

I. PRESENTATION DU DATASET ET DU CONTEXTE :

Voici un exemple de dataset dans le domaine médical :

Le but est de prédire la présence d'un diabète chez un patient en fonction de certaines caractéristiques. Les caractéristiques sont les suivantes :

- Number of times pregnant (Nombre de grossesses)
- Plasma glucose concentration a 2 hours in an oral glucose tolerance test
(Concentration plasmatique de glucose 2 heures après un test de tolérance au glucose oral)
- Diastolic blood pressure (Pression artérielle diastolique)
- Triceps skinfold thickness (Épaisseur du pli cutané du triceps)
- 2-Hour serum insulin (Insuline sérique à 2 heures)
- Body mass index (Indice de masse corporelle)
- Diabetes pedigree function (Fonction de généalogie du diabète)
- Age (Âge)

Le dataset peut être téléchargé depuis le lien suivant :

[kaggle-diabete-dataset](#)

II. DIFFERENTE FONCTIONS NECESSAIRE :

1. Installer les bibliotheques suivantes sous python version minimale : 3.

- **Numpy**
- **Sklearn**
- **Matplotlib**

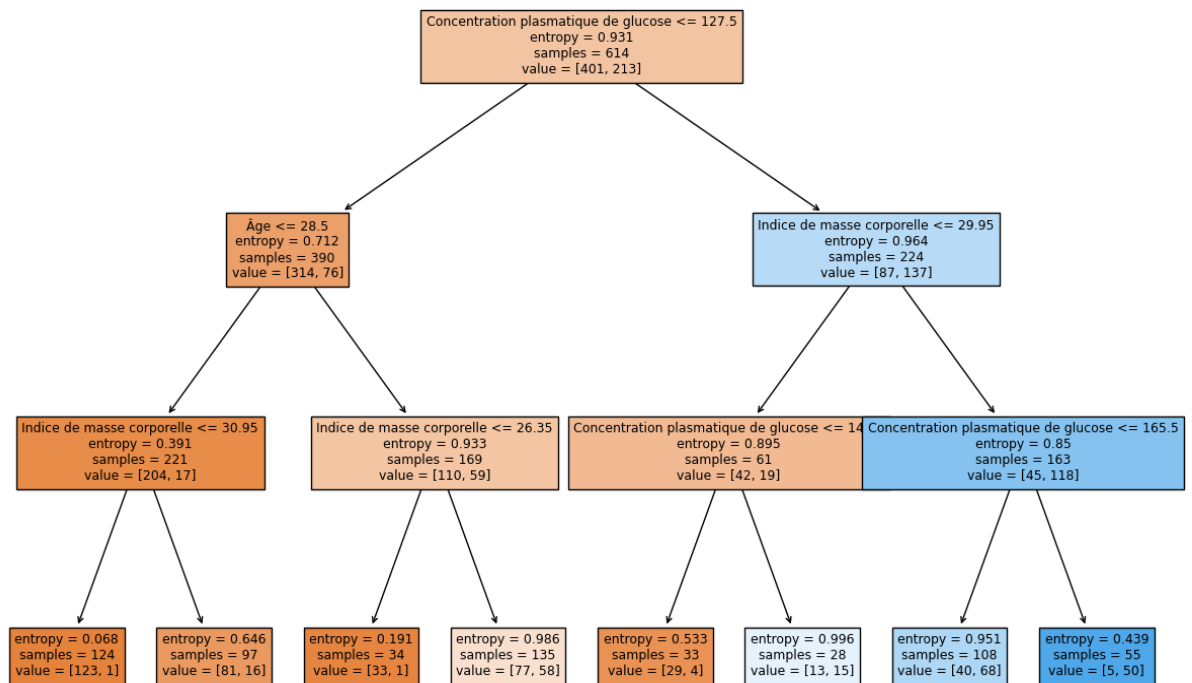
2. les fonctions utilisées:

- **loadtxt()** : charger les données a partir d un fichier
- **data[:, :-1]** : charger des données sans leurs etiquets
- **data[:, -1]** : charger des etiquets ou les differentes classes
- **train_test_split()** : La fonction train_test_split() est une fonction de la bibliothèque scikit-learn (sklearn) en Python. Elle est couramment utilisée pour diviser un ensemble de données en deux sous-ensembles : un ensemble d'entraînement (training set) et un ensemble de test (test set). Cette division est essentielle dans le processus d'apprentissage automatique (machine learning) pour évaluer les performances d'un modèle sur des données non vues auparavant.
- **DecisionTreeClassifier()** : elle permet en general de faire le choix de l algorithme á appliquer(CART, IDE3) .
- **Fit()** : utilisé pour l entrainement

- **predict()** : utilisé pour faire la prediction sur les données
- **score()** : utilisé pour la prediction sur les données
- **plot_tree()** : est une fonction de la bibliothèque scikit-learn (sklearn) en Python qui permet de visualiser un arbre de décision entraîné à l'aide de l'algorithme ID3 ou d'autres algorithmes d'arbres de décision disponibles dans scikit-learn.
- **plt.show()** : utilisé pour afficher l'arbre de decision en question

III. CAPTURES EN FONCTION DES PROFONDEURS

- max_depth=3 pour IDE3:



➤ max_depth=3 pour CART:

