INF4230 – Intelligence artificielle
Examen final – Hiver 2013

Éric Beaudry
Département d'informatique
Université du Québec à Montréal

Lundi 22 avril 2013 – 18h00 à 21h00 (3 heures) – Local SH-2140

Instructions

- Documentation permise : une feuille (recto-verso) de notes personnelles.
- Appareils électroniques : uniquement une calculatrice non programmable et sans dispositif de communication sans-fil.
- Répondez directement sur le questionnaire à l'intérieur des endroits appropriés.
- Si vous croyez qu'une erreur ou qu'une ambiguïté s'est glissée dans le questionnaire, indiquez clairement la supposition que vous avez retenue pour répondre à la question.
- Ne détachez pas les feuilles du questionnaire.
- Le côté verso peut être utilisé comme brouillon. Des feuilles additionnelles peuvent être demandées au surveillant.
- Les questions plus difficiles sont accompagnées d'une étoile ★.

Identification

Nom : Solutionnaire (version préliminaire)

Résultat

Q1 (/6)	Q2 (/5)	Q3 (/9)	Q4 (/7)	Q5 (/4)	Q6 (/4)	TOTAL (/35)

1 Raisonnement probabiliste et construction d'un réseau bayésien [6 points]

Les enquêtes **volontaires** d'opinion, comme les sondages réalisés par Internet sur les intentions de vote aux élections, sont souvent critiquées d'un point de vue scientifique. Pour vérifier si une enquête **volontaire** d'opinion pouvait être biaisée, on a fait un enquête **obligatoire** (tous les individus doivent répondre) en posant les quatre questions suivantes : (Q1) Quel est votre sexe ? (Q2) Avez-vous un téléphone intelligent (*smart phone*) ? (Q3) Aimez-vous le pop-corn ? (Q4) Si la présente enquête était volontaire, auriez-vous répondu ? En compilant les données, on a obtenu la distribution conjointe suivante.

		Homme		Femme	
		Tél. Intelligent	Sans Tél. Intel.	Tél. Intelligent	Sans Tél. Intel.
Répondrait si volontaire	Aime le pop-corn	0,27	0	0,1	0,01
	N'aime pas le pop-corn	0,03	0	0,1	0,01
Refuserait si volontaire	Aime le pop-corn	0,0675	0,1125	0,05	0,09
	N'aime pas le pop-corn	0,0075	0,0125	0,05	0,09

(a1) Calculez la probabilité qu'un **homme** aurait répondu si l'enquête était volontaire. [1 point]

Rappel : $P(a|b) = P(a \wedge b)/P(b)$
$P(RepVol = vrai|Sexe = homme) = P(Sexe = homme, RepVol = vrai)/P(Sexe = homme)$
$= (0.27 + 0.03)/(0.27 + 0.03 + 0.0675 + 0.0075 + 0.1125 + 0.0125) = 0.6$

(a2) Calculez la probabilité qu'une **femme** aurait répondu si l'enquête était volontaire. [1 point]

$P(RepVol = vrai|Sexe = femme) = P(Sexe = femme, RepVol = vrai)/P(Sexe = femme)$
$= (0.1 + 0.1 + 0.01 + 0.01)/(0.1 + 0.1 + 0.01 + 0.01 + 0.05 + 0.05 + 0.09 + 0.09) = 0.44$

