

intro aux algorithmes de l'IA

Quels algorithmes implémentent chaque outil d'IA ?

Stuart Cording (Allemagne)

Explorons les principes et les algorithmes essentiels de l'intelligence artificielle (IA) contemporaine, y compris les réseaux de neurones, l'apprentissage automatique et l'apprentissage profond.

Cette exploration dévoile les vastes capacités de l'IA, qui s'étendent bien au-delà des applications de modèles génératifs tels que ChatGPT.

L'IA est constamment sous les feux de l'actualité. Parfois, c'est pour mettre en lumière les innovations dans le secteur de la santé, où elle permet un diagnostic précoce de maladies graves telles que le cancer, augmentant ainsi les chances de guérison. Cependant, les informations relayées concernent le plus souvent ses aspects négatifs, comme la génération de fausses images (**figure 1**) et vidéos ou les questions éthiques liées à l'utilisation de matériel protégé par des droits d'auteur pour entraîner de nouveaux modèles d'IA. Par ailleurs, les médias grand public abordent rarement les détails techniques tels que les algorithmes qui soutiennent ces outils, le processus de préparation ou encore la manière dont ils sont déployés dans des applications spécialisées, des sujets cruciaux pour les ingénieurs et les technophiles.

IA 101

Lorsque l'on aborde le sujet de l'IA aujourd'hui, plusieurs termes apparaissent rapidement et méritent d'être expliqués : réseaux neuronaux, apprentissage automatique, apprentissage profond et le terme IA lui-même. L'IA est une discipline scientifique qui étudie la manière dont les ordinateurs et les logiciels peuvent être utilisés pour résoudre des problèmes complexes.

Mais, il s'agit également d'un ensemble de technologies, incluant l'analyse des données, le traitement du langage naturel ainsi que des méthodes de prévision et de prédiction.

Elle peut être subdivisée en deux catégories : l'intelligence étroite et l'intelligence générale. L'IA à intelligence étroite est conçue pour réaliser des performances équivalentes à celles de l'homme dans

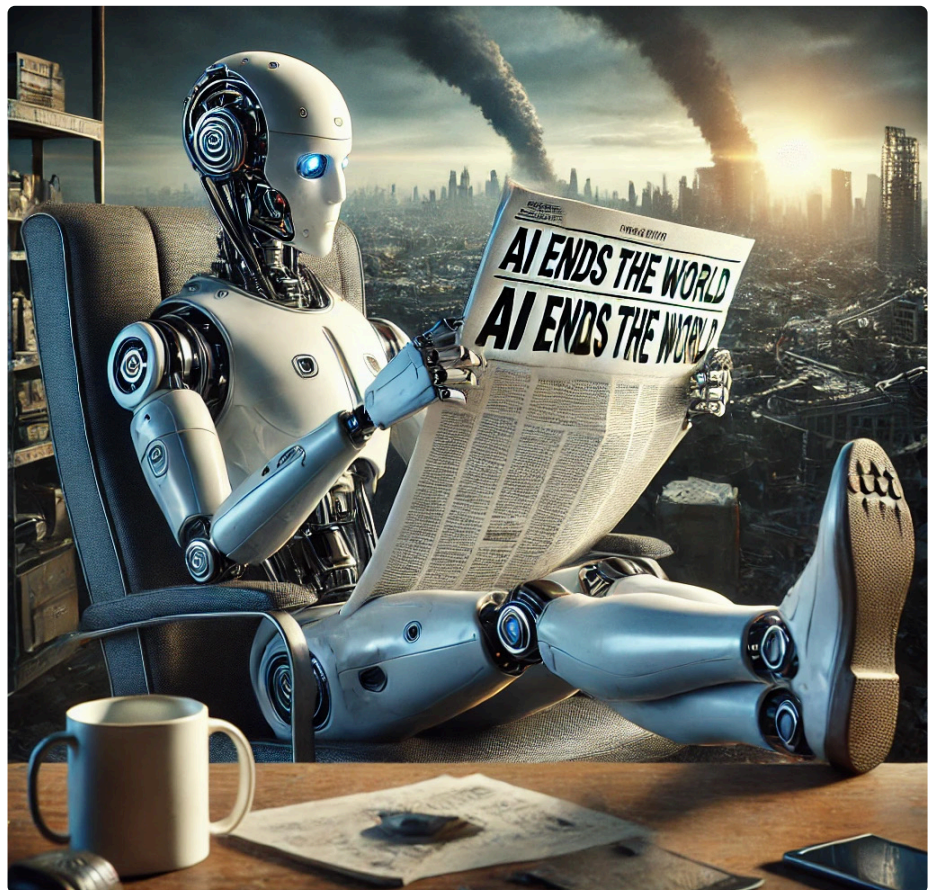


Figure 1. La fin du monde imaginée par l'IA générative. Mais ne vous inquiétez pas, les deux jambes du robot partagent un seul pied, elles n'iront donc pas loin !

