**DS R4.04- 2025**

**Durée : 1h30**

Accès autorisé aux supports de cours, TDs et corrections et uniquement au site geogebra.org

**Exercice 1 : Entreprise d’inspection (8 points)**

Une entreprise a deux catégories d’inspecteurs notés 1 et 2. On veut les affecter à une opération de contrôle de qualité. L’entreprise souhaite inspecter jusqu’à 2800 pièces par journée de travail qui dure 8 heures. Les inspecteurs de 1ère catégorie peuvent inspecter des pièces à raison de 25 par heure avec une précision de 98%, quant aux autres ils vérifient 15 pièces heures avec une précision de 95%. Le gain de l’entreprise pour un inspecteur de 1ère catégorie est de 100 €/ heure alors que celui de la seconde est de 70 €/ heure. A chaque erreur commise par un inspecteur, le coût de perte est de 50 € pour la compagnie. Cette dernière a 11 et 19 inspecteurs respectivement de 1ère et de 2nde catégorie. L’entreprise souhaite affecter de manière optimale ces inspecteurs lui permettant de maximiser ses bénéfices.

1. Ecrivez un programme linéaire ayant pour objectif de maximiser le bénéfice journalier de l’entreprise.

**Soit :**

* x : inspecteur 1
* y : inspecteur 2

**Contraintes de temps :**

* Inspecteur 1 : x ≤ 11

• inspecteur 2 : y ≤ 19

* x≥0,y≥0 (pas d’inspecteur négatif)

**Bénéfices = 200x + 120y**

1. Résoudre ce problème graphiquement à l’aide du site geogebra.org. Il faut faire un imprime écran de l’espace de solutions réalisables et le délimiter.
2. 
3. Donner la solution optimale si elle existe.

Oui

d) L’entreprise souhaite impliquer au minimum 10 inspecteurs de la catégorie 2. Que change cette contrainte dans la formulation de votre programme linéaire ?

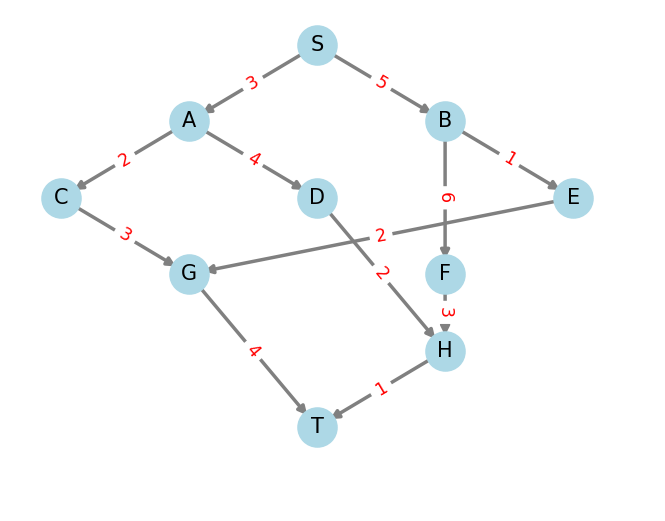
**On ajoute :**

* y ≥ 10 (minimum 10 inspecteurs de la catégorie 2 impliqués)

e) Représenter le nouvel espace de solutions réalisables et donner la solution optimale si elle existe.

**Exercice 2 : A star (5 points)**

Soit le graphe G suivant :



Et l’heuristique h(n) estimant la distance d’un nœud donné vers le nœud T est la suivante :

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nœud** | **S** | **A** | **B** | **C** | **D** | **E** | **F** | **G** | **H** | **T** |
| **Heuristique (h)** | 10 | 7 | 6 | 5 | 4 | 3 | 4 | 2 | 1 | 0 |

Le nœud source est S et le nœud d’arrivée est T. Le coût est indiqué sur chaque arc du graphe et h(n) représente la fonction heuristique associée à chaque nœud du graphe.

1. Simulez l’exécution de l’algorithme A\* et donnez l’ensemble d’itérations avec le contenu des listes « Open » et « Closed » à chaque fois.

