

Machine Learning

Rafael Girodon

20 juin 2024

**Part I**  
**Entrée dans le monde du**  
**Machine Learning**

# 1 Introduction

Afin qu'une intelligence artificielle soit capable de « réfléchir » par elle-même, il est nécessaire de procéder à ce qu'on appelle le « machine learning ». Il s'agit en réalité d'un long processus d'entraînement au cours duquel notre IA va apprendre à analyser les données d'entraînement que nous lui avons préalablement fournies, et au cours duquel elle va aussi apprendre à se débrouiller par elle-même afin d'analyser plus tard des données inconnues. À l'aide de ce procédé, notre intelligence artificielle sera capable d'interpréter ces sus-mentionnées données et ainsi déterminer une réponse adéquate à notre problème. Pour cela, il faudra passer par plusieurs étapes cruciales au bon fonctionnement de l'intelligence ; ce sont toutes ces étapes que je vais vous résumer dans cette présentation. Nous verrons également un exemple de modèle d'IA, bien que ultra-simpliste et simplifié. Sachez enfin qu'il existe un autre moyen d'entraîner son modèle ; il s'agit du Deep Learning. Mais avant tout, nous pouvons nous poser cette question, comment fonctionne une IA ?

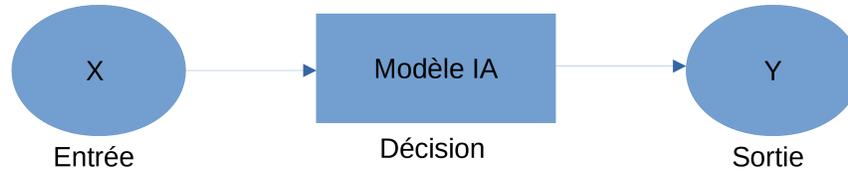


Figure 1: Voilà comment fonctionne une IA

## 2 L'Intelligence Artificielle, c'est quoi ?

En résumé, nous entrons une donnée (ici X) qui va être traitée par le modèle, qui prendra ensuite une décision, en l'occurrence Y dans notre exemple. Ces données peuvent être des images à analyser, du texte, des audios, des vidéos...

Examinons de plus près ces données...

## 3 Les données

Avant de choisir le modèle pour apprendre à l'intelligence à prendre les bonnes décisions, il est nécessaire de lui donner une importante masse de données. Ainsi, la machine aura accès à une base de données solides et variées pour apprendre. Ces données doivent être les plus nombreuses possibles, cela augmentera la capacité et la précision du modèle (plus on lui donne de données et plus il sera capable de prévoir avec précision les résultats, ou de donner une réponse correcte). Ces données doivent être choisies avec soin, car l'utilisation de données erronées sera handicapante pour le système, il ne pourra pas prévoir correctement les issues. Nos données devant être de la plus haute qualité possible, il faut donc impérativement les traiter. Voici quelques exemples de problèmes courants avec nos données :

- Les doublons, ou les valeurs aberrantes (valeurs en double, ou valeurs incohérentes)
- Les données doivent être dans un format unique et exploitable (sinon l'entraînement sera très long)
- Optimiser la tailles de ces données afin qu'elles ne soient pas trop lourdes et ainsi accélérer leur traitement (même problème que précédemment)
- Normaliser ces données (elles doivent être à la même échelle, sinon le système prendra en compte certaines données plus que d'autres)