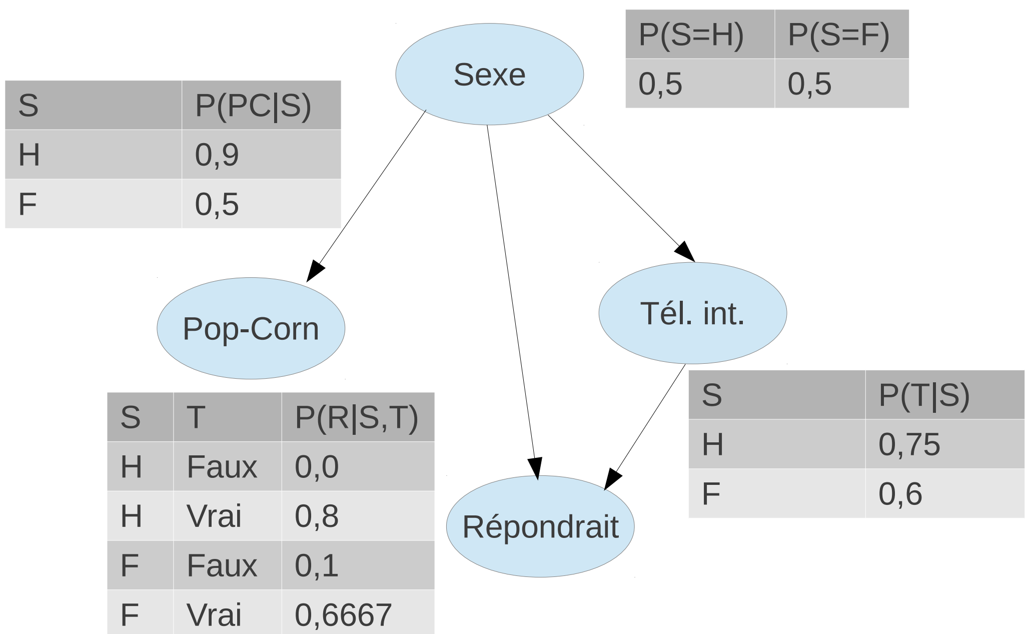
(b1) Calculez la probabilité qu'une **femme** ayant un téléphone intelligent aime le pop-corn. [1 point]

$P(Popcorn = vrai|Sexe = f, TelInt = vrai) = P(Popcorn = vrai, Sexe = f, TelInt = vrai)/P(Sexe = f, TelInt = vrai)$
$(0.1 + 0.05)/(0.1 + 0.1 + 0.05 + 0.05) = 0.5$

(b2) Calculez la probabilité qu'un **homme** ayant un téléphone intelligent aime le pop-corn. [1 point]

$(0.27 + 0.0675)/(0.27 + 0.03 + 0.0675 + 0.0075) = 0.9$

(c) ★ À partir de la distribution conjointe précédente, construisez un réseau bayésien. Ce réseau doit illustrer les relations de dépendances entre les variables. Conseil : commencer sur une feuille brouillon et transcrivez par la suite. [3 points]



P(S=H)	P(S=F)
0,5	0,5

S	P(PC|S)
H	0,9
F	0,5

S	P(T|S)
H	0,75
F	0,6

S	T	P(R|S,T)
H	Faux	0,0
H	Vrai	0,8
F	Faux	0,1
F	Vrai	0,6667

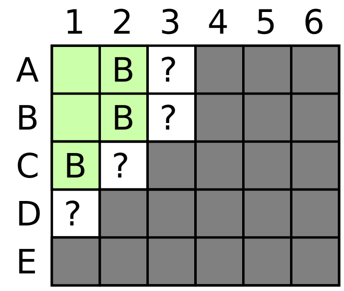
(d) Si on effectuait la même enquête mais de façon volontaire, serait-elle biaisée ? Justifiez. [1 point]

Oui. La possession d'un téléphone intelligent influence la décision de répondre ou non. L'estimation du taux de personnes ayant un téléphone intelligent serait biaisé, c'est-à-dire $P(T) \neq P(T|R = vrai)$.

2 Raisonnement probabiliste / Monde du Wumpus [5 points]

Dans le monde du Wumpus, qui a été introduit dans le livre de référence, un agent doit se déplacer dans une carte initialement inconnue pour atteindre un lingot d'or. L'agent meurt s'il tombe dans une fosse ou touche au Wumpus. L'agent est doté de trois capteurs : il peut percevoir une brise (B), une odeur (S) ou une lueur d'or (L). Quand l'agent perçoit une brise (B), alors il y a au moins une fausse dans l'une des 4 cases voisines (nord, sud, est, ouest). On suppose que les capteurs fonctionnent parfaitement. L'agent peut exécuter quatre actions $\{N, S, E, W\}$ afin de se déplacer.

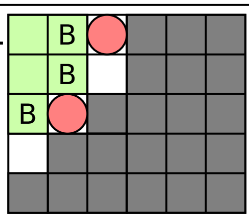
Supposez la situation à droite. À priori, on sait que la grille de 6×5 contient 9 fosses réparties aléatoirement sur la grille à l'aide d'une distribution uniforme. L'agent est initialement placé dans la case A1. Heureusement, il s'est avéré que la case A1 ne contenait pas de fosse. En exécutant la séquence d'actions $\langle S, S, N, E, N \rangle$, l'agent a observé des brises dans les cases C1, B2 et A2, c'est-à-dire l'ensemble d'observations $O = \{B_{C1}, B_{B2}, B_{A2}\}$. L'agent doit maintenant décider quelle sera la prochaine case à explorer parmi les 4 cases marquées d'un point d'interrogation (« ? »).



