux.....	10
5 Conclusion	11
Chapiter 02 : Reseaux De Neurones Convolutionnels.....	13
1 Introduction	14
2 Apprentissage Profond.....	14
2.1 Définition	14
2.2 Principe Fondamental De L'apprentissage Profond	14
2.3 Paramètres De L'apprentissage Profond :	14
3 Neurones Convolutionnel.....	15
3.1 Définition.....	15
3.2 Caractéristiques Des Images Numériques.....	16
3.3 Opérations Des Réseaux De Neurones De Convolution :	18
3.4 Filtre Des Images Numériques	18
4 Principe De Fonctionnement D'un Reseau Cnn.....	22
4.1 Couche De Convolution.....	22

4.2	Couche Pooling	23
4.3	Entièrement Connecté.....	23
4.4	Fonction D'activation	24
4.5	Convnet Les Plus Connus	25
Chapiter 03 : Résultats Et Discussion		27
1	Introduction	28
2	Bases Des Donnes D'images.....	28
3	Logiciels, Librairies Et Matériels Utilisés Dans L'implémentation.....	28
4	Résultats Obtenus Et Discussion	29
4.1	Premier Modèle	29
4.2	Architecture De Modèle 1	29
4.3	Graphe De Précision Et D'erreur Du Model 1.....	30
4.4	Analyse Graphique.....	31
4.5	Deuxième Modèle.....	31
4.6	Architecture De Modèle Proposé	32
4.7	Types Des Filtrés Appliqués Au Deuxième Modèle :.....	32
4.8	Graphe De Précision Et D'erreur Du Model 2.....	33
4.9	Analyse Graphique.....	34
4.10	Tableau De Comparaison Des Résultats.....	35
5	Conclusion	36
Conclusion Générale.....		37
Bibliographie		38

Liste Des Figures

Figure 1:Apprentissage Automatique	5
Figure 8:Paradigmes D'apprentissage Automatique.....	7
Figure 9:Exemple De Classification	7
Figure 10:Exemple De Régression.....	7
Figure 11 :Exemple De Regroupement	8
Figure 12 :Exemple Réduction De La Dimensionnalité.....	8
Figure 2 : Exemple De Régression Logistique.....	9
Figure 3 :Exemple De Arbres De Décision	9
Figure 4:Exemple De Svm.....	10
Figure 5: Exemple De KNN.....	11
Figure 6 :Exemple De K-Means.....	10
Figure 7:Reseux De Neurone	11
Figure 13:Apprentissage Profonde	14
Figure 14 :Abandon Appliqué A Un Réseau Neuronal Standard	15
Figure 15 :Convolution Neural Network	15
Figure 16 :Une Représentation De La Façon Dont Le Filtre Passe Et Les Champs Récepteurs Sont Formés	16
Figure 17 :Extraction De Caractéristiques De Lion	16
Figure 18: Images Sous Forme De Grilles	17
Figure 19:Exemple De Remplissage	18
Figure 20 : Le Fonctionnement De Stride.....	18
Figure 21 :Une Collection De Filtres Populaires Avec Leurs Poids Et Sorties	19
Figure 22 :Filter Median	19
Figure 23 :Normal Convolution Vs Dilated Convolution.....	20
Figure 24 :Filter Cach	20
Figure 25 :Filterc Fermeture	20
Figure 26:Filtrage Inverse : (A) Original Image, (B) Floating Image, (C) Inverse Filtered Image..	21
Figure 27 :Filter Gaufrage	21
Figure 28:Filter Sharpening.....	21
Figure 29: Principe Fondamental De Apprentissage Profond.....	22
Figure 30 : Exemple Explicative Sur L'opération De Convolution.....	22
Figure 31:Pooling Max	23
Figure 32: Couche Fully-Connected.....	23
Figure 33:Fonction D'activation Linear	24
Figure 34:Fonction D'activation Step	24
Figure 35 :Fonction D'activation Sigmoid.....	24
Figure 36 :Fonction D'activation Tanh.....	24
Figure 37:Fonction D'activation Relu.....	25
Figure 38 : Diagramme De Structure Du Réseau Neuronal Convolutionnel Letnet	25
Figure 39 : Diagramme De Structure Du Réseau Neuronal Convolutionnel Googlenet	26
Figure 40 : Diagramme De Structure Du Réseau Neuronal Convolutionnel Alexnet	26
Figure 41 : Graphe Test De Précision Modèle 1.....	30
Figure 42:Test De Précision	30
Figure 45 :Compter Le Nombre D'images Mal Classees	30
Figure 43 :: Graphique D'erreur.....	31
Figure 46 :Les Filtres Appliqués Au Deuxieme Modèle	33
Figure 47 : Graphe De Précision Model 2.....	33
Figure 48:Test De Précision.....	34
Figure 49 :Ratio D'erreur De Classification.....	34
Figure 50 :Compter Le Nombre D'images Mal Classées	34
Figure 52 :Graphe D'erreur	34

Liste des tableaux

Tableau 01 :Résumé du modèle 01.....	30
Tableau 02 :Résumé du modèle 02.....	32
Tableau 03: Tableau de comparaison des résultats	36

Introduction Générale

À l'ère numérique d'aujourd'hui, la classification d'images est devenue une application essentielle dans le domaine de l'intelligence artificielle et de la vision par ordinateur. Les techniques de classification d'images sont utilisées dans un large éventail de domaines tels que la médecine, la sécurité, la conduite autonome, l'analyse de données et la reconnaissance faciale. L'ensemble de données MNIST joue un rôle central en tant que référence pour tester et évaluer les modèles d'apprentissage profond dans la classification des chiffres manuscrits. L'ensemble de données MNIST est l'un des ensembles de données les plus populaires utilisés dans ce domaine, composé de 70 000 images en niveaux de gris de nombres manuscrits de 0 à 9, divisées en 60 000 images pour la formation et 10 000 images pour les tests. Cette étude vise à explorer l'effet de l'utilisation d'une variété de filtres sur l'amélioration des performances d'un modèle de classification d'images à l'aide de l'ensemble de données MNIST. Divers filtres, tels que les filtres de flou, les filtres de dégradé, le gaufrage, la transformation, la détection des contours, etc., seront appliqués pour améliorer la qualité de l'image et extraire des caractéristiques importantes, puis évaluer l'impact de ces améliorations sur la précision du modèle final. Et en comparant sa précision avec un autre modèle qui utilise des filtres par défaut, nous espérons, grâce à cette étude, fournir des informations précieuses sur la manière d'utiliser efficacement les filtres pour améliorer les performances des modèles d'apprentissage en profondeur dans la classification d'images, et ainsi contribuer au développement de modèles plus précis et des applications efficaces dans divers domaines.

- Dans le premier chapitre, nous présenterons la reconnaissance de formes.
- Le deuxième chapitre était consacré à la description des réseaux CNN et le deep learning ainsi leur intérêt dans le domaine de la classification des images.
- Dans le troisième chapitre, nous présenterons la partie expérimentale de notre travail.
- Enfin, nous concluons par une conclusion générale.



Chapitre01 : Reconnaissance de Formes

1 Introduction

Les techniques de reconnaissance de formes et d'apprentissage automatique constituent l'un des principaux domaines de l'informatique, car ces techniques permettent le développement de systèmes capables d'extraire des informations à partir de données et de les analyser de manière intelligente et efficace. Les techniques d'apprentissage automatique comprennent un ensemble de modèles et de méthodes tels que les systèmes informatiques. Tels que l'apprentissage profond et les réseaux de neurones. Dans ce chapitre, nous aborderons les concepts et algorithmes de reconnaissance de formes et d'apprentissage automatique, ainsi que leurs diverses applications dans de nombreux domaines. Nous fournirons également des exemples pratiques pour illustrer comment ces techniques peuvent être utilisées dans le monde réel, l'un des domaines scientifiques les plus vastes et les plus récents aujourd'hui, comme la classification d'images.

2 Prétraitement des données

2.1 Définition

Le prétraitement des données est une étape du processus d'exploration et d'analyse des données qui prend des données brutes et les transforme dans un format pouvant être compris et analysé par les ordinateurs et l'apprentissage automatique. Les données brutes du monde réel sous forme de texte, d'images, de vidéo, etc. sont compliquées. Non seulement il peut contenir des erreurs et des incohérences, mais il est souvent incomplet et n'a pas une conception régulière et uniforme [1].

2.2 Importance du prétraitement des données

Le prétraitement des données est essentiel pour garantir la précision et l'efficacité des algorithmes d'apprentissage automatique. Sans cela, les données ne sont pas prêtes, ce qui affecte négativement les résultats du modèle. Tout comme il est difficile de réaliser une mission sans informations claires, l'absence de traitement des données rend le modèle peu fiable. Par conséquent, les données doivent être nettoyées et transformées correctement pour améliorer les performances du modèle finale prétraitement des données est essentiel dans les premières phases du développement du machine learning [2]. Dans le domaine de l'IA, le prétraitement des données améliore la qualité des données en les nettoyant, en les transformant et en les formatant pour augmenter la précision d'un nouveau modèle tout en minimisant la quantité de calcul nécessaire.

2.3 Caractéristiques du prétraitement des données

Le prétraitement possède de nombreuses fonctionnalités qui en font une étape de préparation importante à l'analyse des données. Voici les deux principales caractéristiques avec une brève explication :

2.3.1 Validation de données

Il s'agit du processus par lequel les entreprises analysent et évaluent les données brutes d'un projet afin de déterminer si elles sont complètes et exactes afin d'obtenir les meilleurs résultats.

2.3.2 Imputation de données

L'imputation des données consiste à saisir les valeurs manquantes et à rectifier les erreurs de données pendant le processus de validation, manuellement ou par programmation, comme l'automatisation des processus métier. Qu'une entreprise utilise une analyse d'applications basée sur des bases de données ou basée sur des règles, le prétraitement permet de garantir des résultats fiables et précis lors de l'analyse des données. Il est important de prétraiter les données lors de l'utilisation d'outils d'apprentissage automatique pour s'assurer que les algorithmes peuvent lire de grands ensembles de données et les interpréter correctement pour une utilisation ultérieure, Informations et claires ordonnées – elles écoutent les données sous forme de 1 et de 0. Il est donc facile de calculer des données structurées, comme des nombres entiers et des pourcentages. Cependant, les données non structurées, sous forme de texte et d'images, doivent d'abord être nettoyées et formatées avant analyse [3].

2.4 Etapes clés du prétraitement des données

Examinons maintenant plus en profondeur les quatre étapes principales du prétraitement des données.

1. Évaluation de la qualité des données :

Examinez attentivement vos données et faites-vous une idée de leur qualité globale, de leur pertinence par rapport à votre projet et de leur cohérence. Il existe un certain nombre d'anomalies de données et de problèmes inhérents à surveiller dans presque tous les ensembles de données, par exemple :

➤ Types de données incompatibles :

La collecte de données à partir de différentes sources nécessite de les uniformiser avant de les formater pour les machines. Par exemple, pour analyser les revenus familiaux de différents pays, il est nécessaire de convertir tous les montants dans une seule devise.

➤ Valeurs de données mixtes :

Les données peuvent présenter des défis tels que des descripteurs de valeurs différents pour les mêmes caractéristiques, des valeurs aberrantes pouvant fausser les résultats, et des données manquantes qui nécessitent un nettoyage.

2. Nettoyage des données :

Le nettoyage des données est particulièrement effectué dans le cadre du prétraitement des données pour nettoyer les données en remplissant les valeurs manquantes, en lissant les données bruitées, en résolvant les incohérences et en supprimant les valeurs aberrantes.

➤ Valeurs manquantes

Voici quelques façons de résoudre ce problème :

Ignorez ces tuples Cette méthode doit être envisagée lorsque l'ensemble de données est énorme et que de nombreuses valeurs manquantes sont présentes dans un tuple. Remplissez les valeurs manquantes Il existe de nombreuses méthodes pour y parvenir, telles que le remplissage manuel des valeurs, la prédiction des valeurs manquantes à l'aide d'une méthode de régression ou des méthodes numériques telles que la moyenne des attributs.

➤ Données bruyantes :

Cela implique de supprimer une erreur aléatoire ou une variance dans une variable mesurée. Cela peut être fait à l'aide des techniques suivantes :

➤ Regroupement :

C'est la technique qui fonctionne sur les valeurs de données triées pour atténuer tout bruit présent. Les données sont divisées en groupes de taille égale et chaque groupe/seau est traité indépendamment. Toutes les données d'un segment peuvent être remplacées par ses valeurs moyennes, médianes ou limites.

➤ Régression :

Cette technique de data mining est généralement utilisée à des fins de prédiction. Cela permet de lisser le bruit en ajustant tous les points de données dans une fonction de régression. L'équation de régression linéaire est utilisée s'il n'y a qu'un seul attribut indépendant ; sinon, des équations polynomiales sont utilisées.

3. Suppression des valeurs aberrantes

Les techniques de clustering regroupent des points de données similaires. Les tuples qui se trouvent en dehors du cluster sont des données aberrantes/incohérentes.

4. Transformation des données :

Avec le nettoyage des données, nous avons déjà commencé à modifier nos données, mais la transformation des données lancera le processus de transformation des données dans le(s) format(s) approprié(s) dont vous aurez besoin pour l'analyse et d'autres processus en aval. Cela se produit généralement dans un ou plusieurs des cas ci-dessous : **Agrégation** , **Normalisation**, **Sélection de fonctionnalité**, **Discréditation** , **Génération de hiérarchie de concepts**

- ✓ **Agrégation** : l'agrégation de données combine toutes vos données dans un format uniforme.
- ✓ **Normalisation** : Pour comparer les données précisément, régularisez-les dans une plage spécifiée. La sélection des fonctionnalités choisit les variables clés pour l'analyse. Trop de fonctionnalités peuvent allonger l'entraînement et réduire la précision.
- ✓ **Discréditation** : la discréditation regroupe les données dans des intervalles plus petits. C'est un peu similaire au regroupement, mais cela se produit généralement après le nettoyage des données. Par exemple, lors du calcul de l'exercice quotidien moyen, plutôt que d'utiliser les minutes et les secondes exactes, vous pouvez regrouper les données pour les classer en 0-15 minutes, 15-30, etc.
- ✓ **Génération de hiérarchie de concepts** : la génération de hiérarchie de concepts peut ajouter une hiérarchie au sein et entre vos fonctionnalités qui n'était pas présente dans les données d'origine. Si votre analyse contient des loups et des coyotes, par exemple, vous pouvez ajouter la hiérarchie de leur genre : canis

5. Réduction des données :

La quantité croissante de données rend leur analyse difficile, même après nettoyage et conversion. Il peut y avoir des données redondantes, notamment dans l'analyse de texte où une grande partie du contenu est inutile. La réduction des données facilite l'analyse, augmente la précision, réduit le stockage requis et aide à identifier les caractéristiques les plus importantes.

- ✓ **Sélection d'attributs** : à l'instar de la discréditation, la sélection d'attributs peut regrouper vos données dans des pools plus petits. Il combine essentiellement des balises ou des fonctionnalités, de sorte que des balises telles que homme/femme et professeur puissent être combinées en professeur masculin/professeur féminin.
- ✓ **Réduction de la nombreuse** : cela facilitera le stockage et la transmission des données. Vous pouvez utiliser un modèle de régression, par exemple, pour utiliser uniquement les données et variables pertinentes pour votre analyse. Réduction de la dimensionnalité : cela, encore une fois, réduit la quantité de données utilisées pour faciliter l'analyse et les processus en aval. Des algorithmes tels que les K-voisins les plus proches utilisent la reconnaissance de formes pour combiner des données similaires et les rendre plus gérables.[4]

3 Apprentissage Automatique

3.1 Définition

Sous-ensemble de l'intelligence artificielle (IA) et de l'informatique, l'apprentissage automatique (ML) traite de l'étude et de l'utilisation de données et d'algorithmes qui imitent la façon dont les humains apprennent. Cela aide les machines à améliorer progressivement leur précision. ML permet aux applications logicielles d'améliorer la précision de leurs prédictions sans être spécifiquement programmées pour le faire. [5] Il estime de nouvelles valeurs de sortie en utilisant des données historiques comme entrée.

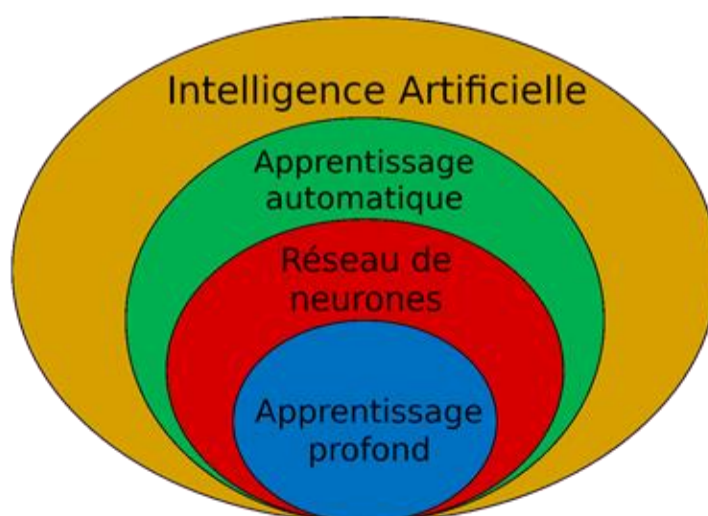


Figure 1:Apprentissage Automatique

3.2 Applications de l'apprentissage automatique

Voici quelques exemples d'apprentissage automatique que vous pouvez rencontrer tous les jours :

- **Reconnaissance vocale** : La reconnaissance vocale convertit la parole humaine en texte écrit, facilitant les recherches vocales et offrant une meilleure accessibilité aux utilisateurs des terminaux mobiles.
- **Service client** : Les agents conversationnels en ligne, comme les bots sur les sites d'e-commerce et les messageries comme Slack et Facebook Messenger, remplacent souvent les agents humains dans l'interaction client. Ils répondent aux questions fréquentes et offrent des conseils personnalisés, modifiant ainsi l'engagement client en ligne.
- **Transactions boursières automatisées** : Conçues pour optimiser les portefeuilles d'actions, les plateformes de transactions boursières à haute fréquence optimisées par l'IA effectuent des milliers, voire des millions, de transactions par jour sans intervention humaine.
- **Services médicaux** : Les méthodes d'apprentissage automatique sont largement utilisées dans le domaine médical. Notamment pour détecter une maladie, planification de la thérapie, recherche médicale, prédiction de la situation de la maladie etc L'utilisation de logiciels basés sur l'apprentissage automatique dans le domaine de la santé apporte une percée dans notre science médicale [6].

