Rapport du mini projet

Kmar Ben Hamad

1ATA

2025/2026

1.Introduction :

Ce mini projet s’inscrit dans le cadre de l’apprentissage supervisé. L’objectif est d’étudier les liens entre la consommation de café et la santé. Le développement a été réalisé en utilisant le langage Python ainsi que les bibliothèques fournies par le module IA01 développé dans le cadre de ce cours.
Ce rapport présente les différentes étapes du projet, les choix techniques effectués, ainsi que les résultats obtenus.

2.Analyse des données :

Lors de l'analyse initiale des variables disponibles, certaines ont été identifiées comme non pertinentes pour les objectifs de cette étude, tandis que d'autres se sont révélées essentielles. Cette section présente les choix effectués en matière de sélection de variables, ainsi que les justifications associées.

2.1. Variables exclues :

Afin de limiter les biais potentiels, les variables suivantes ont été exclues du modèle

- **Pays**

Bien que la consommation de café et la tolérance à la caféine puissent varier selon les pays (par exemple, une différence notable entre l’Italie et d'autres régions), ces variables pourraient introduire une mauvaise généralisation non souhaitée dans le cadre de notre prédiction. De plus, leur pertinence directe pour notre problématique reste limitée.

2.2. Variables retenues :

En revanche, les variables retenues sont celles dont l'impact potentiel sur la santé ou la consommation de caféine est significatif :

- **Age :** il constitue un facteur déterminant dans l'évaluation de la santé générale et peut influencer la tolérance à la caféine.
- **Sexe** : peut influencer certains marqueurs de santé en raison des différences hormonales entre homme et femme.
- **Nombre de verres de café (cafe_verre)** : Ce nombre est directement lié à notre objectif d'étude.
- **Quantité de caféine (cafeine_mg)** : permet de prendre en compte la caféine issue d'autres sources que le café (thé, chocolat, soda, etc.).
- **Qualité du sommeil** : Un aspect fortement influencé par la consommation de caféine.
- **La durée de sommeil** elle constitue un indicateur essentiel de la santé générale. De plus, la consommation de caféine est souvent liée à une réduction du temps de sommeil.
- **Stress** : variable corrélée à la consommation de caféine et à l'état de santé général.
- **Indice de masse corporelle (IMC)**, **rythme cardiaque**, **activité physique,** : facteurs ayant une influence globale sur la santé.
- **Profession** : Le type de profession peut avoir une influence significative sur l'état de santé, notamment en raison des conditions de travail, du niveau de stress, et de facteurs socio-économiques.
- **Consommation régulière de cigarette et alcool** : en raison de leurs effets néfastes bien documentés sur la santé physique et mentale.
- Ces choix visent à garantir une modélisation cohérente, tout en minimisant les risques de biais ou de surinterprétation.

3. Les étapes d'exécution du projet :

3.1. Préparation des données :

3.1.1. Attributs manquants

Après une inspection du document *X_train*, certaines variables se sont révélées manquantes pour certains individus (pour **sexe**, environ 3,3 % des individus n'ont pas soumis leur genre).
Pour cette raison, l'enlèvement de ces individus est primordial.

À ce stade, l'ensemble des données restantes est censé être complet. Cependant, en y regardant de plus près, certains individus ont un âge de 0. Cette valeur n'est pas raisonnable et peut également affecter notre étude. Donc, Il est donc nécessaire de les supprimer.

```
data=lecture_csv(".\data2\X_train.csv")
data_test=lecture_csv(".\data2\X_test.csv")
yt=lecture_csv(".\data2\y_train.csv")
data1=[x for x in data if est_complet(x)]
y_train=[yt[i]["probleme_sante"] for i in range (len(yt)) if
est_complet(data[i])]
y_train=[y_train[i] for i in range (len(y_train)) if data1[i]['age']!='0' ]
data1=[x for x in data1 if x['age']!='0']
```

3.1.2. Encodage des variables qualitatives :

On constate qu'il existe deux types de variables qualitatives ordinales et nominales.

- **Variables qualitatives ordinales :**
  - **Sommeil_qualite**

Excellente	Bonne	Passable	Mauvaise
4	3	2	1

  - **Niveau_stress**

Faible	Moyen	Haut
1	2	3

- **Variables qualitatives nominales :**
  - **Profession**

Vu que quatre catégories différents ont été disposées pour cette caractéristique, on utilisera un encode *one-hot* en suivant la convention suivante.

- [1, 0, 0, 0, 0] : “Sante”
- [0, 1, 0, 0, 0] : “Bureau”
- [0, 0, 1, 0, 0] : “Service”
- [0, 0, 0, 1, 0] : “Etude”
- [0, 0, 0, 0, 1] : “Autre”

- **Cigarette/ Alcool/ sexe**

Ces trois variables ne comportent que deux modalités.
Dans ce cas, un encodage binaire simple est suffisant. Ce type de codage n’introduit aucun biais, car il n’existe ni hiérarchie ni ordre implicite entre les modalités.

