



Ressourcesinformatiques

 + QUIZ


Version en ligne
OFFERTE !
pendant 1 an

Machine Learning et Deep Learning

Des bases à la conception
avancée d'algorithmes

(exemples en Python et en JavaScript)

En téléchargement

 code source

Patrick WAMPÉ





Les éléments à télécharger sont disponibles à l'adresse suivante :
<http://www.editions-eni.fr>
Saisissez la référence ENI de l'ouvrage **RIMLDLALG** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

Avant-propos

Chapitre 1

L'histoire de l'algorithmique

1. Atteindre des objectifs	9
2. Où trouve-t-on les algorithmes ?	11
3. Depuis quand les algorithmes existent-ils ?	13
4. Les algorithmes et les ordinateurs	14
4.1 Des algorithmes inspirés de la nature	15
4.2 L'algorithmique, un jeu de logique inspiré du fonctionnement du cerveau	18

Chapitre 2

Les langages de programmation

1. Le binaire et le programmeur	23
2. Du binaire aux langages de programmation	29
3. Le bon et le mauvais programmeur	36
4. Organigramme et pseudo-code	37
5. Python et JavaScript	40

2—Machine Learning et Deep Learning

Des bases à la conception avancée d'algorithmes

Chapitre 3

Un peu de logique

1. Êtes-vous logique ?	43
2. Les biais cognitifs	44
3. Quelques définitions importantes.	47
4. Les types de raisonnements	49
5. Les quantificateurs	51
6. Résoudre les bogues de façon logique	52
7. Exercices	54

Chapitre 4

Les variables

1. Introduction	57
2. Création et nommage des variables.	58
3. Les types des variables	60
4. Affectation de variables	65
5. Les tableaux à une dimension	67
6. Les tableaux à plusieurs dimensions	68
7. Un premier programme	69
8. Programmer avec Python en utilisant Google Colaboratory.	70
9. Programmer avec JavaScript	75

Chapitre 5
Les opérateurs

- 1. Introduction 81
- 2. Les opérateurs d'affectation 82
- 3. Les opérateurs arithmétiques 82
- 4. Les opérateurs alphanumériques 84
- 5. Les opérateurs logiques 86
- 6. Les opérateurs de comparaison 88

Chapitre 6
Les conditions

- 1. Un booléen sinon rien 91
- 2. Écrire des structures conditionnelles 93
- 3. Les conditions imbriquées 99
- 4. Des conditions avec des choix prédéfinis 103

Chapitre 7
Les boucles

- 1. Introduction 107
- 2. La boucle Pour 108
- 3. La boucle Tant que 111
- 4. La boucle infinie 112
- 5. Les boucles imbriquées 114

4—Machine Learning et Deep Learning

Des bases à la conception avancée d'algorithmes

Chapitre 8

Les fonctions et la programmation orientée objet

1. Introduction	119
2. Un programmeur, ça DRY	119
3. Écrire une procédure	122
4. Les paramètres	124
5. Écrire une fonction	125
6. Les fonctions dans les langages de programmation	126
7. Utiliser des fonctions déjà existantes	130
7.1 Cas du langage Python	130
7.2 Cas du langage JavaScript	134
7.3 Exemple de fonctions très pratiques	136
8. La programmation orientée objet	138
8.1 Exemples de méthodes très pratiques	141
8.2 Créer une première classe	144

Chapitre 9

La complexité algorithmique

1. Introduction	149
2. La complexité spatiale	150
3. La complexité temporelle	151
4. Notation	152
5. Exemple de la recherche d'un élément dans une liste	152

Chapitre 10
La récursivité

- 1. Introduction 155
- 2. Une factorielle récursive..... 156
- 3. Fibonacci récursif 161

Chapitre 11
Les algorithmes d'intelligence artificielle

- 1. Surmonter le paradoxe de Polanyi..... 165
- 2. Quelques applications de l'intelligence artificielle 169
- 3. Les différents types d'intelligence 172
- 4. Comment fonctionnent les algorithmes
d'intelligence artificielle ?..... 173
- 5. Les données, la nourriture des algorithmes
d'intelligence artificielle 178
- 6. Les dangers liés à l'intelligence artificielle 180

Chapitre 12
L'analyse des données

- 1. Introduction 183
- 2. Récupération des données 184
- 3. Les outils de l'analyse de données 187
- 4. Manipulation d'arrays avec NumPy 187
 - 4.1 Importer Numpy 188
 - 4.2 Créer un array 188
 - 4.3 Accéder aux éléments d'un array..... 190
 - 4.4 Obtenir des informations d'un array..... 190
 - 4.5 Modifier le type d'un array 191
 - 4.6 Effectuer des opérations mathématiques sur un array..... 191

