

Voici le support utilisé pour cette séance. Il consiste en trois exercices « type khôlle », le premier sur la **NP**-complétude et les réductions polynomiales ; le second sur l'apprentissage ; et le dernier sur les deux.

En théorie, chaque exercice devrait pouvoir se faire en 1 h (environ). Mais, c'est assez difficile de prévoir la durée d'un sujet (et encore plus un sujet d'oral).

J'ai laissé quelques indications pour les questions plus complexes des exercices.

Table des matières

1	NP-complétude du problème Solitaire.	2
2	Application d'ID_3 à des situations différentes.	6
3	NP-complétude du problème DecisionTree	9

1 NP-complétude du problème Solitaire.

Q1. [*Question de cours.*] Soient P et Q deux problèmes. (Re)démontrer la propriété suivante :

si $P \preceq_p Q$ et que P est un problème **NP**-difficile, alors Q l'est aussi.

On considère un jeu, qu'on appelle « jeu du solitaire ». Ce jeu se joue sur une grille de taille $n \times m$. Sur chaque case de la grille, il peut y avoir :

- ▷ un pion bleu, qu'on représentera par « ● » dans la case ;
- ▷ un pion rouge, qu'on représentera par « ● » dans la case ;
- ▷ aucun pion, qu'on représentera par ne rien avoir dans la case.

Dans ce jeu, on part d'une configuration initiale et, à chaque tour, on retire un pion. Le but est d'arriver à une configuration *gagnante*. Une configuration est dit gagnante si les deux conditions sont vérifiées

1. *contrainte de couverture* : sur chaque ligne, il y a au moins un pion ;
2. *contrainte de monochromie* : sur chaque colonne, tous les pions sont de la même couleurs.

On dit qu'une configuration initiale est *admissible* s'il est possible d'arriver à une position gagnante à partir de cette position initiale.

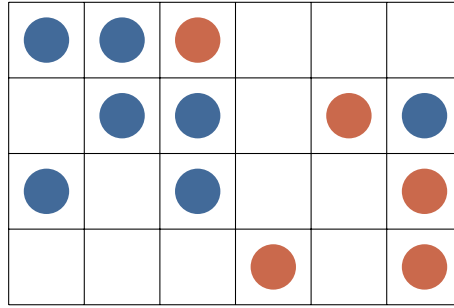


Figure 1.1 | Une configuration initiale admissible

La configuration initiale en figure 1.1 est admissible. En effet, il est possible, en partant de cette configuration, d'arriver dans une position gagnante, comme le montre la figure 1.2.

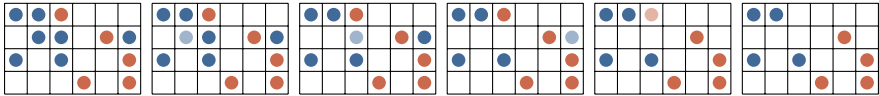


Figure 1.2 | Suite de mouvement permettant de gagner

Q2. Donner un exemple de configuration initiale non vide qui n'est pas admissible.

Formellement, le plateau de jeu, comme décrit précédemment, est représenté par l'ensemble $G = \llbracket 1, n \rrbracket \times \llbracket 1, m \rrbracket$. Une case est identifiée par le couple $(i, j) \in G$ où i est son numéro de ligne et j son numéro de colonne.

Une *configuration* ou *disposition des pions* sur un plateau G est un couple $\mathcal{C} = (B, R)$ où $B, R \subseteq G$ et $B \cap R = \emptyset$. L'ensemble B représente les cases où il y a un pion bleu ; l'ensemble R représente les cases où il y a un pion rouge.

Q3. On considère une configuration $\mathcal{C} = (B, R)$ et une configuration initiale $\mathcal{C}_0 = (B_0, R_0)$. À quelle condition \mathcal{C} peut-elle être atteinte depuis la disposition initiale \mathcal{C}_0 ?

Q4. Soit \mathcal{C} une configuration. À quelle(s) condition(s) \mathcal{C} est-elle gagnante ?

On s'intéresse au problème défini formellement par :

SOLITAIRE	<p>Entrée. Une configuration $\mathcal{C}_0 = (B_0, R_0)$ sur une grille de taille $n \times m$.</p> <p>Sortie. La configuration \mathcal{C}_0 est-elle admissible ?</p>
-----------	---

Dans la suite de cet exercice, on montre que le problème SOLITAIRE est **NP**-complet.

Q5. Justifier que le problème SOLITAIRE est dans **NP**.