3.3 Avantages de l'apprentissage automatique

- **Efficacité et automatisation augmentent** : La technologie d'apprentissage automatique traite efficacement de vastes ensembles de données, facilitant des décisions rapides et précises, et contribuant ainsi à accroître la productivité et à réaliser des économies pour les entreprises.
- **Amélioration des capacités de prise de décision** : Les algorithmes d'apprentissage automatique peuvent identifier des modèles et faire des prédictions sur la base de données historiques, ce qui peut améliorer la précision des décisions et réduire la nécessité d'une supervision humaine.
- **Capacité à traiter de grandes quantités de données** : Les algorithmes d'apprentissage automatique peuvent traiter et analyser de grandes quantités de données, ce qui peut être utile pour les organisations qui ont de grandes quantités d'informations à traiter.
- **Potentiel de réduction des coûts** : En automatisant les processus et en améliorant la prise de décision, l'apprentissage automatique peut permettre aux entreprises de réaliser des économies.
- **Capacité à détecter des modèles et à faire des prédictions** : En utilisant l'apprentissage automatique, vous pouvez identifier des modèles et faire des prédictions basées sur des données historiques [7], ce qui peut être utile pour les prévisions, la détection d'anomalies et d'autres tâches qui nécessitent l'identification de modèles dans les données.

3.4 Inconvénients De L'apprentissage Automatique

- **Coût élevé de la mise en œuvre** : La mise en œuvre de systèmes d'apprentissage automatique peut être coûteuse, car elle nécessite du matériel et des logiciels spécialisés, ainsi que du personnel possédant les compétences nécessaires pour concevoir et entretenir le système.
- **Nécessité de disposer de gros volumes de données** : Le traitement de l'apprentissage automatique nécessite de gros volumes de données pour être efficace, ce qui peut représenter un défi pour les organisations disposant d'ensembles de données limités.
- **Dépendance à l'égard de la qualité des données** : L'efficacité de l'apprentissage automatique et de l'intelligence artificielle dépend fortement de la qualité des données utilisées pour les former. Des données de mauvaise qualité peuvent conduire à des résultats inexacts ou peu fiables.[7]

3.5 Types d'apprentissage automatique

Les paradigmes d'apprentissage automatique diffèrent dans leur approche. Chaque paradigme est caractérisé par le type de données qu'il entre et sort, et le type de tâche ou de problème qu'il est censé résoudre.

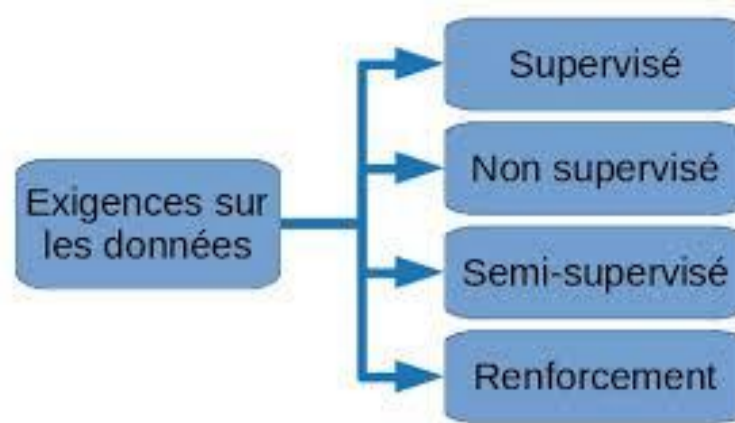


Figure 2:Paradigmes D'apprentissage Automatique

3.5.1 Apprentissage supervise

L'apprentissage supervisé est un type d'apprentissage automatique où un modèle est formé sur la base de données étiquetées, où chaque exemple de données a une variable ou une étiquette cible correspondante. Le modèle vise à trouver des dépendances et des modèles communs entre les données d'entrée et les étiquettes de sortie correspondantes. Des exemples d'algorithmes d'apprentissage supervisé comprennent la régression linéaire [8], les machines à vecteurs de support, les forêts aléatoires et les réseaux de neurones.

Exemples de tâches :

- **Classification** : déterminer l'appartenance à une classe d'un objet. Par exemple, classer les e-mails comme spam ou pas spam.

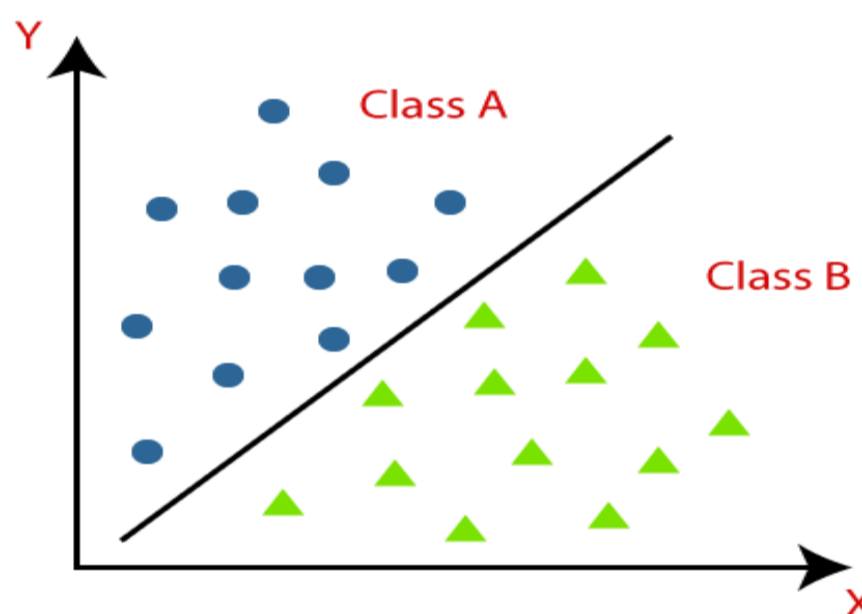


Figure 3:Exemple De Classification

- **Régression**: dans ce type de tâche, le programme informatique est invité à prédire une valeur numérique à partir d'une certaine entrée. Pour résoudre cette tâche, il est demandé à l'algorithme d'apprentissage de sortir une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Ce type de tâche est similaire à la classification, sauf que le format de sortie est différent. Un exemple de tâche de régression est la prédiction du montant attendu des sinistres qu'une personne assurée fera (utilisé pour fixer les primes d'assurance), ou la prédiction des prix futurs des titres.

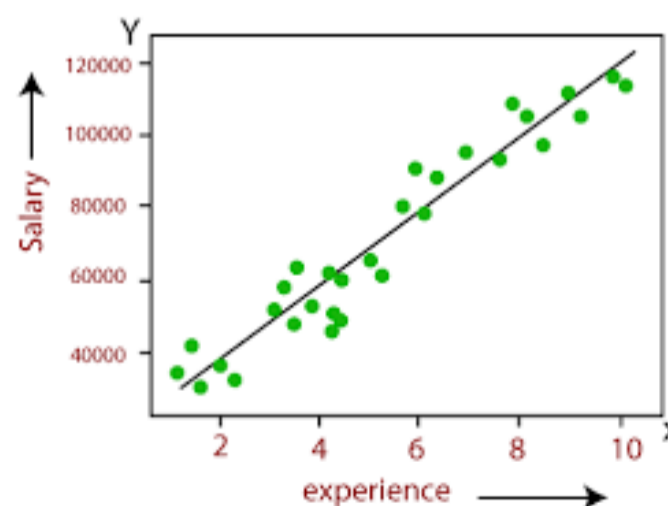


Figure 4:Exemple De Régression

3.5.2 Apprentissage non supervisé

L'apprentissage non supervisé est un type d'apprentissage automatique dans lequel un modèle est formé sur des données non étiquetées sans variables cibles explicites. Au lieu de cela, le modèle recherche des structures, des modèles et des groupes cachés dans les données. Des algorithmes d'apprentissage non supervisés sont utilisés pour le regroupement de données, la réduction de la dimensionnalité, l'analyse d'association et la génération de caractéristiques[8].

Exemples de tâches :

➤ **Regroupement** : regroupement d'objets similaires dans les données. Par exemple, segmenter les clients en fonction de leur comportement d'achat.



Figure 5 : Exemple De Regroupement

➤ **Réduction de la dimensionnalité** :

C'est le processus de réduction du nombre de dimensions dans un ensemble de données, ce qui réduit le nombre total de variables ou d'attributs traités. Cela permet de réduire la complexité et le grand volume de données, ce qui facilite le processus d'analyse et de traitement. Généralement obtenue en appliquant des techniques telles que :

- PCA (Analyse en composantes principales) : elle est utilisée pour transformer l'ensemble de données d'origine en un nouvel ensemble de données contenant le plus petit nombre de dimensions, tout en conservant la plus grande variance dans les données.
- t-SNE (t-Distributed Stochastic Neighbour Embedding) : utilisé pour présenter des données de haute dimension dans une image de basse dimension, tout en conservant la structure relative des données.
- LDA (analyse discriminante linéaire) : elle est utilisée pour transformer l'ensemble de données afin d'obtenir une bonne séparation entre les différentes classes.

Ces techniques sont couramment utilisées dans des domaines tels que l'apprentissage automatique et l'analyse de données, où elles contribuent à réduire le bruit et la complexité des données et à améliorer notre compréhension des informations importantes contenues dans de grands ensembles de données.

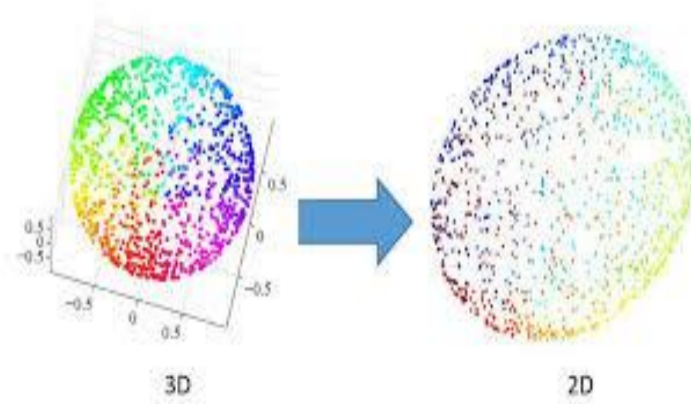


Figure 6 : Exemple Réduction De La Dimensionnalité

3.5.3 Apprentissage par renforcement

L'apprentissage par renforcement est un type d'apprentissage automatique dans lequel un modèle apprend en interagissant avec un environnement et en recevant des commentaires sous forme de récompenses ou de pénalités. Le modèle prend des décisions et ajuste son comportement en fonction de la récompense reçue. Il est activement utilisé dans la robotique, les jeux et le contrôle de système autonome.[8]

Exemples de tâches :

- **Contrôle du robot** : formation d'un robot à effectuer des actions spécifiques dans son environnement pour atteindre les objectifs fixés.
- **Jeux** : former un agent à jouer à des jeux, tels que les échecs ou les jeux vidéo, pour obtenir le score le plus élevé possible.

4 Classifieurs

Il y a tellement d'algorithmes qu'il est presque impossible de les connaître tous. Mais il existe des algorithmes qui sont plus connus et plus utilisés que d'autres. Nous présentons ici les algorithmes les plus connus dans le domaine de l'apprentissage automatique.[9]

4.1 Régression logistique

La régression logistique est adaptée aux tâches de classification, telles que la prédiction des risques de maladie, sur la base de certains facteurs. Elle modélise la probabilité qu'une instance appartienne à une certaine classe. Dans les hôpitaux, par exemple, les médecins utilisent la régression logistique pour calculer le risque d'une certaine maladie sur la base des caractéristiques des patients. Cela permet d'établir un diagnostic plus précis et d'identifier les groupes à risque.[9]

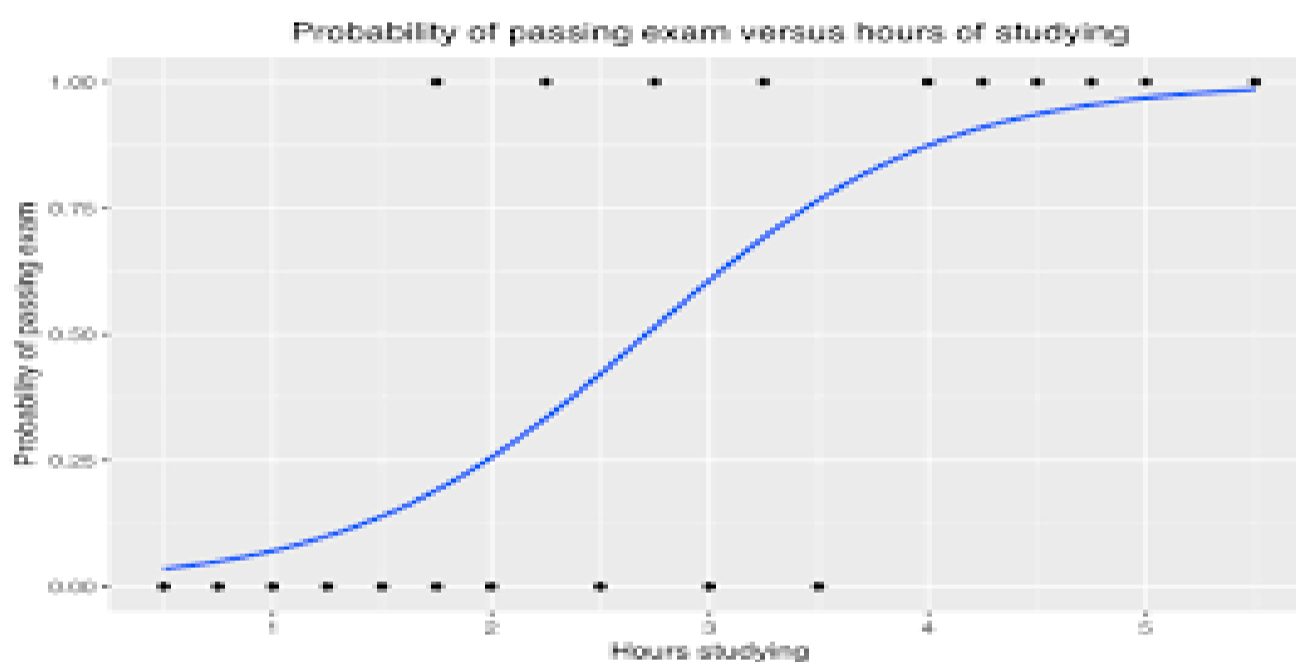


Figure 7 : Exemple De Régression Logistique

4.2 Arbres de décision

Les arbres de décision prennent des décisions sur la base de caractéristiques et divisent l'ensemble de données en sous-groupes. Dans le secteur financier, ils aident par exemple à évaluer les risques de crédit. En analysant les caractéristiques des clients, la banque peut décider à qui elle va accorder des crédits et quelles sont les conditions applicables.

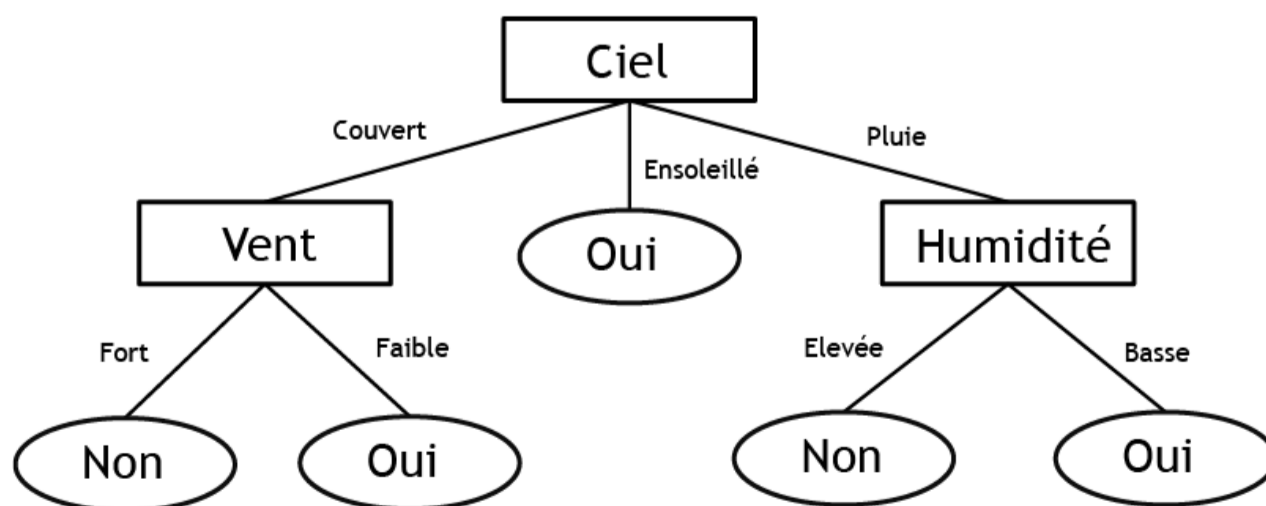


Figure 8 : Exemple De Arbres De Décision

4.3 Machines vectorielles de support

SVM recherche la ligne de séparation optimale entre différentes classes dans l'espace des caractéristiques. Cela est utile dans des domaines tels que la reconnaissance des visages, la classification des textes et les prévisions financières. Par exemple, les entreprises du secteur financier utilisent le SVM pour détecter les transactions frauduleuses[9]. Elle trouve des lignes de démarcation claires entre les modèles de transaction normaux et suspects.

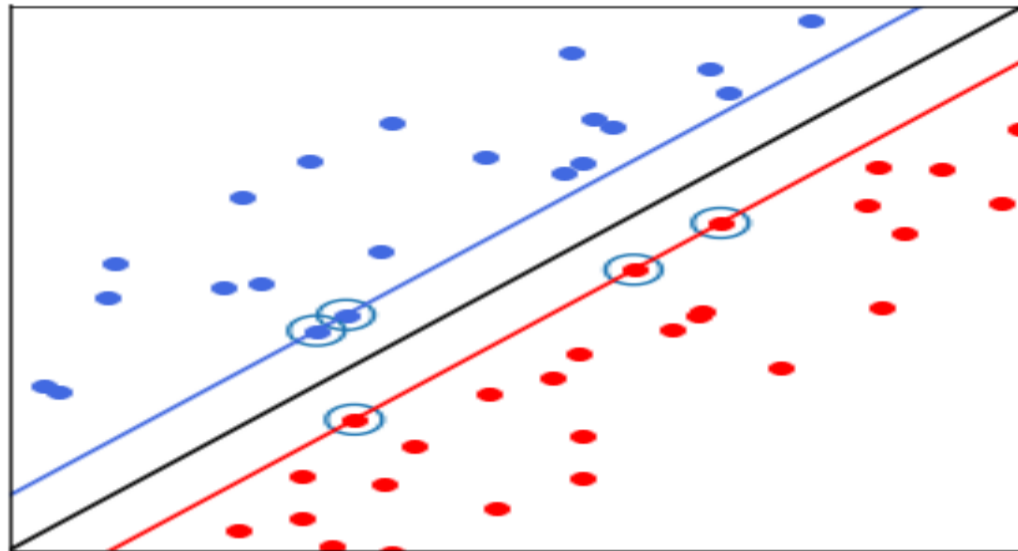


Figure 9: Exemple De SVM

4.4 Clustering K-Means

K-Means regroupe des points de données similaires en clusters en minimisant la distance moyenne. Les applications vont de la segmentation des clients à la compression d'images et à la détection d'anomalies. Dans le secteur du commerce de détail, les commerçants utilisent par exemple le K-Means-Clustering pour segmenter la clientèle. Ils sont ainsi en mesure de développer des stratégies marketing ciblées pour différents groupes de clients.