6—Machine Learning et Deep Learning

Des bases à la conception avancée d’algorithmes

4.7	Les autres méthodes utiles	192
5.	Manipulation de dataframes avec Pandas	193
5.1	Importer Pandas	194
5.2	Créer un dataframe à partir de plusieurs objets	194
5.3	Les séries Pandas	197
5.4	Trier un dataframe	197
5.5	Nettoyer un dataframe	198
5.6	Effectuer des opérations statistiques dans un dataframe	200
6.	Visualisation des données avec Matplotlib et Seaborn	201
7.	Encodage et normalisation des données avec Scikit learn	208
7.1	Introduction	208
7.2	Encodage	209
7.3	Normalisation	210
7.4	Train et test split	211

Chapitre 13

Le machine learning

1.	Définition du machine learning	213
2.	Le métier de data scientist	215
3.	Cas de l’analyse des prix du loyer	218
4.	Découvrir des modèles très utilisés	221
4.1	La régression linéaire	223
4.2	La régression polynomiale	226
4.3	Le clustering K-moyennes	227
4.4	L’analyse en composantes principales	231
4.5	L’algorithme des K plus proches voisins	235
5.	Notre choix de modèle	237
6.	Exemples d’applications	241

- 7. Faire du machine learning avec JavaScript 242
 - 7.1 Math.js 242
 - 7.2 Plotly.js 243
 - 7.3 La régression linéaire simple avec JavaScript 245

Chapitre 14
Le deep learning

- 1. Introduction 247
- 2. Comment ça, des neurones ? 248
- 3. Comment fonctionnent les neurones artificiels ? 249
- 4. Les applications. 255
- 5. Classification supervisée avec du deep learning 255
- 6. D'autres modèles très puissants 261
- 7. La bibliothèque OpenCV 263
- 8. Faire du deep learning avec JavaScript 264

Chapitre 15
L'apprentissage par renforcement

- 1. Introduction 265
- 2. Les bases de l'apprentissage par renforcement 267
- 3. Les processus de décision markoviens 268
- 4. Exemple d'algorithme par renforcement. 269
- 5. Exemples d'applications. 272

- Index 277

Chapitre 11

Les algorithmes d'intelligence artificielle

1. Surmonter le paradoxe de Polanyi

Comme nous l'avons vu dans le chapitre L'histoire de l'algorithmique, un algorithme est une suite d'instructions précises permettant d'atteindre un objectif. Les mots-clés sont : précision et objectif. En effet, un ordinateur est un système électromécanique rigoureux qui exécute des millions voire des milliards d'opérations par minute. Il ne fait que ce qui a été explicitement programmé au travers des lignes d'un code du programme. Il ne tolère ni ne gère l'ambiguïté. Si une nouvelle situation non mentionnée dans le programme apparaît, l'ordinateur plante, ou alors ne fait rien, attendant sagement et passivement que le programmeur ajoute les instructions lui permettant de gérer ce cas de figure.

L'ambiguïté apparaît quand les instructions ne sont pas précises ou manquent de clarté, et le langage naturel est bien trop ambigu par moments. N'avez-vous jamais eu du mal à comprendre votre interlocuteur alors qu'aucun mot utilisé ne vous était inconnu ? Savez-vous toujours discerner l'ironie du second degré ? Et même si vous devinez l'intention, êtes-vous certain de placer correctement la ponctuation ? Car elle impacte grandement le sens d'une phrase. Un exemple connu qui illustre son importance est la phrase du comte d'Auteroche lors de la bataille de Fontenoy (1745). La voici sans ponctuation : "Messieurs les Anglais tirez les premiers". Le comte a-t-il voulu dire "Messieurs ! Les Anglais ! Tirez les premiers !" ou plutôt "Messieurs les Anglais, tirez les premiers !" ?

166 — Machine Learning et Deep Learning

Des bases à la conception avancée d'algorithmes

C'est d'ailleurs pour réduire l'ambiguïté du langage naturel que des langages de programmation ont été créés. Leur syntaxe rigoureuse et leur implémentation des mécanismes de la logique de programmation rendent le code prévisible, et non pas versatile. Cela permet même à un programme de mettre en place une stratégie de gestion d'erreurs lui stipulant de s'arrêter pour nous proposer des pistes de résolution pour le bogue en cours, voire pour le résoudre automatiquement. Preuve qu'il a "compris" ce que nous "voulions" faire, qu'il a compris le sens de nos instructions. À condition bien sûr que les instructions pour gérer ce cas de figure aient été renseignées dans le programme.

