TP1 : Importance de la préparation des données pour le développement d'un outil de prédiction de la survie des patients souffrant d'insuffisance cardiaque (2H)

Contexte clinique

Les maladies cardiovasculaires, responsables du plus grand nombre de décès à l'échelle mondiale, entraînent environ 17,9 millions de pertes de vie annuelles, représentant ainsi 31 % de l'ensemble des décès dans le monde. L'insuffisance cardiaque, fréquemment associée à ces affections, constitue un événement courant. Les maladies cardiovasculaires peuvent être prévenues en agissant sur les facteurs de risque comportementaux tels que la consommation d'alcool, le tabagisme, une alimentation inadéquate, l'obésité et l'inactivité physique, et identifier précocement les patients exposés à un risque de décès élevé permettrait donc d'adapter leur prise en charge.

Le jeu de données étudié dans ce TP contient des caractéristiques cliniques, comportementales et démographiques qui ont été montrées comme associées à la mortalité liée à l'insuffisance cardiaque (voir Tableau 1). Ce jeu nous servira à mettre en place un outil de prédiction (basé sur une modélisation par forêt aléatoire) de la survie des patients souffrant d'insuffisance cardiaque.

Tableau 1 Description des variables du jeu de données

Variable	Description
age	Age du patient
anemia	Diminution des globules rouges ou de l'hémoglobine
creatinine_phosphokinase	Taux de l'enzyme CPK dans le sang (mcg/L)
diabetes	Présence de diabète
ejection_fraction	Pourcentage de sang quittant le cœur à chaque contraction
high_blood_pressure	Présence d'hypertension
platelets	Plaquettes dans le sang (kiloplaquettes/mL)
serum_creatinine	Taux de créatinine sérique dans le sang (mg/dL)
serum_sodium	Taux de sodium sérique dans le sang (mEq/L)
sex	Sexe du patient
smoking	Consommation de tabac
time	Période de suivi (jours)
DEATH_EVENT	Décès du patient pendant la période de suivi

I. Les ressources

Récupérer l'archive *TP1_gestion_donnees_manquantes.zip* sur Moodle. Elle contient :

- ***0_prise_en_main_dataset.py*** : le script à compléter qui vous permettra de répondre aux questions de la partie II.
- ***1_dev_modeles.py*** : le script à compléter qui vous permettra de répondre aux questions des parties III et IV.

- ***helping_functions.py*** : un ensemble de fonctions prêtes à l'emploi et à utiliser pour certaines questions de ce TP.
- ***heart_failure_dataset.csv*** : le jeu de données étudié.

II. Prise en main du jeu de données (1 heure)

Dans cette partie, il s'agit de se familiariser avec le jeu de données et de comprendre le contenu de chacune des variables. Cette étape nous servira aussi à juger de la qualité du jeu de données. Un ensemble de questions vous est proposé pour vous guider dans l'analyse. Répondez à ces questions dans votre rapport en incluant les graphiques, les nombres qui vous permettent d'y répondre (***0_prise_en_main_dataset.py***).

1. Combien d'observations contient le jeu de données ?
2. Préciser les variables du jeu de données ainsi que leurs types.
3. Quelle est la variable cible ? Combien de patients sont décédés ? Combien de patients ont survécu ?
4. Commenter le contenu de chacune des variables pour les patients décédés et pour les survivants en rapportant les graphes de visualisation (boxplot, histplot…) ou les valeurs qui vous ont permis de répondre.
    a. Est-ce que les proportions sont balancées ?
    b. Que peut-t 'on dire des distributions (dispersion, symétrie, variabilité, outliers, balancement…) ? Est-ce que les distributions sont similaires pour les deux groupes ?
    c. Est-ce que les valeurs observées vous semblent cohérentes avec vos connaissances sur ces variables ?
    d. Le jeu de données est-t 'il complet ? Combien d'observations sont complètes ? Quel est le pourcentage de données manquante par variable ?
    e. Les variables sont-t 'elles corrélées entre elles ou avec la cible ?

III. Développement d'un modèle en excluant les observations ayant des données manquantes (15 minutes)

Surprise ! Le jeu de données n'est pas complet. Ici, nous allons observer les performances qu'aurait un modèle de prédiction si nous utilisions une méthode simple de préparation des données : toutes les observations non complètes sont exclues (Modèle 0).