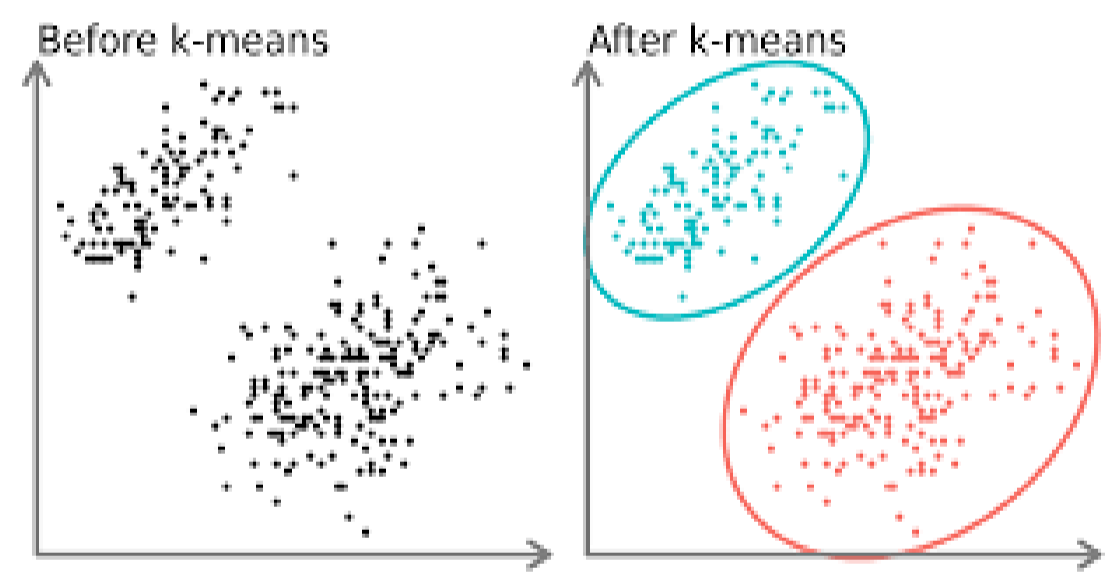


Figure 10 : Exemple De K-Means

4.5 Réseaux neuronaux

Les réseaux neuronaux sont composés de neurones interconnectés et apprennent des modèles complexes. Ils sont largement utilisés dans la reconnaissance d'images, le traitement du langage et le trafic.

Un exemple pratique : Dans l'industrie automobile, les réseaux neuronaux sont utilisés dans les véhicules autonomes. Ils apprennent à comprendre des modèles de trafic complexes et soutiennent ainsi la prise de décision adaptative du véhicule.[9]

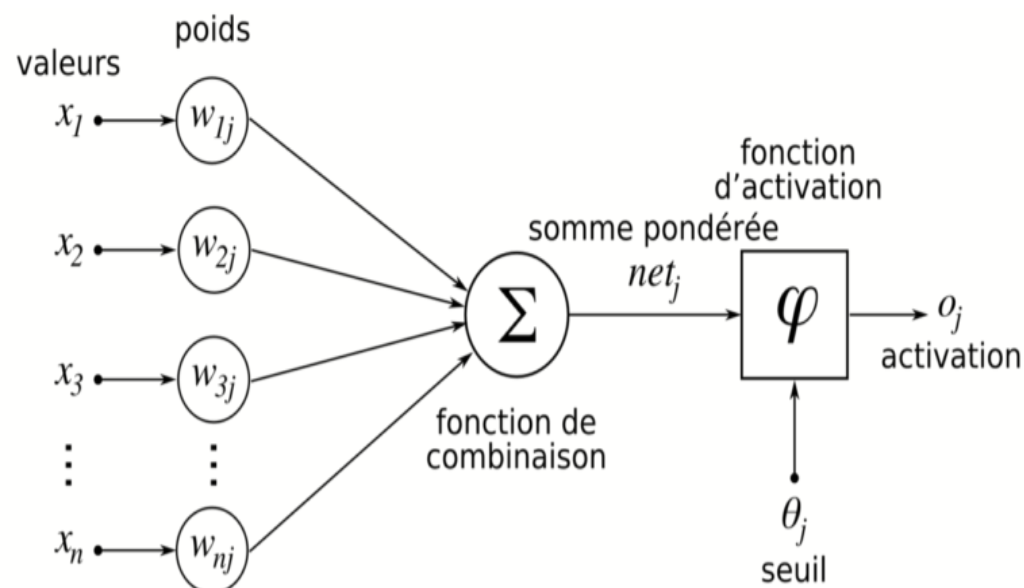


Figure 11:Reseaux De Neurones

4.6 K plus proches voisins

L'algorithme k-NN (k-Nearest Neighbors) est une méthode d'apprentissage supervisé dans laquelle une instance est classée en fonction de la majorité de ses k plus proches voisins. La proximité est souvent définie par la distance euclidienne mesurée. Dans la pratique, la détection d'anomalies dans la sécurité des réseaux, par exemple, utilise les k-NN pour identifier les activités suspectes en se basant sur des modèles à proximité du trafic réseau.[10]

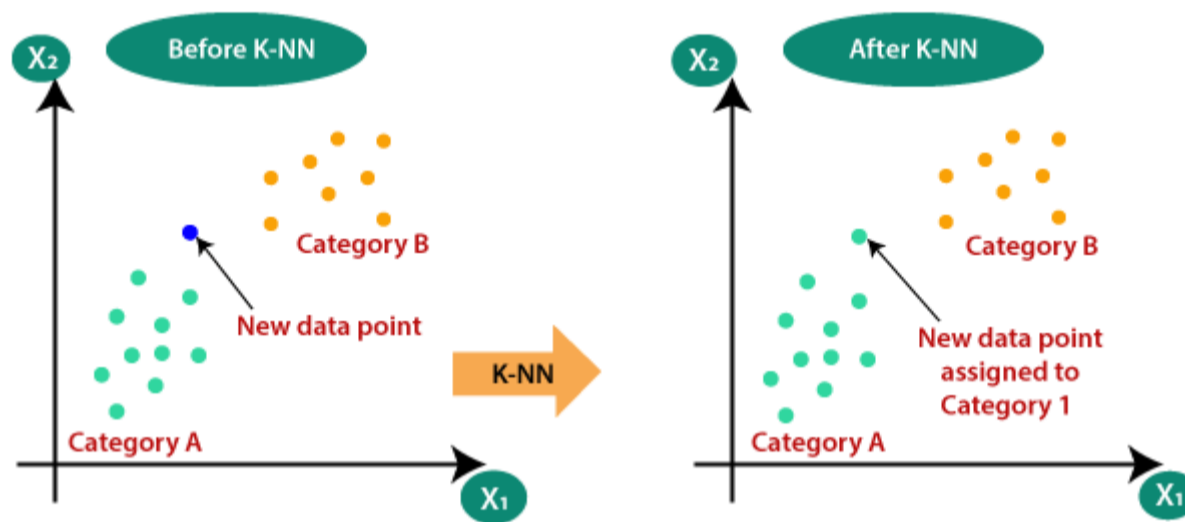


Figure 12:Exemple De Knn

5 Conclusion

L'apprentissage automatique est le présent et l'avenir de l'informatique, et la plupart des percées technologiques récentes ont été réalisées avec cet outil qui prouve sa puissance et son efficacité jour après jour. Dans ce chapitre nous avons défini l'apprentissage automatique ainsi que ses paradigmes. Nous avons aussi mentionné un ensemble de ses algorithmes tel que les réseaux de neurones artificiels. Dans le prochain chapitre nous allons parler d'un type de réseaux de neurones qui sont largement utilisées dans le domaine de vision par ordinateur et plusieurs autres domaines qui sont les réseaux de neurones convolutionnels.

Chapitre 02 : Réseaux de neurones convolutionnels

1 Introduction

A partir de l'année 2012, une révolution se produit grâce à un algorithme de Deep Learning (apprentissage profond) qui a explosé tous les records. Il s'agit d'un réseau de neurones convolutif appelé alexNet, et depuis, les réseaux de neurones convolutionnels sont devenus dominants dans diverses tâches de vision par ordinateur. Ils suscitent un intérêt dans une variété de domaines, y compris la classification d'images, les moteurs de recherche, les systèmes de recommandation, les réseaux sociaux, la reconnaissance faciale et d'autres. Les réseaux de neurones convolutionnels sont conçu pour apprendre automatiquement et de manière adaptative les caractéristiques et les hiérarchies spatiales d'entités en utilisant plusieurs blocs de construction, tels que des couches de convolution et des couches entièrement connectées. Dans ce chapitre nous donnons une perspective sur les concepts de base des réseaux de neurones convolutionnels et leurs fonctionnements.

2 Apprentissage Profond

2.1 Définition

L'apprentissage profond (Deep Learning) est une méthode d'intelligence artificielle (IA) qui apprend aux ordinateurs à traiter les données d'une manière inspirée par le cerveau humain. Les modèles d'apprentissage profond peuvent reconnaître des modèles complexes dans les images, le texte, les sons et d'autres données pour produire des informations et des prédictions précises. Vous pouvez utiliser des méthodes d'apprentissage profond pour automatiser des tâches qui nécessitent généralement l'intelligence humaine, comme décrire des images ou transcrire un fichier son en texte [11].

Apprentissage profond et réseaux de neurones



Figure 13: Apprentissage Profonde

2.2 Principe fondamental de l'apprentissage profond

Un réseau neuronal convolutionnel se compose d'une couche d'entrée et d'une sortie, ainsi que de plusieurs couches cachées. Les couches cachées d'un RNC consistent généralement en une série de couches convolutionnels qui sont convoluées avec une multiplication ou un autre produit scalaire. Les couches convolutionnels sont généralement suivies d'une fonction d'activation (généralement la fonction RELU) et ensuite suivie de convolutions supplémentaires telles que des couches de mise en commun, des couches entièrement connectées et des couches de normalisation, appelées couches cachées car leurs entrées et sorties sont masquées par la fonction d'activation et la convolution finale [12].

2.3 Paramètres De L'apprentissage Profond :

Il existe deux techniques de régularisation très populaires et efficaces appelées L2 et abandon :

a) Régularisation

Diminution des Poids permet d'améliorer la généralisation lors de l'apprentissage du modèle. De la sorte, le modèle va être performant à la fois sur les données d'entraînements mais aussi sur toutes les autres données, Comme expliqué dans cet article, lors de l'apprentissage, certaines poids du modèle augmentent tandis que d'autres diminuent.

Le but est de diminuer la valeur des poids trop important. C'est une sorte de handicap que l'on donne au poids qui augmentent trop vite. Grâce à cet handicap, le modèle doit performer ailleurs que sur les neurones associés à ces poids. Ainsi, d'autres poids vont pouvoir progresser pour transmettre de l'information.[13]

b) Abandon :

Le terme « abandon » ((dropout)) fait référence à l'abandon des nœuds (couche d'entrée et couche cachée) dans un réseau neuronal (comme le montre la figure 1). Toutes les connexions aller et retour avec un nœud abandonné sont temporairement supprimées, créant ainsi une nouvelle architecture réseau à partir du réseau parent. Les nœuds sont supprimés avec une probabilité d'abandon de p.[12]

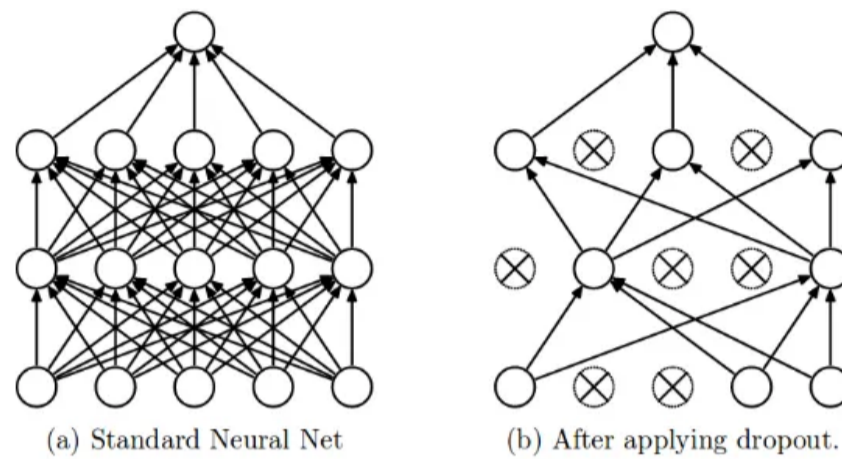


Figure 14 :Abandon Appliqué A Un Réseau Neuronal Standard (Image De Nitish)

3 Neurones Convolutionnel

3.1 Définition

Réseau de neurone convolutionnel (CNN) est un algorithme d'apprentissage profond (deep learning) qui peut prendre une image comme entrée, attribuer une importance (poids et biais apprenants) à divers aspects / objets de l'image et être capable de différencier les uns des autres. Le prétraitement requis dans un ConvNet est beaucoup plus faible par rapport à d'autres algorithmes de classification. Alors que dans les méthodes primitives, les filtres sont conçus à la main, avec un entraînement suffisant, les réseaux de neurones convolutionnels ont la capacité d'apprendre ces filtres / caractéristiques eux-mêmes. L'architecture d'un ConvNet est analogue à celle du modèle de connectivité des neurones dans le cerveau humain et a été inspirée par l'organisation du Visual Cortex. Les neurones individuels ne répondent aux stimuli que dans une région restreinte du champ visuel. Une collection de ces champs se chevauche pour couvrir toute la zone visuelle [13].

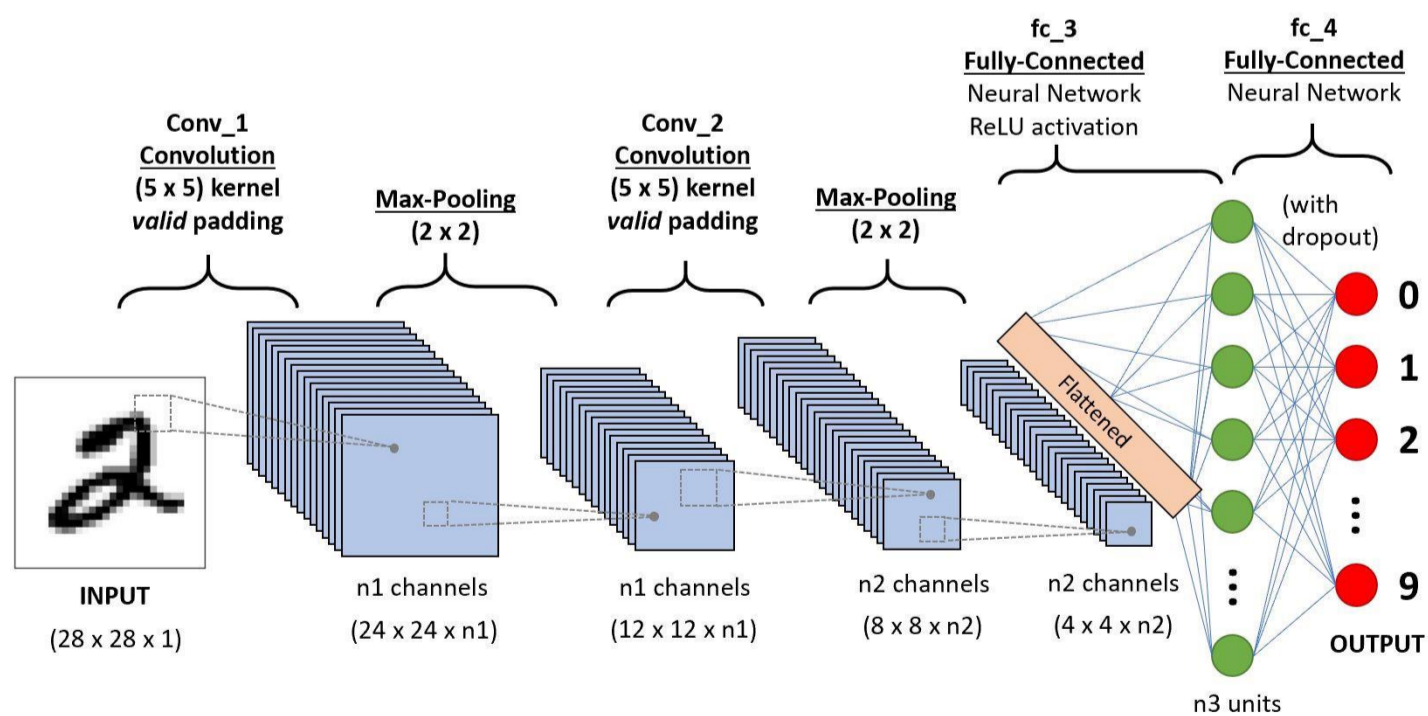


Figure 15 : Réseau Neuronal A Convolution

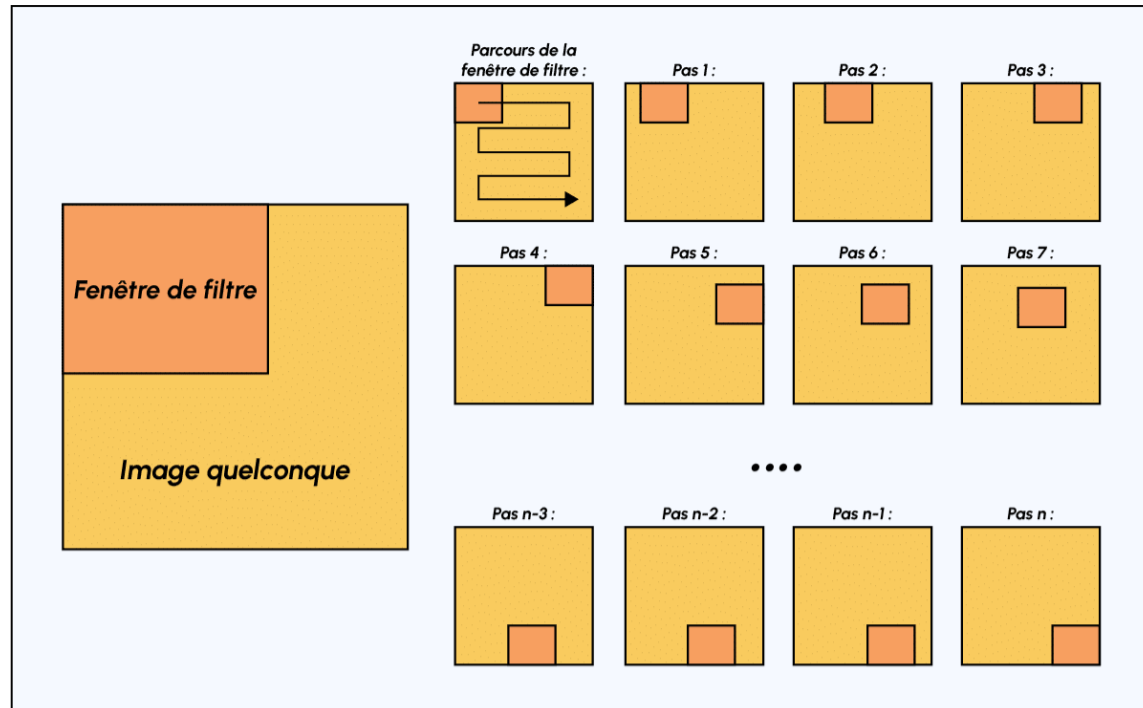


Figure 16 : Une Représentation de la façon dont le filtre passe et les champs récepteurs sont formés

3.2 Caractéristiques des images numériques

Les réseaux de neurones convolutifs sont aujourd’hui les éléments constitutifs des tâches de classification d’images utilisant l’apprentissage automatique. Cependant, une autre tâche très utile qu’ils effectuent avant la classification consiste à extraire les caractéristiques pertinentes d’une image. L’extraction de fonctionnalités est la manière dont les CNN reconnaissent les modèles clés d’une image afin de la classer. Mais d’abord, afin de formaliser ces concepts CNN, nous devons parler de l’espace des pixels.

a) Espace de pixels :

L’espace pixel est exactement ce que son nom l’indique : c’est l’espace où l’image est convertie en une matrice de valeurs, où chaque valeur correspond à un pixel individuel. Par conséquent, l’image originale que nous voyons, lorsqu’elle est transmise à CNN, est convertie en une matrice de nombres. Dans les images en niveaux de gris, ces nombres vont généralement de 0 (noir) à 255 (blanc), et les valeurs intermédiaires sont des nuances de gris. Dans cet article, toutes les images ont été normalisées, c’est-à-dire que chaque pixel a été divisé par 255 afin que sa valeur se situe dans l’intervalle [0, 1].

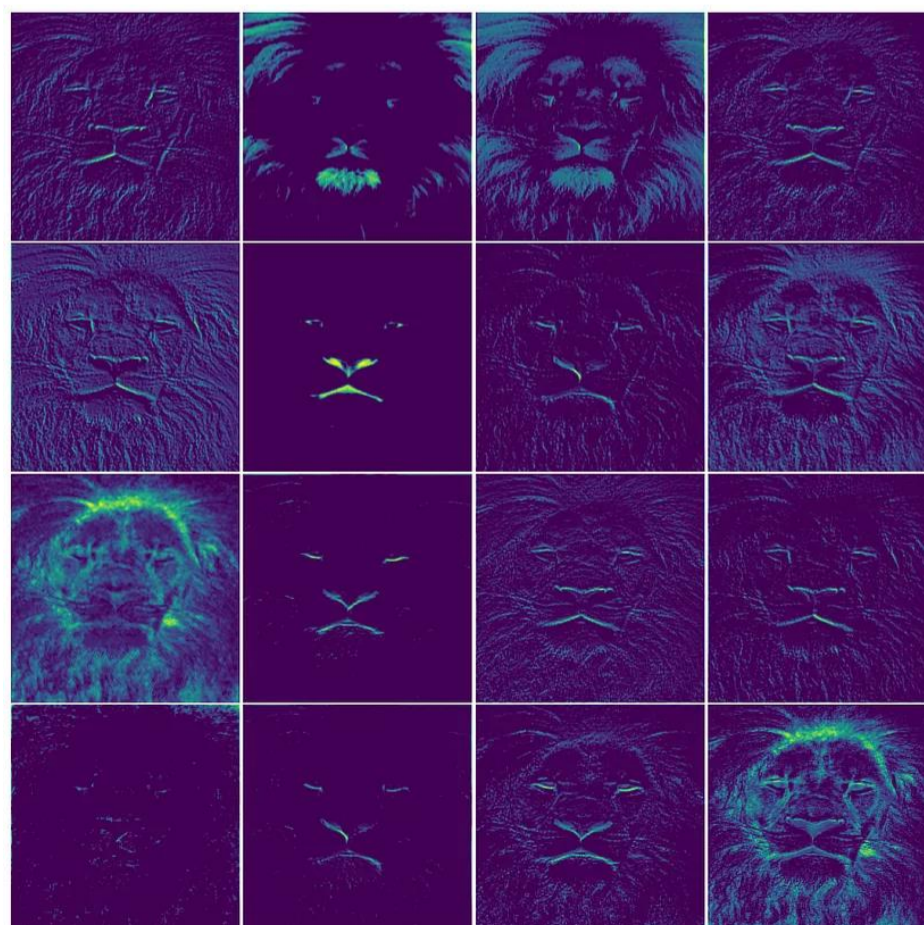


Figure 17 : Extraction de caractéristiques de lion

3.3 Opérations des réseaux de neurones de convolution :

1. Remplissage

Le remplissage (padding) est le processus consistant à ajouter des couches de zéros ou d'autres valeurs en dehors des données réelles dans une matrice d'entrée[16]. L'objectif principal du remplissage est de préserver la taille spatiale de l'entrée afin que la sortie après l'application de filtres (noyaux) reste de la même taille, ou de l'ajuster en fonction des dimensions de sortie souhaitées comme indiqué dans l'image 19.

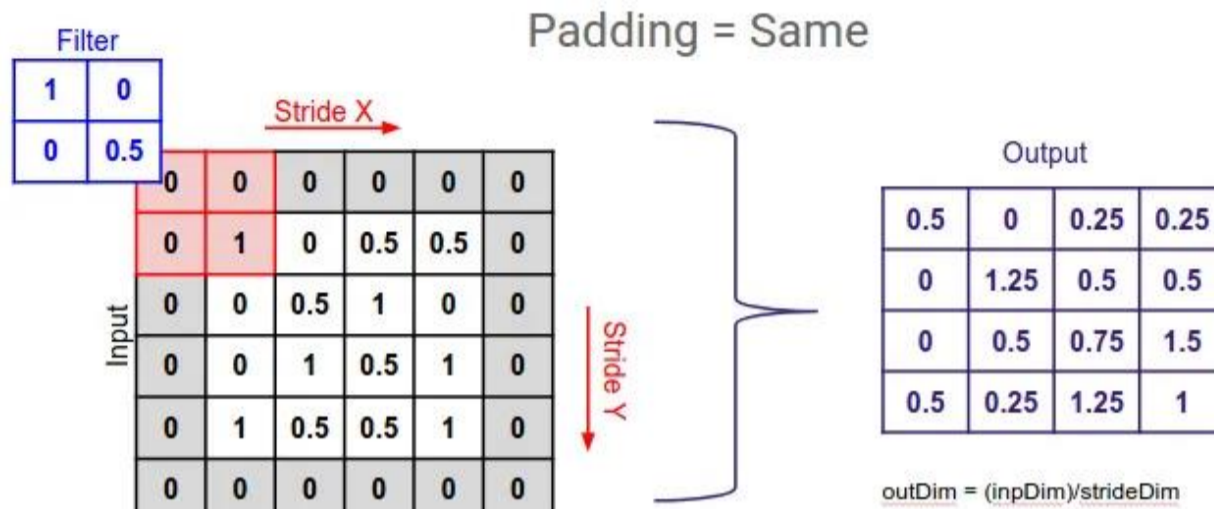


Figure 19 : Exemple De Remplissage

2. Stride:

Stride est un enseignant dictant le mouvement du noyau, ou du candidat, via des données d'entrée, telles que l'image. Lors de l'exécution d'un processus payant, l'étape détermine le nombre d'unités que le candidat tourne à chaque étape. Cette contradiction peut être horizontale, verticale ou les deux, selon la formation de l'étape[15]. Vous produirez une étape plus grande après une sortie plus petite et réduit efficacement l'image comme indiqué dans l'image .

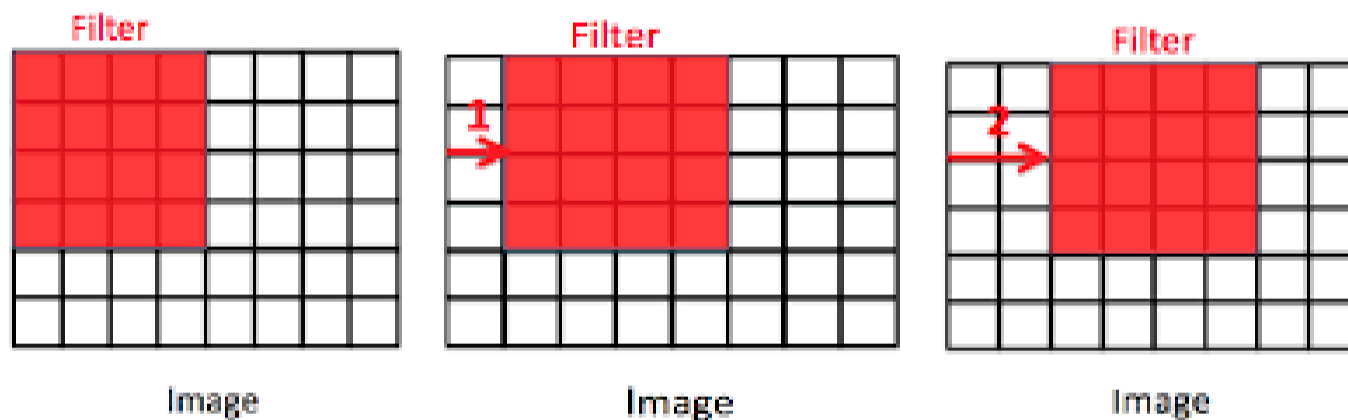


Figure 20 : Le Fonctionnement de stride.

3.4 Filtre des images numériques

- **Gaussien** : Le filtre gaussien est une technique de traitement d'image utilisée pour lisser les images et réduire le bruit aléatoire. Ce filtre est basé sur l'application d'une distribution gaussienne (distribution normale) pour répartir les poids sur les pixels environnants, ce qui permet de réduire les détails fins inutiles tout en préservant les bords et les détails importants de l'image. Il est appliqué en passant un noyau gaussien sur l'image, attribuant un poids plus élevé aux pixels les plus proches du point central et un poids plus faible aux pixels les plus éloignés.[17]

$$h(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (3)$$

- **Sharpen** : Les filtres de netteté sont un type de traitement d'image qui améliore le contraste entre les pixels voisins, rendant les bords et les détails plus visibles et définis. Les filtres de netteté peuvent également réduire les effets du bruit, du flou ou des artefacts de compression qui dégradent la qualité des images numériques. Les filtres de netteté peuvent être appliqués à l'image entière ou à des régions spécifiques, en fonction de vos besoins et préférences [18].
- **Détection des contours Sobel** : la détection des contours consiste simplement à essayer de trouver les régions d'une image où nous avons un changement brusque d'intensité ou un changement brusque de couleur, une valeur élevée indique un changement abrupt et une valeur faible indique un changement superficiel.
- **Détection de contour laplacien** : Le filtre laplacien est un filtre dérivé du second ordre utilisé en détection de contour, dans le traitement d'images numériques. Dans les filtres dérivés du 1^{er} ordre, nous détectons séparément le bord ainsi que les directions horizontale et verticale, puis combinons les deux. Mais en utilisant le filtre laplacien, nous détectons les bords de toute l'image en même temps.[19]
- **Identité** : les filtres d'identité sont un mécanisme de filtrage de données amélioré que vous pouvez utiliser pour filtrer les données d'une application mobile en fonction des champs dynamiques renvoyés par un fournisseur d'identité [19].

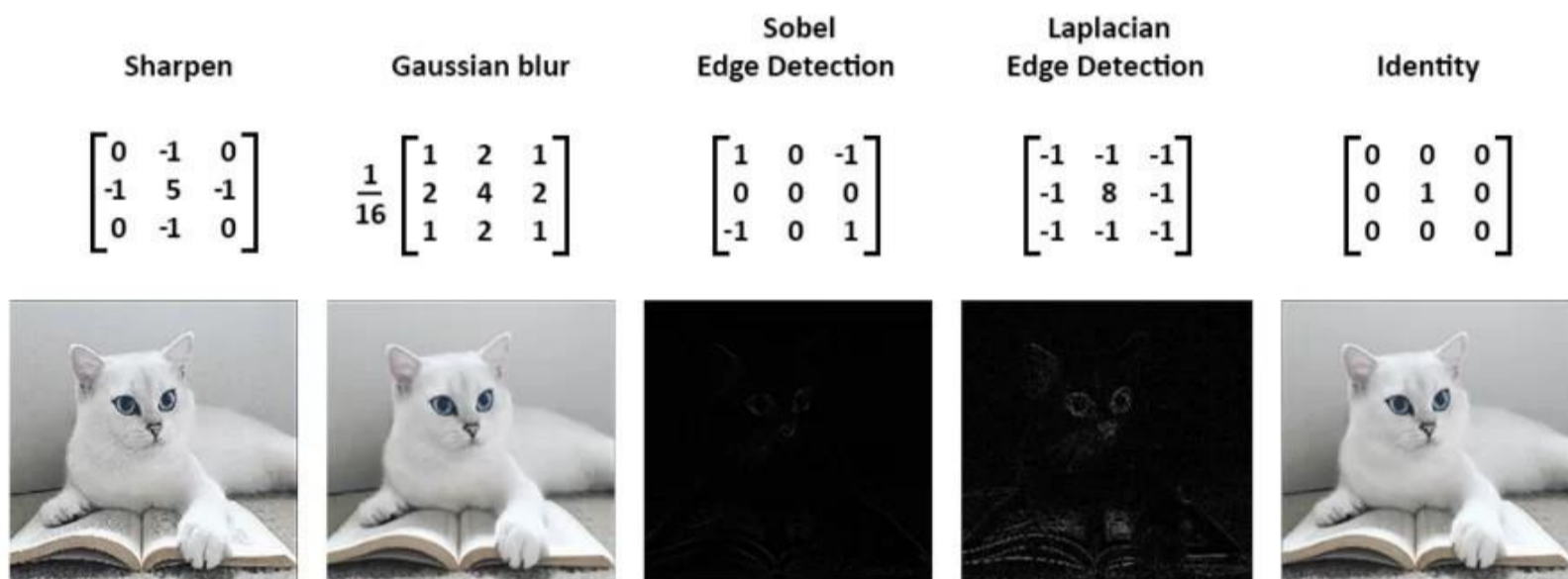


Figure 21 : Une Collection De Filtres Populaires Avec Leurs Poids Et Sorties

- **Filtre médian** : Le filtre médian est un autre filtre très courant, qui réduit le bruit et préserve les bords. Il peut réparer les pixels ou même les petites zones dont la couleur est endommagée ou manquante.[20]



Figure 22 : Filtre Median

- **La dilatation** : Opérations de base de la morphologie mathématique. Développée à l'origine pour les images binaires, elle a été étendue d'abord aux images en niveaux de gris, puis aux réseaux complets. L'opération de dilatation utilise généralement un élément structurant pour sonder et étendre les formes contenues dans l'image d'entrée.[21]

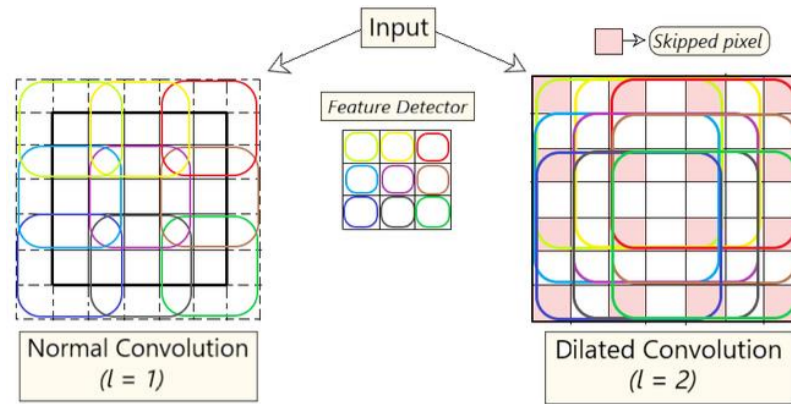


Figure 23 : Convolution normale contre convolution dilatée

- **Clahé** : La méthode CLAHE est utilisée pour améliorer le contraste des images. Elle diffère de l'égalisation d'histogramme ordinaire en ce sens que la méthode adaptative calcule plusieurs histogrammes, chacun correspondant à une section distincte de l'image, et les utilise pour redistribuer les valeurs de luminosité de l'image. Il peut donc améliorer le contraste local et améliorer la définition des bords dans chaque région d'une image [22].



Figure 24 : Filtre de CLAHE

- **Fermeture** : Les opérateurs morphologiques – dilater, éroder, ouvrir et fermer – peuvent être appliqués via le filtrage d'image pour agrandir ou réduire les régions de l'image, ainsi que pour supprimer ou remplir les pixels de limite des régions de l'image. Ces opérateurs de base, qui traitent les objets dans l'image d'entrée en fonction des caractéristiques codées dans l'élément structurant sélectionné.[23], sont décrits ci-dessous. Des filtres morphologiques supplémentaires incluent les transformations en forme de chapeau. Gradient morphologique et morphologique de Laplace.



Figure 25 :Filtre de fermeture

- **Filtrage inverse** :Est une technique utilisée dans le traitement du signal et le traitement d'images pour récupérer un signal ou une image originale à partir d'une version dégradée ou déformée. Il est basé sur l'idée d'inverser les effets d'un filtre ou d'un processus de dégradation connu.[24]



Figure 26: Filtrage inverse : (A) image Original, (B) image Floating, (C) image Inverse filtered

- Gaufrage** : Le gaufrage d'image est une technique d'infographie dans laquelle chaque pixel d'une image est remplacé soit par une lumière, soit par une ombre, en fonction des limites claires/foncées de l'image originale. Les zones à faible contraste sont remplacées par un fond gris. L'image filtrée représentera le taux de changement de couleur à chaque emplacement de l'image originale[25]. L'application d'un filtre de gaufrage à une image donne souvent une image ressemblant à un gaufrage en papier ou en métal de l'image originale, d'où son nom.