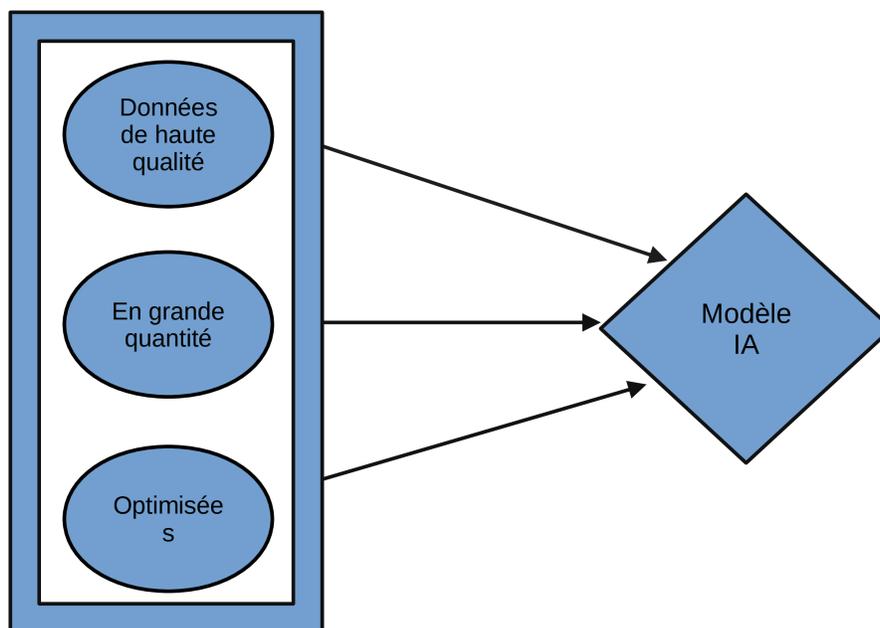


Figure 2: La Database, un espace optimisé

En résumé, voici un rapide schéma expliquant les points importants sur ces données

## 4 Les modèles, paramètres et hyperparamètres

Certes, nous avons des données à analyser. Mais maintenant, il nous faut de quoi les traiter, un programme qui saura les interpréter et donner une réponse appropriée. Il faudra également que ce programme soit capable d'intégrer des données tout a fait inconnues et de les classer immédiatement, de manière correcte bien entendu. On appelle ce programme un « modèle » . Il en existe beaucoup, et chacun peut servir à résoudre un problème différent. Nous verrons juste après comment fonctionne l'un d'entre eux, la régression linéaire.

### 4.1 Les paramètres et hyperparamètres

Il est aussi important de savoir qu'un modèle dispose de paramètres, qui seront justement déterminés par l'intelligence artificielle (soit le modèle) durant l'entraînement de ce dernier. De plus, chaque modèle dispose de ses propres « hyperparamètres », qu'il est très important de bien ajuster avant de passer à l'entraînement du modèle. Et oui, contrairement aux paramètres vus précédemment, ceux-ci sont

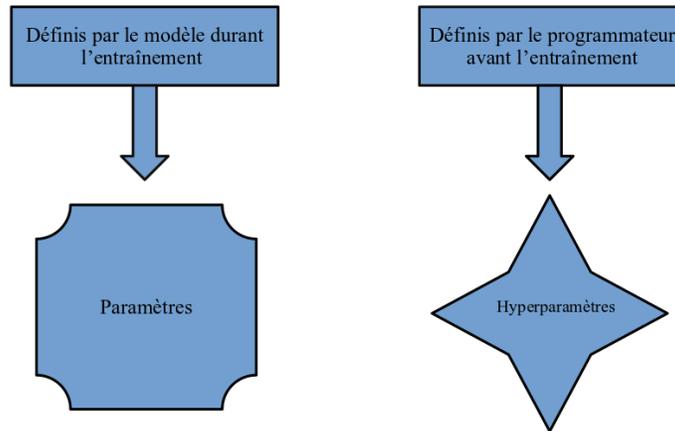


Figure 3: Paramètres et Hyperparamètres

directement ajustés par le programmeur. Il est très important de bien les ajuster, car de mauvais hyperparamètres peuvent causer principalement ces deux types de problèmes :

- L'overfitting, soit le surplus de paramètres ; le modèle va "trop" bien apprendre, il sera donc très fort avec les données de la data base, mais il sera très médiocre avec les nouvelles données
- L'underfitting, soit le manque de paramètres ; le modèle n'a pas fini son entraînement, il est médiocre avec les données de la data base et avec les nouvelles