5. Suivez les instructions présentes dans ***1_dev_modeles.py*** (L57→L126)
    a. Combien reste-t-il de données après le nettoyage ?
6. Commenter la figure contenant les performances obtenues par le Random Forest lors de la validation croisée stratifiée.
    a. Quels sont les hyperparamètres qui ont été retenus au préalable ?
    b. Quelles sont les performances ? Rapporter les dans le tableau (Modèle 0)
    c. Commenter.

IV. Développement de modèles avec imputation des valeurs manquantes (30 minutes)

On ne vous la fait pas, vous aviez compris que ça ne serait pas super comme performances. Comment fait-t ’on pour mieux gérer ces valeurs manquantes ? Nous allons étudier plusieurs pistes et rapporter les différents résultats pour les comparer et les discuter dans la prochaine section.

1. Piste 1 : imputation des valeurs manquantes par la valeur médiane de la variable pour les variables numériques et par la valeur la plus fréquente pour les variables catégorielles.
    a. Suivez les instructions présentes dans ***1_dev_modeles.py*** (L128→L199)
    b. Commenter la figure contenant les performances obtenues par le Random Forest lors de la validation croisée stratifiée.
        i. Quels sont les hyperparamètres qui ont été retenus au préalable ?
        ii. Quelles sont les performances ? Rapporter les dans le tableau (Modèle 1)
        iii. Commenter ces résultats.

2. Piste 2 : imputation des valeurs manquantes en utilisant un K-plus proche voisins (k=10).
    a. Suivez les instructions présentes dans ***1_dev_modeles.py*** (L200→fin)
    b. Commenter la figure contenant les performances obtenues par le Random Forest lors de la validation croisée stratifiée.
        i. Quels sont les hyperparamètres qui ont été retenus au préalable ?
        ii. Quelles sont les performances ? Rapporter les dans le tableau (Modèle 2)
        iii. Commenter ces résultats.

V. Discussion (15 minutes)

Métrique	Modèle 0	Modèle 1	Modèle 2	Modèle Inconnu
concordance équilibrée				0.82 +/- 0.05
sensibilité				0.81 +/- 0.04
spécificité				0.83 +/- 0.08

1. Classer ces modèles en fonction de leurs performances.
2. Quelles sont leurs limites ?
3. A votre avis, pourquoi le modèle inconnu est meilleur que les autres ?

VI. Pour aller plus loin…

Ce TP a été construit à partir du jeu de données exploité par Chicco et al. dans leur étude : *Davide Chicco, Giuseppe Jurman: Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. BMC Medical Informatics and Decision Making 20, 16 (2020).*

Si vous avez du temps, vous pouvez lire la contribution initiale et comprendre quelles modifications du jeu de données ont été faites pour le « salir ». Ces auteurs avaient de la chance, leur jeu initial était bien propre ! Vous pouvez aussi comparer le jeu de données initial avec vos jeux de données une fois imputés.

https://www.kaggle.com/datasets/andrewmvd/heart-failure-clinical-data

Annexe 1 : Métriques dérivées de la matrice de confusion

Matrice de confusion

Auto\Ref	0	1
0	VN	FN
1	FP	VP

Vrai Positifs (VP) : nombre d'échantillons correctement classés comme «positif»
Vrai Négatifs (VN) : nombre d'échantillons correctement classés comme «négatif»
Faux Positifs (FP) : nombre d'échantillons classés comme « positif » au lieu de «négatif»
Faux Négatifs (FN) : nombre d'échantillons classés comme « négatif" au lieu de «positif»

Métriques utilisées dans le TP

$$S_e = \frac{VP}{VP + FN}$$

$$S_p = \frac{VN}{VN + FP}$$

$$B_a = \frac{S_e + S_p}{2}$$

Sensibilité (Se) mesure la capacité du modèle à détecter correctement les exemples positifs

Spécificité (Sp) mesure la capacité du modèle à détecter correctement les exemples négatifs.

Balanced Accuray (Ba), « concordance équilibrée **».** Elle représente la moyenne arithmétique de la sensibilité (capacité à détecter correctement les exemples positifs) et de la spécificité (capacité à détecter correctement les exemples négatifs), offrant une mesure plus équilibrée des performances du modèle, particulièrement en présence de déséquilibres entre les classes.