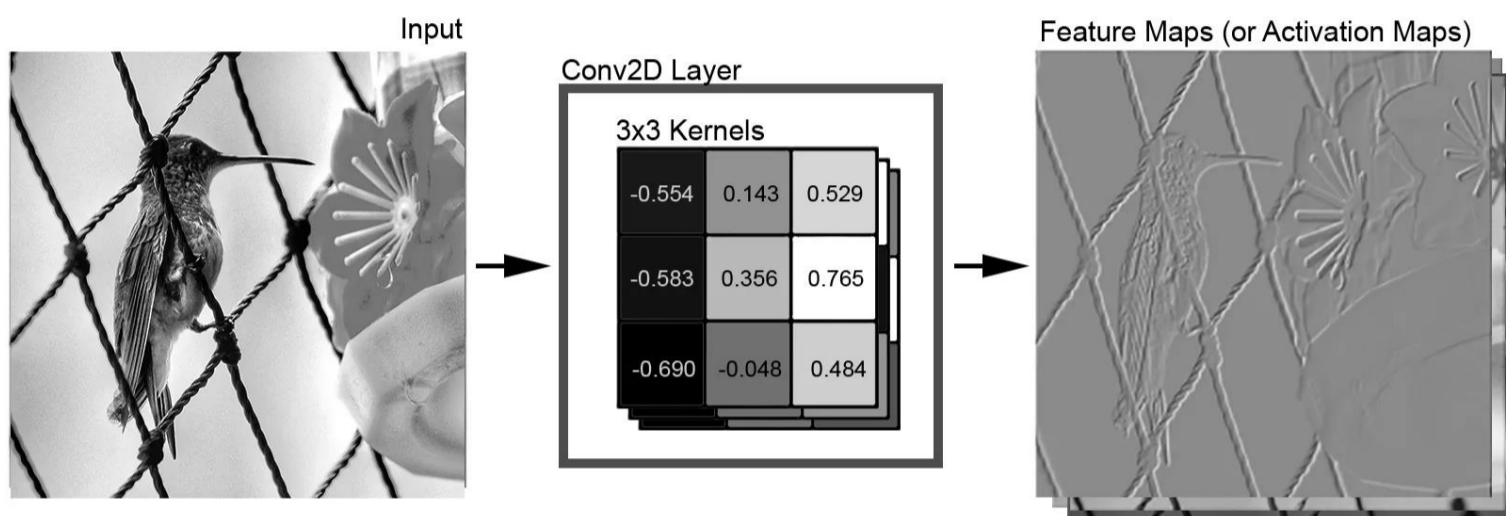


Figure 27 : Filtre de gaufrage

- Affûtage du filtre** : Les filtres de netteté sont un type de traitement d'image qui améliore le contraste entre les pixels voisins, rendant les bords et les détails plus visibles et définis. Les filtres de netteté peuvent également réduire les effets du bruit, du flou ou des artefacts de compression qui dégradent la qualité des images numériques. Les filtres de netteté peuvent être appliqués à l'image entière ou à des zones spécifiques, selon vos besoins et vos préférences.[26]

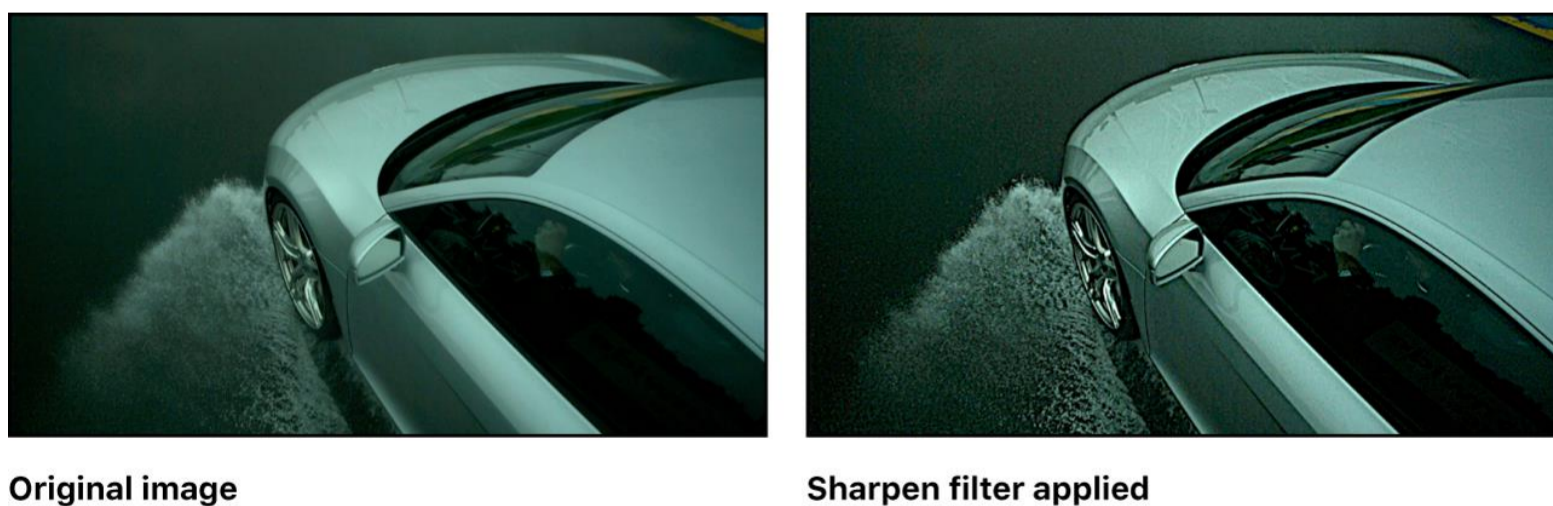


Figure 28: Filtre de Sharpening

4 Principe de fonctionnement d'un réseau CNN

Un réseau de neurones convaincants applique un ensemble d'opérations dans une image pour extraire et classer les informations comme indiqué sur la figure29. Ces types d'opérations sont les suivants:La convolution

- Le Pooling
- La fonction d'activation de type ReLU
- L'opération Flattening
- Couche entièrement connectée (fully-connected)

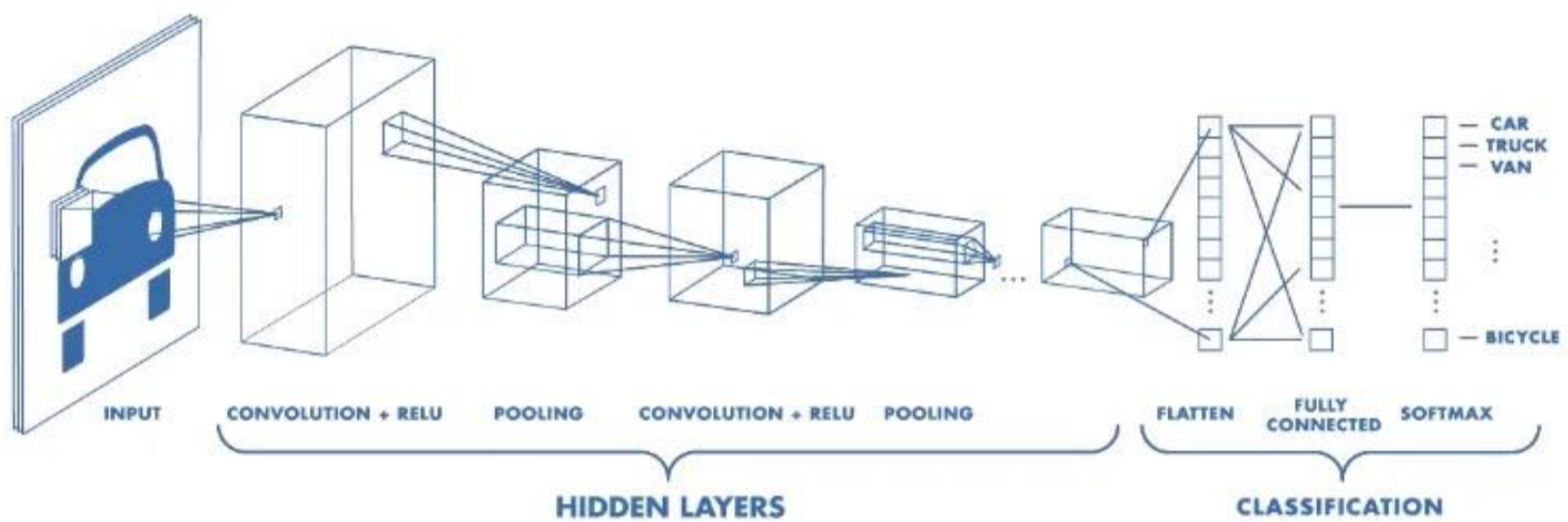


Figure 29: Principe Fondamental De l' Apprentissage Profond

4.1 Couche de convolution

Est la composante clé des réseaux de neurones convolutifs, et constitue toujours au moins leur première couche. Son but est de repérer la présence d'un ensemble de *features* dans les images reçues en entrée. Pour cela, on réalise un filtrage par convolution : le principe est de faire « glisser » une fenêtre représentant la *feature* sur l'image, et de calculer le produit de convolution entre la *feature* et chaque portion de l'image balayée. Une *feature* est alors vue comme un filtre : les deux termes sont équivalents dans ce contexte. La couche de convolution reçoit donc en entrée plusieurs images, et calcule la convolution de chacune d'entre elles avec chaque filtre. Les filtres correspondent exactement aux *features* que l'on souhaite retrouver dans les images. On obtient pour chaque paire (image, filtre) une carte d'activation, ou *featuremap*, qui nous indique où se situent les *features* dans l'image : plus la valeur est élevée, plus l'endroit correspondant dans l'image ressemble à la *feature*. [27]

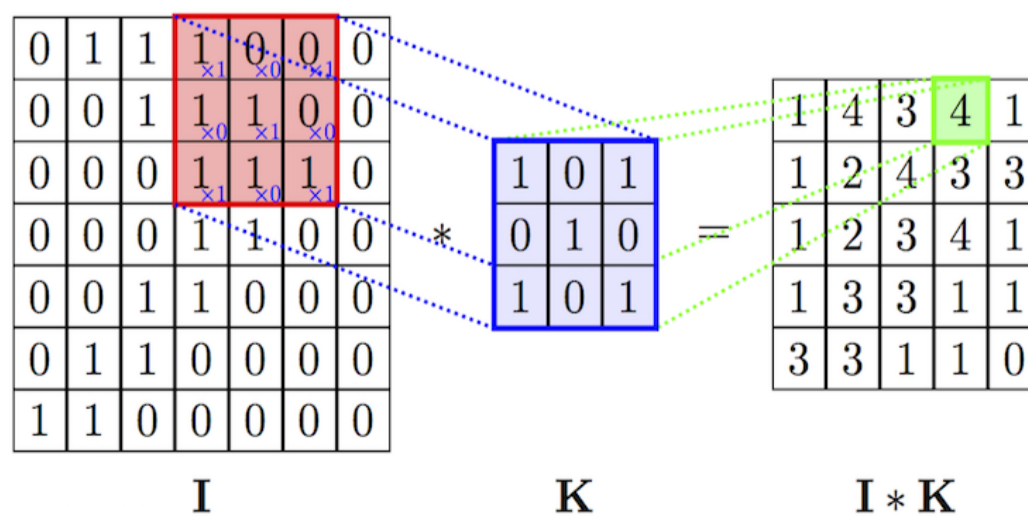


Figure 30 : Exemple Explicative Sur L'opération De Convolution

4.2 Couche Pooling

Ce type de couche est souvent placé entre deux couches de convolution : elle reçoit en entrée plusieurs *featuremaps*, et applique à chacune d'entre elles l'opération de *pooling*. L'opération de *pooling* consiste à réduire la taille des images, tout en préservant leurs caractéristiques importantes.

Pour cela, on découpe l'image en cellules régulières, puis on garde au sein de chaque cellule la valeur maximale. En pratique, on utilise souvent des cellules carrées de petite taille pour ne pas perdre trop d'informations. Les choix les plus communs sont des cellules adjacentes de taille 2×2 pixels qui ne se chevauchent pas, ou des cellules de taille 3×3 pixels, distantes les unes des autres d'un pas de 2 pixels (qui se chevauchent donc). On obtient en sortie le même nombre de *feature maps* qu'en entrée, mais celles-ci sont bien plus petites. 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. Les valeurs maximales sont repérées de manière moins exacte dans les *feature maps* obtenues après *pooling* que dans celles reçues en entrée – c'est en fait un grand avantage ! En effet, lorsqu'on veut reconnaître un chien par exemple, ses oreilles n'ont pas besoin d'être localisées le plus précisément possible : savoir qu'elles se situent à peu près à côté de la tête suffit.

Ainsi, la couche de *pooling* rend le réseau moins sensible à la position des *features* : le fait qu'une *feature* se situe un peu plus en haut ou en bas, ou même qu'elle ait une orientation légèrement différente ne devrait pas provoquer un changement radical dans la classification de l'image.[27]

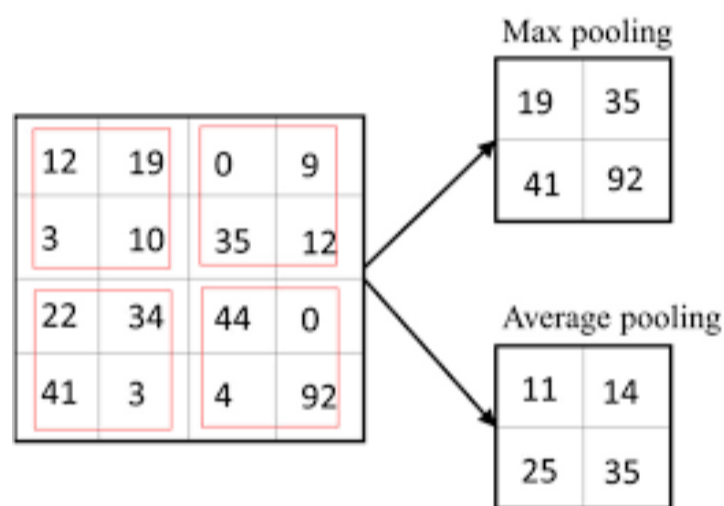


Figure 31: Pooling Max

4.3 Entièrement connecté

La couche de Entièrement connecté (FC) s'applique sur une entrée préalablement aplatie où chaque entrée est connectée à tous les neurones. Les couches de fully connected sont typiquement présentes à la fin des architectures de CNN et peuvent être utilisées pour optimiser des objectifs tels que les scores de classe[28].

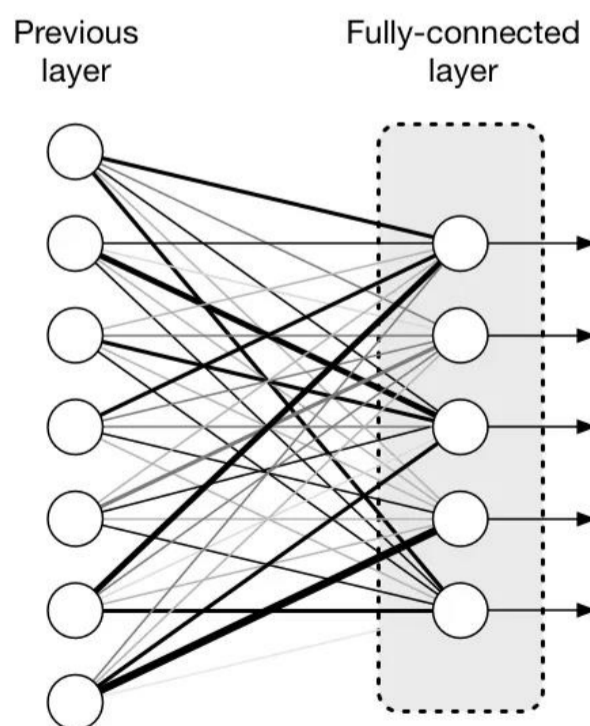


Figure 32: Couche Fully-Connected

4.4 Fonction d'activation

Voici les principales fonctions d'activations que l'on peut trouver dans le domaine des réseaux de neurones :

- **Linear** : Utilisé en couche de sortie pour une utilisation pour une régression. On peut la caractériser de nulle, puisque les unités de sortie seront identiques à leur niveau d'entrée Intervalle de sortie $(-\infty; +\infty)$. [29]

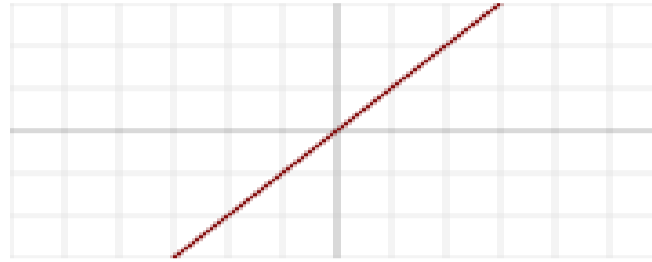


Figure 33:Fonction D'activation Linear

- **Step** : Elle renvoi tout le temps 1 pour un signal positif, et 0 pour un signal négatif. [29]

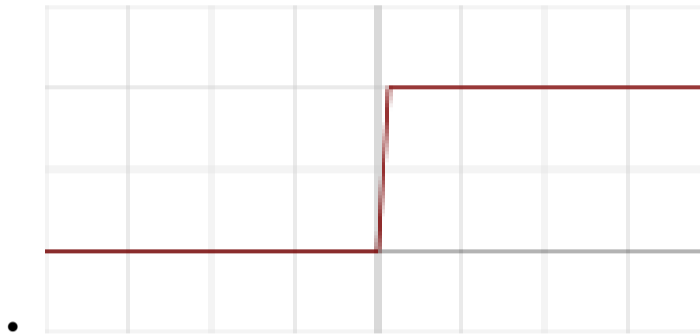


Figure 34:Fonction D'activation Step

- **Sigmoid (logistic)** : Fonction la plus populaire depuis des décennies. Mais aujourd'hui, elle devient beaucoup moins efficace par rapport à d'autre pour une utilisation pour les couches cachées. Elle perd de l'information due à une saturation que cela soit pour la phase de feedforward ou de back propagation, en donnant des effets non linéaires au réseau due à un paramètre unique. Elle a aussi des soucis de gradient 0 avec des entrées étant très large, même si le soucis est minimalisé avec les système utilisant des batch par lots (mini batch) [29]. Utilisé en couche de sortie pour de la classification binaire Intervalle de sortie : $\{0,1\}$

