DiabetesUFO

ML simplon sur du dataset medical pour prévoir la maladie du diabéte; avec Anna, Olivier et Fidel :)

ML1

Dataset utilisé

Le dataset utilisé provient de la source: https://archive.ics.uci.edu/dataset/529/early+stage+diabetes+risk+prediction+dataset

Il contient des informations médicales et des symptômes rapportés par des patients, ainsi qu'un diagnostic final indiquant s'ils sont atteints de diabète ou non.

Attention: nous utilisons les données de `data/train_with_id.csv` pour l'apprentisage automatisé..

Description du dataset

Champ	Signification médicale / métier	Intérêt métier
`ID`	Identifiant unique du patient	Juste technique
`Age`	Âge du patient	Le risque de diabète augmente avec l'âge
`Gender`	Sexe du patient	Le diabète de type 2 est légèrement plus fréquent chez les hommes, mais dépend aussi des habitudes de vie
`Polyuria`	Urines fréquentes	Signe typique du diabète (le sucre attire l'eau dans les urines)
`Polydipsia`	Soif excessive	Autre symptôme clé (le corps se déshydrate à cause de la polyurie)
`sudden weight loss`	Perte de poids rapide	Indique un diabète mal contrôlé (le corps brûle les graisses)
`weakness`	Fatigue ou faiblesse générale	Résulte d'un manque d'énergie dû au glucose non utilisé

`Polyphagia`	Faim excessive	Le corps “pense” manquer d’énergie malgré le sucre sanguin élevé
`Genital thrush`	Mycose génitale	Très fréquent chez les diabétiques à cause du sucre dans les urines
`visual blurring`	Vision trouble	Causée par des fluctuations du taux de sucre
`Itching`	Démangeaisons	Liées à des infections cutanées ou mycoses
`Irritability`	Irritabilité accrue	Conséquence possible des variations de glycémie
`delayed healing`	Cicatrisation lente	Caractéristique du diabète (affecte les vaisseaux et nerfs)
`partial paresis`	Faiblesse musculaire partielle	Peut signaler une neuropathie diabétique
`muscle stiffness`	Raideur musculaire	Parfois associée à un mauvais métabolisme du glucose
`Alopecia`	Perte de cheveux	Effet secondaire possible d’un déséquilibre hormonal
`Obesity`	Obésité	Facteur de risque majeur du diabète de type 2
`class`	Diagnostic final (Positive/ Negative)	Cible à prédire

Notes sur les modifications des données

- Les noms de colonnes ont été modifiés pour respecter le snakecase ainsi que tout en minuscules.
- Les valeurs des données ont été modifiées en “booléens” à valeur entière (0/1) pour faciliter l’apprentissage automatique.

Bibliotéques ajoutées

Librairie	Description courte	Commande d’installation	Utilisation principale
Matplotlib	Bibliothèque de base pour créer des graphiques 2D (courbes, histogrammes, scatter plots, etc.)	`pip install matplotlib`	Visualisation personnalisée et fine des données

Pandas	Outil essentiel pour la manipulation, le nettoyage et l'analyse de données tabulaires (DataFrames)	`pip install pandas`	Chargement, transformation et agrégation de données
Seaborn	Extension de Matplotlib qui simplifie la création de graphiques statistiques attrayants	`pip install seaborn`	Visualisations statistiques (heatmaps, boxplots, pairplots, etc.)

ML2

Bibliotéques ajoutées

Librairie	Description courte	Commande d'installation	Utilisation principale
Scikit-learn	Bibliothèque essentielle pour le machine learning en Python, intégrant de nombreux algorithmes de classification, régression et clustering.	`pip install scikit-learn`	Entraînement, optimisation et évaluation de modèles de machine learning supervisés et non supervisés.
Joblib	Outil performant pour la sérialisation, la sauvegarde et le chargement de modèles Python, notamment ceux créés avec Scikit-learn.	`pip install joblib`	Sauvegarde et restauration rapide des modèles entraînés pour le déploiement ou la réutilisation.

Objectifs et observations

Utilisation de Scikit-learn et méthode de classification

Nous avons utilisé la bibliothèque **Scikit-learn** (ajoutée aux bibliothèques mentionnées ci-dessus) afin de mettre en œuvre un modèle d'apprentissage supervisé de type **arbre de**

décès, afin de mettre en œuvre un modèle d'apprentissage supervisé de type **arbre de décision** (*Decision Tree Classifier*).
L'objectif était de **déterminer la classe** — positive ou négative — des patients atteints de diabète à partir du jeu de données d'entraînement `diabetes_clean.csv`.
Le fichier de test `test_clean.csv`, quant à lui, ne contenait pas la colonne `class`, celle-ci devant être prédite par le modèle.

Pour l'optimisation du modèle, nous avons employé la méthode **Grid Search** via `GridSearchCV()`, en testant plusieurs métriques de performance :
- **accuracy**
- **balanced accuracy**
- **f1-score** (pertinente pour la classification binaire) - **TopK Accuracy** évalue si la bonne classe se trouve parmi les *K* prédictions les plus probables du modèle

Après comparaison, nous avons choisi de retenir la métrique **accuracy**, car elle produisait un score identique à celui du `f1-score` (≈ 0.99) tout en réduisant le temps de calcul d'environ deux secondes.
La métrique `topk` a été écartée en raison d'un score légèrement inférieur (≈ 0.981).

Rendu

Les éléments suivants :

- `./data/model_UFO.ipynb` → Notebook Jupyter contenant l'ensemble du code d'analyse, de préparation des données et d'entraînement du modèle.
- `./model/diabeast.pkl` → Modèle final sauvegardé (format binaire) à l'aide de *Joblib* pour une réutilisation ou un déploiement ultérieur.
- `./README.md` → Fichier descriptif du projet bonus. :)

Liens Documentation

- Scikit-Learn sur grid_search: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html
- Joblib Doc: https://joblib.readthedocs.io/en/stable/