Q6. Démontrer que le problème SOLITAIRE est **NP**-difficile. Pour cela, on admettra la **NP**-difficulté du problème CNF-SAT rappelé ci-dessous.

CNF-SAT	<p>Entrée. Une formule φ sous forme normale conjonctive.</p> <p>Sortie. La formule φ est-elle satisfiable ?</p>
---------	---

Indication 1.

Pour une formule φ sous n -CNF contenant m variables, on construit une grille ayant n lignes et m colonnes.

Indication 2.

Équivalent de cette condition pour la grille construite ?
Satisfait une clause de φ , c'est dire qu'un littéral est évalué à vrai.

Indication 3.

Dans une valuation ρ , chaque variable vaut soit vrai, soit faux. Équivalent de cette condition pour la grille construite ?

Indication 4.

Avoir uniquement des pions bleus dans la colonne j représente le fait qu'on choisisse x_j évalué à vrai. Avoir uniquement des pions rouges dans la colonne j représente le fait qu'on choisisse x_j évalué à faux.



Q7. Conclure quant à la **NP**-complétude du problème **SOLITAIRE**.

— Fin de l'exercice 1 —

2 Application d' ID_3 à des situations différentes.

Q1. [*Questions de cours.*]

1. Définir ce qu'est l'apprentissage supervisé/non-supervisé.
2. Rappeler les algorithmes d'apprentissage vus en cours.
3. Parmi ces algorithmes, lesquels réalisent de l'apprentissage supervisé/non-supervisé.

Dans cet exercice, on s'intéresse à l'algorithme ID_3 appliqué au problème de classification suivant : des objets sont répartis en deux classes, notées «  » et «  », selon deux attributs prenant des valeurs continues dans $[0, 1]^2$. On pourra ainsi représenter ces objets par des points (carrés ou hexagones) dans le plan.

Q2. Peut-on appliquer ID_3 directement aux données initiales afin d'obtenir un modèle permettant de les classer ? Expliquer votre réponse.

Q3. Proposer une méthode naïve permettant d'appliquer ID_3 sur un tel jeu de données. Est-ce raisonnable ?

- Q4.** Dans la suite, on souhaite classer les données à l'aide d'un arbre de décision obligatoirement binaire. Donner une méthode moins naïve que celle de la question 3 permettant d'atteindre ce but.

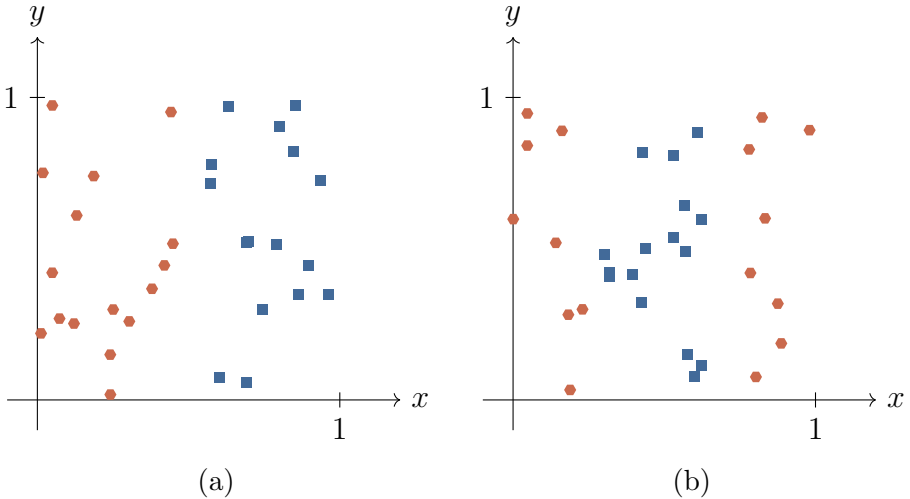


Figure 2.1 | Deux exemples de jeux de données

- Q5.** Quel résultat obtiendrait-on avec cette méthode sur la figure 2.1a ? Et sur la figure 2.1b ? Est-ce satisfaisant ?
- Q6.** Proposer une méthode permettant de mieux classer les données dans le cas où celles-ci sont réparties de manière similaire à celles de la figure 2.1b. Argumenter quant à sa terminaison.
- Q7.** Dessiner la frontière entre les deux classes que la méthode de la question 6 calculerait sur la figure 2.2a. Est-ce satisfaisant ?