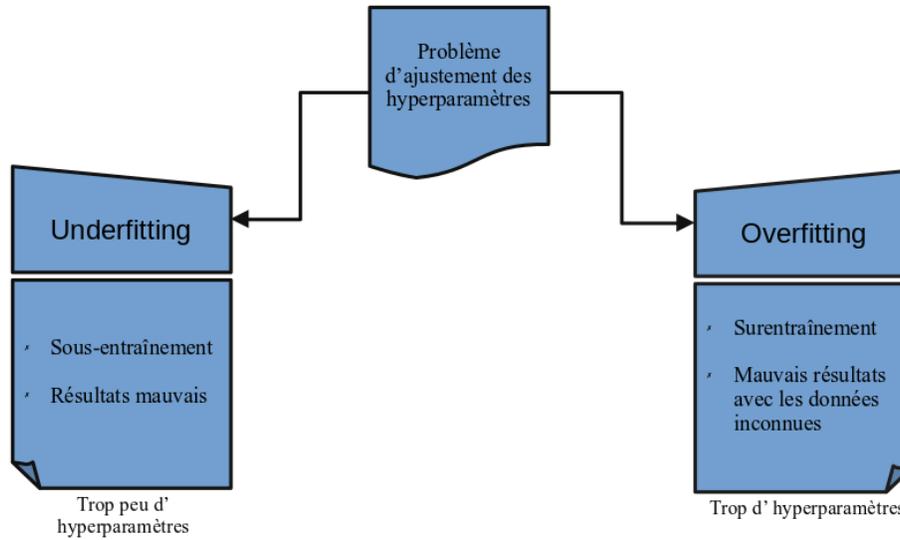


Figure 4: Overfitting et Underfitting, des problèmes récurrents

## 5 Les modèles

### 5.1 La régression linéaire

La régression linéaire est un modèle possible permettant de faire du machine learning. Des données sont représentées sur un graphique simple. Ensuite, la machine va tracer une droite en essayant d'être le plus proche possible des sus-dits points afin de créer une fonction linéaire. Elle calculera ensuite la moyenne des écarts et essaiera de les réduire au maximum en changeant les paramètres adéquats (en l'occurrence nos paramètres sont ceux d'une fonction linéaire, soit  $a$  et  $b$  ; pour rappel, une fonction linéaire s'écrit

$$y = ax + b$$

Ceci était un exemple simple s'appliquant à des fonctions linéaires très simples. Cela va sans dire que les machines learning avancées contiennent beaucoup plus de valeurs en database, l'expression de la fonction est ainsi beaucoup plus longue et fastidieuse. De même, les calculs afin de minimiser l'écart sont beaucoup plus complexes). Pour réaliser ces tâches beaucoup plus lourdes, on peut utiliser des GPU (des cartes graphiques).

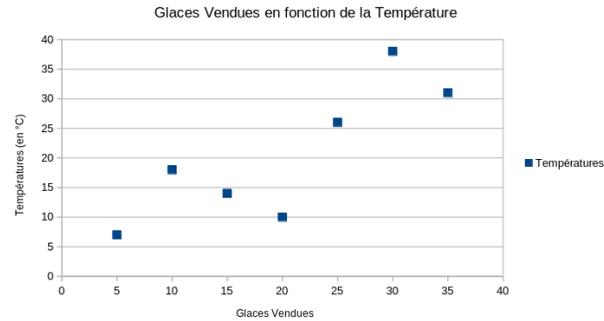


Figure 5: Nos observations

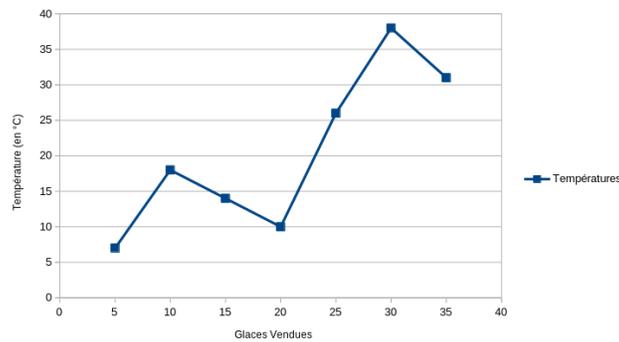
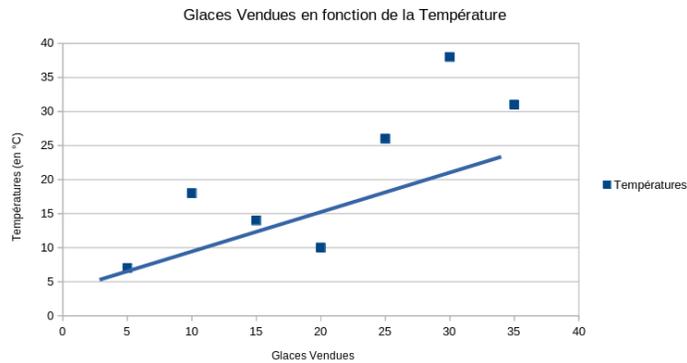


Figure 6: La courbe

## 5.2 Exemple de régression linéaire

Nous avons donc un glacier (naturellement vendant des glaces). Nous répertorions le nombre de glaces vendues par jour en fonction de la température moyenne de la journée, et nous mettons tous les résultats dans le graphique ci-contre. Nous pouvons constater que la fonction représentée semble croissante (ce qui paraît tout à fait logique). En réalité, nous avons donc donné des données qui serviront à l'entraînement de notre modèle ; soit plus précisément un nombre  $n$  de couples de données  $x_i$  et  $y_i$ , que nous appellerons ensemble  $D$ .

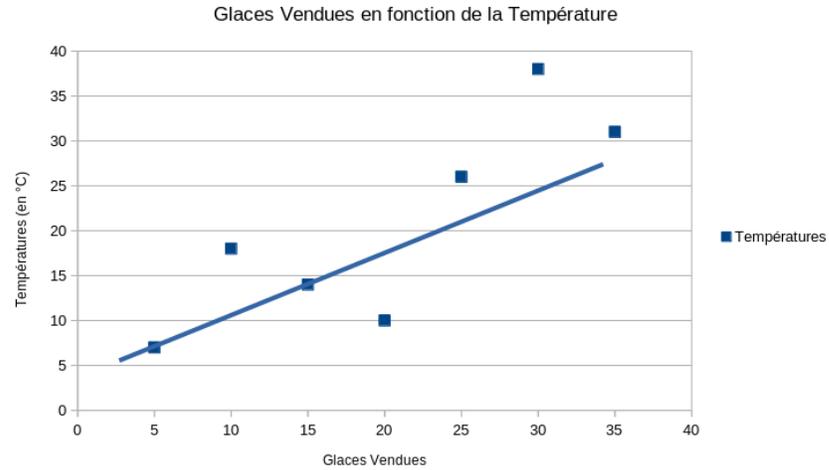
Nous pouvons maintenant tracer la courbe en reliant nos points, la fonction n'est pas du tout linéaire. Pourtant, on peut distinguer environ à l'œil nu où passerait cette fonction linéaire...



Effectivement, nous sommes assez loin de notre ligne imaginaire si nous plaçons notre courbe de fonction linéaire de cette façon. Pourtant, notre modèle va commencer ses calculs avec cette base. Pour rappel, cette fonction linéaire a pour expression :

$$y = ax + b$$

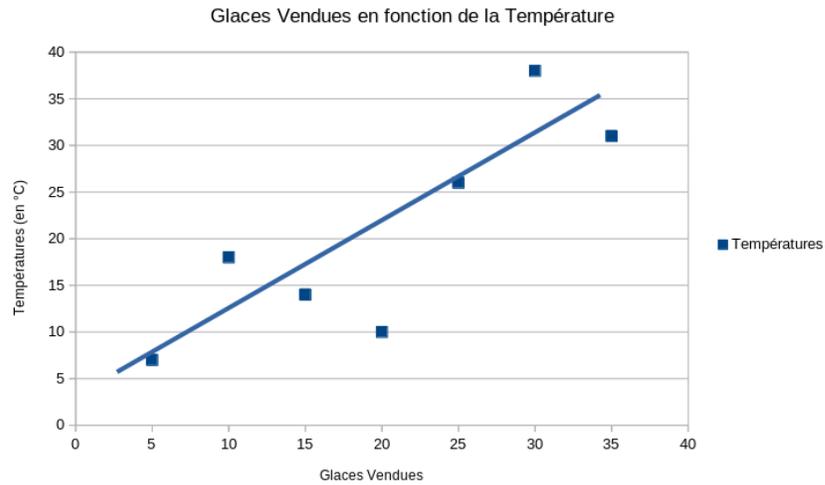
Donc, notre module va essayer de déterminer la valeur de a et de b en fonction de la valeur de x et y, qu'il connaît déjà dans tout les cas. Grâce à ces calculs, il va pouvoir essayer de tracer une nouvelle courbe...



