

République Algérienne Démocratique et Populaire  
Ministère de l'enseignement Supérieur et de la Recherche Scientifique  
Université de Mohamed El Bachir El Ibrahimi de Bordj Bou Arréridj  
Faculté des Mathématiques et d'Informatique  
Département d'informatique



## **MEMOIRE**

Présenté en vue de l'obtention du diplôme

**Master en informatique**

Spécialité : Ajoutez votre spécialité ici

## **THEME**

Le thème de ce mémoire ICI

*Présenté par :*

NOM ET PRENOM DE L'ETUDIANT 1

NOM ET PRENOM DE L'ETUDIANT 2

*Soutenu publiquement le : jj/mm/aaaa*

*Devant le jury composé de :*

**Président :** .....

**Examineur :** .....

**Encadreur :** .....

**2022/2023**

# Dédicace

Écrivez votre dédicace ici  
Une courte dédicace est préférable

# Remerciement

Gardez-le aussi court et direct que possible.

# Résumé

Écrivez votre résumé en français ici. Présentation en quelques lignes du contenu du rapport. Visez 1 paragraphe, mais ne pas dépasser 2 paragraphes. Présentez l'objectif du projet, les résultats obtenus et leur importance.

# Abstract

Write your abstract in English here.



# Table des matières

<b>Liste des abréviations</b>	<b>ix</b>
<b>Liste des figures</b>	<b>x</b>
<b>Liste des tableaux</b>	<b>xi</b>
<b>Liste des Algorithmes</b>	<b>xii</b>
<b>1 Introduction Générale</b>	<b>1</b>
1.1 Introduction . . . . .	1
1.2 L'apprentissage profondeur (Deep Learning) . . . . .	2
1.3 Types dApprentissage Automatique . . . . .	2
1.3.1 Apprentissage Supervisé . . . . .	2
1.3.2 1.3.2. Apprentissage Non-Supervisé . . . . .	3
1.4 Algorithmes de Classification . . . . .	3
1.4.1 Les k plus proches voisins (KNN) . . . . .	3
1.4.2 k-means . . . . .	4
1.4.3 Régression Logistique . . . . .	5
1.4.4 Machine à Vecteurs de Supports (S V M) . . . . .	5
1.5 L'apprentissage profondeurd (Deep Learning) . . . . .	6
1.6 Modèles du Deep Learning . . . . .	6
1.6.1 Réseau de neurones à convolution (CNN) . . . . .	6
1.6.2 Réseaux Neuronaux Récurrents (RNN) . . . . .	7
1.6.3 Réseaux Long Short-Term Memory (LSTM) . . . . .	7
1.7 Conclusion . . . . .	8

<b>2</b>	<b>Un guide</b>	<b>9</b>
2.1	Introduction . . . . .	9
2.2	La structure générale . . . . .	9
2.3	Du pronom désignant l’auteur du rapport . . . . .	9
2.4	Du pronom désignant le lecteur ou une personne en général . . . . .	10
2.5	Conclusion . . . . .	10
<b>3</b>	<b>Le format</b>	<b>11</b>
3.1	Texte, paragraphes, les titres, et les sous titres . . . . .	11
3.2	Code source et Algorithmes . . . . .	11
3.3	Formules mathématiques . . . . .	12
3.4	Les listes . . . . .	13
3.5	Remarques . . . . .	13
<b>4</b>	<b>Figures, tableaux et références</b>	<b>15</b>
4.1	... . . . .	15
4.2	Les tableaux . . . . .	15
4.3	Les figures . . . . .	15
4.4	Les références . . . . .	17
4.5	... . . . .	17
<b>5</b>	<b>Conclusion générale (2 pages max)</b>	<b>18</b>
5.1	Contributions . . . . .	18
5.2	Critique du travail . . . . .	18
5.3	Travaux futurs et perspectives . . . . .	18
	<b>Références</b>	<b>18</b>
<b>A</b>	<b>Titre de l’annexe ici</b>	<b>20</b>
1.1	. . . . .	20
1.1.1	. . . . .	20
<b>B</b>	<b>Titre de l’annexe ici</b>	<b>21</b>
2.1	. . . . .	21
2.1.1	. . . . .	21