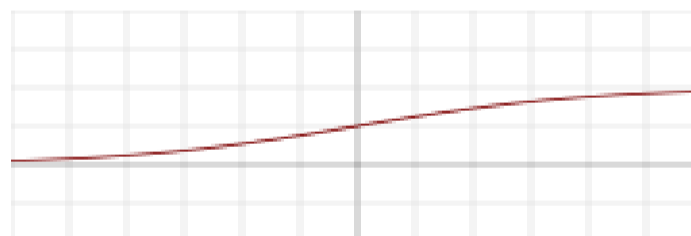


Figure 35 :Fonction D'activation Sigmoid

- **TanH** : Utilisé pour des LSTM pour des données en continue Intervalle de sortie : $(-1,1)$

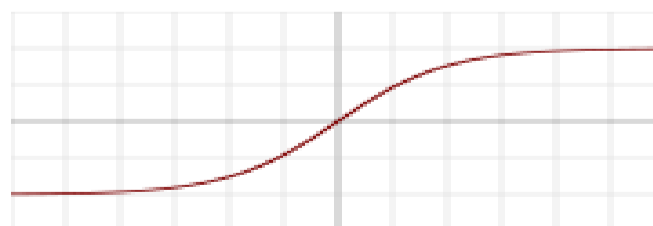


Figure 36 :Fonction D'activation Tanh

- **ReLU (RectifiedLinear Unit)** : Ce sont les fonctions les plus populaires de nos jours. Elles permettent un entrainement plus rapide comparé aux fonctions sigmoid et tanh, étant plus légères. Attention au phénomène de ‘Dying ReLU’, auquel on préférera les variations de ReLU. Plus d’informations en fin d’article. Très utilisé pour les CNN, RBM, et les réseaux de multi perceptron Intervalle de sortie $(0;+\infty)$. [29]

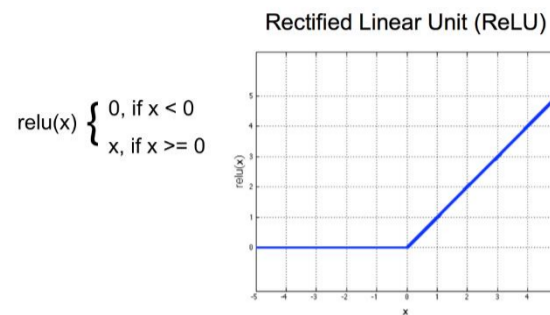


Figure 37:Fonction D’activation Relu

4.5 Conv Net les plus connus

Il existe différentes architectures de Réseaux de Neurones Convolutionnels disponibles qui ont été essentielles dans la construction d’algorithmes qui alimentent et alimenteront l’IA dans un avenir prévisible. Notamment :

LeNet

Est une structure de réseau neuronal convolutionnel proposée par leCun et al. En 1998.[1] En général, lenet fait référence à lenet-5 et est un réseau neuronal convolutionnel simple. Les réseaux neuronaux convolutionnels sont une sorte de réseau neuronal à propagation directe dont les neurones artificiels peuvent répondre à une partie des cellules environnantes dans la plage de couverture et fonctionnent bien dans le traitement d’images à grande échelle [30]

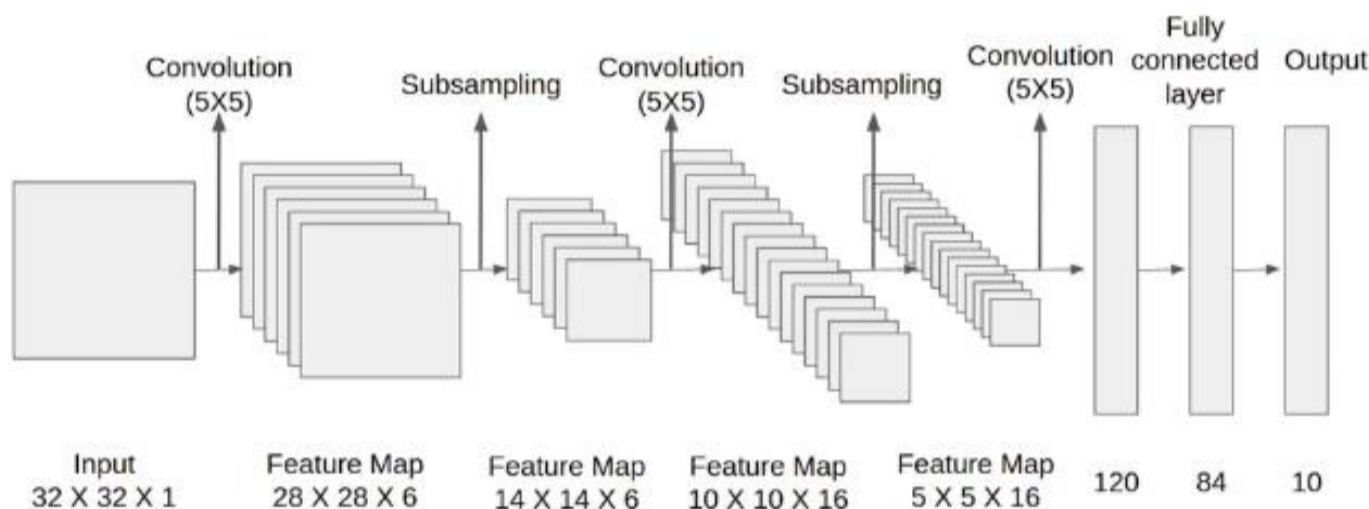


Figure38 : Diagramme de structure du réseau neuronal convolutionnel Net

Google Net

Il s’agit d’un réseau neuronal convolutionnel de 22 couches de profondeur. Vous pouvez charger une version pré-entraînée du réseau formé sur le réseau d’images ou sur les ensembles de données places365. Le réseau formé sur le réseau d’images classe les images en 1 000 catégories d’objets, telles que le clavier, la souris, le crayon et de nombreux animaux. Le réseau formé sur places365 est similaire au réseau formé sur le réseau d’images, mais classe les images en 365 catégories de lieux différentes, telles que le champ, le parc, la piste et le hall d’entrée. Ces réseaux ont appris différentes représentations de caractéristiques pour une large gamme d’images. Les deux réseaux pré-entraînés ont une taille d’entrée d’image de 224 x 224 [31].

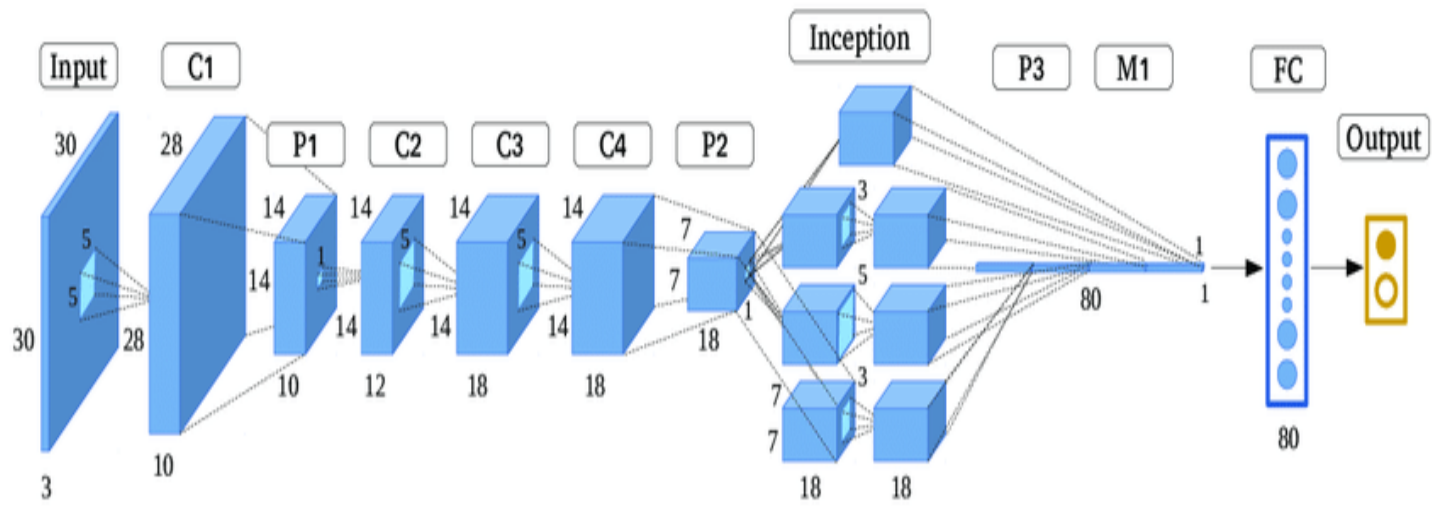


Figure 39 : Diagramme De Structure Du Réseau Neuronal Convolutionnel GoogLeNet

AlexNet

Le 30 septembre 2012, le réseau a participé au concours ImageNet Large Scale Visual Recognition Challenge. Le réseau a obtenu une erreur de 15,3 % dans le top 5, soit plus de 10,8 points de pourcentage de moins que le deuxième. Le principal résultat de l'article original était que la profondeur du modèle était essentielle à ses hautes performances, qui étaient coûteuses en termes de calcul, mais rendues possibles grâce à l'utilisation d'unités de traitement graphique (GPU) pendant la formation.[32]

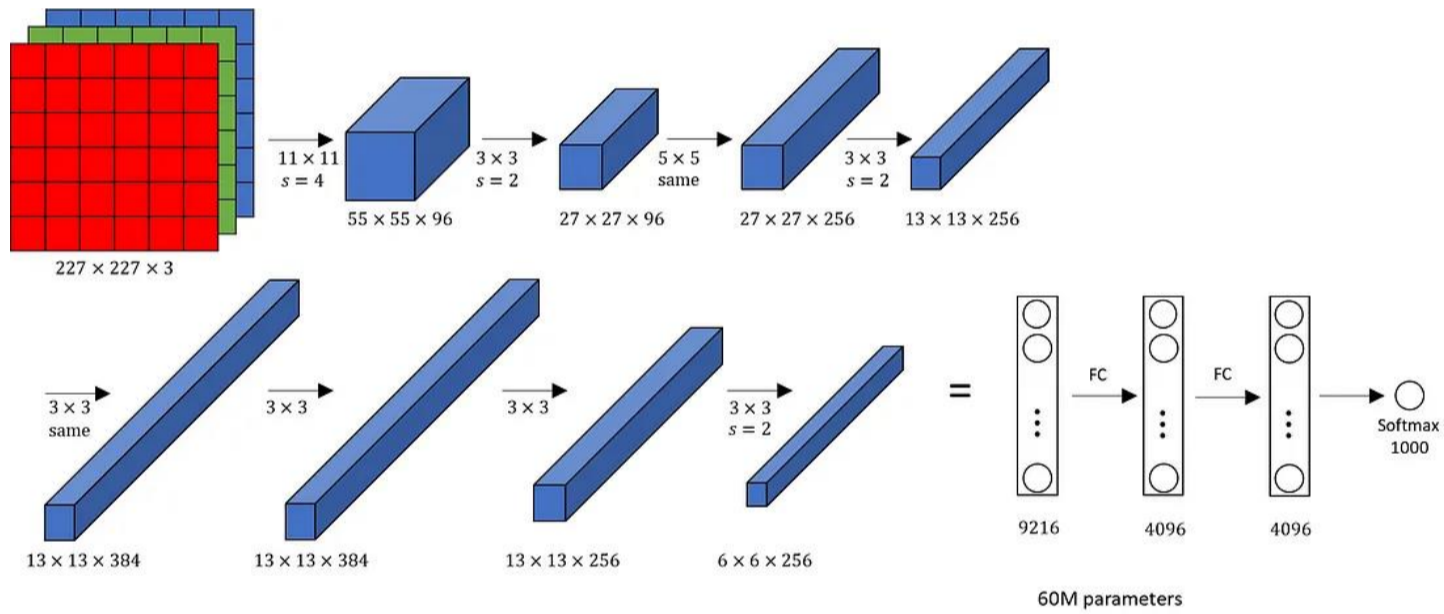


Figure 40 : Diagramme De Structure Du Réseau Neuronal Convolutionnel Alexnet

5 Conclusion

Les réseaux de neurones profonds, tels que les CNN, ont transformé de nombreux domaines tels que la classification, l'optimisation et le traitement de Big Data. Ces avancées ont propulsé l'intelligence artificielle vers des niveaux de complexité et de sophistication jamais atteints auparavant, avec des applications remarquables dans la reconnaissance d'images, l'analyse de texte, la reconnaissance vocale, Cependant, des défis persistent, notamment en matière de sécurité des données, de transparence et d'équité dans les systèmes intelligents, d'optimisation des modèles profonds et de gestion de l'interaction entre le Big Data et les capacités de calcul requises.



Chapitre 03 : Application et Résultats

1 Introduction

Ce chapitre se concentre sur la création d'algorithmes d'apprentissage profond pour la classification d'images à l'aide de réseaux de neurones. Nous appliquons le modèle CNN à notre base de données d'images, qui est l'ensemble de données MNIST. Pour de meilleurs résultats, nous utilisons les bibliothèques TensorFlow et Keras disponibles dans l'environnement Python pour la classification et la photographie automatisées. Pour améliorer les performances des modèles, nous utilisons certaines techniques d'augmentation des données et éliminons les problèmes d'ajustement excessifs. Nous avons utilisé la base de données MNIST et divisé les données en deux ensembles de données : l'ensemble d'entraînement et l'ensemble de test. Nous avons comparé sa précision avec un autre modèle utilisant des filtres par défaut.

2 Bases des données d'images

Notre base de données MNIST est une collection de données populaire. Il contient un groupe d'images numériques en noir et blanc de nombres dessinés à la main (de 0 à 9) comme le montre la figure 41. Il se compose de 60 000 photos d'entraînement et de 10 000 photos d'essai. Chaque image de l'image 28 x 28 pixels.

```

Training data shapes: X=(60000, 28, 28), y=(60000,)
Testing data shapes: X=(10000, 28, 28), y=(10000,)

```

figure 41 : ensemble de données mnist

3 Logiciels, librairies et matériels utilisés dans l'implémentation

a. Python

Python est un langage de programmation open source le plus employé par les informaticiens. Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels. En effet, parmi ses qualités, Python permet notamment aux développeurs de se concentrer sur ce qu'ils font plutôt que sur la manière dont ils le font. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens[33]. Ainsi, développer du code avec Python est plus rapide qu'avec d'autres langages .

b. Tensorflow

Tensorflow est un framework de programmation pour le calcul numérique qui a été rendu Open Source par Google en novembre 2015. Depuis sa création, le Tensorflow n'a cessé de gagner en popularité, pour devenir très rapidement l'un des framework les plus utilisés pour le Deep Learning et donc les réseaux de neurones. Son nom est notamment inspiré du fait que les opérations courantes sur des réseaux de neurones sont principalement faites via des tables de données multidimensionnelles, appelées Tenseurs (Tensor). Un Tensor à deux dimensions est l'équivalent d'une matrice. Aujourd'hui, les principaux produits de Google sont basés sur Tensorflow : Gmail, google Photos, Reconnaissance de voix [34].

c. Keras

Keras est un réseau de neurones de haut niveau, écrite en Python et capable de fonctionner sur Tensorflow ou Theano. Il a été développé en mettant l'accent sur l'expérimentation rapide. Il a été développé dans le cadre de l'effort de recherche du projet ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System), et son principal auteur et mainteneur est François Chollet, un ingénieur Google. En 2017, l'équipe Tensorflow de Google a décidé de soutenir Keras dans la bibliothèque principale de Tensorflow. Chollet a expliqué que Keras a été conçue comme une interface plutôt que comme un cadre d'apprentissage end to end. Il présente un ensemble d'abstractions de niveau supérieur et plus intuitif qui facilitent la configuration des réseaux neuronaux indépendamment de la bibliothèque informatique de backend [35].

d. Jupyter Notebook

Jupyter Notebook est destiné à la formation et à la recherche dans l'apprentissage automatique. Cette plateforme permet d'entraîner des modèles de Machine Learning directement dans le cloud. Sans donc avoir besoin d'installer quoi que ce soit sur notre ordinateur à l'exception d'un navigateur[36].

e. les ressources matérielles

Lors de la recherche, j'utilise diverses ressources pour trouver les objets recherchés et développer le projet. Ces ressources comprennent des ordinateurs et des logiciels utilisés pour analyser les données et développer des modèles informationnels. PC Dell Équipé de GPU Intel Core i5 de 8 génération et 16 Go de RAM à 2,30 GHz, avec logiciel d'analyse des données de Python (TensorFlow et Keras), et d'autres livres scientifiques utilisés pour répondre aux algorithmes nécessaires pour l'analyse et l'application automatique.

