

Réseaux de neurones convolutifs pour le diagnostic médical

Le choix de ce sujet a été motivé par le désir de comprendre le fonctionnement de cette technologie qui est au cœur de notre société, de démystifier l'intelligence artificielle, qui est souvent source de peurs et d'aprioris, alors qu'elle peut être un atout pour notre santé.

L'un des défis majeurs en médecine est le diagnostic précoce des maladies cardiovasculaires, du cancer ou des pneumonies. En aidant les professionnels de santé à la décision, les réseaux de neurones et les réseaux de neurones convolutifs sont des atouts majeurs face à ce défi de santé publique.

Positionnement thématique (ETAPE 1)

INFORMATIQUE (Informatique pratique), MATHÉMATIQUES (Mathématiques Appliquées).

Mots-clés (ETAPE 1)

Mots-Clés (en français)	Mots-Clés (en anglais)
<i>Réseau de neurone convolutif</i>	<i>Convolutional Neural Network (CNN)</i>
<i>Apprentissage supervisé</i>	<i>Supervised learning</i>
<i>Algorithme de rétropropagation du gradient</i>	<i>Backpropagation algorithm</i>
<i>Perceptron multicouche</i>	<i>Multilayer perceptron</i>
<i>Diagnostic médical</i>	<i>Medical diagnosis</i>

Bibliographie commentée

Dès 1950, Alan Turing se pose la question de savoir si les ordinateurs ont la capacité d'être doués d'intelligence humaine grâce au test de Turing. Jusqu'aux années 70, de nombreux algorithmes sont inventés en ce sens, à visée essentiellement industrielle. En 1976 puis en 1986, des machines d'aide à la décision à destination des médecins sont conçus mais le manque de données et de puissance de calcul fait que l'usage de ces technologies reste limité. Il faut attendre les années 2010 pour que, grâce aux réseaux de neurones convolutifs (CNN) et à l'ère de la Big Data, des algorithmes réussissent à poser des diagnostics médicaux, à prédire des risques d'accidents cardiovasculaires ou encore à évaluer l'effet d'un traitement sur l'évolution de la maladie d'Alzheimer. [1]

L'apprentissage supervisé est un domaine de l'intelligence artificielle visant à créer un algorithme qui apprend une fonction prédictive (comme classier des images de tumeurs cancéreuses) à partir d'exemples annotés. Pour ce faire, les réseaux de neurones sont un atout majeur. [2]

La composante principale d'un réseau de neurone est le neurone formel. Par analogie avec le neurone biologique, c'est une fonction caractérisée par un poids et un biais, et qui à un vecteur associe une combinaison affine de ses composantes réelles. Les problèmes étudiés étant rarement linéairement séparable comme le XOR (ou exclusif), on applique en sortie du neurone formel une

fonction non linéaire dite d'activation. Il en existe plusieurs dont l'efficacité dépend du problème.

[2]

Le réseau de neurone le plus utilisé est le perceptron multicouche. Il est composé de couches successives de neurones formels. Les neurones d'une même couche ne sont pas connectés entre eux. Chaque neurone d'une couche donnée prend en entrée le vecteur sortie de la couche précédente et est connecté en sortie à tous les neurones de la couche suivante. La couche d'entrée lit le signal fourni au système, et la couche de sortie donne la réponse du réseau. Chaque couche étant fonction de la précédente, la propagation directe est la fonction qui calcule de proche en proche la sortie du réseau en fonction de son entrée : c'est ce qui permet au réseau de faire une prédiction à partir de données. **[2] [3]**

L'objectif est donc de trouver des valeurs de poids internes et de biais tel que la sortie calculée soit la plus proche possible de la sortie désirée. Pour ce faire, on commence par générer aléatoirement des poids et biais initiaux. Après une propagation directe, on calcule l'erreur (par exemple quadratique) de la sortie obtenue en fonction de celle désirée, puis grâce à l'algorithme de rétropropagation, on calcule de proche en proche (de droite à gauche) le gradient des différentes couches. Pour minimiser l'erreur, on choisit un pas (ou taux d'apprentissage) et on met à jour les poids et biais dans le sens de minimisation de l'erreur. On répète ce processus un grand nombre de fois. **[2] [3]**