# Liste des abréviations

**IA** Intelligence Artificielle.

**MI** Math et Informatique.

(Cette liste est optionnelle, voici un exemple)

# Table des figures

1.1	Les différentes méthodes d'apprentissage automatique. . . . .	2
1.2	Pour $k = 3$ la classe majoritaire du point central est la classe B, mais si on change la valeur du voisinage $k = 6$ la classe majoritaire devient la classe A . .	4
1.3	L'algorithme k-means regroupe les données en k cluster, ici $k = 3$ . Les centres de gravité sont représentés par de petit cercle . . . . .	4
1.4	La différence entre la régression logistique et la régression linéaire. . . . .	5
1.5	Les vecteurs de support, hyperplan et la marge. . . . .	5
1.6	Réseau de neurones avec de nombreuses couches convolutives. . . . .	7
1.7	Architecture de RNN. . . . .	7
1.8	: Le module répétitif dans un LSTM. . . . .	8
4.1	Un exemple d'une figure. . . . .	16
4.2	Un exemple d'une figure avec deux sous-figures. . . . .	16

# Liste des tableaux

4.1	Un exemple d'un tableau. . . . .	15
-----	----------------------------------	----

# List of Algorithms

1	An algorithm with caption . . . . .	12
---	-------------------------------------	----

# Chapitre 1

## Introduction Générale

Chaque rapport doit commencer par une introduction générale dans laquelle le contexte du projet est clairement expliqué. Cette introduction devrait également inclure l'objectif du projet et le plan du reste du rapport. Cette introduction ne devrait pas dépasser 2 pages. Soyez concis et clair, et écrivez uniquement ce qui est nécessaire à écrire.

### 1.1 Introduction

L'apprentissage automatique, également connu sous le nom de *machine Learning*, est une sous-discipline de l'intelligence artificielle (IA) qui se concentre sur le développement de techniques permettant aux systèmes informatiques d'apprendre à partir de données, sans être explicitement programmés. Plutôt que de suivre des instructions programmées, les algorithmes d'apprentissage automatique utilisent des modèles statistiques pour identifier des motifs et tirer des conclusions à partir des données d'entrée. [1] Dans ce chapitre, nous nous intéressons aux techniques d'apprentissage automatique. Dans une première partie, nous présentons les deux types d'apprentissage automatique : l'apprentissage supervisé et l'apprentissage non supervisé. Ensuite, nous présentons les méthodes de classification supervisées qui ont été utilisées dans notre étude notamment, SVM, Logistique Régression, Arbres de décision et K-plus proche voisins et à la fin de chapitre, nous vous présentons Définition de Deep Learning et Modèles du Deep Learning (CNN, RNN et LSTM).

## 1.2 L'apprentissage profondeur (Deep Learning)

L'intérêt de l'apprentissage automatique a augmenté au cours de la dernière décennie, pour tout le discours sur l'apprentissage automatique, il y a beaucoup de conflits entre ce que la machine peut faire et ce que nous souhaitons. D'une façon générale, l'apprentissage automatique est un type d'intelligence artificielle (IA), c'est une science qui permet aux ordinateurs d'apprendre sans être explicitement programmés [1] Arthur Samuel, 1959. Plus précisément, l'apprentissage automatique fait référence au développement, l'analyse et l'implémentation de méthodes qui permettent à une machine (au sens large) d'évoluer et de remplir des tâches associées à une intelligence artificielle grâce à un processus d'apprentissage. Cet apprentissage permet d'avoir un système qui s'optimise en fonction de l'environnement, les expériences et les résultats observés. Dans le domaine médicale, l'apprentissage automatique a été conçu pour réaliser l'analyse de données médicales, surtout lorsque l'évolution numérique a fourni des moyens (capteurs) peu coûteux permettant de recueillir et de stocker des informations importantes liées aux patients et maladies. Par exemple, les algorithmes d'apprentissage sont utiles au médecin lors du diagnostic des patients, afin d'améliorer la vitesse, la précision et la fiabilité de son diagnostic.