4 Résultats obtenus et discussion

4.1 Premier modèle

Il se compose d'une couche multiple et de deux couches inférieures. L'échantillon contient deux couches complètement connectées. Taille de L'image d'entrée est de 28 x 28 pixels et l'image est d'abord transférée vers la première couche torsadée. Cette couche Il se compose de 16 filtres 3 x 3, suivis d'une fonction de lecture en référence. Cette fonction oblige les cellules nerveuses à retourner Valeurs positives, Ensuite, nous appliquons une sous-vente pour réduire la taille de l'image et donc la quantité. Paramètres et compte. Quand nous produisons Cette couche, nous aurons 16 fonctionnalités 14x14.L'apprentissage et la formation (formation) et les tests (tests) augmentent. Avec le nombre de répétitions, cela reflète cela à chaque période, Le modèle apprend plus d'informations. Si la précision est réduite, nous avons besoin de plus d'informations pour apprendre notre modèle, alors nous sommes Par conséquent, le nombre de répétitions et vice versa doivent être augmentés. De même, l'erreur du lecteur est diminué avec le nombre de répétitions et d'erreur Le test augmente avec le nombre d'âges. L'erreur de classification dans la formation est de 0,11 et la précision du test est de 0,98 comme le montre **la figure 41 et la figure 42, et le tableau 1.**

4.2 Architecture de modèle 1

Nous avons proposé un modèle simple et obtenu des résultats bons et acceptables. Notre modèle se compose d'une couche convolutive et d'une sous-couche. Échantillonnage et abandon de (0,4) et deux couches entièrement connectées. La taille de l'image d'entrée est de 28 x 28 et l'image passe d'abord à travers le premier calque. envelopper. Cette couche est composée de 16 filtres 3x3, pour chacune de nos couches

La convolution suit la fonction d'activation ReLU, et cette fonction force les neurones à le faireRenvoi des valeurs positives, après cette convolution nous appliquons sous échantillonnage, Réduisez la taille de l'image ainsi que la quantité de paramètres et de calculs. Quand tu sortiras de ça Couche, nous aurons 16 cartes de fonctionnalités d'une taille de 14x14. comme le montrent tableau 01

- **Tableau de Résumé du modèle 01**

Layer (type)	Output Shape
Conv2d_1(Conv2D)	(None, 26, 26, 16)
Max_pooling2d_2(maxpooling2D)	(None, 13, 13, 16)
Flatten_2(flatten)	(None ,2704)
Dense_4(dense)	(None ,64)
Dense_5(dense)	(None ,10)

Tableau 01 :Résumé du modèle 01

4.3 Graphe de précision et d'erreur du model 1

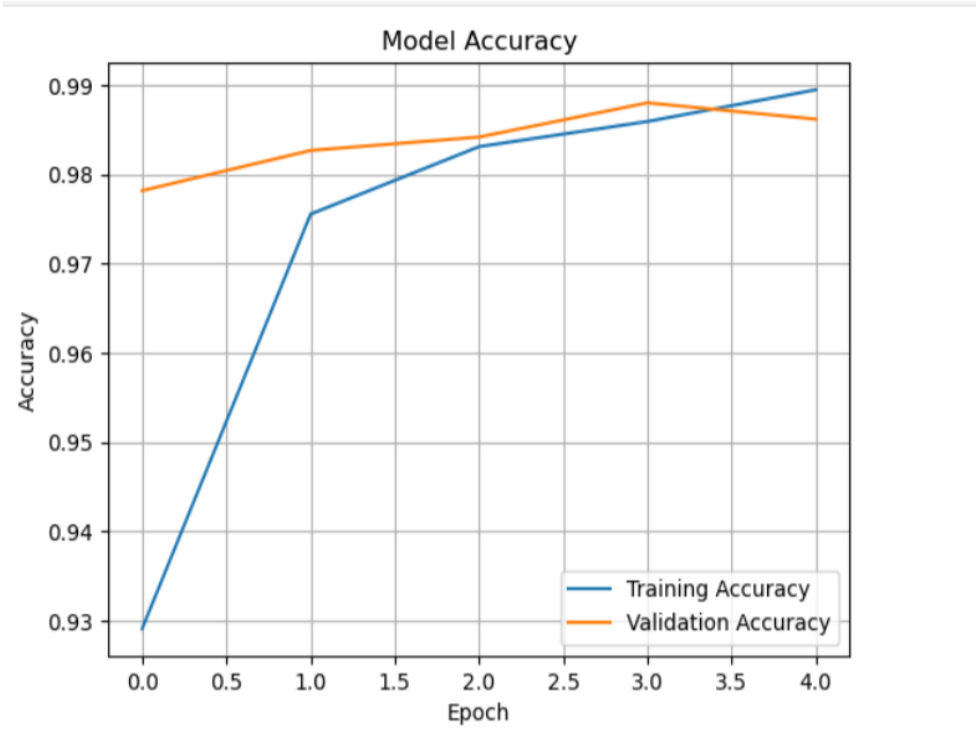


Figure 41 : Graphe test de Précision Modèle 1

```
Epoch 1/10
844/844 ————— 9s 8ms/step - accuracy: 0.8649 - loss: 0.4807 - val_accuracy: 0.9730 - val_loss: 0.1005
Epoch 2/10
844/844 ————— 6s 8ms/step - accuracy: 0.9714 - loss: 0.0968 - val_accuracy: 0.9810 - val_loss: 0.0750
Epoch 3/10
844/844 ————— 6s 7ms/step - accuracy: 0.9826 - loss: 0.0615 - val_accuracy: 0.9853 - val_loss: 0.0578
Epoch 4/10
844/844 ————— 7s 8ms/step - accuracy: 0.9863 - loss: 0.0448 - val_accuracy: 0.9862 - val_loss: 0.0544
Epoch 5/10
844/844 ————— 6s 7ms/step - accuracy: 0.9901 - loss: 0.0348 - val_accuracy: 0.9867 - val_loss: 0.0552
Epoch 6/10
844/844 ————— 7s 8ms/step - accuracy: 0.9922 - loss: 0.0273 - val_accuracy: 0.9880 - val_loss: 0.0496
Epoch 7/10
844/844 ————— 6s 8ms/step - accuracy: 0.9936 - loss: 0.0208 - val_accuracy: 0.9867 - val_loss: 0.0552
Epoch 8/10
844/844 ————— 6s 7ms/step - accuracy: 0.9945 - loss: 0.0185 - val_accuracy: 0.9868 - val_loss: 0.0574
Epoch 9/10
844/844 ————— 7s 8ms/step - accuracy: 0.9964 - loss: 0.0127 - val_accuracy: 0.9852 - val_loss: 0.0662
Epoch 10/10
844/844 ————— 6s 7ms/step - accuracy: 0.9968 - loss: 0.0113 - val_accuracy: 0.9878 - val_loss: 0.0573
313/313 ————— 1s 3ms/step - accuracy: 0.9803 - loss: 0.0626
Test accuracy: 0.9840999841690063
```

Figure 42: test de précision

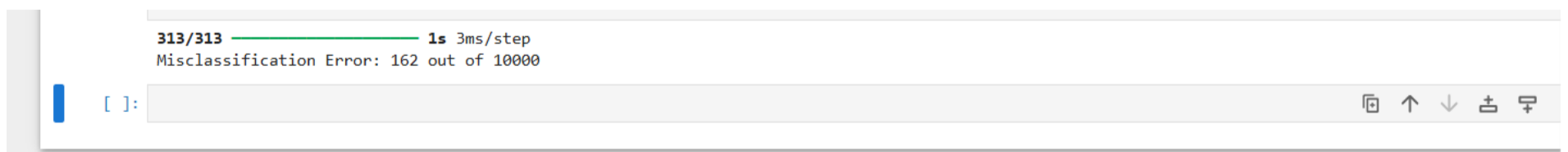


Figure 43 :Compter Le Nombre D'images Mal Classées

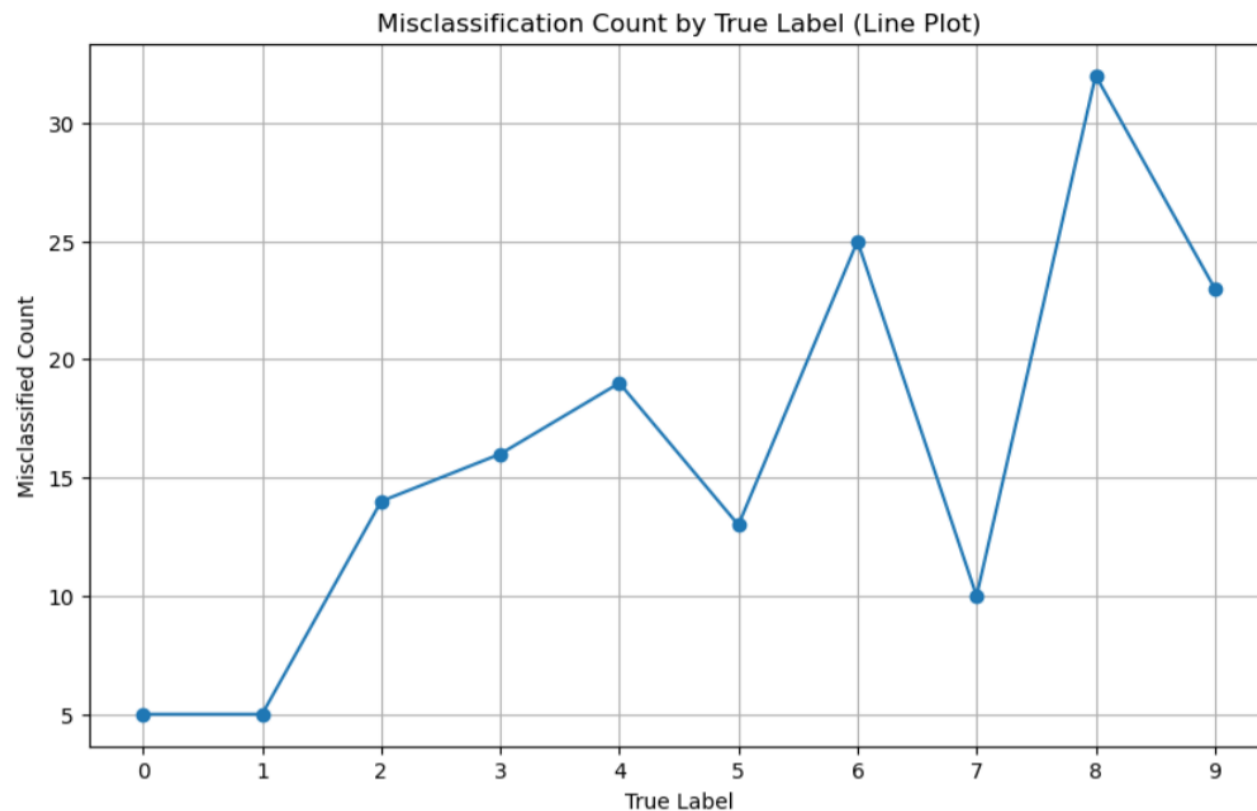


Figure 44 :Graphique D'erreur

4.4 Analyse graphique

En analysant le graphique, les observations suivantes peuvent être tirées :

1. Catégorie « 9 » :

- Il contient le plus grand nombre d'erreurs de classification, avec un nombre d'erreurs supérieur à 30. Cela indique que le modèle a de grandes difficultés à classer correctement cette catégorie.

2. Catégories « 8 », « 7 » et « 6 » :

- Il contient un grand nombre d'erreurs de classification, compris entre 20 et 30. Ces catégories présentent d'autres défis importants pour le modèle.

3. Catégories « 3 » et « 5 » :

- Il contient un nombre moyen d'erreurs de classement, compris entre 10 et 20. Ce qui signifie que le modèle a des performances moyennes dans la classification de ces catégories.

4. Catégories « 2 », « 0 » et « 1 » :

- Il contient le moins d'erreurs de classification, le nombre d'erreurs étant d'environ 10 ou moins pour chacune. Cela indique que le modèle fonctionne bien dans la classification de ces catégories par rapport aux autres catégories.

Résultat

Ce graphique montre qu'il existe une variation dans les performances du modèle entre les différentes catégories. Les catégories telles que « 9 », « 8 » et « 7 » nécessitent des améliorations supplémentaires dans le modèle en raison du grand nombre d'erreurs de classification. Alors que des catégories telles que « 0 » et « 1 » montrent de bonnes performances du modèle.

4.5 Deuxième modèle

Il se compose d'une couche convolutive, d'une couche de pooling et de deux sous-couches. Le processus d'échantillonnage se compose de deux couches entièrement connectées. La taille de l'image d'entrée est de 28 x 28 pixels et l'image est d'abord transférée vers la première couche convolutive. Cette couche est composée de 16 filtres d'une taille de 3x3. La couche d'activation convolutive est suivie de la fonction RELU. Cette fonction oblige les neurones à renvoyer des valeurs positives et on applique ensuite un sous-échantillonnage pour réduire la taille de l'image et donc la quantité de paramètres et de calculs. Lorsque nous produisons cette couche, nous aurons 16 cartes de caractéristiques d'une taille de 14x14. Nous passons par la couche convolutive de 16 filtres de taille 3*3, sur la figure 47 nous pouvons voir les résultats de la classification, la validité

(précision) de l'apprentissage, de la formation et des tests est augmentée. Le nombre d'itérations reflète le fait qu'à chaque période, le modèle apprend plus d'informations. Si la précision diminue, nous avons besoin de plus d'informations pour faire apprendre

notre modèle, nous devons donc augmenter le nombre d'itérations et vice versa. De même, l'erreur de formation diminue avec le nombre d'itérations et l'erreur de test augmente avec le nombre d'époques. Pour la formation, l'erreur de classification est de 0,26, la précision est de 0,91, l'erreur d'évaluation est de 0,91 % et l'erreur de test est de 1,47, comme le montrent les figures 47 et 48

4.6 Architecture de modèle proposé

Nous avons proposé un modèle simple et obtenu des résultats bons et acceptables. Notre modèle se compose d'une couche convolutive et d'une sous-couche. Échantillonnage et abandon de (0,4) et deux couches entièrement connectées. La taille de l'image d'entrée est de 28 x 28 et l'image passe d'abord à travers le premier calque envelopper. Cette couche est composée de 16 filtres 3x3, pour chacune de nos couches

La convolution suit la fonction d'activation ReLU, et cette fonction force les neurones à le faire Renvoie des valeurs positives, après cette convolution nous appliquons sous échantillonnage, Réduisez la taille de l'image ainsi que la quantité de paramètres et de calculs. Quand tu sortiras de ça Couche, nous aurons 16 cartes de fonctionnalités d'une taille de 14x14. comme le montrent tableaux 02

- **Tableau de Résumé du modèle 02**

Layer (type)	Output Shape
Conv2d_1(Conv2D)	(None, 26, 26, 16)
Max_pooling2d_2(maxpooling2D)	(None, 13, 13, 16)
Flatten_2(flatten)	(None ,2704)
Dense_4(dense)	(None ,64)
Dense_5(dense)	(None ,10)

