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 NECCESSAIRE :

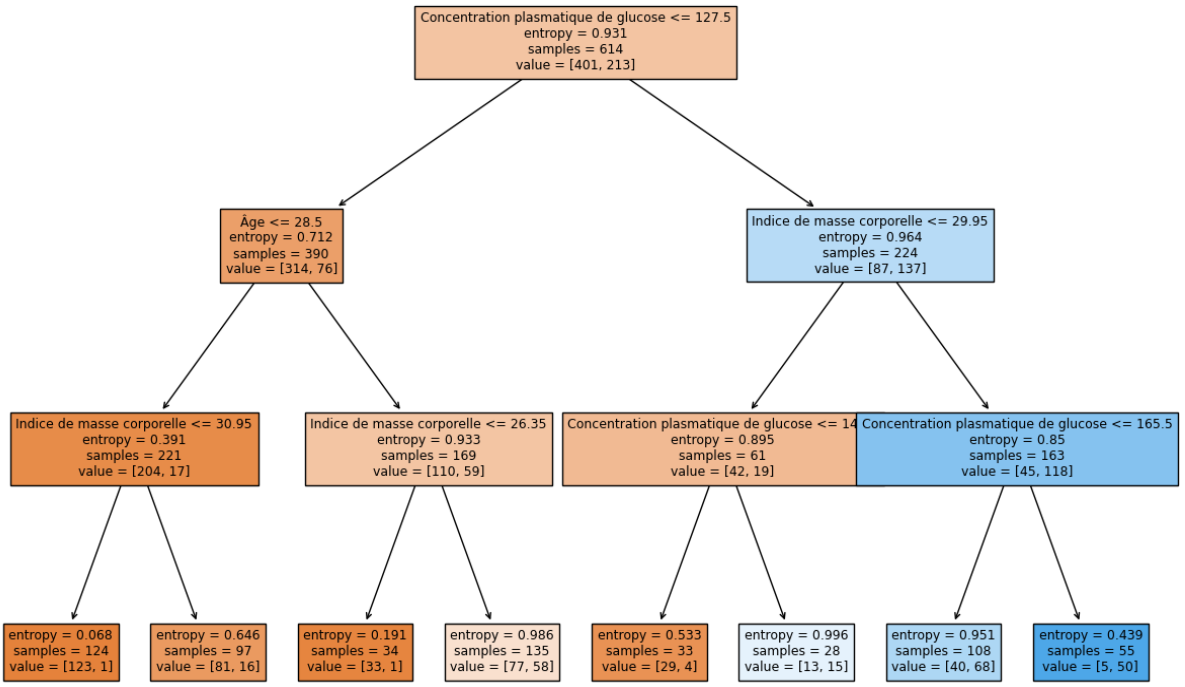
1. Installer les bibliotheques suivantes sous python version minimale : 3.
   - **Numpy**
   - **Sklearn**
   - **Matplotlib**

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

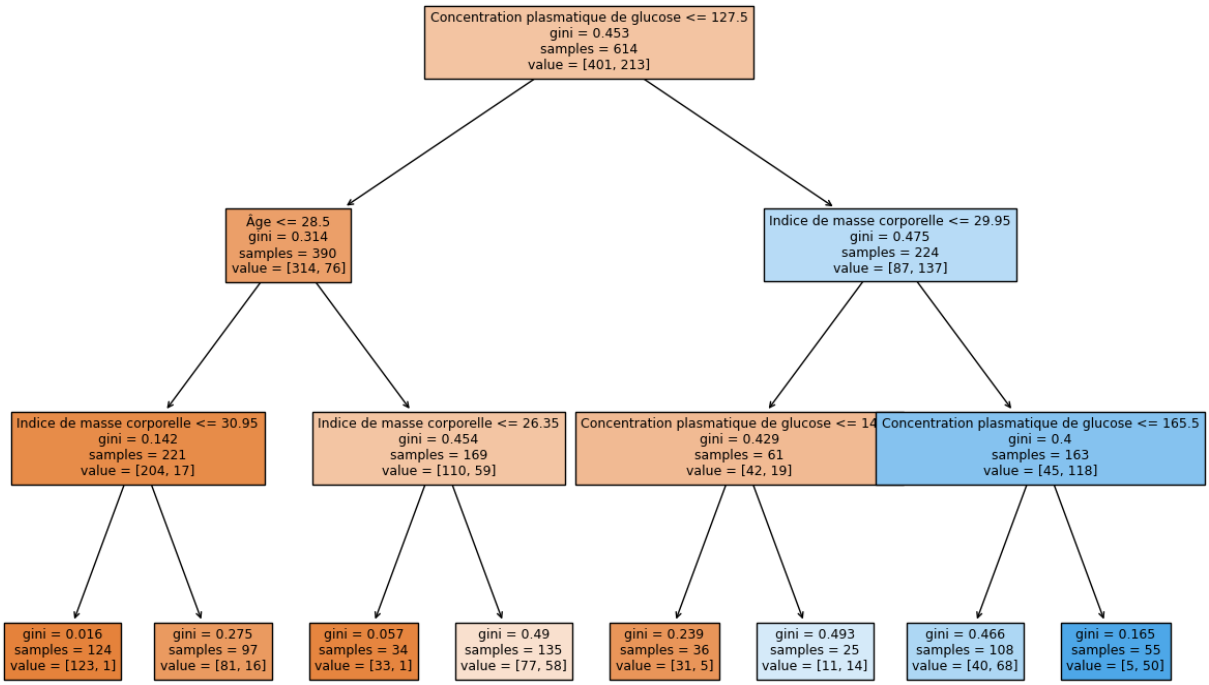
- **predict()** : utilisé pour faire la prediction sur lea données
- **score() :** utilisé pour la prediction sur les données
- **plot_tree()** : est une fonction de la bibliothèque scikit-learn (sklearn) en Python qui permet de visualiser un arbre de décision entraîné à l'aide de l'algorithme ID3 ou 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:



IV. INTERPRETATION DES DONNEES DANS LE CADRE DE IDE3 :

Les valeurs d'entropie, de sample et de value dans les feuilles d'un arbre de décision sont des indicateurs qui permettent d'interpréter le fonctionnement de l'arbre et les prédictions qu'il produit.

L'entropie est une mesure de l'impureté d'un ensemble de données. Plus l'entropie est élevée, plus l'ensemble de données est diversifié et plus il est difficile de prendre une décision. Dans le cas d'un arbre de décision, l'entropie est utilisée pour sélectionner la caractéristique qui maximise le gain d'information lors de la division des données en sous-ensembles. Une faible entropie indique une séparation claire entre les classes de sortie et donc une décision plus facile à prendre.

Le sample : est le nombre d'échantillons (lignes) dans l'ensemble de données correspondant à la feuille de l'arbre. Il indique le nombre d'observations qui ont été classées dans cette feuille.

La value : est un tableau qui indique le nombre d'observations pour chaque classe de sortie dans l'ensemble de données correspondant à la feuille de l'arbre. Par exemple, si la variable cible est binaire (0 ou 1), le tableau de value peut être [30, 10],indiquant qu'il y a 30 observations de la classe 0 et 10 observations de la classe 1 dans cette feuille.

Ainsi, en regardant les valeurs ***d'entropie, de sample et de value dans*** les feuilles de l'arbre de décision, on peut interpréter les prédictions produites par l'arbre. Par exemple, une feuille avec une faible entropie et un grand nombre d'observations peut indiquer une décision claire et probablement précise. Une feuille avec une entropie élevée et un petit nombre d'observations peut indiquer un manque de données ou une difficulté à prendre une décision précise.

En résumé, l'entropie, le sample et la value sont des indicateurs qui permettent d'interpréter les feuilles de l'arbre de décision et les prédictions qu'il produit. Ils peuvent aider à identifier les caractéristiques les plus discriminantes et à évaluer la qualité des prédictions.

V. SENS DE LECTURE DU GRAPHE RESULTANT :

Pour lire l'arbre de décision en langage naturel, vous pouvez suivre les flèches en partant de la racine jusqu'à chaque feuille et interpréter les décisions prises à chaque nœud. Voici comment interpréter un exemple simple d'arbre de décision binaire :

La racine de l'arbre de décision est le premier nœud. Il représente l'ensemble de données complet et n'a pas de condition de division. Dans notre exemple, la racine peut être interprétée comme suit : "Si la caractéristique X est inférieure ou égale à 0.5, passez au nœud fils de gauche, sinon passez au nœud fils de droite."

Les nœuds suivants sont des nœuds internes, qui représentent une condition de division sur une caractéristique. Chaque nœud contient une flèche entrante qui indique la caractéristique utilisée pour diviser les données, et deux flèches sortantes qui indiquent les deux sous-ensembles de données résultants. Par exemple, un nœud peut être interprété comme suit : "Si la caractéristique Y est inférieure ou égale à 1.0, passez au nœud fils de gauche, sinon passez aunœud fils de droite."

Les feuilles de l'arbre de décision sont les derniers nœuds, qui ne contiennent pas de condition de division. Chaque feuille représente une décision finale sur la classe de sortie. Par exemple, une feuille peut être interprétée comme suit : "Si les caractéristiques X et Y sont inférieures ou égales à leurs seuils respectifs, la sortie est de classe 0. Sinon, la sortie est de classe 1."

Pour interpréter l'arbre de décision produit par votre code, vous pouvez suivre le même processus en lisant chaque nœud et chaque feuille en termes de conditions de division et de décisions sur les classes de sortie. Les flèches étiquetées vous indiqueront quelles caractéristiques ont été utilisées pour diviser les données en sous-ensembles. Vous pouvez également prendre en compte les valeurs d'entropie, de sample et de value dans les feuilles pour évaluer la qualité des prédictions de l'arbre.

En résumé, pour lire l'arbre de décision en langage naturel, il suffit de suivre les flèches en partant de la racine jusqu'à chaque feuille et d'interpréter les conditions de division et les décisions sur les classes desortie. Dans le cas de votre code, vous pouvez utiliser les étiquettes des flèches pour identifier les caractéristiques utilisées pour diviser les données en sous-ensembles, et les valeurs d'entropie, de sample et de value pour évaluer la qualité des prédictions. Vous pouvez donc interpréter l'arbre de décision en termes de décisions logiques sur les caractéristiques et les classes de sortie, ce qui peut vous aider à comprendre le fonctionnement de l'algorithme et à évaluer sa précision.

VI. jj