Initialisation

• Open List : {(A, f=10, g=0, h=10)}

• Closed List : {}

|  |  |
| --- | --- |
| Nœuds à explorer - étapes | Nœuds visités et fermés |
| (S, 10 , vide) | - |
| (A, 2+7, S), (B, 5+6, S) | (S, 10, vide) |
| (C, 3+2+5, A) ,(D, 4+3+4,A), (B, 5+6, S) | (A, 2+7, S) |
| (G,3+2+3+2), (B, 5+6, S) | (C, 3+2+5, A) |
| (T,3+2+3+4+0,G), (B, 5+6, S) | (G,3+2+3+2) |

1. Donnez la solution finale de l’algorithme A\*

**La solution finale est S 🡪 A🡪 C 🡪 G 🡪 T**

1. L’heuristique h(n) est-elle admissible ? justifiez.

**Un heuristique est admissible si la valeur de l’heuristique à partir d’un point donné X du graphe est toujours inférieure à tous les chemins possibles de X 0 à la destination. Dans notre cas elle vérifie bien la condition, l’heuristique est ainsi admissible.**

**Exercice 3 : Souscription à une Salle de Sport (7 points)**

Voici un exemple de jeu de données sur le profilage de personnes ayant souscrit ou non à une salle de sport. Ce jeu de données contient des informations sur différents facteurs qui pourraient influencer la décision d’une personne à souscrire à un abonnement.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Âge** | **Genre** | **Revenus** | **Fréquence d'exercice** | **Objectif** | **Souscrire** |
| 1 | Jeune | Homme | Élevé | Rare | Perte de poids | Oui |
| 2 | Jeune | Femme | Modeste | Régulier | Musculation | Oui |
| 3 | Adulte | Homme | Moyen | Régulier | Santé générale | Non |
| 4 | Senior | Femme | Modeste | Rare | Santé générale | Non |
| 5 | Adulte | Femme | Élevé | Régulier | Perte de poids | Oui |
| 6 | Jeune | Homme | Modeste | Rare | Musculation | Non |
| 7 | Senior | Homme | Moyen | Régulier | Santé générale | Oui |
| 8 | Adulte | Femme | Modeste | Régulier | Perte de poids | Non |
| 9 | Jeune | Femme | Élevé | Rare | Musculation | Oui |
| 10 | Senior | Femme | Moyen | Rare | Santé générale | Non |
| 11 | Adulte | Homme | Élevé | Régulier | Musculation | Oui |
| 12 | Jeune | Femme | Modeste | Régulier | Perte de poids | Oui |
| 13 | Senior | Homme | Modeste | Rare | Santé générale | Non |
| 14 | Adulte | Femme | Moyen | Régulier | Musculation | Oui |

#### 1. Calculer l'entropie initiale pour la variable cible **"Souscrire"** (Oui/Non). Donner la formule et résultat de calcul.

H(S) = - ∑ (p\_i \* log2(p\_i)) où p\_i est la proportion d'instances appartenant à la classe i.  
 - Nombre de 'Oui' : 8

- Nombre de 'Non' : 6

2. Calculer la proportion de chaque classe :

- p(Oui) = 8/14 ≈ 0.57

- p(Non) = 6/14 ≈ 0.43

3. Appliquer la formule de l'entropie :

H(S) = - (0.57 \* log2(0.57) + 0.43 \* log2(0.43)) ≈ 0.98

#### 2. Calculer le gain d'information pour chaque caractéristique possible :

* **Âge** (Jeune, Adulte, Senior)
* **Genre** (Homme, Femme)
* **Revenus** (Élevé, Moyen, Modeste)
* **Fréquence d'exercice** (Rare, Régulier)
* **Objectif** (Perte de poids, Musculation, Santé générale)

### Gain d'Information pour la Caractéristique "Age"

Entropie conditionnelle pour chaque valeur de 'Age' :  
- Jeune :  
 - Instances : 1 (Non), 4 (Oui)  
 - Entropie : H(Jeune) = - ((1/5) \* log2(1/5) + (4/5) \* log2(4/5)) ≈ 0,72  
  
- Adulte :  
 - Instances : 2 (Non), 3 (Oui)  
 - Entropie : H(Adulte) = - ((2/5) \* log2(2/5) + (3/5) \* log2(3/5)) ≈ 0.97  
  
- Senior :  
 - Instances : 3 (Non), 1 (Oui)  
 - Entropie : H(Senior) = - ((1/4) \* log2(1/4) + (3/4) \* log2(3/4)) ≈ 0.81  
  
Entropie pondérée :  
H(Age) = (5/14) \* 0.72 + (5/14) \* 0.97 + (4/14) \* 0.81 ≈ 0.84

Gain d'information :  
IG(Age) = H(S) - H(Age) = 0.98 - 0.84 = 0.14

### Gain d'Information pour la Caractéristique "Genre"

Entropie conditionnelle pour chaque valeur de Genre:  
- Homme :  
 - Instances : 3 (Non), 3 (Oui)  
 - Entropie : H(Elevé) = - ((3/3) \* log2(3/3)+ (3/3) \* log2(3/3)) = 0  
  
- Femme :  
 - Instances : 3 (Non), 5 (Oui)  
 - Entropie : H(Moyen) = - ((3/8) \* log2(3/8) + (5/8) \* log2(5/8)) ≈ 0,95   
  
Entropie pondérée :  
H(Genre) = (6/14) \* 0 + (8/14) \* 0.95 ≈ 0,54  
  
Gain d'information :  
IG(Genre) = H(S) - H(Genre) = 0.98 - 0.54 = 0.44

### Gain d'Information pour la Caractéristique "Revenus"

Entropie conditionnelle pour chaque valeur de 'Revenus' :  
- Elevé :  
 - Instances : 0 (Non), 4 (Oui)  
 - Entropie : H(Elevé) = - ((0/4) \* log2(0/4) + (4/4) \* log2(4/4)) = 0  
  
- Moyen :  
 - Instances : 2 (Non), 2 (Oui)  
 - Entropie : H(Moyen) = - ((2/4) \* log2(2/4) + (2/4) \* log2(2/4)) = 1  
  
- Modeste :  
 - Instances : 4 (Non), 2 (Oui)  
 - Entropie : H(Modeste) = - ((4/6) \* log2(4/6) + (2/6) \* log2(2/6)) ≈ 0.92  
  
Entropie pondérée :  
H(Revenus) = (4/14) \* 0 + (4/14) \* 1 + (6/14) \* 0.92 ≈ 0.68  
  
Gain d'information :  
IG(Revenus) = H(S) - H(Revenus) = 0.98 - 0.68 = 0.3

### Gain d'Information pour la Caractéristique "Fréquence d'exercice"

Entropie conditionnelle pour chaque valeur de **Fréquence d'exercice**:  
- Rare:  
 - Entropie : H(Rare) ≈ 0,92  
  
- Régulier :  
 - Entropie : H(Régulier) ≈ 0.81  
  
Entropie pondérée :  
H(**Fréquence d'exercice**) = (6/14) \* 0,92 + (8/14) \* 0,81 ≈ 0.86

Gain d'information :  
IG(**Fréquence d'exercice**) = H(S) - H(**Fréquence d'exercice**) = 0.98 - 0.84 = 0.12

### Gain d'Information pour la Caractéristique " **Objectif** "

Entropie conditionnelle pour chaque valeur de **Objectif**:  
- Perte de poids:  
 - Entropie : H(Perte de poids) ≈ 0,81  
  
- Musculation :  
 - Entropie : H(Musculation) ≈ 0.72  
  
- Santé générale:  
 - Entropie : H(Santé générale) ≈ 0.72  
  
Entropie pondérée :  
H(**Objectif**) = (4/14) \* 0,81 + (5/14) \* 0,72+ (5/14) \* 0,72 ≈ 0.74

Gain d'information :  
IG(**Objectif**) = H(S) - H(**Objectif**) = 0.98 - 0.74 = 0.24

#### Quelle caractéristique offre le plus grand gain d'information ?

C’est la fréquence d’exercice.

#### 4. Construire l'arbre de décision :

* Donner le code du programme python permettant de construire l'arbre de décision en question ainsi que la méthode de calcul de l’entropie utilisée.
* Donner l’arbre de décision résultat.