un domaine spécifique, comme les jeux d'échecs ou de go. De nombreuses implémentations de l'IA peuvent être considérées comme égales ou supérieures à celles des humains, comme celles utilisées pour classer des objets dans des images. Cependant, bien que ces outils d'IA soient exceptionnellement intelligents et capables, on ne les confondrait pas avec un être humain. L'IA générale est souvent considérée comme une panacée. Les performances d'une telle IA seraient indiscernables de celles d'un être humain, imitant sa compréhension du monde qui l'entoure et ses réactions émotionnelles. Toutefois, la pertinence de développer une telle technologie est un sujet de débat sociétal. Il convient de noter qu'en théorie, il serait possible de créer une IA superintelligente surpassant les capacités cognitives humaines. Bien que cela soit concevable, pour l'instant, ce sujet est plus approprié aux discussions philosophiques et aux récits de science-fiction. Pour comprendre la mise en œuvre sous-jacente de toutes les technologies d'IA, nous devons explorer le niveau le plus bas - les réseaux neuronaux. Le perceptron multicouche (MLP), modelé sur les neurones biologiques, utilise les mathématiques et un processus appelé rétropropagation pour régler progressivement le neurone afin qu'il reconnaisse des modèles [1]. L'élément essentiel est l'utilisation d'une couche de cellules cachées entre l'entrée et la sortie, qui rend possible le processus de réglage (**figure 2**). En conséquence, le neurone peut classer des entrées qu'il serait difficile, voire impossible, de mettre

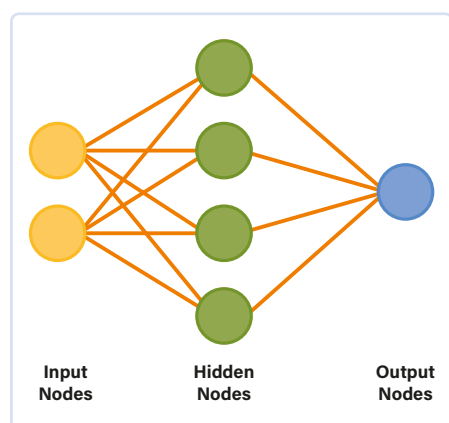


Figure 2. Le perceptron multicouche se compose de nœuds d'entrée, de nœuds de sortie qui fournissent des résultats de classification, et de nœuds cachés qui apprennent la classification souhaitée grâce à l'entraînement.

Traffic Lights in RGB Color Space

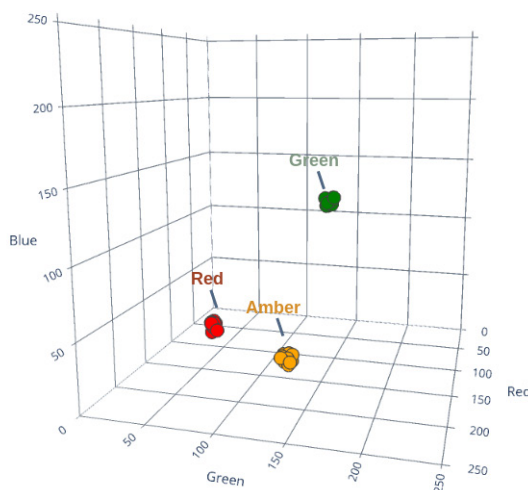


Figure 3. Les grappes de valeurs RGB peuvent être interprétées, ou classées, comme des couleurs.

en œuvre à l'aide de simples instructions if-else dans un langage de programmation. Un bon exemple est la classification des couleurs capturées par un capteur RGB. L'orange est considéré comme étant composé à 100 % de rouge et à 65 % de vert [2]. Le violet est composé de 93 % de rouge, 51 % de vert et 93 % de bleu. Cependant, lorsqu'un capteur RGB est placé devant un échantillon de couleur, il ne fournit pas toujours des résultats parfaits et précis en termes de pourcentage et de valeur correspondant exactement à ces couleurs. Au lieu de cela, les trois valeurs mesurées tendent à fluctuer autour de la valeur idéale. L'éclairage complique encore la détection, influençant les mesures qui diminuent sous un faible éclairage et augmentent quand l'éclairage est plus intense. Les réseaux neuronaux peuvent être entraînés pour s'adapter à des variations raisonnables et à des situations d'éclairage, en regroupant des valeurs (**figure 3**) approchées pour classer les couleurs [3].