## 1.3 Types d'Apprentissage Automatique

Il existe fondamentalement quatre types d'apprentissage automatique : Supervisé, semi-supervisé et non-supervisé et Apprentissage par renforcement. Dans notre étude, nous utilisons l'apprentissage supervisé pour construire des modèles pour la prédiction des maladies. Dans la suite de cette section, nous allons présenter les deux types d'apprentissage les plus utilisés qui sont l'apprentissage supervisé et l'apprentissage non supervisé.

unsup.JPG unsup.bb

FIGURE 1.1 : Les différentes méthodes d'apprentissage automatique.

### 1.3.1 Apprentissage Supervisé

Dans l'apprentissage supervisé, l'ordinateur est fourni avec des exemples d'entrées qui sont étiquetés avec les sorties souhaitées. Le but de cette méthode est que l'algorithme puisse apprendre en comparant sa sortie réelle avec les sorties apprises pour trouver des erreurs

et modifier le modèle en conséquence. L'objectif des algorithmes d'apprentissage supervisé est d'apprendre une fonction qui mappe les vecteurs de caractéristiques (entrées) aux étiquettes (sortie), sur la base d'exemples de paires entrée-sortie.

Comme il est illustré dans la Figure 1 l'apprentissage supervisé peut être utilisé pour deux types de tâches principales : la classification et la régression. Les algorithmes de classification cherchent à prédire la classe ou la catégorie à laquelle appartient une donnée d'entrée tandis que les algorithmes de régression servent à prédire une valeur numérique continue à partir de variables d'entrée. Dans ce travail, nous nous intéressons aux méthodes de classification.

### **1.3.2 1.3.2. Apprentissage Non-Supervisé**

Dans l'apprentissage non supervisé, les données sont non étiquetées, de sorte que l'algorithme d'apprentissage trouve tout seul des points communs parmi ses données d'entrée. Les données non étiquetées étant plus abondantes que les données étiquetées, les méthodes d'apprentissage automatique qui facilitent l'apprentissage non supervisé sont particulièrement utiles. L'objectif de l'apprentissage non supervisé peut être aussi simple que de découvrir des modèles cachés dans un ensemble de données. Les plus fréquents problèmes connus dans ce type sont :

- Le clustering qui consiste à regrouper un ensemble d'éléments hétérogènes sous forme de sous-groupes homogènes.
- La réduction de dimension qui consiste à prendre des données dans un espace de grande dimension, et à les remplacer par des données dans un espace de plus petite dimension sans perdre la variance [3]

## **1.4 Algorithmes de Classification**

Dans cette partie, nous présentons les algorithmes de classification [4] [2] [5].

### **1.4.1 Les k plus proches voisins (KNN)**

L'algorithme des K-Nearest Neighbors (KNN) (K plus proches voisins) est un algorithme de classification supervisé. Chaque observation de l'ensemble d'apprentissage est représentée par un point dans un espace à  $n$  dimensions ou  $n$  est le nombre de variables prédictives. Pour prédire

la classe d'une observation, on cherche les  $k$  points les plus proches de cet exemple. La classe de la variable cible, est celle qui est la plus représentée parmi les  $k$  plus proches voisins. Il existe des variantes de l'algorithme où on pondère les  $k$  observations en fonction de leur distance à l'exemple dont on veut classer [?], les observations les plus éloignées de notre exemple seront considérées comme moins importantes.

FIGURE 1.2 : Pour  $k = 3$  la classe majoritaire du point central est la classe B, mais si on change la valeur du voisinage  $k = 6$  la classe majoritaire devient la classe A

- **Avantages**
  - Simple à concevoir .
- **Inconvénients**
  - Sensible aux bruits.
  - Pour un nombre de variables prédictives très grands, le calcul de la distance devient très coûteux.

## 1.4.2 k-means

L'algorithme des  $k$ -moyennes ( $k$ -means) est un algorithme non supervisé. Chaque observation est représentée par un point dans un espace à  $n$  dimensions où  $n$  est le nombre de variables descriptives. À partir d'un ensemble d'apprentissage de  $M$  observations  $[X^{(1)}, \dots, X^{(M)}]$  cet algorithme va répartir

Choisir  $k$  points qui représentent la position moyenne des clusters.

répéter jusqu'à stabilisation des points centraux :

- affecter chacun des  $M$  points au plus proche des  $k$  points centraux .
  - mettre à jour les points centraux en calculant les centres de gravité des  $k$  clusters .
- **Avantages**
    - implémentable pour des grands volumes de données.
  - **Inconvénients**
    - Le choix du paramètre  $k$  n'est pas découvert mais choisi par l'utilisateur.
    - La solution dépend des  $k$  centres de gravité choisis lors de l'initialisation.