Néanmoins, un tel algorithme est peu efficace avec une image en entrée, car tous les neurones sont connectés entre eux entre couches adjacentes. Inspiré du fonctionnement du cortex visuel des vertébrés, les CNN limitent ces connexions. Les neurones formels deviennent des filtres, qui calculent la sortie à l'aide de l'opération de convolution en fonction d'hyperparamètres. On ajoute d'ailleurs des couches pooling qui réduisent la taille des images en gardant leurs caractéristiques importantes. Elles améliorent l'efficacité du réseau et permettent d'éviter le surapprentissage. Fonction d'activation et d'erreur, poids et biais initiaux, structure du réseau, nombre d'itérations, pas et hyperparamètres sont autant de paramètres à choisir expérimentalement pour augmenter l'efficacité du réseau. **[4] [5]**

Afin de diagnostiquer rapidement le cancer de la peau et ainsi fortement augmenter les chances de guérison, on utilise des CNN. L'entraînement de tels réseaux ont souvent permis d'avoir un taux de succès de diagnostic supérieur à celui de dermatologues. **[6]**. La base de données HAM10000 propose les images nécessaires à cette étude. **[7]**

Problématique retenue

Nous étudierons la manière de construire un perceptron multicouche et un CNN sur Python à partir de zéro. Nous expérimenterons l'effet du changement de paramètres du réseau sur son apprentissage et interpréterons les résultats associés. Nous analyserons enfin comment un CNN permet d'établir un diagnostic médical et interpréterons son efficacité.

Objectifs du TIPE

1. Comprendre la structure d'une base de données. Explorer et visualiser la donnée pour la nettoyer.
2. Construire en programmation orientée objet sur python et avec numpy une classe perceptron multicouche et un réseau de neurone convolutif dont les caractéristiques sont personnalisables.
3. Utiliser ces réseaux de neurones sur des problèmes afin de mettre en évidence leurs propriétés.
4. Entraîner un réseau de neurone convolutif à classifier des images de lésions cutanées afin de prédire un diagnostic. Modifier les paramètres du réseau afin de minimiser son erreur. Représenter graphiquement les résultats et les interpréter.

Références bibliographiques (ETAPE 1)

- [1] KAUL, VIVEK, ENSLIN, SARAH, ET GROSS, SETH A : History of artificial intelligence in medicine. *Gastrointestinal endoscopy* : 2020, vol. 92, no 4, p. 807-812.
<https://www.sciencedirect.com/science/article/pii/S0016510720344667>
- [2] WIKISTAT : Réseaux de neurones : <https://www.math.univ-toulouse.fr/~besse/Wikistat/pdf/st-m-app-rn.pdf>
- [3] JULIE DELON : Introduction aux réseaux de neurones et à l'apprentissage profond : https://helios2.mi.parisdescartes.fr/~ebirmele/depots/Enseignements/GrandeDimension/neural_networks.pdf
- [4] STANFORD UNIVERSITY : CS231n Convolutional Neural Networks for Visual Recognition : <https://cs231n.github.io>
- [5] SAMUEL BÉDARD-VENNE : Réseau de neurones artificiels et applications à la classification d'images : <https://archipel.uqam.ca/11700/1/M15636.pdf>
- [6] DAS, KINNOR, COCKERELL, CLAY J., PATIL, ANANT, ET AL : Machine Learning and Its Application in Skin Cancer : 2021, vol. 18, no 24, p. 13409. <https://www.mdpi.com/1660-4601/18/24/13409>
- [7] HARVARD DATAVERSE : The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions : <https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/DBW86T>

DOT

- [1] [Février-Mars 2021] *Etude de la théorie des réseaux de neurones et du perceptron multicouche*
- [2] [Avril-Juin 2021] *Implémentation d'un réseau de neurone sous Python, et compréhension de son fonctionnement à l'aide de problèmes simples (comme XOR)*
- [3] [Septembre-Octobre 2021] *Etude et implémentation d'un réseau de neurone convolutif sous Python*
- [4] [Novembre 2021] *Résolution du MNIST (lecture d'images de chiffres par un réseau de neurone convolutif)*
- [5] [Décembre-Janvier 2021] *Exploration et traitement des données HAM10000, mais résolution*

trop lente avec le réseau implémenté

[6] *[Février 2022] Décision de l'utilisation de Keras (module Python implémentant des réseaux de neurones rapides) avec les mêmes couches et fonctions d'erreurs codées auparavant pour une résolution plus rapide*

[7] *[Février - Mars 2022] Aboutissement à des résultats insatisfaisants pour le problème du HAM10000 avec des précisions très grandes pour certaines classes, très petites (ou nulles) pour d'autres*

[8] *[Avril-Mai 2022] Découverte et prise en compte du déséquilibre de classe, puis résolution du HAM10000. Comparaison de différentes architectures de réseau de neurones convolutif et choix de celle avec la meilleure efficacité.*