Apprentissage automatique versus apprentissage profond

La plupart des outils d'IA actuels sont en fait classés dans la catégorie de l'apprentissage automatique et constituent donc un sous-ensemble de l'IA. Ils sont généralement entraînés sur des données structurées. Par exemple, les images sont étiquetées de manière à identifier clairement leur contenu, que ce soit en nommant l'objet présent (concombre, pizza, pain) ou en fournissant un contexte supplémentaire, tel que le stade de croissance d'un légume, l'emplacement d'un plat cuisiné ou le nombre de calories qu'il contient. Il est également important de noter que

ces algorithmes, comme les humains, ne sont pas parfaits et ne fournissent pas de solutions absolues aux problèmes qu'ils abordent. Ils évaluent plutôt la probabilité que l'entrée corresponde à quelque chose qu'ils ont été entraînés à reconnaître. Par exemple d'un algorithme de ML formé aux images d'animaux pourrait être très sûr (plus de 90 %) qu'une photo représente une vache, mais pourrait également indiquer une certaine probabilité (65 %) qu'il s'agisse d'un panda et une plus faible encore (35 %) qu'il s'agisse d'un blaireau. Si vous lui présentez l'image d'un pingouin et si cet animal n'a jamais été inclus dans son apprentissage, l'algorithme ne le reconnaîtra pas comme tel.

La plupart des applications d'IA embarquées ou de pointe peuvent être considérées comme des implémentations de ML. Un bon exemple est le Izoelektro RAM-1 [4]. Ce dispositif surveille les parafoudres pour la surveillance du réseau électrique [5]. Il utilise un nRF9160 de Nordic Semiconductor basé sur Arm Cortex-M33, signalant le courant de fuite, la température excessive, l'inclinaison des poteaux/tours, et d'autres

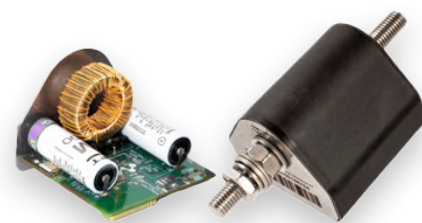


Figure 4. Le RAM-1 utilise l'apprentissage automatique pour détecter les surtensions dans les réseaux électriques. Grâce à l'IdO, le modèle de détection peut être mis à niveau. (Source : Izoelektro)

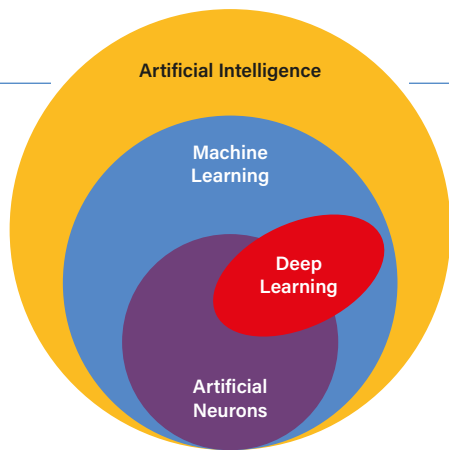


Figure 5. La hiérarchie qui va de l'IA aux neurones artificiels. L'apprentissage profond utilise des réseaux avec de nombreuses couches pour mettre en œuvre les modèles d'IA les plus avancés d'aujourd'hui, tels que le traitement du langage naturel.

événements de manipulation (par exemple, les coups de foudre). Le développement initial de l'algorithme a été réalisé avec la plateforme Edge Impulse [6], et il continue d'évoluer en intégrant des données provenant de nouveaux événements anormaux détectés par le réseau de dispositifs installés (figure 4).