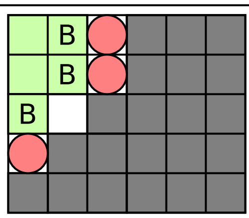
(a) Un **agent purement logique** peut-il tirer une conclusion sur l'une des cases inconnues marquées d'un point d'interrogation ? Si oui, laquelle ? [2 points]

Oui. Un agent logique peut conclure qu'il y a une fosse dans la case A3.
Il peut aussi conclure que le Wumpus est dans aucune des cases marquées d'un point d'interrogation, car aucune odeur n'a été perçu dans les cases voisines. On ne peut rien conclure quant à la présence d'or dans ces cases.

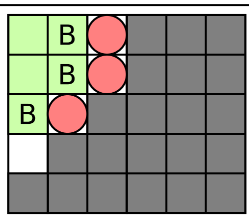
(b) ★ Quelle serait la prochaine case visitée par un **agent probabiliste** ? Indices : énumérez les 5 modèles possibles qui collent aux observations et calculez leur probabilité. Conseil : commencez par faire les calculs sur un brouillon. [3 points]

m1


P(m1|O) =α 9/25 * 16/24 * 8/23* 15/22 =9*16*8*15 /(25*24*23*22)

m2


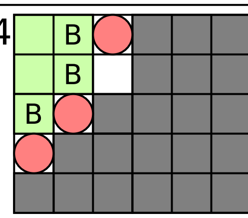
P(m2|O) =α 9/25 * 8/24 * 16/23* 7/22 =9*8*16*7 /(25*24*23*22)

m3


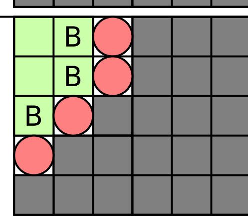
P(m3|O) =α 9/25 * 16/24 * 8/23* 7/22 =9*16*8*7 /(25*24*23*22)

Prochaine case à visiter : B3 ou D1

Case à éviter : A3 et C2

m4


P(m4|O) =α 9/25 * 8/24 * 7/23* 16/22 =9*8*7*16 /(25*24*23*22)

m5


P(m5|O) =α 9/25 * 8/24 * 7/23 * 6/22 =9*8*7*6 /(25*24*23*22)

$P(F_{A3}|O)$=P(m1|O)+P(m2|O) +P(m3|O)+P(m4|O)+P(m5|O) = 1.0

$P(F_{B3}|O)$=P(m2|O)+P(m3|O)+P(m5|O) =α' (9*8*16*7+9*16*8*7+9*8*7*6) =α' 19152

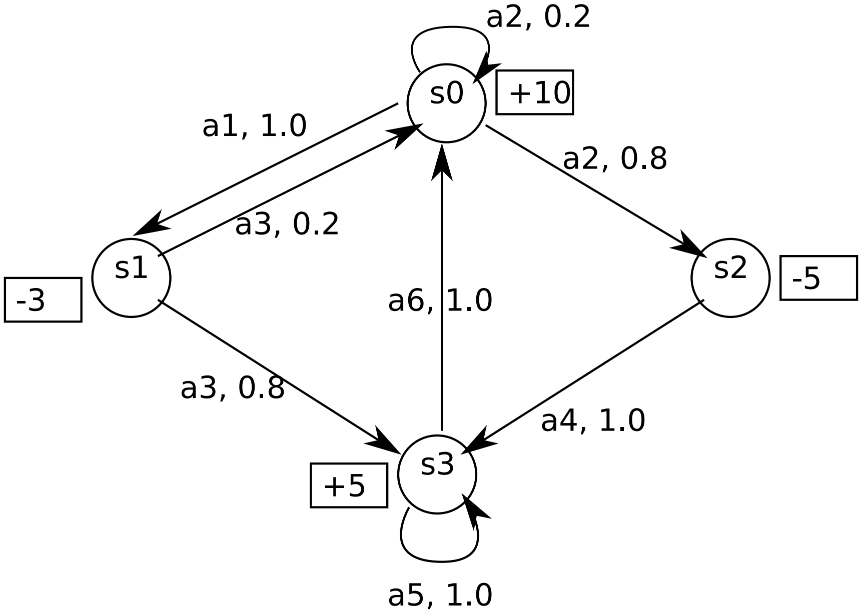
$P(F_{C2}|O)$=P(m1|O)+P(m3|O)+P(m4|O)+P(m5|O) =α' 36432