Tableau 02:Résumé du modèle 02

4.7 Types des filtres appliqués au deuxième modèle :

Filtre1: blur

Filter2: gaussaian blur

Filtre 3: median blur

Filtre 4: bilateral

Filtre 5: sobel x

Filtre 6: sobel y

Filtre 7: laplacian

Filtre 8: canny edge detector

Filtre 9: Unsharp Masking

Filtre 10: Thresholding

Filtre 11: Adaptive Thresholding

Filtre 12: Dilation

Filtre 14: Gradient

Filtre 15: Histogram Equalization

Filtre 16: dilatation

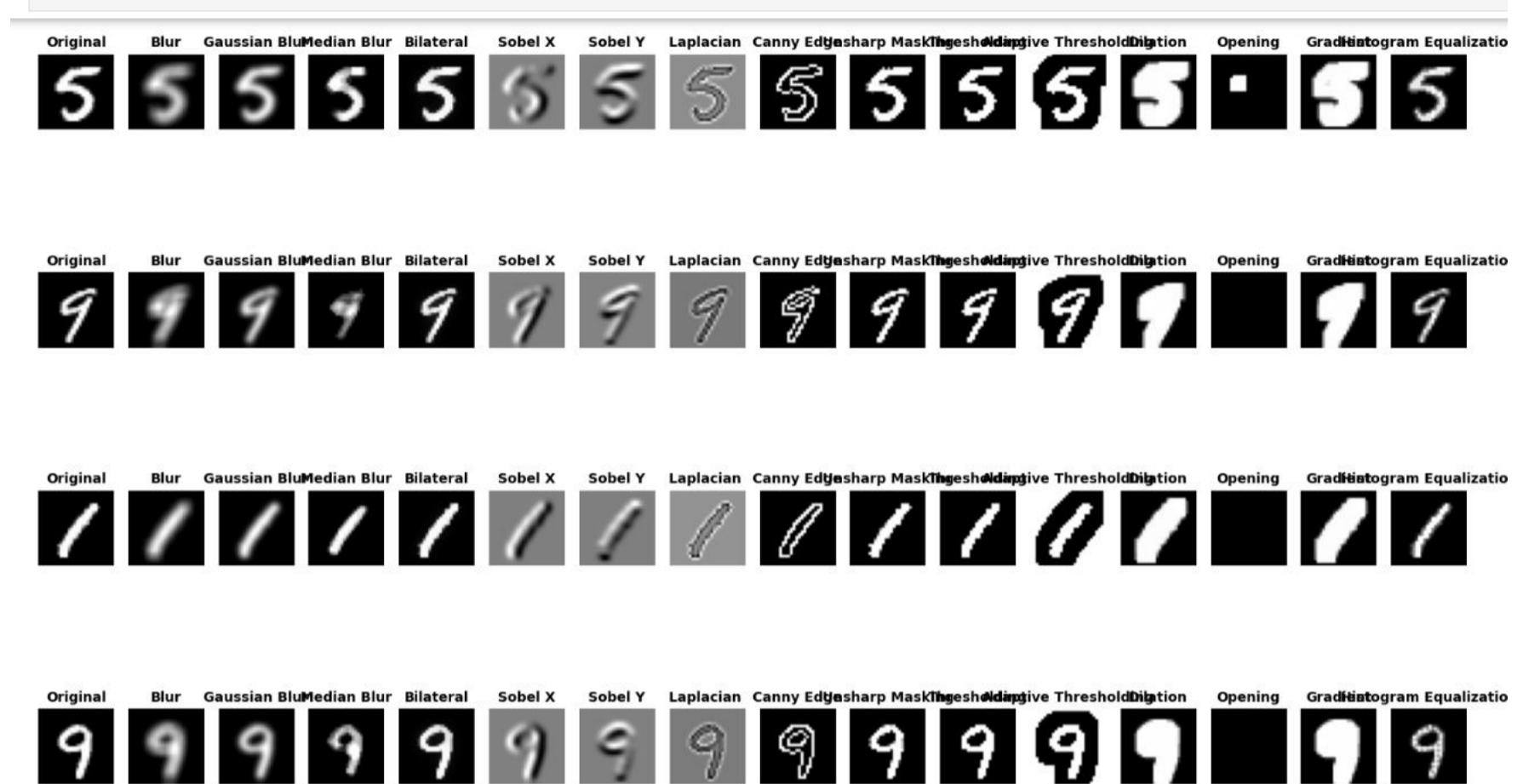


Figure 45 :Les Filtres Appliqués Au Deuxieme Modèle

4.8 Graphe de précision et d'erreur du model 2

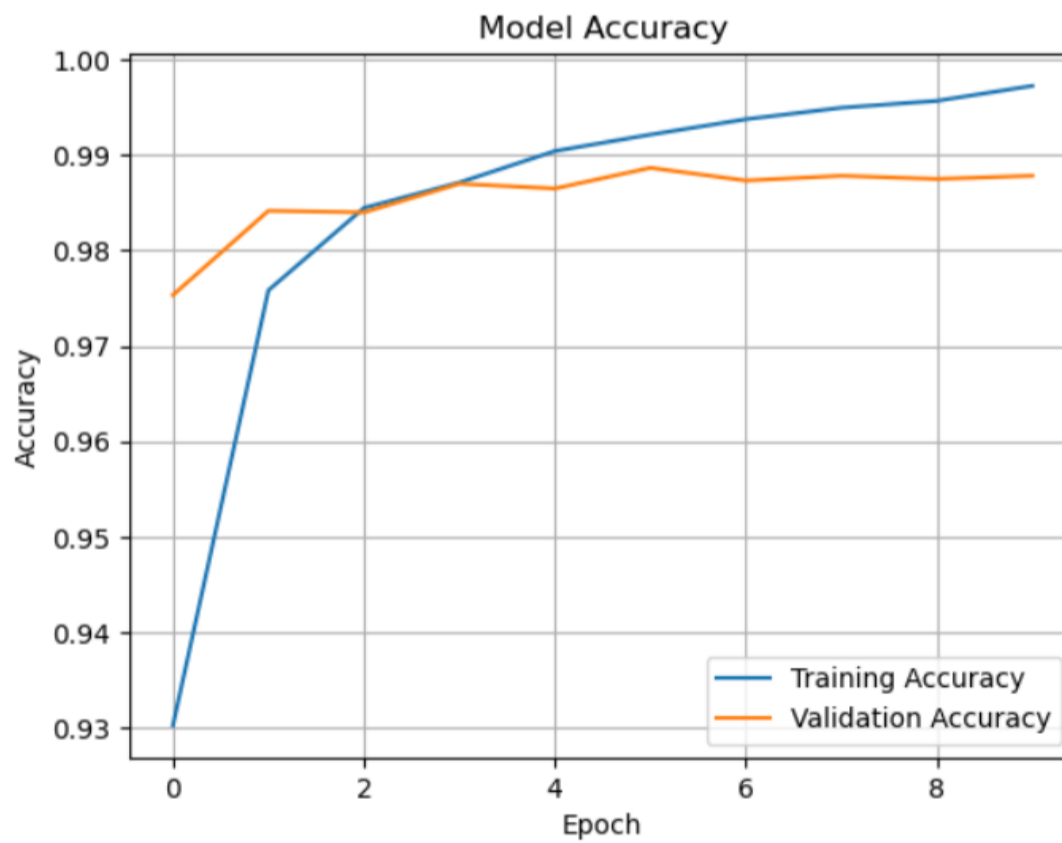


Figure 46 : Graphe De Précision Model 2

```

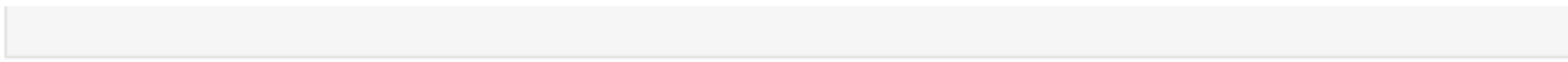
Epoch 1/10
844/844 ————— 4s 3ms/step - accuracy: 0.6723 - loss: 1.2198 - val_accuracy: 0.8890 - val_loss: 0.3774
Epoch 2/10
844/844 ————— 2s 3ms/step - accuracy: 0.8721 - loss: 0.4186 - val_accuracy: 0.9007 - val_loss: 0.3251
Epoch 3/10
844/844 ————— 3s 3ms/step - accuracy: 0.8847 - loss: 0.3678 - val_accuracy: 0.9083 - val_loss: 0.3025
Epoch 4/10
844/844 ————— 2s 3ms/step - accuracy: 0.8924 - loss: 0.3419 - val_accuracy: 0.9112 - val_loss: 0.2868
Epoch 5/10
844/844 ————— 2s 2ms/step - accuracy: 0.9008 - loss: 0.3168 - val_accuracy: 0.9157 - val_loss: 0.2701
Epoch 6/10
844/844 ————— 2s 3ms/step - accuracy: 0.9070 - loss: 0.2964 - val_accuracy: 0.9197 - val_loss: 0.2572
Epoch 7/10
844/844 ————— 2s 3ms/step - accuracy: 0.9109 - loss: 0.2791 - val_accuracy: 0.9247 - val_loss: 0.2426
Epoch 8/10
844/844 ————— 2s 3ms/step - accuracy: 0.9162 - loss: 0.2628 - val_accuracy: 0.9275 - val_loss: 0.2323
Epoch 9/10
844/844 ————— 2s 2ms/step - accuracy: 0.9229 - loss: 0.2443 - val_accuracy: 0.9313 - val_loss: 0.2239
Epoch 10/10
844/844 ————— 2s 3ms/step - accuracy: 0.9273 - loss: 0.2343 - val_accuracy: 0.9343 - val_loss: 0.2145
313/313 ————— 1s 2ms/step - accuracy: 0.9147 - loss: 0.2618
Test accuracy: 0.9258000254631042
    
```

Figure 47: Test De Précision

```

313/313 ————— 1s 2ms/step
Misclassification Error: 1.47%
    
```

Figure 48 : Ratio D'erreur De Classification



```

313/313 ————— 1s 2ms/step
Misclassification Error: 147 out of 10000
    
```

Figure 49 : Compter Le Nombre D'images Mal Classées

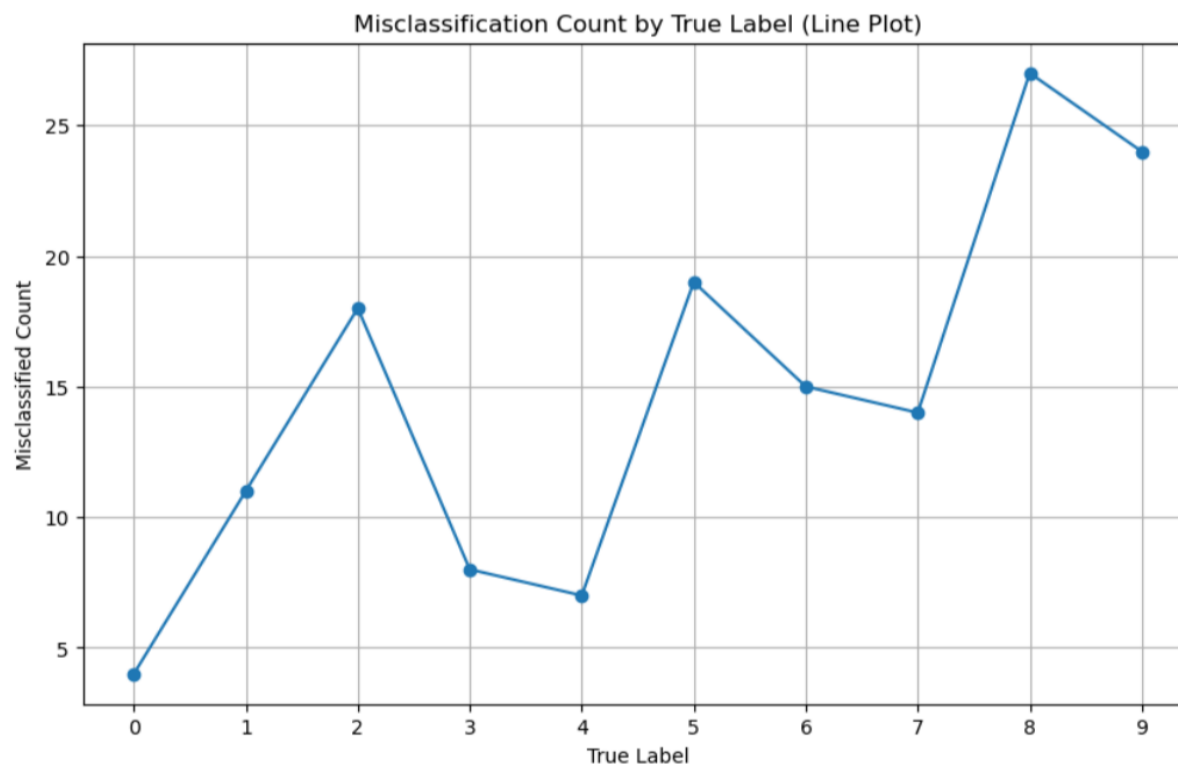


Figure 50 : Graphe D'erreur

4.9 Analyse graphique

Il contient le plus grand nombre d'erreurs de classification, puisque le nombre d'erreurs dépasse 120. Cela indique que le modèle a de grandes difficultés à classer correctement cette catégorie.

2. Catégories « 3 », « 7 », « 8 » et « 9 » :

Chacun contient un grand nombre d'erreurs de classification, allant de 80 à 100. Ces catégories présentent d'autres défis importants pour le modèle.

3. Catégories « 2 », « 5 » et « 6 » :

- Il contient un nombre moyen d'erreurs de classement, compris entre 40 et 60. Ce qui signifie que le modèle a des performances moyennes dans la classification de ces catégories.

4. Catégories « 0 » et « 1 » :

- Il contient le moins d'erreurs taxonomiques, environ 30 chacune. Cela indique que le modèle fonctionne bien dans la classification de ces catégories par rapport aux autres catégories.

Un résultat

Ce graphique montre qu'il existe une variation dans les performances du modèle entre les différentes catégories. Les catégories telles que « 4 » et « 9 » nécessitent d'autres améliorations du modèle en raison du grand nombre d'erreurs de classification. Alors que des catégories telles que « 0 » et « 1 » montrent de bonnes performances du modèle.

Les figures ci-dessus représentent la courbe de précision et de perte d'entraînement du modèle CNN avec un nombre d'Epochs égale à 10. Nous pouvons voir que la précision d'entraînement augmente à chaque itération jusqu'à atteindre 98.5% de précision, ceci explique qu'à chaque itération le modèle apprend plus d'information sur les images, pour mieux les classer et améliorer la précision ainsi que la performance du modèle. D'autre part la Figure 42, montre que l'erreur de classification diminue au fur et à mesure à chaque itération.

4.10 Tableau de comparaisant des résultats

Model	Nombre de Couch	Images bien classés	Images mal classés	Taux de précision	Taux d'erreur	Temps d'exécution
01	01	9842	158	0.98%	0,061%	1,7min
02	01	9184	816	0.92%	0,2945%	1,9min

Tableaux 3 :Tableau de comparaison des résultats

Les résultats obtenus en termes d'apprentissage sont exprimés avec précision, les tests de vérification et de précision, et enfin des erreurs au moment de la mise en œuvre. Le temps varie en fonction de la base après avoir utilisé les données de base selon le premier modèle, le degré de précision obtenu est de 98,39%. D'un autre côté, avec le deuxième modèle, nous avons pu obtenir un petit degré de 92,1%. Le résultat ne s'est pas amélioré dans le deuxième modèle après avoir appliqué un ensemble de filtres proposés.

Remqraue:

Je voulais essayer une application de 32 filtres ou plus sur notre modèle CNN, mais en raison des capacités limitées de l'ordinateur sur lesquelles je travaille, je ne pouvais pas le faire. L'appareil actuel n'a pas suffisamment de capacité en termes de traitement Et la mémoire nécessaire pour effectuer des opérations complexes, j'ai entravé ma capacité à mener les expériences requises et à obtenir les résultats souhaités. sur les performances du modèle plus efficacement.

5 Conclusion

Dans cette étude, nous avons exploré l'amélioration des performances d'un réseau neuronal entièrement intégré (CNN) dans la classification d'images à l'aide de l'ensemble de données MNIST. 16 filtres différents ont été appliqués aux images pour améliorer la capacité discriminante du modèle. Les résultats ont montré que les performances du modèle de réseau neuronal convolutif sur lequel nous avons proposé des filtres étaient de 92% pour cent, légèrement inférieures aux performances du modèle qui utilisait des filtres par défaut et dont la précision des performances était de 98% pour cent. Cette recherche indique l'importance de l'utilisation de filtres techniques pour améliorer la reconnaissance des images, comme l'utilisation de divers filtres pour les reconnaître. Cette approche peut contribuer à améliorer les performances des applications d'IA qui s'appuient sur la classification d'images, ouvrant la porte à des utilisations nouvelles et avancées dans des domaines tels que la reconnaissance faciale, l'imagerie médicale, la surveillance, les images spatiales, etc. Cette étude conclut que l'utilisation d'une variété de filtres de reconnaissance peut être un moyen efficace d'améliorer la capacité des modèles neuronaux à reconnaître les images avec plus de précision, et constitue un domaine qui mérite des recherches plus approfondies et des développements plus approfondis à l'avenir.

Conclusion générale

À la fin de cette recherche, nous pouvons souligner l'importance du processus de préparation des données, et de l'optimisation des modèles utilisant des réseaux de neurones convolutifs (CNN) et des filtres pour améliorer les images afin d'obtenir des résultats fiables dans le domaine de l'apprentissage automatique. La préparation des données est une étape essentielle et vitale pour garantir la qualité et l'exactitude des résultats, car elle nécessite de nettoyer et de traiter les données de manière appropriée avant de les utiliser pour entraîner des modèles. Les réseaux de neurones convolutifs (CNN), se sont révélés efficaces pour traiter les données d'image et en extraire des caractéristiques, ce qui en fait un outil puissant de classification d'images et de reconnaissance de formes, CNN représente un changement de paradigme dans la façon dont les images sont traitées grâce à sa capacité à apprendre automatiquement des caractéristiques à partir de données brutes. L'utilisation de filtres pour améliorer les images avant de les intégrer au modèle s'est avérée efficace pour améliorer la précision et les performances du modèle. Des filtres tels que Flou gaussien, Sobel, Laplacien et Canny Edge Detection, entre autres, contribuent à améliorer la qualité des images et à mettre en évidence des caractéristiques importantes, ce qui contribue à améliorer la capacité du modèle à apprendre et à reconnaître plus efficacement les modèles.

En fin de compte, on peut dire que l'intégration d'une bonne préparation des données, l'utilisation de réseaux de neurones et l'application de techniques d'amélioration d'image représentent un cadre intégré pour créer des modèles d'apprentissage automatique efficaces et précis. Ces pratiques sont non seulement essentielles pour obtenir des résultats fiables, mais ouvrent également de nouveaux horizons pour la recherche et le développement dans le domaine de l'apprentissage profond et de l'analyse des données d'images.

Bibliography

- [1] T. G. Mesevage, 24 mai 2021. Available: <https://monkeylearn.com/blog/data-preprocessing/>.
- [2] F. Khan, «astra,» 10 mai 2024 Available: <https://www.astera.com/fr/type/blog/data-preprocessing/>.
- [3] D. Vernon, «Data Preprocessing in Machine Learning: A Beginner's Guide,» *simplelearn*, p. 10, 28 septembre 2023.
- [4] G. Lawton, 1 janvier 2022. Available: <https://www.techtarget.com>.
- [5] «What is Machine Learning?,» *.geeksforgeeks.*, p. 1, 26 mai 2024.
- [6] "machine-learning," *IBM*, p. 1, 2024.
- [7] R. Team, «Avantages et inconvénients de l'apprentissage automatique,» *Avantages De L'apprentissage Automatique*, p. 1, 3 avril 2023.
- [8] G. Lawton, «Apprentissage supervisé et non supervisé : les différencier et les combiner,» p. 5, 14 octobre 2020.
- [9] J. Schäfer, «konfuzio,» 31 janvier 2024. Available: <https://konfuzio.com/fr/algorithmes-dapprentissage-automatique/>.
- [10] Mr.Mint, "machine learning," 2024. Available: mrmint.fr/introduction-k-nearest-neighbors. [Accessed 1 mai 2024].
- [11] «1,» 2024. Available: <https://aws.amazon.com/what-is/deep-learning/>.
- [12] T. KELDENICH, "inside machine learning," 11 JUILLET 2021. Available: [inside-machinelearning](https://inside-machinelearning.com)
- [13] "towardsdatascience. Available: <https://towardsdatascience.com/regularization-in-deep-learning->.
- [14] R. Awati, "resolution, Available: [techtarget](https://www.techtarget.com). [Accessed 2024].
- [15] B. Dickson, "bdtechtalks," What are convolutional neural networks (CNN)?, 6 janvier 2020. Available: <https://bdtechtalks.com/2020/01/06/convolutional-neural-networks-cnn-convnets/>.
- [16] "Padding (Machine Learning)," Available: <https://deeptai.org/machine-learning-glossary-and-terms/padding>.
- [17] "MACHINE VISION," 28 2 2013 Available: <https://forums.ni.com/t5/Machine-Vision/gaussian-filter/td-p/2441104>.
- [18] d. pattel, "How can you apply sharpening filters to enhance the edges and details of digital images?".
- [19] c. d. i. a. l. r. d. neurones., "dSPACE.univ-tlemcen.dz,". Available: <http://dSPACE.univ-tlemcen.dz/bitstream/112/12235/1/Classification-des-images-avec-les-reseaux-de-neurones.pdf>.
- [20] T. S. Huang, G. J. Yang and G. Y. Tang, ""A fast two-dimensional median filtering algorithm", " IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, Available: <https://www.uio.no/studier/emner/matnat/ifi/INF2310/v12/undervisningsmateriale/artikler/Huang-et-al-median>. [Accessed 2024].
- [21] "Morphologie Mathématique,". Available: perso.esiee.fr/~perretb/I5FM/TAI/morpho. [Accessed 2024].
- [22] "Égalisation d'histogramme adaptatif limité par contraste (CLAHE)," Available: <https://siril.readthedocs.io/fr/latest/processing/clahe>.
- [23] A. Gramaglia, "Sharpening filters," 14 7 2018. Available: <https://www.albertogramaglia.com/sharpening-filters-for-image-processing>
- [24] "Morphologie Mathématique,". Available: perso.esiee.fr/~perretb/I5FM/TAI/morpho. [Accessed 2024].
- [25] "Wand image.emboss in Python," *geeksforgeeks*, Available: [geeksforgeeks](https://www.geeksforgeeks.com/wand-image-emboss-in-python/). [Accessed 2024].
- [26] d. pattel, "How can you apply sharpening filters to enhance the edges and details of digital images?".
- [27] P. Monasse, "Découvrez les différentes couches d'un CNN," *openclassrooms*, [Online]. Available: <https://openclassrooms.com/fr/courses/4470531-classez-et-segmentez-des-donnees-visuelles/5083336-decouvrez-les-differentes-couches-dun-cnn>. [Accessed 2024].
- [28] A. A. e. S. Amidi, "Apprentissage profond,". Available: <https://stanford.edu/~shervine/l/fr/teaching/cs-230/pense-bete-reseaux-neurones-convolutionnels>.
- [29] B. Maurice, "deeplylearning," 22 janvier 2021. Available: <https://deeplylearning.fr/cours-theoriques/>.
- [30] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "*Gradient-based learning applied to document recognition*", Proceedings of the IEEE, 2020.
- [31] Christian Szegedy, *GoogLeNet*, paperswithcode, 2021.
- [32] A. Krizhevsky, I. Sutskever and G. E. Hinton, "*ImageNet classification with deep convolutional neural networks*", england: Communications of the ACM, 2019.
- [33] "python, Available: <https://www.python.com/>. [Accessed 2024].
- [34] "tensorflow," Available: <https://www.tensorflow.org/>. [Accessed 2024].

- [35] "keras," Available: <https://keras.io/>. [Accessed mai 2024].
- [36] "google colab," Available: <https://colab.research.google.com/>. [Accessed 2024].