L'apprentissage profond (Deep Learning, DL) va encore plus loin. Si l'on se réfère à notre MLP, une seule couche de poids entre l'entrée et la sortie suffit pour un réglage initial. Toutefois, les tâches plus complexes nécessitent un réglage plus précis, d'où la nécessité d'ajouter des couches cachées. Dès lors que l'on utilise au moins deux couches cachées, on parle d'apprentissage profond (figure 5).

Approches de l'apprentissage

Les données étiquetées sont les plus faciles à traiter, mais elles posent un problème important : qui se chargera de l'étiquetage ? Des plateformes telles qu'Amazon Mechanical Turk permettent aux utilisateurs de traiter de grands ensembles de données [7] ou même de tester les performances des modèles d'IA existants. Le coût de chaque opération d'étiquetage est minime, ce qui rend l'étiquetage économiquement accessible pour de nombreuses entreprises. Toutefois, si les données à traiter sont de nature propriétaire ou si elles comportent des informations confidentielles, il faudra envisager une alternative appropriée.

Les données étiquetées peuvent être utilisées pour effectuer un apprentissage supervisé. Le modèle d'IA reçoit des données d'entrée (texte, son, image) et des données de sortie attendues, et à force de répéti-

tion, le modèle s'améliore. L'apprentissage est considéré comme complet lorsque le modèle atteint un niveau de précision prédéfini. Comme c'est le cas pour les humains, ce niveau ne peut atteindre la perfection.

Un modèle d'IA qui obtient toutes les réponses correctes lors de l'analyse des données d'apprentissage est considéré comme souffrant d'un surajustement. Par exemple, si le modèle est censé reconnaître des feux de circulation sur des images, il peut commencer à déduire l'existence d'un feu de circulation sur la base d'anomalies dans les images d'apprentissage, telles que la lumière du soleil se reflétant sur un environnement en plastique ou la présence d'un portique. Lorsque le modèle d'IA est ensuite exposé à des images ne faisant pas partie de l'ensemble d'apprentissage, la simple présence d'un portique ou d'un point lumineux dans l'image peut faire en sorte que l'image soit étiquetée comme contenant un feu de circulation, même si aucun feu n'est visible. Par conséquent, les développeurs d'algorithmes d'IA doivent

déterminer le juste milieu entre l'ajustement excessif et l'ajustement insuffisant (figure 6).

Pour la plupart des entreprises, la réalité est qu'elles disposent de beaucoup de données, mais qu'elles ne sont pas étiquetées. Par exemple, un véhicule d'essai peut avoir collecté des masses de données brutes de capteurs lors de la conduite d'un véhicule prototype. La feuille de calcul Excel (oui, c'est encore un format de données courant !) contiendra les noms des capteurs et leurs unités de mesure, mais sans aucun contexte, comme le fait que le véhicule avait cinq passagers adultes ou que la route était mouillée.

L'apprentissage non supervisé permet de résoudre ce problème. Bien que la tâche soit, de toute évidence, plus difficile et plus exigeante en termes de traitement, il est possible de réduire la dimensionnalité des données et de trouver des modèles et des grappes. Ainsi, pour revenir à l'exemple précédent, l'IA peut classer les virages, l'accélération et la décélération et même proposer d'autres modèles, tels que

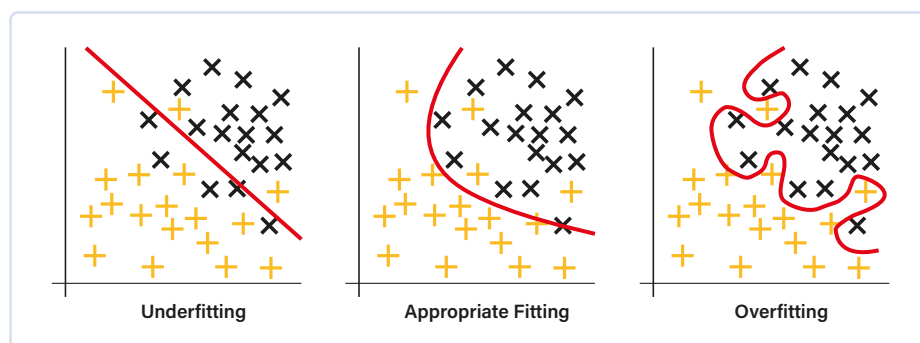


Figure 6. Entraînez trop longtemps, et le modèle d'IA peut être entravé par un surajustement ; trop peu d'entraînement conduit à une mauvaise classification des données appliquées aux entrées.