FIGURE 1.3 : L'algorithme  $k$ -means regroupe les données en  $k$  clusters, ici  $k = 3$ . Les centres de gravité sont représentés par de petits cercles



### 1.4.3 Régression Logistique

L'analyse de régression est souvent utilisée pour faire des prédictions, comprendre les variables indépendantes par rapport à la variable dépendante et étudier la forme de leur relation. Dans des circonstances limitées, l'analyse de régression peut être utilisée pour déduire la relation causale entre la variable indépendante et la variable dépendante. La régression est un algorithme robuste lorsqu'il s'agit de classer des ensembles de problèmes, et a une fonction logistique (fonction sigmoïde) au cur de celui-ci. Dans cet algorithme, les valeurs d'entrée sont combinées en fonction de coefficients ou de poids pour donner les valeurs de sortie/prédites. [ ? ].

FIGURE 1.4 : La différence entre la régression logistique et la régression linéaire.

- **Avantages**

- Ses résultats sont faciles à interpréter.

- **Inconvénients**

- La phase d'apprentissage peut être longue car l'optimisation des coefficients est complexe.
- Sa linéarité empêche la prise en compte des interactions entre les variables.

### 1.4.4 Machine à Vecteurs de Supports (S V M)

Support Vector Machine (SVM) est l'un des algorithmes d'apprentissage supervisé les plus populaires, utilisé pour les problèmes de classification et de régression. Cependant, il est principalement utilisé pour les problèmes de classification dans l'apprentissage automatique. Le but de l'algorithme SVM est de créer la meilleure ligne ou limite de décision qui peut séparer l'espace à  $n$  dimensions en classes afin que nous puissions facilement mettre le nouveau point de données dans la bonne classe à l'avenir.

FIGURE 1.5 : Les vecteurs de support, hyperplan et la marge.

- **Avantages**

- Il permet de traiter des problèmes de classification non linéaire complexe.

- Les SVM constituent une alternative aux réseaux de neurones car plus faciles à entraîner.
- **Inconvénients**
  - Les SVM sont souvent moins performants que les forêts aléatoires.

## 1.5 L'apprentissage profond (Deep Learning)

Le Deep Learning ou apprentissage profond est un type d'intelligence artificielle dérivé du machine Learning (apprentissage automatique) où la machine est capable d'apprendre par elle-même, contrairement à la programmation où elle se contente d'exécuter à la lettre des règles prédéterminées. Le deep learning est une sous-discipline de l'intelligence artificielle qui se concentre sur l'apprentissage automatique de représentations de données à plusieurs niveaux, souvent basées sur des architectures de réseaux de neurones artificiels composés de nombreuses couches (d'où le terme "profond") permettant de modéliser et d'abstraire des données complexes et de capturer des structures hiérarchiques. Ces réseaux sont entraînés sur de grandes quantités de données à l'aide d'algorithmes d'optimisation tels que la rétropropagation, afin d'apprendre à extraire automatiquement des caractéristiques pertinentes des données [?].

## 1.6 Modèles du Deep Learning

On va présenter dans cette section les modèles de deep learning utilisés dans notre proposition (à savoir les CNN, RNN et les LSTM).

### 1.6.1 Réseau de neurones à convolution (CNN)

Le nom Réseau de neurones à convolution indique que le réseau emploie une opération mathématique appelée la convolution. Les réseaux de convolution sont un type spécialisé de réseaux neuronaux qui utilisent la convolution à la place de la multiplication matricielle générale dans au moins une de leurs couches. Les CNN sont l'un des meilleurs algorithmes d'apprentissage

pour faire l'opération de convolution qui aide à l'extraction de fonctionnalités utiles à partir de points de données corrélés localement. La sortie des noyaux convolutifs est ensuite affectée à l'unité de traitement non linéaire (fonction d'activation), qui non seulement aide à apprendre les abstractions, mais intègre également la non-linéarité dans l'espace des fonctionnalités. La topologie de CNN est divisée en plusieurs étapes d'apprentissage composées d'une combinaison des couches convolutives, des unités de traitement non linéaires et des couches de sous-échantillonnage (Jarrett et al., 2009) [?].