Pas encore, mais nous voyons que nous nous rapprochons du but. Maintenant, notre module va essayer de voir l'écart qu'il lui reste à combler. Pour cela, un calcul assez complexe (alors que nous sommes sur une fonction linéaire très simple) ;

$$\operatorname{argmin}(D, \theta)_{a, b}$$

Cette fonction assez barbare va en fait calculer la moyenne d'écart entre tout les points et la droite qu'il vient de tracer. Ensuite, il va recommencer l'étape précédente en essayant de réduire cet écart encore et encore, jusqu'à...



Nous y sommes, la fonction linéaire est placée, l'écart entre les points et cette fonction a été minimisé au maximum. Maintenant, en demandant au modèle le nombre de glaces vendues, si on lui indique la température, il devrait être capable de donner une valeur moyenne approchée correcte. Même chose avec l'opération inverse (en lui donnant un nombre de glaces vendues il serait capable de prédire la température).

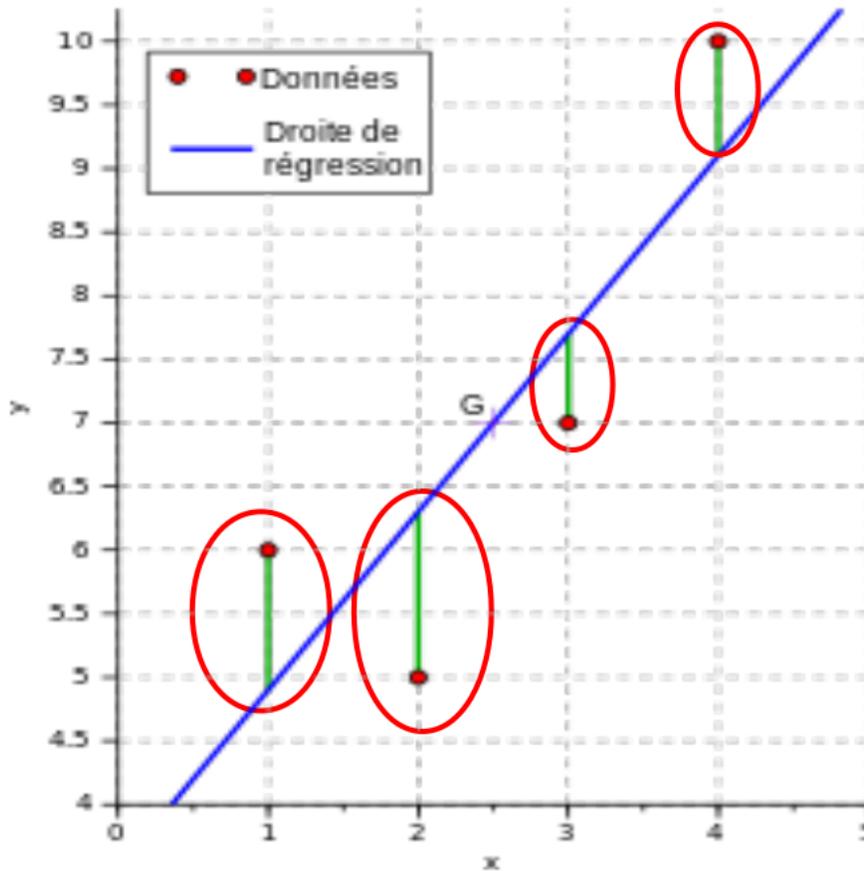
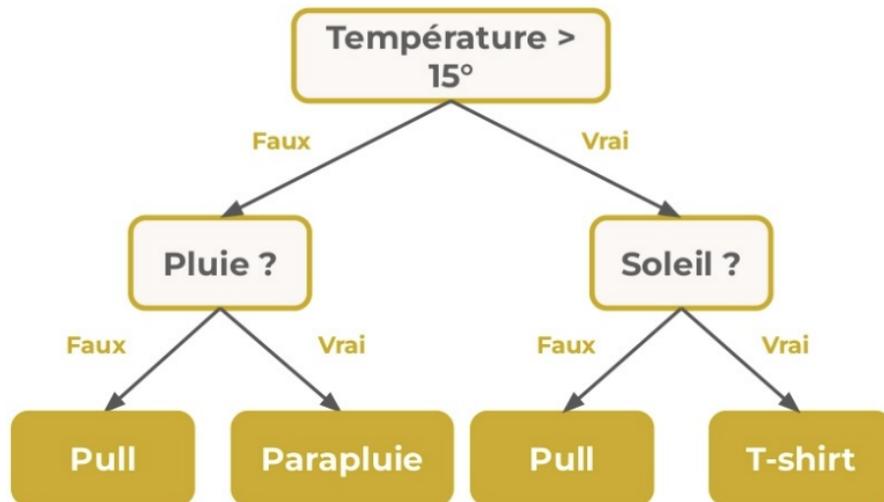