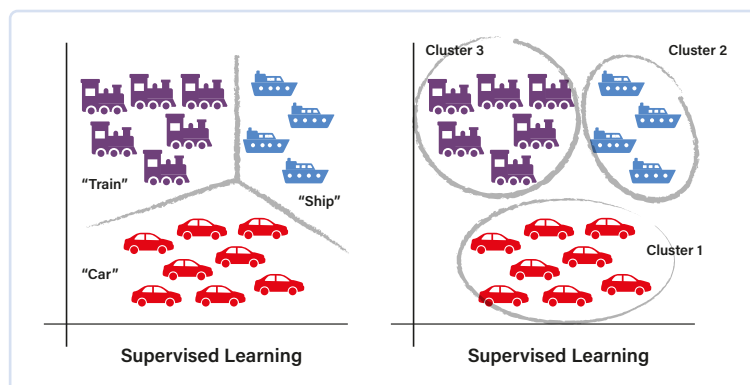


Figure 7.

la présence de pluie ou de neige pendant l'essai routier. Bien entendu, l'IA ne peut pas suggérer les termes « pluie » et « neige ». Au lieu de cela, les experts qui analysent les résultats verront que des groupes spécifiques s'alignent sur des circonstances de test spécifiques et les étiqueteront en conséquence (**figure 7**).

Les deux méthodes ci-dessus peuvent être combinées, en utilisant des données étiquetées et des données non étiquetées dans une approche d'apprentissage semi-supervisé. Les résultats de la formation peuvent également être réintégrés dans les données de formation. Toutefois, une surveillance et une intervention humaines sont nécessaires, car ces résultats peuvent contenir des erreurs.

L'apprentissage par renforcement est une autre approche. Utilisé dans des environnements simulés, un modèle reçoit des observations concernant son environnement. L'IA choisit des actions possibles et reçoit une récompense si le modèle se rapproche de l'objectif souhaité. Après de nombreuses itérations, l'IA finit par apprendre la tâche souhaitée. L'apprentissage par renforcement a été utilisé dans le traitement du langage naturel et dans les IA conçues pour jouer à des jeux. Google DeepMind a utilisé cette technique pour réduire le nombre d'instructions d'assembleur nécessaires à la mise en œuvre d'un algorithme de tri (**figure 8**). Une récompense a été attribuée au code fonctionnant correctement qui utilisait également moins d'instructions d'assembleur que l'algorithme original [8].

Les algorithmes d'IA

Les modèles de base de l'IA se divisent en deux grands types. Les premiers sont des modèles de classification. Ceux-ci décomposent les données d'entrée en une réponse probable/improbable, répondant à une question telle que « Cette image contient-elle un canard ? » ou « Cette couleur est-elle jaune ? » Ensuite, il y a les modèles de régression, qui génèrent une réponse numérique à des questions telles que « Combien d'Arduino Uno allons-nous vendre le mois prochain ? » Des modèles plus complexes permettent la création d'outils tels que ChatGPT, capable de dialoguer avec l'utilisateur, ou DALL-E, qui peut générer des images à partir de descriptions textuelles. Ces modèles reposent sur une variété d'algorithmes qui, parfois

Original Sort Code	Improved Sort Code	
Memory[0] = A	Memory[0] = A	Collect three values for sorting. Save to registers.
Memory[1] = B	Memory[1] = B	
Memory[2] = C	Memory[2] = C	
mov Memory[0] P	mov Memory[0] P	Sort values.
mov Memory[1] Q	mov Memory[1] Q	
mov Memory[2] R	mov Memory[2] R	
mov R S	mov R S	
cmp P R	cmp P R	
cmovg P R	cmovg P R	
cmovl P S	cmovl P S	Return the result to memory.
mov S P	mov S P	
cmp S Q	cmp S Q	
cmovg Q P	cmovg Q P	
cmovg S Q	cmovg S Q	
mov P Memory[0]	mov P Memory[0]	
mov Q Memory[1]	mov Q Memory[1]	
mov R Memory[2]	mov R Memory[2]	

Figure 8. L'IA a économisé une seule instruction d'assembleur (mov S P) dans cet algorithme de tri sans impact sur le fonctionnement.