FIGURE 1.6 : Réseau de neurones avec de nombreuses couches convolutives.

### 1.6.2 Réseaux Neuronaux Récurrents (RNN)

Les Réseaux Neuronaux Récurrents (RNN) sont une variante très importante de réseaux neuronaux lourdement utilisés dans le traitement de langue naturelle. Ils sont appelés récurrents car ils effectuent la même tâche pour chaque élément d'une séquence, la sortie dépendant des calculs précédents. Une autre façon de penser aux RNN est qu'ils ont une mémoire qui capture des informations sur ce qui a été calculé jusqu'à présent. En théorie, les RNN peuvent utiliser des informations dans des séquences arbitrairement longues, mais en pratique, ils se limitent à ne regarder que quelques étapes en arrière. Les RNNs sont une classe de réseaux de neurones qui permettent aux prédictions antérieures d'être utilisées comme entrées, par le biais d'états cachés.

FIGURE 1.7 : Architecture de RNN.

### 1.6.3 Réseaux Long Short-Term Memory (LSTM)

Ce sont un type spécial de RNN, capable d'apprendre les dépendances à long terme. Ils ont été introduits par (Hochreiter et Schmidhuber, 1997), et ont été affinés et popularisés par de nombreuses personnes dans les travaux suivants. Ils fonctionnent extrêmement bien sur une grande variété de problèmes et sont

maintenant largement utilisés. Les LSTM sont explicitement conçus pour éviter le problème de dépendance à long terme. Se souvenir des informations pendant de longues périodes est pratiquement leur comportement par défaut, pas quelque chose qu'ils ont du mal à apprendre ! Tous les réseaux de neurones récurrents ont la forme d'une chaîne de modules répétitifs de réseau de neurones [ ? ]. Les LSTM ont des mécanismes intégrés qui leur permettent de mémoriser et d'oublier sélectivement des informations au fil du temps, ce qui les rend plus efficaces pour traiter des séquences à long terme. Ils sont couramment utilisés dans des applications nécessitant une compréhension à long terme, comme la traduction automatique, la génération de texte et d'autres tâches séquentielles complexes

FIGURE 1.8 : : Le module répétitif dans un LSTM.

## 1.7 Conclusion

Dans ce chapitre, nous avons présenté les algorithmes d'apprentissage automatique. Après avoir présenté les deux principaux types d'apprentissage automatique, une description détaillée de chaque méthode de classification a été fournie, expliquant le principe de fonctionnement de chaque méthode ainsi que ses avantages et inconvénients. Nous avons également identifié les techniques d'apprentissage profond et les bases de leur travail, qui constitueront la base de nos prochains travaux. Le chapitre suivant présente quelques travaux connexes où les méthodes d'apprentissage automatique ont été appliquées dans le domaine de la santé.

# **Chapitre 2**

## **Un guide**

### **2.1 Introduction**

Chaque chapitre doit commencer par une courte introduction et une courte conclusion. Suivez soigneusement les conseils de votre superviseur lorsque vous rédigez vos introductions et vos conclusions.

### **2.2 La structure générale**

Un rapport comprend une introduction générale, suivi d'un chapitre de l'état de l'art. Dans le troisième chapitre, vous expliquez l'architecture ou la méthodologie que vous avez utilisée. La mise en œuvre (implémentation) est expliquée et les résultats sont discutés dans le chapitre 4. Dans la conclusion générale [chapitre 5], décrivez la contribution de votre projet, ainsi que les critiques et les limites de votre travail, suivies d'éventuelles extensions et perspectives.

Ajoutez toutes les références utilisées à la fin de votre rapport après la conclusion générale. Enfin, vous pouvez ajouter vos annexes (si vous en avez) après les références.

