



Fenofitia Nomenjanahary

Kajy University: Informatique

Algorithmes et structures de données

Author: Dimby Rabearivony

Date: 25 novembre 2024

Version: 1.1



"Ny fahalalana no valamparihiko."

Table des matières

1	Structures de données fondamentales	3
1.1	Variables	3
	Déclaration de variables	3
	Initialisation des variables	3
	Utilisation des variables	3
	Portée des variables	4
1.2	Pointeurs	5
	Déclaration de pointeurs	5
	Initialisation de pointeurs	5
	Utilisation de pointeurs	5
	Double pointeurs	6
	Utilisation avancée des pointeurs	6
	Pointeurs et mémoire dynamique	7
1.3	Types de données	7
	Types de données de base	7
	Types de données dérivés	8
	Typedef	9
1.4	Exercices	11
	Variables et types de données	12
	Pointeurs	12
	Tableaux	12
	Chaînes de caractères	13
	Structures	13
2	Algorithmes de tri et de recherche	14
2.1	Qu'est-ce qu'un algorithme ?	14
2.2	Récursion	14
2.3	Algorithmes de tri	15
2.4	Algorithmes de recherche	17
2.5	Exercices	17
3	Complexité algorithmique	20
3.1	Les bornes asymptotiques	20
3.2	La notation O	21
3.3	Les notations Ω et θ	21

3.4	Complexité de certains algorithmes	22
	Temps constant ($O(1)$)	22
	Temps linéaire ($O(n)$)	22
	Temps quadratique ($O(n^2)$)	22
	Temps exponentiel ($O(2^n)$)	22
3.5	Exercices	23
4	Structures de données avancées	25
4.1	Listes chaînées	25
	Création d'une liste chaînée	25
	Insertion d'un nœud dans la liste chaînée	25
	Suppression d'un nœud de la liste chaînée	26
	Résumé	26
4.2	Piles	26
	Construction d'une pile	27
	Insertion d'un élément dans une pile (empilage)	27
	Suppression d'un élément dans une pile (dépileage)	27
	Vérification si la pile est vide	27
	Utilisations courantes des piles	28
	Résumé	28
4.3	Files	28
	Construction d'une file	29
	Insertion d'un élément dans une file (enfilage)	29
	Suppression d'un élément dans une file (défilage)	29
	Vérification si la file est vide	29
	Utilisations courantes des files	29
	Résumé	30
4.4	Arbres binaires	30
	Construction d'un arbre binaire	31
	Insertion dans un arbre binaire	31
	Utilisations courantes des arbres binaires	31
	Algorithmes de parcours des arbres binaires	31
	Résumé	32
4.5	Graphes	34
	Construction d'un graphe	35
	Ajout d'arêtes (liens) dans un graphe	35
	Utilisations courantes des graphes	35
	Algorithmes de parcours de graphes	35

Résumé	36
5 Applications d’algorithmes en IA	37
5.1 Introduction	37
5.2 Algorithmes d’Apprentissage Supervisé	37
Régression Linéaire	37
Régression Logistique	38
Clustering avec K-means	39

Chapitre 5

Applications d'algorithmes en IA

5.1 Introduction

L'intelligence artificielle repose sur des algorithmes sophistiqués pour prendre des décisions, analyser des données et résoudre des problèmes complexes. Ce chapitre explore les algorithmes populaires de l'IA en mettant l'accent sur leur fonctionnement étape par étape et sur des exemples concrets qui illustrent leur application.

5.2 Algorithmes d'Apprentissage Supervisé

Régression Linéaire

La **régression linéaire** est utilisée pour prédire une valeur numérique continue (comme un prix ou une température) en fonction d'une ou plusieurs variables.

- **Idée de base** : Imaginez un nuage de points sur un graphique où l'axe x représente une variable (par exemple, la taille d'une maison) et l'axe y représente le résultat (comme le prix de la maison). La régression linéaire cherche à tracer une ligne qui passe le plus près possible de ces points.
- **Étapes de l'algorithme** :
 1. **Préparer les données** : Regroupez les données en paires (x, y) .
 2. **Déterminer l'équation** : Une ligne est définie par $y = ax + b$.
 3. **Ajuster la ligne** : Trouvez a et b pour minimiser l'erreur.
 4. **Prédire** : Utilisez $y = ax + b$ pour de nouvelles valeurs de x .

TABLE 5.1 – Données pour la régression linéaire

Surface (m ²)	Prix (MGA)
50	70,000,000
75	95,000,000
100	130,000,000

Exercice : Implémenter la Régression Linéaire

1. Écrivez un programme en C qui :
 - Accepte les paires (x, y) en entrée.

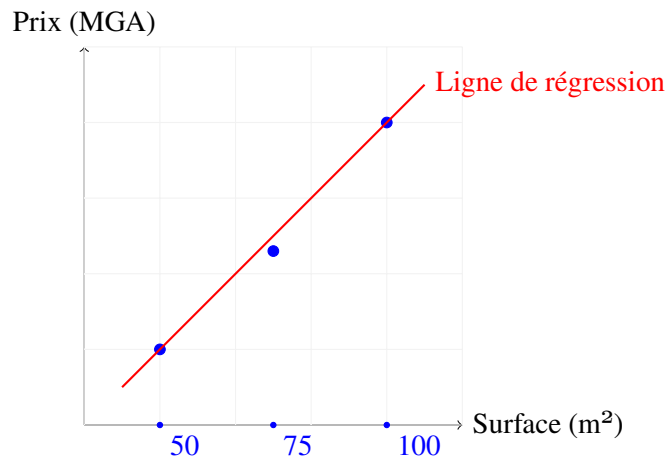


FIGURE 5.1 – Régression linéaire sur des données simulées.

