TD3. Algorithmes d'intelligence artificielle

Exercice 1. k plus proches voisins

On considère le jeu de données classifiées ci-contre. Les données sont représentées par 3 coordonnées entières : x, y et z, et l'ensemble des classes est $Y = \{A, B\}$. On munit l'espace des données, ici $X = \mathbb{R}^3$ de la norme infinie :

$$d((x_1, y_1, z_1), (x_2, y_2, z_2)) = \max\{|x_1 - x_2|, |y_1 - y_2|, |z_1 - z_2|\}$$

Exécuter l'algorithme des 3 plus proches voisins pour classifier les points $(2, 1, 1)$ et $(3, 0, 0)$.

x	y	z	classe
0	1	0	A
0	3	1	B
0	2	1	A
3	2	1	B
1	3	1	B
2	1	0	A
1	0	2	B
3	2	3	A

Exercice 2. ID3

1. On considère le jeu de données classifiées ci-dessous. Les données sont représentées par 3 coordonnées booléennes x, y et z, et ces données sont classifiées au moyen de 3 classes A, B, C.

x	y	z	Classe
F	F	F	A
F	F	V	B
F	V	F	B
F	V	V	C
V	F	F	B
V	F	V	C
V	V	F	C
V	V	V	C

Exécuter l'algorithme ID3 pour fabriquer un arbre de décision pour ce jeu de données.

2. Soit $n \in \mathbb{N}$, proposer un jeu de données de $X = \{V, F\}^n$, classifiées dans $Y = \{V, F\}$ (donc deux classes seulement), telle que tout arbre de décision contient nécessairement $2^{n+1} - 1$ nœuds.

Exercice 3. k-moyenne

Considérons le jeu de données suivant de points unidimensionnels :

$$\{6, 14, 22, 38, 46, 54, 60\}$$

1. Appliquer l'algorithme des k-moyennes sur ce jeu de données, en supposant $k = 2$, et les centres initiaux $c_0 = 6$ et $c_1 = 25$.
2. On dit qu'une classification est bien séparée si pour chaque point x il n'existe pas de point y dans sa classe et il n'existe pas de point z dans une autre classe tel que $d(x, y) > d(x, z)$.
La classification obtenue est-elle bien séparée ?

Exercice 4. Implémentation des k-moyennes
Dans cet exercice, on souhaite implémenter l'algorithme des k moyennes en `C`.

1. Écrire une fonction `dist` qui prend en entrée deux points (stockés sous la forme d'un tableau), leur dimension, et renvoie la distance euclidienne entre les deux points. On pourra utiliser la fonction `sqrt` définit dans le fichier d'en tête `<math.h>`.

2. Écrire une fonction `tirage_initial` qui prend en entrée un ensemble de points, la taille de cet ensemble, et un entier k, et renvoie un tableau contenant k points tirés aléatoirement dans cet ensemble.

Pour alléger l'implémentation, on introduit une fonction auxiliaire qui détermine quelle classe attribuer au point x étant donnés les k barycentres en cours.
2. Écrire une fonction `determine_classe` qui prend en entrée un point `x`, sa dimension, et un tableau de k points, et renvoie l'indice du point le plus proche de `x` dans le tableau.

3. Écrire une fonction `k_moyenne` qui implémente l'algorithme des k moyennes en `C`. (À vous de déterminer la signature de la fonction).
Indication : Pour vérifier qu'un barycentre a été modifié, on ne comparera pas les coordonnées entre le nouveau barycentre et l'ancien entre elles, puisque ce sont des flottants. On préférera calculer la distance euclidienne entre le nouveau barycentre et l'ancien, puis s'assurer qu'il est bien inférieur à un seuil (par exemple `1e-6`).

Exercice 5. Classification hiérarchique ascendante
On a collecté cinq caractéristiques morphologiques sur 7 types de plantes, et on a calculé les distances euclidiennes entre les vecteurs descripteurs de chaque paire possible de plante. Le tableau des distances est le suivant :

	P1	P2	P3	P4	P5	P6	P7
P1	0	33	37	24	31	36	39
P2		0	42	22	39	42	35
P3			0	41	45	30	42
P4				0	41	32	40
P5					0	46	48
P6						0	34
P7							0

Appliquer l'algorithme de classification hiérarchique ascendante avec critère du lien maximum pour classer ces plantes en trois catégories.