### **2.3 Du pronom désignant l'auteur du rapport**

Utilisez *je* ou *nous* pour désigner l'auteur du mémoire.

**Exemple :** Dans ce chapitre, nous introduisons la notation utilisée pour le reste du mémoire.

## **2.4 Du pronom désignant le lecteur ou une personne en général**

Utilisez *on* ou *z* pour désigner le lecteur ou une personne en général.

**Exemple 1 :** *On* note que cette liste est longue.

**Exemple 2 :** Dans la phase de programmation, *on* doit tout d’abord obtenir une spécification précise du programme.

## **2.5 Conclusion**

Ne dépassez pas 5 phrases dans les conclusions de vos chapitres.

# Chapitre 3

## Le format

### 3.1 Texte, paragraphes, les titres, et les sous titres

Pour le texte dans tout le rapport, utilisez le Times New Roman, taille 12. Justifiez votre texte et laissez un peu d'espace au début de chaque paragraphe. L'espace entre les lignes est de 1,5, et vous devez ajouter des espaces avant et après les paragraphes pour augmenter la lisibilité.

Pour les titres et les sous-titres, le style proposé par la classe `reportz` doit être gardé.

### 3.2 Code source et Algorithmes

Les packages `algorithm`, `algorithmicx`, `algpseudocode` et `algorithm2e` peuvent être utilisés pour rédiger des algorithmes avec  $\text{\LaTeX}$ . Veuillez vous référer au lien suivant pour plus de détails sur l'utilisation de ces packages : <https://fr.overleaf.com/learn/latex/Algorithms>

L'algorithme 1 illustre un exemple simple d'un algorithme produit à l'aide du package `algorithm2e`.

Pour les codes sources des programmes et afin de bien les afficher, le package `listings` peut être utilisé. Veuillez consulter le lien suivant pour plus de détails : [https://fr.overleaf.com/learn/latex/Code\\_listing](https://fr.overleaf.com/learn/latex/Code_listing)

Voici l'exemple suivant qui illustre un code java simple affiché à l'aide du package `listings`. Le style utilisé pour formater ce code tel qu'il apparaît

---

**Algorithm 1** : An algorithm with caption

---

**Data** :  $n \geq 0$   
**Result** :  $y = x^n$

```
1  $y \leftarrow 1$ ;  
2  $X \leftarrow x$ ;  
3  $N \leftarrow n$ ;  
4 while  $N \neq 0$  do  
5   if  $N$  is even then  
6      $X \leftarrow X \times X$ ;  
7      $N \leftarrow \frac{N}{2}$ ; /* This is a comment */  
8   else  
9     if  $N$  is odd then  
10       $y \leftarrow y \times X$ ;  
11       $N \leftarrow N - 1$ ;  
12    end  
13  end  
14 end
```

---

est défini dans le préambule du document.

```
1 class HelloWorldApp {  
2     public static void main(String[] args) {  
3         System.out.println("Hello World!"); // Display the  
4         string.  
5         for (int i = 0; i < 100; ++i) {  
6             System.out.println(i);  
7         }  
8     }
```

### 3.3 Formules mathématiques

$\text{\LaTeX}$  est très pratique pour écrire des mathématiques. En fait, cette fonctionnalité est l'un des aspects les plus importants qui font du  $\text{\LaTeX}$  un choix incontournable pour la rédaction de documents techniques. Le lien suivant montre les commandes les plus élémentaires nécessaires pour commencer à écrire des mathématiques à l'aide du  $\text{\LaTeX}$  : [https://fr.overleaf.com/learn/latex/Mathematical\\_expressions](https://fr.overleaf.com/learn/latex/Mathematical_expressions)



Voici un exemple :

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \times \begin{pmatrix} e \\ f \end{pmatrix} = \begin{pmatrix} ae + bf \\ ce + df \end{pmatrix} \quad (3.1)$$

### 3.4 Les listes

Il est souvent nécessaire de présenter de l'information sous forme synthétique ou sous forme de séquence. Les listes sont un excellent outil pour présenter ce genre d'information. Celles-ci peuvent être numérotées ou non numérotées. Différents types de listes peuvent être utilisés dans  $\text{\LaTeX}$  :

- L'environnement `itemize` pour créer des listes non numérotées,
- L'environnement `enumerate` pour créer des listes numérotées,
- L'environnement `description` pour créer des listes de description.