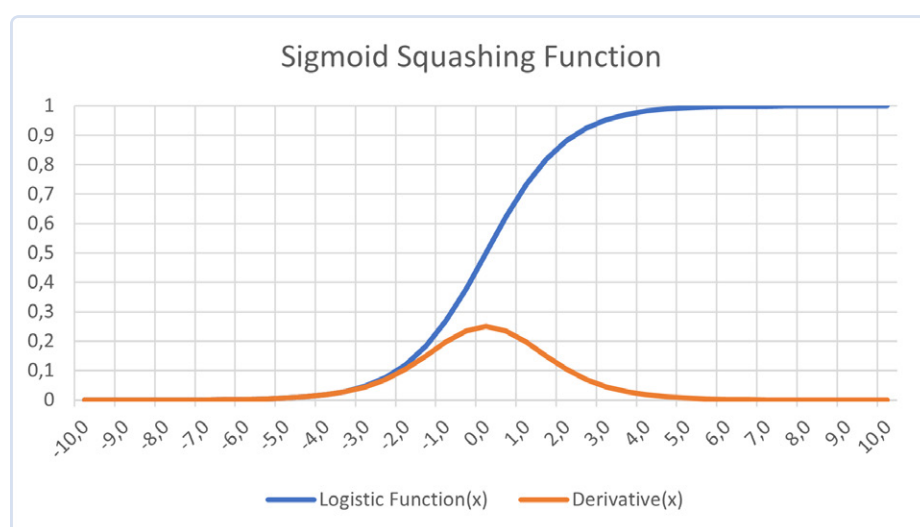


Figure 9. La fonction sigmoïde aide à la classification en poussant rapidement les valeurs d'entrée vers un résultat « probable » ou « improbable ».

combinés, répondent spécifiquement aux exigences de la tâche.

Régression logistique

Les premières expériences avec les neurones artificiels ont rencontré un obstacle notable. Ils étaient performants pour classer les informations pouvant être séparées en deux groupes distincts à l'aide d'une ligne droite, grâce à l'utilisation de fonctions linéaires, exprimées mathématiquement par $f(x) = ax + b$ si Cependant, ce type de fonctionnalité présente des limites lorsque les données à classer ne sont pas linéairement séparables, par exemple quand elles se regroupent de manière circulaire, situation où ces modèles linéaires échouent.

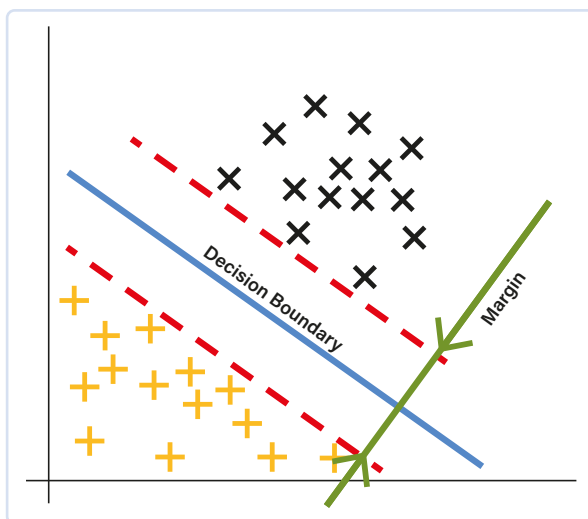
La régression logistique utilise des fonctions qui rapprochent rapidement les valeurs d'entrée de 0 ou de 1. Un exemple est la fonction sigmoïde, $s(z) = 1 \div (1 + \exp$

$(-z))$, qui renvoie 0,5 pour une entrée de 0,0, proche de 1 pour une entrée de 10, et proche de 0 pour une entrée de -10,0 (**figure 9**). Grâce à ces propriétés mathématiques, les réseaux de neurones peuvent, par de nombreuses répétitions, être entraînés à des tâches de classification relativement simples, comme celle de savoir si un capteur RGB se trouve devant la couleur violette (probable/improbable).

Machines à vecteurs de support (SVM)

Si vous jetez sur votre tapis des M&M rouges suivis de M&M bleus, vous constateriez probablement que vous pouvez placer une règle entre les deux groupes de couleurs. La règle sépare les deux groupes et son angle sur le tapis peut être représenté par un vecteur (une ligne avec une magnitude et une direction). C'est ainsi que fonctionnent, à la base, les machines

Figure 10. Avec les SVM, les données sont classées de part et d'autre d'une limite de décision, définie comme un vecteur, avec une marge (distance entre les groupes opposés les plus proches).