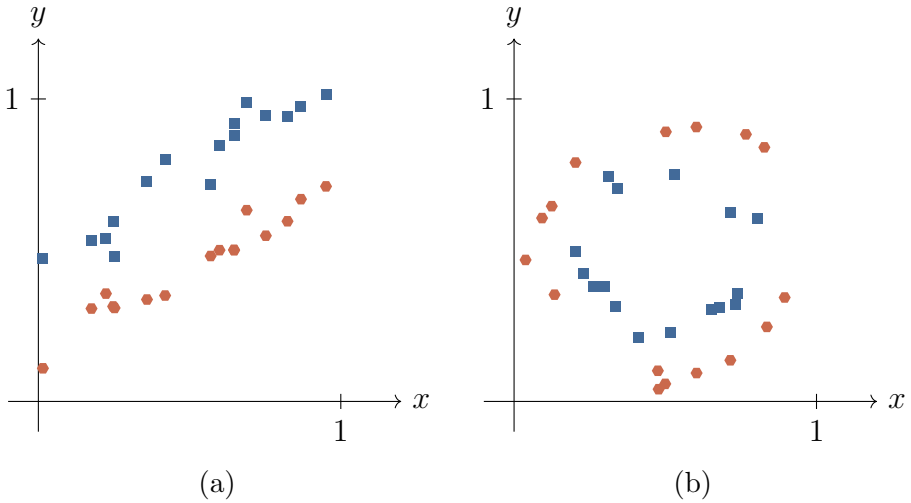


Figure 2.2 | Deux exemples supplémentaires de jeux de données

Q8. Expliquer comment transformer le jeu de données de la figure 2.2a de sorte à pouvoir tracer une frontière bien plus simple à l’aide de la méthode décrite à la question 6.

Indication 1.

Penser à une transformation géométrique, qui permet de se ramener aux hypothèses de la question 6.

Q9. Et dans le cas de la figure 2.2b, que peut-on faire ?

Indication 2.

Utiliser un changement de coordonnées très utilisé en physique.

— Fin de l’exercice 2 —

3 NP-complétude du problème DecisionTree

Q1. [*Question de cours.*] Pour quelles valeurs de $k \in \mathbb{N}$, le problème k -SAT, est-il **NP**-complet ? Justifier votre réponse.

k -SAT		Entrée. Une formule φ sous k -CNF. Sortie. La formule φ est-elle satisfiable ?
----------	--	---

Dans cet exercice, on souhaite démontrer la **NP**-complétude du problème **DECISION-TREE** défini ci-dessous.

DECISION-TREE		Entrée. Un jeu de données $\mathcal{D} = \{x_1, \dots, x_n\} \subseteq \mathbb{B}^{p+1}$ et un réel K ; Sortie. Existe-t-il un arbre de décision pour \mathcal{D} de coût inférieur ou égal à K ?
---------------	--	--

Le coût d'un arbre de décision A est défini comme la somme des longueur des chemins partant de la racine à chacune de ces feuilles.

Q2. Justifier que le problème **DECISION-TREE** est dans **NP**.

Afin de démontrer la **NP**-difficulté de **DECISION-TREE**, on va procéder à une réduction au problème **EXACT-COVER-3**.

Étant donnés un ensemble X et un sous-ensemble \mathcal{R} de $\wp(X)$, on dit que $\mathcal{R}^* \subseteq \mathcal{R}$ est une *couverture exacte* de X dès lors que, pour tout élément $x \in X$, il existe un unique $R \in \mathcal{R}^*$ tel que $x \in R$.

EXACT-COVER-3

Entrée. Un ensemble X et $\mathcal{R} \subseteq \wp(X)$ tel que tout élément de \mathcal{R} est un sous-ensemble ayant *exactement* trois éléments.
Sortie. Existe-t-il $\mathcal{R}^* \subseteq \mathcal{R}$ une couverture exacte de l'ensemble X ?

On admet la **NP**-difficulté d'EXACT-COVER-3, qui peut se déduire (par réduction) de la **NP**-difficulté de 3DM (*3 dimensional matching*).

Considérons une entrée (X, \mathcal{R}) du problème EXACT-COVER-3. On essaie de trouver un sous-ensemble \mathcal{S} de \mathcal{R} tel que

$$\bigsqcup_{R \in \mathcal{S}} R = X.$$

On rappelle que la réunion \sqcup (et par extension \bigsqcup) désigne la réunion d'ensembles disjoints deux-à-deux.