- 1 : oui
- 0 : non
- 1 : Femme
- 0 : Homme

```
x_train=[]
for x in data1:
    l=[]
    for i in carac:
        if i == 'sommeil_qualite':
            if x[i]=='Excellente':
                l.append(4)
            elif x[i]=='Bonne':
                l.append(3)
            elif x[i]=='Passable':
                l.append(2)
            elif x[i] == "Mauvaise":
                l.append(1)
        elif i=='profession':
            if x[i] == "Sante":
                l += [1, 0, 0, 0,0]
            elif x[i] == "Bureau":
                l += [0, 1, 0, 0,0]
            elif x[i] == "Service":
                l += [0, 0, 1, 0,0]
            elif  x[i] == "Etude":
                l += [0, 0, 0, 1,0]
```

```
            else:
                l+=[0,0,0,0,1]
        elif i== 'sexe':
            if x[i]== 'Femme':
                l.append(1)
            else:
                l.append(0)
        elif i == 'niveau_stress':
            if x[i]=='Faible':
                l.append(1)
            elif x[i]=='Moyen':
                l.append(2)
            elif x[i] == "Haut":
                l.append(3)
        elif i == 'cigarette':
            if x[i]=='Oui':
                l.append(1)
            else:
                l.append(0)
        elif i == "alcool":
            if x[i]=='Oui':
                l.append(1)
            else:
                l.append(0)
        else:
            l.append(float(x[i]))
    x_train.append(l)
```

3.2. Choix du modèle d'apprentissage

Dans le cadre de ce projet, le modèle arbre de décision a été jugé plus pertinent pour cette étude que le modèle **k-plus proches voisins** (KPPV) pour plusieurs raisons **:**

- **Robustesse aux types de données** : l'arbre de décision peut gérer efficacement des variables de nature mixte (quantitatives, qualitatives nominales et ordinales), ce qui correspond bien à la diversité des données disponibles.
- **Prétraitement minimal** : contrairement à KPPV, l'arbre de décision ne nécessite pas la normalisation ou la mise à l'échelle des variables, ce qui simplifie la phase de préparation des données.

3.3. Validation et choix de la profondeur :

Vu la disposition d'une seule base de donnée (x_train, y_train), il est judicieux de mettre de côté un ensemble de validation (20% de notre data. Afin d'assurer une estimation fiable, on opter pour la méthode de validation croisée (pour les 80% restantes) dans le but d'entrainer notre modèle et trouver la meilleure profondeur. Le principal inconvénient observé concerne le temps d'exécution, vu l'utilisation de l'arbre de décision.

```
x_t, y_t, x_valf, y_valf = partition_train_val(x_train, y_train, 1 / 5)
arbre_test=arbre_train(x_t,y_t)
```

3.4. Choix de la profondeur :

Dans le cadre d'évaluation des performances du modelés, le taux d'erreur a été calculer pour différentes profondeurs d'arbre allant de 0 à 20.

Les résultats montrent que la profondeur optimale celle qui minimise le taux d'erreur est égale à p=3 avec une taux d'erreur égale à 0.001. Cela suggère que le modèle parvient à un excellent compromis entre biais et variance.

```
k=5
xk,yk = partition_val_croisee(x_t,y_t, 5)

prof = list(range(11)) + [float("inf")]
erreur_cv = [0] * len(prof)

for i in range(k):
    X_val, y_val = xk[i], yk[i]
    X_train, y_train = [], []
    for j in range(k):
        if j != i:
            X_train += xk[j]
            y_train += yk[j]
    arbre = arbre_train(X_train, y_train)
    for j, p in enumerate(prof):
        y_pred_val = arbre_pred(X_val, arbre, max_prof=p)
        erreur_cv[j] += taux_erreur(y_val, y_pred_val) / k
```

Taux d'erreur pour prof=0 ; e=0.406

Taux d'erreur pour prof=1 ; e=0.142

Taux d'erreur pour prof=2 ; e=0.061

Taux d'erreur pour prof=3 ; e=0.001

Taux d'erreur pour prof=4 ; e=0.001

Taux d'erreur pour prof=5 ; e=0.002

Taux d'erreur pour prof=6 ; e=0.002

Taux d'erreur pour prof=7 ; e=0.002

Taux d'erreur pour prof=8 ; e=0.002

Taux d'erreur pour prof=9 ; e=0.002

Taux d'erreur pour prof=10 ; e=0.002

Taux d'erreur pour prof=inf ; e=0.002

Profondeur optimale : prof = 3

3.5. Validation avec l'ensemble de validation :

Une fois la profondeur optimale déterminée (p = 3) à l'aide de la validation croisée, le modèle a été évalué sur l'ensemble de validation afin de tester sa performance sur des données non vues. Avec cette configuration, le taux d'erreur obtenu sur l'ensemble de validation est de **0.0052** confirmant ainsi la bonne capacité de généralisation du modèle.

```
y_pred_valf=arbre_pred(x_valf,arbre_test,bestp_e)
print(taux_erreur(y_valf,y_pred_valf))
```

3.6. Construction y_pred pour x_test :

Après la préparation du x_test, on construit y_pred avec la profondeur optimale et on utilise ecriture_csv_projet(y_pred, fichier) pour générer le fichier y_test.csv.

4. conclusion

Ce mini-projet nous a permis de mettre en pratique les notions fondamentales de l'apprentissage supervisé à travers l'étude du lien entre la consommation de café et la santé.

Grâce à une analyse rigoureuse des données et un choix méthodique des variables pertinentes, nous avons pu concevoir un modèle d'arbre de décision à la fois simple et performant.

Les résultats obtenus démontrent qu'une profondeur de 3 permet d'atteindre un excellent compromis entre biais et variance, avec un taux d'erreur très faible (0.0052 sur l'ensemble de validation).

Au-delà des performances techniques, ce travail illustre l'importance du prétraitement des données et de la sélection judicieuse des variables dans la réussite d'un modèle prédictif.