à vecteurs de support. Toutefois, dans un espace à deux dimensions, cette règle imaginaire est aussi large que possible pour maximiser la marge (distance) entre les deux groupes lorsque le vecteur est calculé (figure 10). Une fois déployé dans un modèle d'IA, nous pouvons désormais classer les M&Ms rouges et bleus.

À ce niveau superficiel, ce n'est pas différent d'une fonction linéaire. Les SVM deviennent intéressants lorsque les données comportent davantage de dimensions. Par exemple, supposons que vos données puissent être représentées sur les axes X, Y et Z. Dans ce cas, vous n'êtes pas obligé d'utiliser une fonction linéaire. Dans ce cas, vous ne cherchez plus une ligne pour séparer les classes, mais un hyperplan (un plan bidimensionnel dans un espace tridimensionnel). Ils conviennent également aux cas où certains de vos M&M rouges et bleus ont dérivé vers le groupe opposé. Les SVM nécessitent plus de calculs que la régression logistique, mais sont plus performants sur les données comportant de nombreuses dimensions et sur les ensembles de données non structurés.

Arbres de décision et forêts aléatoires

Les arbres de décision sont, à un niveau simple, similaires à la création d'organigrammes. Ils utilisent une stratégie de division et de conquête pour classer un ensemble de données en autant de groupes que nécessaire. Lorsqu'un utilisateur applique l'algorithme d'IA à sa tâche, il parcourt l'organigramme pour fournir une réponse. Appliqué à des images, il pourrait commencer par diviser l'ensemble de données en arrière-plans, puis en couleurs dominantes, et enfin en caractéristiques d'objet (oreilles, pattes, rayures).

La décision de diviser l'ensemble de

données à chaque nœud dépend de l'entropie des échantillons examinés. Par exemple, si de nombreuses images contiennent du rouge et que quelques-unes sont bleues, l'entropie de « contient du rouge » est élevée. L'ajout d'un nœud « L'image contient-elle du rouge ? » permet au modèle d'IA de traiter rapidement et efficacement les images rouges. Si l'ensemble de données est partagé à 50/50 sur la question rouge/bleu, il recherchera une classification alternative avec une entropie plus élevée.

Les arbres de décision posent un problème d'ajustement excessif et de biais. L'ajustement excessif est combattu en élaguant les nœuds sur les caractéristiques de faible importance au cours de l'apprentissage. Le surajustement et le biais peuvent être traités à l'aide de l'algorithme de la forêt aléatoire qui utilise en outre le bagging, pour réduire la variance de l'ensemble des données, et le caractère aléatoire.

Réseaux neuronaux convolutifs (CNN)

Pour les applications de vision artificielle et de reconnaissance d'objets, les CNN à apprentissage profond sont utilisés. Ces algorithmes multicouches commencent par utiliser la convolution pour trouver des caractéristiques (lignes droites, courbes, couleurs) dans l'image. Viennent ensuite des couches de mise en commun qui réduisent essentiellement l'échantillonnage des résultats de la convolution. Le réseau profond comprend généralement de nombreuses couches de convolution et de mise en commun, autant que nécessaire pour obtenir la classification souhaitée. La dernière couche est appelée couche entièrement connectée. C'est à ce stade que l'algorithme renvoie une classification.

L'un des avantages des CNN est qu'ils fonctionnent bien même si la position de

l'objet recherché se déplace par rapport aux exemples de l'objet dans les données d'apprentissage.

Réseaux neuronaux récurrents (RNN) et transformateurs

Les images sont des moments statiques dans le temps, mais certaines applications nécessitent de comprendre ce qui s'est passé avant, comme dans le cas de la parole et de l'écriture. Par exemple, si j'écris « John a construit un amplificateur. Il l'a fait fonctionner du premier coup », nous savons que « il » fait référence à John et que « il » fait référence à l'amplificateur. Les RNN sont beaucoup plus performants pour traiter ce type de tâches temporelles, notamment le traitement du langage naturel, la traduction et la reconnaissance de l'écriture manuscrite. La plupart des réseaux neuronaux transmettent simplement toutes leurs données à l'étape suivante au fur et à mesure qu'ils traitent les entrées. Les RNN s'écartent de cette approche en mettant en œuvre une « mémoire » dans certaines couches cachées, en conservant des informations pour des décisions futures, et même en anticipant les données futures, telles que les mots au-delà de ceux qui sont actuellement traités, pour prendre leurs décisions. Les transformateurs améliorent les RNN et ont rendu possibles de grands modèles linguistiques tels que ChatGPT. Ils utilisent des mécanismes d'attention plus performants lors de l'analyse du langage, en particulier lorsque des relations doivent être trouvées entre les mots de différentes phrases plutôt qu'entre les mots d'une même phrase.