Quel est le problème alors ? Le fait est que parfois décrire une instruction est une tâche plus ardue qu'elle n'y paraît. Par exemple, pourriez-vous rédiger de manière précise la suite d'instructions qui vous permet de reconnaître un visage ? Si vous dites qu'il faut le regarder et vous remémorer les visages que vous connaissez, vous êtes déjà dans l'ambiguïté. En effet, comment faites-vous concrètement et précisément pour vous remémorer un souvenir tel qu'un visage ? Autre exemple : comment faites-vous pour lire un livre ? Savez-vous l'expliquer, juste en utilisant les concepts de variables, conditions, boucles et fonctions ? Peut-être y arriverez vous si ces concepts vous sont familiers. Mais l'expérience montre que la somme d'instructions nécessaire pour cette tâche est incroyablement importante et complexe. Comment pourrait-il en être autrement puisque vous-même ne parvenez pas systématiquement à reconnaître une image précise ou l'ironie dans un texte ? Les concepts d'image, de lecture et d'ironie sont ambigus, tout ce qu'un ordinateur déteste, car il ne sait pas gérer l'ambiguïté.



Cette situation illustre le paradoxe de Karl Polanyi (1886 - 1964), économiste austro-hongrois.

Selon lui, "nous en savons plus que ce que nous pouvons exprimer". Autrement dit, l'être humain réalise des tâches qu'il aurait toutes les peines du monde à expliquer de manière claire et précise. Les humains sont des machines extraordinaires de complexité, dirigées par un cerveau que l'on qualifie comme l'élément le plus complexe de l'univers. Mais pour la plupart de nos tâches, nous fonctionnons en mode automatique. Nous faisons, mais nous ne savons pas exactement comment nous le faisons. La complexité des tâches que nous effectuons est encapsulée dans des mécanismes inconscients qui représentent la partie immergée de l'iceberg. Si nous ne pouvons pas l'expliquer, il ne peut y avoir ni clarté ni précision, et par conséquent, la programmation devient impossible, ou du moins extrêmement compliquée. Comment dépasser ce paradoxe ? Est-il possible de permettre à la machine de "penser" par elle-même ?

168 — Machine Learning et Deep Learning

Des bases à la conception avancée d'algorithmes

C'est la question que se pose, en 1950, le mathématicien britannique Alan Turing (1912 - 1954). Bien avant lui, au XVII^e siècle, des scientifiques rêvaient déjà de fabriquer une machine capable de résoudre des problèmes logiques ou de mathématiques, bref une "machine à penser". Turing, lui, a eu l'idée géniale d'élaborer un test d'intelligence pour les machines, appelé test de Turing. Durant ce test, un humain communique avec deux entités, un autre humain et une machine, avec l'objectif de discerner qui est l'humain et qui est la machine. La machine passe le test avec succès si elle arrive à se rendre indiscernable de l'humain lors de l'échange. Il s'agit donc d'un "jeu d'imitation". Plus le leurre dure longtemps, plus la machine peut être jugée "intelligente". Comment Turing a-t-il eu l'idée d'un test de cette nature ? Il s'est inspiré des travaux du philosophe anglais Thomas Hobbes (1588 - 1679). Ce dernier, en empiriste convaincu, explique dans son ouvrage *Léviathan*, que la pensée et ses conséquences (paroles, actions) ne sont en fait que le résultat d'un calcul. Le terme "calcul" est pris dans un sens très général, les opérations pouvant être de nature mathématique ($2+2=4$) ou simplement logique (syllogisme). La conclusion logique à laquelle on arrive en suivant le raisonnement de Hobbes est qu'une machine à calculer peut être convertie en machine à penser.

C'est dans ce contexte qu'apparaît l'intelligence artificielle, mot inventé par John McCarthy et Marvin Lee Minsky, deux des pionniers de cette discipline visant à surmonter le paradoxe de Polanyi. Leur objectif est basé sur l'hypothèse audacieuse inspirée des travaux de Hobbes selon laquelle "toute activité intellectuelle peut être décrite avec suffisamment de précision pour être simulée par une machine". Les deux ingénieurs définissent l'intelligence artificielle de cette façon : "La construction de programmes informatiques qui s'adonnent à des tâches qui sont, pour l'instant, accomplies de façon plus satisfaisante par des êtres humains, car elles demandent des processus mentaux de haut niveau tels que : l'apprentissage perceptuel, l'organisation de la mémoire et le raisonnement critique." Il s'agit d'une intelligence dans le sens où elle est capable d'autorégulation, d'adaptation, notamment du fait qu'elle gère dans une certaine mesure l'ambiguïté.