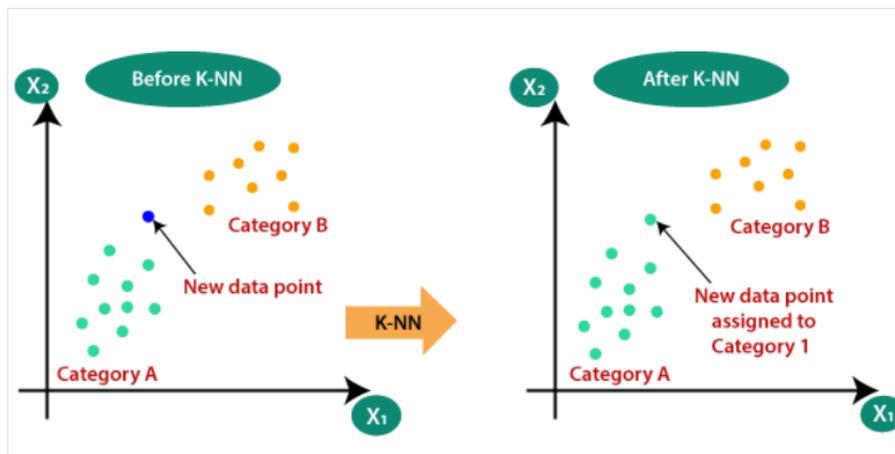
Figure 7: La régression linéaire

### 5.3 Comment fonctionne ce calcul ?

Intéressons nous un instant à cette commande  $\arg \min$ . À quoi sert-elle en réalité ? Elle va nous servir à calculer la moyenne d'écart entre chacun de nos points et la droite linéaire tracée ; en prenant l'exemple ci-contre, nous allons calculer la moyenne (de taille) des tout les bâtons verts entourés (voir ci-contre). Nous allons maintenant observer le fonctionnement d'autres modèles d'Intelligence Artificielle.



L'arbre de décision est un algorithme très simple, fonctionnant à base de conditions vérifiées ou non. Dans cet exemple, il prévoit seulement la tenue à porter en fonction de variables très simples comme la température ou la présence de pluie ; il sera généralement codé avec un algorithme CART.



L'algorithme K-NN (soit K-Plus proches Voisins) va classer les données dans un graphique et les classifier (ici en fonction de leur couleur). Ainsi, lorsqu'on lui donnera une nouvelle donnée, il sera capable de prévoir sa catégorie en fonction de sa position et donc des autres données les plus proches. Le calcul reste très simple ; il s'agit ni plus ni moins d'une distance euclidienne.

# Part II

## Les réseaux de neurones

## 6 Introduction

Il existe une autre possibilité afin d'entraîner un modèle à réaliser des prédictions que le Machine Learning : il s'agit du Deep Learning, littéralement apprentissage "profond". En effet, cette manière essaye de se rapprocher au maximum du fonctionnement du cerveau humain et pour cela, les réseaux de neurones sont utilisés... Mais en quoi consistent-ils ? C'est ce que je vais vous résumer (en simplifiant) dans cet article.

## 7 Le réseau de neurones chez l'humain

Avant de nous intéresser à comment ces réseaux fonctionnent sur un ordinateur, nous allons d'abord voir comment ils fonctionnent chez l'humain. Les réseaux de neurones humains sont un aspect fascinant de notre cerveau. Pour comprendre cela de manière simplifiée, imaginez notre cerveau comme une immense toile d'araignée composée de milliards de petites cellules appelées neurones. Ces neurones sont connectés entre eux par des branches appelées dendrites et axones, formant ainsi un réseau complexe.