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.tree import DecisionTreeClassifier

from sklearn import tree

import matplotlib.pyplot as plt

from sklearn.tree import export\_text

# Données

data = {

    'Age': ['Jeune','Jeune','Adulte','Senior','Adulte','Jeune','Senior','Adulte','Jeune','Senior','Adulte','Jeune','Senior','Adulte'],

    'Genre': ['Homme','Femme','Homme','Femme','Femme','Homme','Homme','Femme','Femme','Femme','Homme','Femme','Homme','Femme'],

    'Revenus': ['Élevé','Modeste','Moyen','Modeste','Élevé','Modeste','Moyen','Modeste','Élevé','Moyen','Élevé','Modeste','Modeste','Moyen'],

    'Fréquence exercice': ['Rare','Régulier','Régulier','Rare','Régulier','Rare','Régulier','Régulier','Rare','Rare','Régulier','Régulier','Rare','Régulier'],

    "Objectif": ['Perte de poids','Musculation','Santé générale','Santé générale','Perte de poids','Musculation','Santé générale','Perte de poids','Musculation','Santé générale','Musculation','Perte de poids','Santé générale','Musculation'],

    "Souscrire" : ['Oui','Oui','Non','Non','Oui','Non','Oui','Non','Oui','Non','Oui','Oui','Non','Oui']

}

# Conversion des données en DataFrame

df = pd.DataFrame(data)

# Encodage des variables catégoriques

df\_encoded = pd.get\_dummies(df.drop("Souscrire", axis=1))

target = df["Souscrire"].apply(lambda x: 1 if x == "Oui" else 0)

# Division des données en ensembles d'entraînement et de test

X\_train, X\_test, y\_train, y\_test = train\_test\_split(df\_encoded, target, test\_size=0.2, random\_state=42)

# Création de l'arbre de décision

model = DecisionTreeClassifier(criterion="entropy", max\_depth=3, random\_state=42)

model.fit(X\_train, y\_train)

# Affichage de l'arbre de décision

plt.figure(figsize=(10, 10))

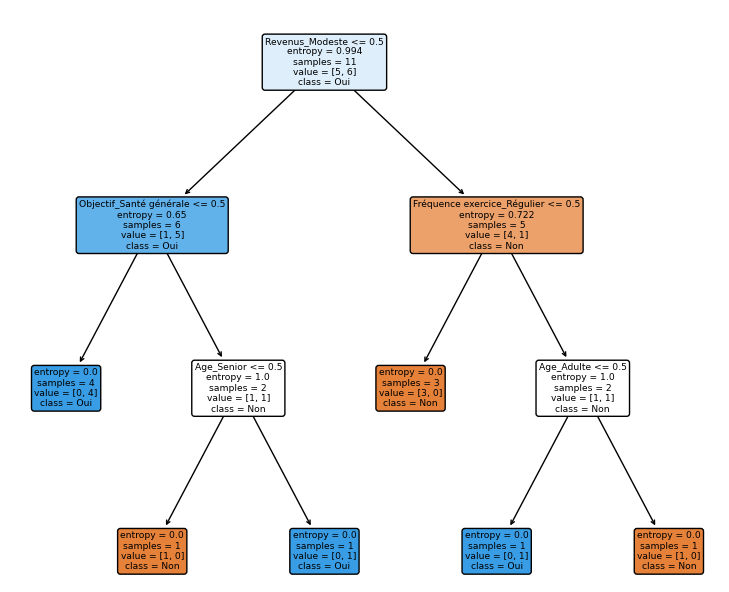
tree.plot\_tree(model, feature\_names=df\_encoded.columns, class\_names=["Non", "Oui"], filled=True, rounded=True)

plt.show()

# Affichage du texte de l'arbre de décision

tree\_rules = export\_text(model, feature\_names=list(df\_encoded.columns))

print(tree\_rules)



#### 8. Quelle décision pour ces nouvelles données ?

* Utiliser l'arbre de décision généré dans la question précédente pour prédire si les clients suivants souscriraient à un abonnement :
  + **Client 1** : Âge = Jeune, Genre = Femme, Revenus = Modeste, Fréquence d'exercice = Régulier, Objectif = Perte de poids
* OUI
  + **Client 2** : Âge = Senior, Genre = Homme, Revenus = Élevé, Fréquence d'exercice = Rare, Objectif = Santé générale
* OUI