Ces avancées sont le résultat de discussions philosophiques entre experts, mais également d'avancées théoriques et pratiques significatives. En effet, en 1955-1956 a été créé le programme Logic Theorist, capable de prouver 38 théorèmes sur 52 des *Principia Mathematica*. En 1957, Frank Rosenblatt invente le perceptron, qui est un algorithme d'apprentissage de classifieurs binaires. Le perceptron peut être considéré comme le type de réseau de neurones le plus simple. Il y a également eu le robot ELIZA, développé par Joseph Weizenbaum entre 1964 et 1966, et dont le but est de simuler un psychothérapeute appliquant les méthodes de Carl Rogers. ELIZA reformule en questions les affirmations du patient et les lui pose.

2. Quelques applications de l'intelligence artificielle

Des applications qui auraient été pour ainsi dire impossibles avec des algorithmes utilisant les mécanismes classiques deviennent possibles grâce à l'intelligence artificielle. Cela permet non seulement de réaliser de nouvelles tâches, jusque-là l'apanage des humains, mais aussi de les surpasser dans des domaines toujours plus nombreux. Cela est dû notamment aux quatre éléments suivants dans lesquels les machines nous surpassent :

- Puissance : comparés aux humains, les ordinateurs sont plus résistants et endurants. De plus, ils possèdent une puissance de calcul supérieure bien plus élevée.
- Fiabilité : ils stockent l'information de manière fiable et durable, et cela en très grande quantité au travers de dispositifs toujours plus petits.
- Rapidité : ils sont extrêmement rapides, effectuant des millions de calculs à la seconde.
- Connexion : grâce au réseau Internet qui parcourt le monde, notamment au travers de câbles continentaux, ils peuvent entrer en contact avec des milliards d'autres machines intelligentes et autonomes pour s'envoyer des informations qui circulent à la vitesse de la lumière.

170 — Machine Learning et Deep Learning

Des bases à la conception avancée d'algorithmes

Voici une liste non exhaustive des prouesses de l'intelligence artificielle :

- reconnaître des images
- conduire des voitures
- deviner le style d'une musique
- dessiner une pochette d'album
- jouer à des jeux vidéo (échec, go, Mario Kart...)
- composer de la musique
- écrire un article de journal
- acheter et vendre des actions en bourse
- prendre la place de l'arbitre dans une compétition de gymnastique
- diagnostiquer des maladies
- traduire un message oral en texte après avoir déterminé la langue parlée
- etc.

Détaillons quelques exemples issus de deux secteurs distincts, tout en gardant à l'esprit que le champ d'application de l'intelligence artificielle est pour ainsi dire illimité :

- Industrie : ce secteur utilise déjà massivement des produits issus de l'intelligence artificielle. On pense par exemple aux ateliers de fabrication automatisés dans lesquels des robots industriels s'affairent à la fabrication d'objets divers et variés. Un domaine très intéressant concerne la maintenance prédictive, qui consiste à anticiper les pannes d'équipement en fonction du cycle de vie et de l'usage des composants. Il s'agit non seulement d'une problématique de sécurité, mais également d'une question financière. En effet, plutôt que de changer un composant ou un ensemble de composants à date fixe, ce qui peut mener à des actions de maintenance excessives susceptibles de se révéler dangereuses, on personnalise les actions de maintenance pour maximiser la sécurité et donc minimiser les coûts, l'intelligence artificielle permettant d'obtenir une probabilité d'apparition d'une défaillance en fonction des données récoltées par des capteurs et analysées en temps réel. Ces capteurs peuvent alimenter le modèle d'intelligence artificielle avec des données issues de mesures acoustiques, thermographiques ou encore d'analyses vibratoires.

- Commerce : pour vendre plus et mieux, les entreprises veulent analyser et anticiper les comportements clients, dans le but de les regrouper par similarité et de leur proposer ainsi les produits les plus adaptés ou des produits de substitution, et cela au meilleur moment. Cela a également un effet sur la gestion des stocks, car savoir ce que les clients veulent vraiment permet de prévoir la demande d'un produit particulier ou d'un type de produit. L'intelligence artificielle permet d'analyser un ensemble de données (météo, emplacement du magasin, type d'achat, période de l'année...) pour atteindre ces objectifs, et d'autres encore, comme prévoir l'impact d'une campagne publicitaire ou promotionnelle sur le chiffre d'affaires. Parmi les entreprises qui font appel à l'intelligence artificielle, on peut citer Amazon, qui a mis au point un système basé sur les réseaux de neurones (deep learning) nommé DSSTNE (*Deep Scalable Sparse Tensor Network Engine*), dont l'objectif est de recommander des produits en fonction de l'historique des clients (historique qui comprend les produits achetés ou simplement les pages consultées). Il y a également Netflix, qui utilise massivement un système de recommandation de films très perfectionné. Non seulement ce système recommande des films en utilisant la méthode du filtrage collaboratif, mais il analyse également les recherches des utilisateurs pour orienter la stratégie Netflix de création de films, l'entreprise étant passée du statut de simple distributeur de vidéos à producteur à succès.