Il n'y a pas que ChatGPT et Midjourney

L'IA a permis des avancées significatives, modifiant la manière dont nous abordons certaines tâches et rendant possibles des activités auparavant difficiles. En médecine, l'intelligence artificielle améliore significativement l'analyse des scanners médicaux et accélère le développement de nouveaux médicaments. Pour les entreprises, elle est capable d'analyser les journaux d'accès aux réseaux pour détecter les cyberattaques, anticiper les menaces, et même examiner le code source. Dans le domaine des systèmes embarqués, l'apprentissage automatique peut se substituer à des filtres analogiques complexes et à l'utilisation de



filtres numériques énergivores. Cependant, l'attention médiatique se concentre souvent sur les aspects les moins flatteurs de l'IA, comme les fausses vidéos génératives ou les erreurs de ChatGPT lors d'opérations simples. Bien que certaines inquiétudes et risques éthiques méritent une attention sérieuse, il ne faudrait pas pour autant renoncer aux nombreux bénéfices apportés par l'intelligence artificielle.

Comme toute technologie, l'IA peut être utilisée à des fins bénéfiques, malveillantes ou néfastes. Il serait judicieux de consacrer davantage de temps à comprendre le fonctionnement interne de cette techno-

logie, comme nous l'avons exploré ici, et de l'appliquer à des défis que les humains trouvent fastidieux, complexes, ou à des tâches sujettes aux erreurs humaines. Internet offre une abondance de tutoriels, guides et exemples sur la manière de construire vos propres modèles d'IA ou de les entraîner à traiter des ensembles de données qui résistaient à d'autres méthodes. Le moment est peut-être venu d'expérimenter avec l'IA et de relever les défis que vous envisagez. ◀

240564-04

À propos de l'auteur

Stuart Cording est à la fois ingénieur et journaliste, avec plus de 25 ans d'expérience dans l'industrie électronique. Il est spécialisé dans la création de contenu vidéo, mettant l'accent sur les analyses techniques approfondies et la compréhension des technologies. Il porte un intérêt particulier à la technologie elle-même, à son intégration dans les applications finales et aux perspectives d'évolution future. Vous pouvez découvrir de nombreux articles récents de Stuart Cording sur le site d'Elektor à l'adresse www.elektormagazine.com/cording.

Questions ou commentaires ?

Contactez Elektor (redaction@elektor.fr).

LIENS

- [1] S. Cording, "Understanding the Neurons in Neural Networks (Part 1): Artificial Neurons," elektormagazine.com, May 2021: <https://www.elektormagazine.com/articles/neural-networks-part-1-artificial-neurons>
- [2] RGB in % for orange: <https://www.farb-tabelle.de/en/rgb2hex.htm?q=orange>
- [3] S. Cording, "Understanding the Neurons in Neural Networks (Part 3): Practical Neurons," elektormagazine.com, June 2021: <https://www.elektormagazine.com/articles/neural-networks-part-3-practical-neurons>
- [4] Izoelektro RAM-1: <https://www.ram-center.com/>
- [5] Edge Impulse, "The New AI- Powered Grid": https://cdn.prod.website-files.com/618cdeef45d18e4ef2fd85f3/621cef966699cbc24cdae67e_Smart-Grid-Monitoring.pdf
- [6] Edge Impulse: <https://edgeimpulse.com/>
- [7] B. Kopp, "Getting Started with Using Amazon Mechanical Turk to Label Data," U.S. Bureau of Labor Statistics, September 2019: <https://www.fcs.gov/assets/files/docs/gasp/Kopp.pdf>
- [8] D. J. Mankowitz, A. Michi, "AlphaDev discovers faster sorting algorithms," Google DeepMind, June 2023: <https://deepmind.google/discover/blog/alphadev-discovers-faster-sorting-algorithms>
- [9] Incorrect count of "r" characters in the word "strawberry": <https://community.openai.com/t/incorrect-count-of-r-characters-in-the-word-strawberry/829618>

Rejoignez notre communauté



www.elektormagazine.fr/community

elektor
design > share > earn