Les neurones sont les cellules de base du cerveau. Chaque neurone peut se connecter à des milliers d'autres neurones via des points de contact appelés synapses. Les synapses permettent la transmission des signaux électriques et chimiques entre les neurones. Imaginez que vous percevez quelque chose (comme un son, une image, de la douleur...), vos neurones vont recevoir ces informations sous forme de signaux électriques. Ces signaux voyagent le long des axones, traversent les synapses et sont reçus par les dendrites des neurones voisins.

Notre cerveau apprend ainsi et se développe en modifiant les connexions entre les neurones, un processus laborieux appelé plasticité synaptique. Plus une connexion est utilisée, plus elle devient forte. C'est ainsi que nous apprenons de nouvelles choses et renforçons nos souvenirs.

Donc, les réseaux de neurones travaillent ensemble pour effectuer des tâches complexes comme la pensée, le mouvement et la sensation. Chaque partie du cerveau a des réseaux spécialisés pour des fonctions spécifiques, comme la vision, l'audition ou le langage.

## 8 Les réseaux de neurones et le Deep learning

### 8.1 Les réseaux de neurones

Les réseaux de neurones sont une technologie fascinante inspirée du fonctionnement du cerveau humain. Ils sont au cœur de nombreuses innovations en intelligence artificielle (IA) et permettent à des machines d'apprendre et de résoudre des problèmes complexes. Mais qu'est-ce qu'un réseau de neurones exactement ?

Imaginez un réseau de neurones comme un modèle informatique qui imite la façon dont les neurones du cerveau humain traitent l'information. Il est composé de "neurones" artificiels connectés entre eux, formant un réseau similaire à une toile d'araignée. Chaque neurone reçoit des informations, les traite, et transmet le résultat aux autres neurones du réseau.

Nous avons donc des semblants de neurones numériques, qui tout comme dans le cerveau vont recevoir des données, les traiter et les envoyer à un ou plusieurs autre(s) neurone(s). Ces neurones sont de plus organisés en couches ; la couche d'entrée (qui reçoit les données brutes), les couches cachées (qui vont donc traiter nos données et réaliser une tâche précise par rapport à ces données) et enfin la couche de sortie qui pourra nous donner une réponse au problème initial (dans le cadre d'un chatbot une réponse à ce qu'on lui a dit, ou dans le cadre d'un système de reconnaissance faciale si nous sommes sur l'image ou pas).

Nos données vont donc se "propager" de neurones en neurones vers l'avant (vers la couche de sortie donc). Elles seront soumises à chaque neurones à des fonctions spécifiques les modifiant avant de les envoyer vers les neurones suivants.

### 8.2 Apprentissage

Notre réseau de neurones va devoir optimiser ses liens entre ses sus-dits neurones, afin d'évaluer ses performances de réussite à la tâche effectuée, il calculera une "fonction de perte". Il va donc tout simplement voir l'écart entre ses prédictions et les réponses justes. Il essaiera ensuite de minimiser cet écart afin qu'il soit le plus faible possible.

Tout ceci aura donc lieu durant notre entraînement, a cours duquel notre modèle va donc calculer comment ajuster le "poids" de chaque neurone afin d'obtenir la perte et la précision de prédiction dite "accuracy" respectivement la plus basse et la plus haute possible.

Notre modèle va donc avoir besoin de data (de données) d'entraînement qu'il est nécessaire de lui fournir afin qu'il s'entraîne. Comme pour le machine learning, ces données doivent le plus optimisées possible.

A l'instar d'un modèle fonctionnant à l'aide du Machine Learning, nous allons devoir définir un hyperparamètre très important : le taux d'apprentissage.

Prenons un exemple simple avant de commencer.

Imaginez que vous vous trouvez en haut d'une vallée. Vous voulez vous retrouver au point le plus bas de cette vallée. Si vous prenez un petit "taux d'apprentissage", vous allez faire des petits pas, ce que vous permettra de descendre lentement mais sûrement et ainsi trouver le point le plus bas à coup sûr. Cependant, cela prendra beaucoup de temps. A l'inverse, si vous choisissez un "taux d'apprentissage" élevé, cela reviendra à faire de grands sauts afin de descendre plus vite, mais augmentant par la même occasion le risque de rater le point le plus bas, et le risque de chutes.