- Calcule les coefficients a et b .
 - Prédit y pour une nouvelle valeur x .
2. Utilisez les données du tableau ci-dessus et prédisez y pour $x = 80$.

Régression Logistique

La **régression logistique** est utilisée pour prédire une probabilité binaire, comme "oui/non" ou "succès/échec".

- **Idée de base** : Contrairement à la régression linéaire, la régression logistique utilise une *fonction sigmoïde* pour transformer un score z en une probabilité :

$$P = \frac{1}{1 + e^{-z}}$$

- **Étapes de l'algorithme** :
 1. Calculez z avec $z = a \cdot x + b$.
 2. Appliquez la fonction sigmoïde pour obtenir P .
 3. Comparez P avec un seuil (e.g., 0,5) pour décider "oui" ou "non".

TABLE 5.2 – Exemple pour la régression logistique

Temps (minutes)	Pages visitées	z
12	7	$z = 2.3$

Exercice : Implémenter la Régression Logistique

1. Écrivez un programme en C qui :
 - Calcule z pour une valeur donnée.
 - Transforme z en P à l'aide de la fonction sigmoïde.
 - Compare P avec un seuil pour afficher "Oui" ou "Non".
2. Utilisez les données du tableau ci-dessus et prédisez si le client achètera.

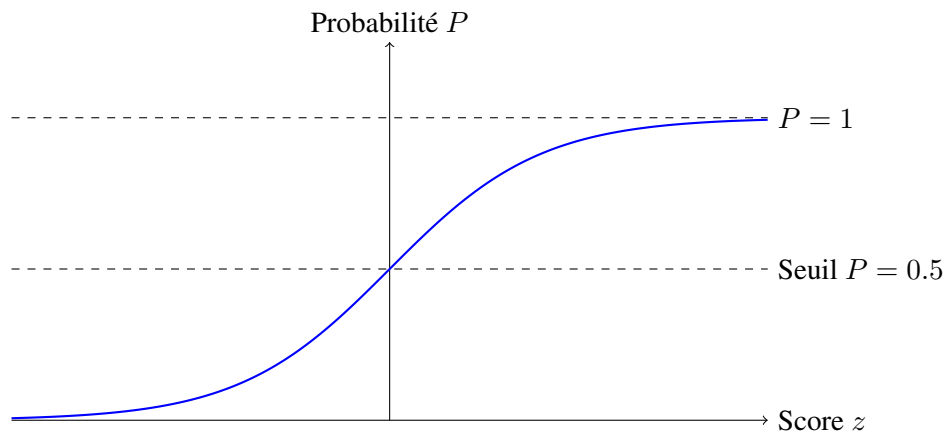


FIGURE 5.2 – Courbe sigmoïde pour la régression logistique.

Clustering avec K-means

L'algorithme **K-means** regroupe des données en plusieurs groupes appelés *clusters*.

- **Idée de base** : Divisez les données en K groupes, chaque groupe ayant un centre (*centroïde*).
- **Étapes de l'algorithme** :
 1. Choisissez K centroïdes aléatoires.
 2. Assignez chaque point au centroïde le plus proche.
 3. Recalculez les centroïdes en prenant la moyenne des points.
 4. Répétez jusqu'à ce que les centroïdes ne changent plus.

TABLE 5.3 – Données pour K-means

Point	Coordonnées (x, y)
A	(1, 2)
B	(3, 4)
C	(5, 6)
D	(8, 9)

Exercice : Implémenter K-means

1. Écrivez un programme en C qui :
 - Accepte une liste de points (x, y) .
 - Initialise 2 centroïdes aléatoires.
 - Assigne chaque point au centroïde le plus proche.
 - Recalcule les centroïdes et affiche les groupes finaux.
2. Utilisez les données du tableau ci-dessus.

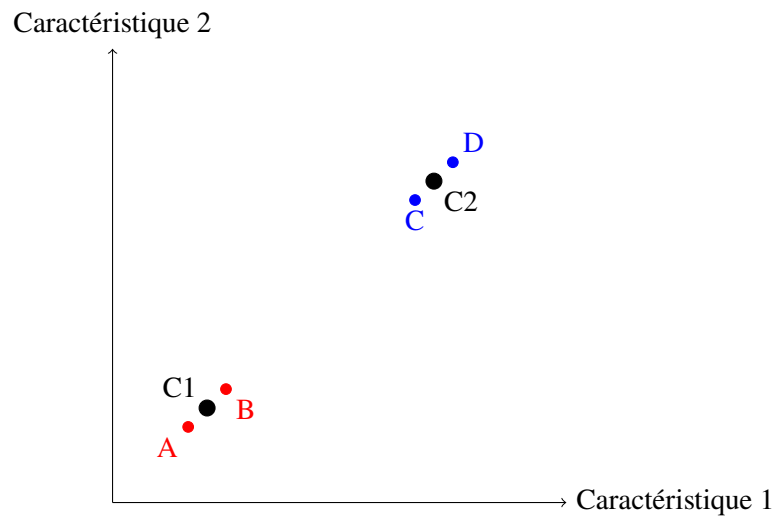


FIGURE 5.3 – Clustering K-means avec deux clusters.