Vous pouvez vous référer au lien suivant pour plus de détails sur la composition et la personnalisation des listes dans  $\text{\LaTeX}$  : <https://fr.overleaf.com/learn/latex/Lists>

Voici un exemple de liste numérotée :

1. Cette liste est créée à l'aide de l'environnement `enumerate`.
  - 1.1 Ce style permet de présenter l'information de façon hiérarchisée et en séquence ;
2. Ce style propose une numérotation alignée à gauche mais un texte indenté.

Voici un exemple de liste non numérotée :

- Cette liste utilise l'environnement `itemize`.
  - Par défaut, des puces différentes sont définies pour les quatre premiers niveaux hiérarchiques.
- Si vous le désirez, vous pouvez changer les puces proposées.

### 3.5 Remarques

- Utilisez les chevrons et l'italique pour les termes d'une langue étrangère : par exemple, schéma de conception (ń design pattern ž).

- Vous pouvez utiliser l’italique ou le gras pour mettre en évidence des termes. Toutefois, il convient de les utiliser de manière uniforme, et avec parcimonie.

# Chapitre 4

## Figures, tableaux et références

### 4.1 ...

Chaque figure et chaque tableau doit être référencé. L'ajout des figures et des tableaux à l'aide du  $\text{\LaTeX}$  est simple. Ce chapitre présente quelques exemples de ce processus d'ajout.

### 4.2 Les tableaux

Le lien suivant explique en détail la manière avec laquelle doit être faite la création et la personnalisation des tableaux à l'aide du  $\text{\LaTeX}$  : <https://fr.overleaf.com/learn/latex/Tables>

Le tableau 4.1 illustre un exemple d'un tableau.

TABLE 4.1 : Un exemple d'un tableau.

Methods	Result 1	Result 2
Method 1	0.67	0.74
Method 2	0.86	0.90

### 4.3 Les figures

Veuillez vous référer au lien suivant pour une description détaillée sur la façon d'insérer des images dans votre document  $\text{\LaTeX}$  et la façon de les référencer dans le texte : [https://fr.overleaf.com/learn/latex/Inserting\\_Images](https://fr.overleaf.com/learn/latex/Inserting_Images)

La figure 4.1 illustre une figure qui a été ajoutée juste pour montrer un exemple et la figure 4.2 illustre une figure principale avec deux sous-figures 4.2a and 4.2b.

## Genetic Algorithms

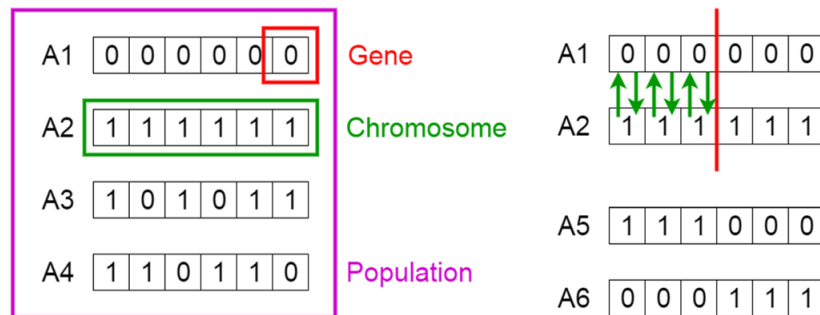


FIGURE 4.1 : Un exemple d'une figure.