Sur le plan numérique, cela se traduit de la même façon :

1. **Petit taux d'apprentissage** : Vous faites de petits pas, ce qui vous permet de descendre lentement mais sûrement, en évitant de tomber ou de manquer le point le plus bas. Cependant, cela peut prendre beaucoup de temps.
2. **Grand taux d'apprentissage** : Vous faites de grands pas, descendant rapidement, mais risquant de dépasser le point le plus bas ou de trébucher sur des rochers, ce qui peut vous empêcher de trouver la position optimale.

## 9 Conclusion

C'est donc ainsi que fonctionne un réseau de neurones numérique. Effectivement, le processus est très proche de celui humain, si l'on excepte bien entendu les sentiments, la morale...

Effectivement, ces derniers posent un problème récurrent dans le monde de l'intelligence artificielle : la "fairness". C'est le fait que en accumulant des données, des modèles peuvent être biaisés par ces données.

Prenons un exemple simple ; vous connaissez sans aucun doute Chat GPT, ce chatbot si répandu maintenant. Et bien, sa base de données s'étendant sur une assez vaste partie de Internet, son modèle a été biaisé par du contenu raciste, homophobe, grossophobe... Il est donc possible de lui faire dire, si l'on lui donne des instructions précises, tout ces biais déplacés. Effectivement, cela pose un problème important ; comment conserver l'intégrité de ces modèles ?

*Dans un monde injuste, comment faire  
pour que les IA ne le soient pas?*

# Part III

## La vie au labo

## 10 Contexte

Afin de comprendre comment s'organisait la vie au labo ; j'ai réalisé un petit questionnaire que j'ai ensuite proposé à différentes personnes travaillant au laboratoire dans des domaines différents. Ce questionnaire oriente ses questions sur divers thèmes, comme le parcours (études, diplômes, expériences professionnelles...), leur travail actuel (que ce soit des projets, des thèses...) ou encore des questions plus personnelles. J'ai ensuite repertorié leurs réponses dans la partie suivante.

### 10.1 Le questionnaire

Voici les différentes personnes ayant répondu à mes question :

Olivier ALATA — Thomas GAUTRAIS — Julie DEBIESSE — Corinne FOURIER

#### 10.1.1 Situation

Quel est votre métier, ou votre statut au laboratoire ?

Enseignant-chercheur

Ingénieur développement applications

Responsable d'administration

Maître de conférence

De quelle équipe faites-vous parti(e) ? Images, computer vision

Service Administration

Optical Design and Image reconstitution

#### 10.1.2 Parcours

Depuis combien de temps travaillez-vous au laboratoire ?

2011

Fin 2015

2014

2000

Quel a été votre parcours(études, exp. pro) avant d'arriver ici ?

prépa CPGE, école d'ingénieur, maître de conférence

prépa, école d'ingé, laboratoire d'astrophysique, IRISA de rennes

école de commerce, secteur culturel, secteur administratif

Master, Agreg, Thèse au laboratoire

### 10.1.3 Travail Actuel

Sur quel sujet travaillez-vous ?

Images, analyse données image

Gestion du cluster, développement d'application pour le labo, projet de recherche

RH, finances, instances du labo (règlement), sécurité/hygiène

Recherche, imagerie numérique

Donnez vous des cours à côté de votre travail au labo ?

Enseigne à télécom

Enseigne à Telecom l'imagerie numérique

### 10.1.4 Questions persos

Quelles sont les raisons qui vous ont poussées à choisir ce métier, cette orientation professionnelle?

Enseigner, métier complet (beaucoup d'aspects)r

informatique, choix de spé fin école d'ingé

secteur culturel, stimulation, dynamique

enseignement, stage en labo

Estimez-vous que la barrière de la langue peut-être problématique au déroulement de votre travail ?

pas handicapant, élèves mieux préparés maintenant

pas handicapant

vrai frein notamment pour les contrats, pour l'administratif

handicapant au début, notamment donner des cours en anglais

Quels sont les aspects de votre travail que vous aimez le plus?

enseigner, apprendre, chercher, avoir des contacts internationaux

variété de contact (personnes rencontrées), variété de travail (tâches effectuées)

variété, enrichissement culturel, stimulation intellectuelle

encadrement des jeunes, travail et contacts à l'international, organiser son travail et ses collabs

A l'inverse, quels sont les aspects de votre travail que vous aimez le moins ?

métier parfois (trop) complet

Ingénieur développement applications

beaucoup de tableaux chiffrés à remplir, tâche longue

corriger des copies