$P(F_{D1}|O)$=P(m2)+P(m4)+P(m5) =α' 19152

Explications : Il reste à découvrir : 25 cases, 9 fosses et $30 - 5 - 9 = 16$ cases sans fosse. Dans les calculs pour $P(m1|O)$ à $P(m5|O)$, les numérateurs indiquent le nombre cases restantes avec fosse (9, 8, 7, 6) ou sans fosses (16, 15) ; le dénominateur indique le nombre de cases restantes (25, 24, 23, 23). Comme il y a 5 modèles possibles, on trouve α qui normalise $(P(m1|O) + P(m2|O) + P(m3|O) + P(m4|O) + P(m5|O)) = 1.0$.

3 Processus décisionnels markoviens (MDP) (9 points)

Soit le processus de décision markovien ci-bas. Chaque transition est étiquetée d'une action (a_x) et d'une probabilité de transition. La récompense de chaque état est spécifiée dans un petit rectangle.



(a) Dans le tableau ci-dessous, indiquez la valeur des états à la fin de chacune des trois itérations de l'algorithme d'itération par valeurs. Un facteur d'escompte (*discount factor*) de $\gamma = 0.9$ est utilisé. Toutes les valeurs sont initialisées à zéro (itération 0). [3 points]

(b) Dans la dernière colonne du tableau ci-dessous, indiquez la politique extraite en utilisant les valeurs obtenues immédiatement après la deuxième itération. [2 points]

État	Itér.#0	Itération #1	Itération #2	Itération #3	π
s_0	0	+10	8.2	12.16	a1
s_1	0	-3	2.4	8.56	a3
s_2	0	-5	-0.5	7.6	a4
s_3	0	+5	14.0	17.6	a5

(c) Dans l'équation de Bellman des MDPs, à quoi sert le facteur d'escompte (*discount factor*) γ ? [2 points]

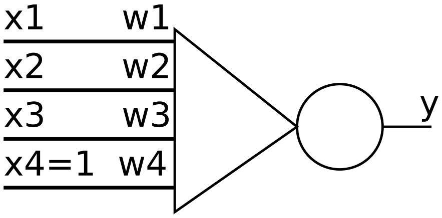
Le facteur d'escompte sert à pondérer les récompenses immédiates et futures. On utilise généralement $0 \leq \gamma < 1$. La pondération est entre autre nécessaire lorsque l'horizon de temps est infini.

(d) ★ Imaginez qu'on ajoute la règle suivante dans le jeu de serpents et échelles du TP4 : **un joueur peut exécuter l'action de lancer les deux dés (a_{DD}) au maximum N fois**. Expliquez comment vous adapteriez la politique #1, la politique permettant d'atteindre la dernière case le plus rapidement possible. [2 points]

Il suffit de modifier la représentation d'un état. Dans le TP4 original, un état est simplement une position. Dans la version modifiée, un état est maintenant une paire $(position, X)$ où X est le nombre de lancé de deux dés restant. L'état initial est $(0, N)$. Lorsqu'on applique l'action a_{DD} dans un état $s = (x, n)$, on bascule dans un ensemble d'états $s = (x', n-1)$.

4 Réseaux de neurones artificiels [7 points]

Soit le réseau de neurones suivant et les données d'entraînement suivantes.



	Échantillons			
x_1	1	2	1	4
x_2	0	0	0	0
x_3	4	4	2	1
y	+1	+1	-1	-1

Fonction d'activation : $y = sign(\sum_{i=1}^{4}(w_i \cdot x_i))$

(a) Simulez une itération complète de l'algorithme d'apprentissage du perceptron en utilisant les données d'entraînement ci-haut. Le pas d'apprentissage est fixé à 0.2 et les poids initiaux sont : $w_1 = 0$, $w_2 = 0$, $w_3 = 0$ et $w_4 = 0$. [2 points]

Ici, on suppose que $sign(0) = 0$. Les supposions comme $sign(0) = +1$ ou $sign(0) = -1$ sont également acceptées.