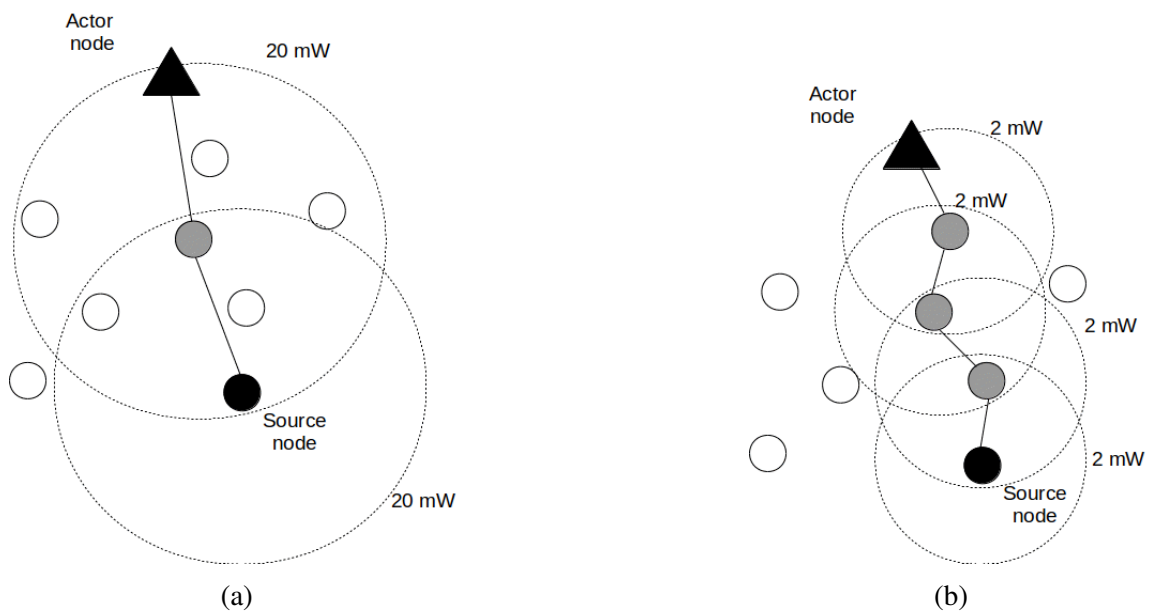


FIGURE 4.2 : Un exemple d'une figure avec deux sous-figures.

## 4.4 Les références

Les listes de références sont gérées en  $\text{\LaTeX}$  à l'aide de l'outil  $\text{\BibTeX}$  logiciel de gestion de références bibliographiques développé principalement à cet effet. Voici le lien suivant qui explique en détail comment utiliser  $\text{\BibTeX}$  :

[https://fr.overleaf.com/learn/latex/Bibliography\\_management\\_with\\_bibtex](https://fr.overleaf.com/learn/latex/Bibliography_management_with_bibtex)

Veuillez suivre le style de référence IEEE, pour cela, vous pouvez choisir, par exemple, le style de référence  $\text{\IEEEtranN}$ , ce dernier qui nécessite l'invocation du package  $\text{\natbib}$  en ajoutant `\usepackage[numbers]{natbib}` au préambule.

[2], [3], [4], [5], [1], ...

## 4.5 ...

Acronym of "Intelligence Artificielle" : IA

Meaning of MI : Math et Informatique

# **Chapitre 5**

## **Conclusion générale (2 pages max)**

### **5.1 Contributions**

Insérez un texte décrivant les contributions de votre projet.

### **5.2 Critique du travail**

Insérez un texte faisant une critique du travail.

### **5.3 Travaux futurs et perspectives**

Insérez un texte décrivant les extensions possibles du travail et les perspectives.

# Références

- [1] T. M. Mitchell, *Machine Learning*. McGraw-Hill, 1997.
- [2] D. E. Knuth, “Literate programming,” *The Computer Journal*, vol. 27, no. 2, pp. 97–111, 1984.
- [3] F. Mittelbach, M. Gossens, J. Braams, D. Carlisle, and C. Rowley, *The L<sup>A</sup>T<sub>E</sub>X Companion*, 2nd ed. Addison-Wesley Professional, 2004.
- [4] L. Lamport, *L<sup>A</sup>T<sub>E</sub>X : a Document Preparation System*, 2nd ed. Massachusetts : Addison Wesley, 1994.
- [5] M. Lesk and B. Kernighan, “Computer typesetting of technical journals on UNIX,” in *Proceedings of American Federation of Information Processing Societies : 1977 National Computer Conference*, Dallas, Texas, 1977, pp. 879–888.

# **Annexe A**

## **Titre de l'annexe ici**

### **1.1**

#### **1.1.1**



# **Annexe B**

## **Titre de l'annexe ici**

### **2.1**

#### **2.1.1**