Posons $X' = X \sqcup \{a, b, c\}$, où $a, b, c \notin X$ puis

$$\mathcal{R}' = \mathcal{R} \sqcup \{ \{x\} \mid x \in X' \}.$$

Ainsi, \mathcal{R}' ne contient que des sous-ensembles de cardinal 1 ou 3. L'intuition est que nos éléments de \mathcal{R}' représentent des tests à effectuer sur les éléments de X .

Q3. On pose $f(n)$ le coût minimal d'un arbre de décision sur un jeu de données à n éléments. Justifier la relation de récurrence, pour $n \geq 4$,

$$f(n) = \min_{i \in [1,3]} (f(n-i) + f(i) + n).$$

Indication 1.

L'indice i représente la taille du sous-ensemble choisi comme test à la racine.

n	1	2	3	4	5	6	7	8
$f(n)$	0	2	5	8	12	16	20	25

Q4. Montrer que, pour $n \geq 7$, un ensemble à trois éléments doit être choisi à la racine d'un arbre de décision de coût optimal.

Q5. Comment passer d'un ensemble \mathcal{R}' de tests à un ensemble de données $\mathcal{D} \subseteq \mathbb{B}^{p+1}$?

Indication 2.

On note $d = \text{card } \mathcal{X}'$ et on définit, coordonnée par coordonnée, un vecteur $b_x \in \mathbb{B}^{p+1}$ qui représente un élément $x \in X$.

Q6. Étant donné une couverture exacte $\{T_{i_1}, \dots, T_{i_k}\}$, justifier que la forme de l'arbre de décision correspondant est un quasi-peigne (comme montré sur la figure 3.1).

Q7. L'arbre en figure 3.1 vérifie-t-il la condition donnée en question 4? Quel est le coût de l'arbre? En déduire l'optimalité de l'arbre.

Indication 3.

Démontrer que le coût de l'arbre est d'exactement $f(|X|)$.

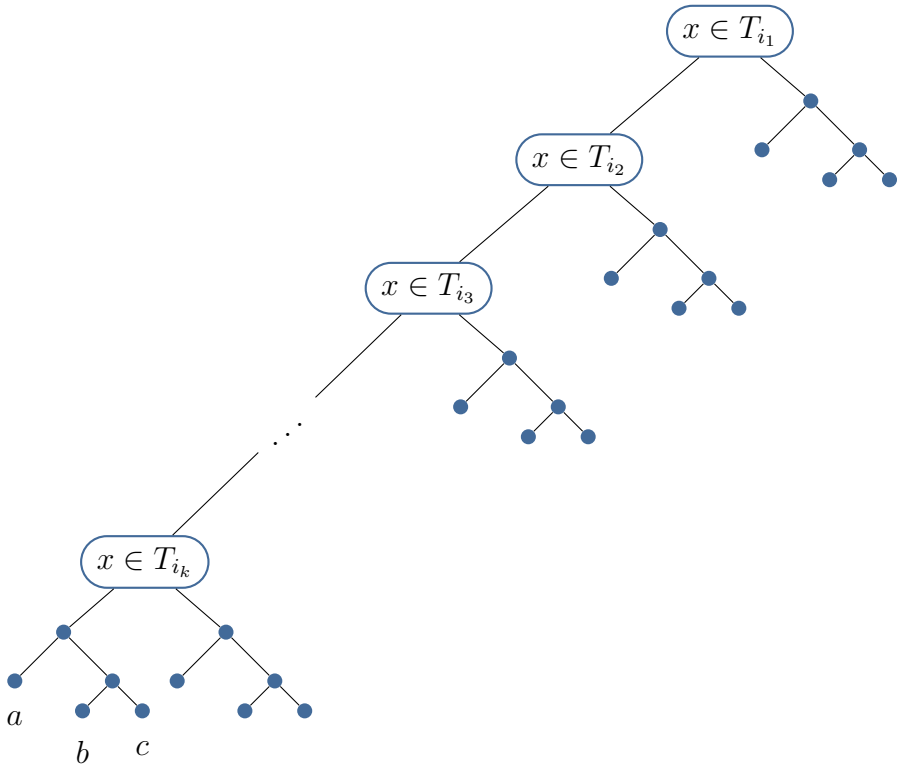


Figure 3.1 | *Arbre de décision correspondant*

- Q8.** Dédire de la construction et des questions précédentes que l'on a réalisé une réduction polynomiale.
- Q9.** Conclure quant à la **NP**-complétude du problème **DECISION-TREE**.

— *Fin de l'exercice 3* —