Itération #1 / Échantillon #1 $(1,0,4) \rightarrow +1$:
- Sortie : $sign(0 \cdot 1 + 0 \cdot 0 + 0 \cdot 4 + 0 \cdot 1) = 0$
- Erreur : $(+1) - (0) = +1$
- Correction : $\Delta w = 0.2 \cdot +1 \cdot (1,0,4,1) = (0.2,0,0.8,0.2)$
- Poids : $(0.2,0,0.8,0.2)$

Itération #1 / Échantillon #2 $(2,0,4) \rightarrow +1$:
- Sortie : $sign(0.2 \cdot 2 + 0 \cdot 0 + 0.8 \cdot 4 + 0.2 \cdot 1) = 1$
- Erreur : $(+1) - (+1) = 0$
- Correction : $\Delta w = 0.2 \cdot 0 \cdot (2,0,4,1) = (0,0,0,0)$
- Poids : $(0.2,0,0.8,0.2)$

Itération #1 / Échantillon #3 $(1,0,2) \rightarrow +1$:
- Sortie : $sign(0.2 \cdot 1 + 0 \cdot 0.8 + 0.2 \cdot 2 + 0.2 \cdot 1) = 1$
- Erreur : $(-1) - (+1) = -2$
- Correction : $\Delta w = 0.2 \cdot -2 \cdot (1,0,2,1) = (-0.4,0,-0.8,-0.4)$
- Poids : $(-0.2,0,0,-0.2)$

Itération #1 / Échantillon #4 $(4,0,1) \rightarrow +1$:
- Sortie : $sign(-0.2 \cdot 4 + 0 \cdot 0 + 0.0 \cdot 1 + -0.2 \cdot 1) = -1$
- Erreur : $(-1) - (-1) = 0$
- Correction : $\Delta w = 0.2 \cdot 0 \cdot (1,0,2,1) = (0,0,0,0)$
- Poids : $(-0.2,0,0,-0.2)$

(b) Est-ce que ce neurone artificiel est capable d'apprendre correctement le jeu de données d'entraînement ci-haut ? Justifiez votre réponse. [1 point]

Oui.
Pour que ce neuronne (qui est un perceptron) puisse apprendre les données d'entraînement, ces données doivent être linéairement séparables. C'est le cas ici. Par exemple, il suffit possible de séparer les données d'entraînement en considérant x_3. Exemple de poids : $w_1 = 0$, $w_2 = 0$, $w_3 = 1$ et $w_4 = -3$.

(c) Nommez et expliquez brièvement deux faiblesses des réseaux de neurones ? [2 points]

1. Sur-apprentissage. Lorsque mal paramétré, un réseau de neurones artificiels peut sur-apprendre, c'est-à-dire qu'il estime trop bien la relation entre les entrées et sortie des exemples d'entraînement, et ce, sans être capable de généraliser (extrapôler) la relation. Cela peut arriver lorsqu'on fixe un trop grand nombre de neurones dans la ou les couches cachées.

2. Fonctionnement difficilement explicable. Il peut être difficile de comprendre pourquoi le réseau de neurones fonctionne. Pourquoi les poids obtenus après la phase d'apprentissage donne le bon résultat ? Il peut être difficile d'expliquer les résultats obtenus ou de prouver théoriquement (plutôt qu'empiriquement) son fonctionnement correct.

3. Non infaillible. Le réseau peut se comporter incorrectement lorsqu'on lui présente de nouvelles données trop éloignées des données d'entraînement. Pour certaines applications, il peut être difficile d'avoir un jeu de données d'entraînement couvrant bien l'ensemble des données possibles.

(d) Les réseaux de neurones artificiels peuvent être appliqués à diverses problématiques. Expliquez brièvement comment vous utiliseriez un réseau de neurones artificiels pour faire de la reconnaissance optique de caractères. Indiquez ce que seraient le(s) entrée(s) et le(s) sortie(s). [2 points]

Entrées : matrices de pixels (un neurone d'entrée par pixel). Les valeurs peuvent être des bits (0 ou 1), des niveaux de gris (0.0 à 1.0, ou 0 à 255), etc.
Une couche cachée.
Sortie : il y a un neurone de sortie par caractère possible. Si on veut classifier les lettres de l'alphabet, on aura 26 neurones de sortie.
Fonctionnement : on entraîne le réseau sur des échantillons de caractères. Il faut généralement fournir plusieurs échantillons pour chaque caractère. Une fois entraîné, on numérise un texte et le découpe lettre par lettre. Pour chaque lettre à reconnaître, on envoie la valeur des pixels dans le réseau et on prend le neurone de sortie ayant la plus grande valeur d'activation pour déterminer quelle lettre a été reconnue.
Cette technique peut être améliorer de diverses façons. Par exemple, lorsqu'il y a des ambiguïtés sur plusieurs lettres consécutives, on peut recourir à un dictionnaire pour proposer le mot le plus probable.

5 Questions générales [4 points]

(a) En IA, il existe deux grandes familles d'approches : connexionnistes et symboliques. Expliquez brièvement la différence entre ces deux familles. [2 points]

Symboliques : famille d'approches, généralement plus «pragmatiques», qui consistent à modéliser des problèmes et situations à l'aide de symboles. Ensuite, problèmes sont résolus à l'aide d'algorithmes qui raisonnent sur ces symboles. Exemples : résolution de problèmes dans des espaces d'états, CSP, MDP, raisonnement probabiliste, etc. Les approches symboliques ont besoin d'experts pour spécifier les symboles.
Connexionnistes : famille d'approches, souvent inspirées de la biologie ou de la nature, qui consistent à interconnecter de nombreuses petites unités simples dans le but d'espérer l'émergence de capacités plus complexes ou plus «intelligentes» sans que celles-ci ne soient explicitement programmées. Exemples : réseaux de neurones artificiels, algorithme génétiques, etc.

(b) Expliquez brièvement ce qu'est un réseau bayésien dynamique. Dans quelles circonstances l'utilise-t-on ? [1 point]

Un réseau bayésien (RB) sert à modéliser une sitation à l'aide d'un graphe où les sommets sont des variables aléatoires et les arêtes des relations de dépendance. Un RBD généralise un RB afin de modéliser des situations qui évoluent dans le temps. Pour un RB ayant ensemble de variables aléatoires $\{X^1, X^2, ..., X^n\}$, son RBD équivalent contiendra ces variables aléatoires $\{X_t^1, X_t^2, ..., X_t^n\}$ pour chaque instant t. Le temps t doit être discret. Une ou plusieurs variables aléatoires X^i dépendent des états précédents dans la même variable. Par exemple X_t^1 peut dépendre de X_{t-1}^1.

(c) Sous quelle(s) condition(s) l'algorithme A* trouvent-il une solution optimale ? [1 point]

— **Heuristique admissible**.
— La solution doit exister.
— Espace d'états fini.
— ...

6 Résolution de problème : Jeu démineur [4 points]

Le jeu Démineur (*Minesweeper*) est un jeu de réflexion dans lequel le joueur doit localiser des mines dans une grille. Ce jeu était autrefois inclus avec le système d'exploitation Microsoft Windows. Plusieurs implémentations libres existent, dont Gnome Mines illustrée à droite.

Les mines sont initialement distribuées aléatoirement sur la grille. À priori, le joueur connaît seulement le nombre de mines dans la grille. Lorsque le joueur clique sur une case de la grille, il y a deux résultats possibles. Si la case contient une mine, celle-ci explose et le joueur perd la partie. Si la case ne contient pas de mine, la case est découverte et un chiffre indique le nombre de mines contenus dans les 8 cases voisines (le zéro est généralement pas indiqué). Le jeu se termine lorsque le joueur a cliqué sur toutes les cases sans mine, c'est-à-dire que les cases restantes sont celles où les mines ont été placées. Le but du jeu est de terminer le plus rapidement possible.

(a) Pour chacune des six dimensions suivantes, encerclez la caractéristique représentant mieux l'environnement du jeu Démineur ? Si vous hésitez entre deux caractéristiques, vous pouvez nuancer votre pensée en écrivant une courte explication. Elle sera considérée lors de la correction. [2 points]

1. Observabilité **partielle** vs totale
2. **Déterministe** vs stochastique
3. Épisodique vs séquentielle*

*Les 2 sont acceptés, pourvu que ce soit cohérent avec (b).

4. **Statique** vs dynamique
5. **Discret** vs continu
6. **Agent unique** vs multi-agent

(b) En vous basant sur les techniques d'IA vue dans le cours, identifiez celle qui vous semble la plus appropriée pour implémenter un joueur artificiel pour le jeu de Démineur. Expliquez brièvement votre démarche. [2 points]

Approche : raisonnement probabiliste.
Pour résoudre ce problème, on peut calculer la probabilité qu'une mine se trouve sous chaque case non découverte. Cela ressemble à la façon de décider dans le monde du Wumpus sous incertitude.

/*********** Fin de l'examen ! **********/