WRITTEN BY MR. HUYNH NAM
www.flex.edu.vn

THỰC HÀNH CÂY QUYẾT ĐỊNH

Reqirments: Features used: Age, Sex, Blood Pressure, and Cholesterol of 200 patients. The classifier is built to find a proper drug for a new patient among 5 drugs.

---------- TẢI DỮ LIỆU

1. Import các thư viện cần thiết

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn import metrics

from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OneHotEncoder
```

2. Hãy tải dữ liệu từ file drug200.csv lưu vào data frame **df**

---------- HIỂU DỮ LIỆU

3. Kiểm tra thông tin dữ liệu, dùng hàm info
4. Mô tả dữ liệu, dùng hàm describe
5. Kiểm tra kiểu dữ liệu, dùng types
6. Kiểm tra tên cột, dùng columns
7. Kiểm tra dữ liệu trống, dùng df.isnull().sum()
8. Kiểm tra dữ liệu trùng, dùng df.duplicated().sum()

---------- THAO TÁC DỮ LIỆU VÀ HIỆU CHỈNH

9. Đổi tên cột phù hợp ngữ nghĩa

```
df.rename(columns = { 'Na_to_K' : 'Sodium_to_Potassium', 'BP' : 'Blood_Pressure'}, inplace = True)
df['Sex'].replace({'M': 'Male', 'F': 'Female'},inplace = True)
df['Sodium_to_Potassium'] = df['Sodium_to_Potassium'].round(0)
df['Sodium_to_Potassium'] = df['Sodium_to_Potassium'].astype(int)
```

---------- KHẢO SÁT DỮ LIỆU VỚI KỸ THUẬT EDA

10. Thống kê số lượng người thuộc 5 nhóm tuổi cao nhất

```
age_values = df['Age'].value_counts()
top_age = age_values.head(5)
df_top_age = pd.DataFrame({'Age' : top_age.index, 'Count': top_age.values})
df_top_age
```

11. Tiến hành EDA dữ liệu. Sinh viên quan sát các biểu đồ và cho nhận xét

```
# EDA
def create_plot(ax, x, data, plot_type='count', y=None, palette='Set2'):
    if plot_type == 'count':
        sns.countplot(x=x, data=data, palette=palette, ax=ax)
    elif plot_type == 'bar':
        sns.barplot(x=x, y=y, data=data, palette=palette, ax=ax)

    ax.set_title(f'Plot of {x}' if plot_type == 'count' else f'Bar Plot of {x} and {y}')

    for p in ax.patches:
        ax.annotate(f'{int(p.get_height())}',
                    (p.get_x() + p.get_width() / 2., p.get_height()),
                    ha='center', va='baseline',
                    fontsize=10, color='black',
                    xytext=(0, 1),
                    textcoords='offset points')
# Create the figure and subplots
plt.figure(figsize=(10, 10))

# Define plot configurations
plot_configs = [
    {'x': 'Sex', 'data': df, 'plot_type': 'count'},
    {'x': 'Blood_Pressure', 'data': df, 'plot_type': 'count'},
    {'x': 'Cholesterol', 'data': df, 'plot_type': 'count'},
    {'x': 'Drug', 'data': df, 'plot_type': 'count'},
    {'x': 'Age', 'y': 'Count', 'data': df_top_age, 'plot_type': 'bar'}
]

# Loop through plot configurations to create subplots
for i, config in enumerate(plot_configs):
    ax = plt.subplot(3, 3, i + 1)
    create_plot(ax, **config)

plt.tight_layout()
plt.show()
```

12. Tiến hành khảo sát dữ liệu outlier và cho nhận xét

```
plt.figure(figsize=(5, 5))
sns.boxplot(x='Sex', y='Sodium_to_Potassium', data=df)
plt.title('Boxplot of Sex by Sodium_to_Potassium')
plt.show()
```

---------- XÂY DỰNG MÔ HÌNH TRÍ TUỆ NHÂN TẠO

13. Tạo tập dữ liệu đặc trưng (feature) và mục tiêu (target)

```
X = df.iloc[:,:-1]
y = df.iloc[:,-1]
```

14. Tiến hành One Hot Encoder cho đặc trưng Sex và Label Encoder cho các categorical feature còn lại. Sinh viên giải thích tại sao làm như vậy?

```
#OneHotEncoder
onehot_encoder_sex = OneHotEncoder(sparse_output=False)
one_hot_encoded  = onehot_encoder_sex.fit_transform(X[['Sex']])
one_hot_df = pd.DataFrame(one_hot_encoded, columns=onehot_encoder_sex.get_feature_names_out(['Sex']))
X = pd.concat([X, one_hot_df], axis=1)
X = X.drop('Sex', axis=1) # Drop the original categorical columns

# Labelled encoder
labelled_encoder_blood_pressure = LabelEncoder()
labelled_encoder_cholesterol = LabelEncoder()
X['Blood_Pressure'] = labelled_encoder_blood_pressure.fit_transform(X['Blood_Pressure'])
X['Cholesterol'] = labelled_encoder_cholesterol.fit_transform(X['Cholesterol'])
```

15. Sinh viên phân tách thành tập dữ liệu train và test với tỉ lệ 60:40 , hệ số ngẫu nhiên là 42

```
X_train, X_test, y_train, y_test =
```

16. Xây dựng mô hình với các hyperparameter tunning (siêu tham số điều chỉnh mô hình) như sau: criterion="entropy", max_depth=6 và lamx_leaf_nodes=10. Sinh viên giải thích các siêu tham số

```
dtc = DecisionTreeClassifier(
```

17. Tiến hành huấn luyện (train) mô hình trên tập dữ liệu huấn luyện
18. Hiện thị danh sách các lớp mục tiêu phân lớp
19. Vẽ sơ đồ mô hình cây phân lớp. Sinh viên giải thích các số liệu trên từng node và leaf

```
from sklearn.tree import plot_tree

feature_cols = X_train.columns

plt.figure(figsize=(20,10))
plot_tree(dtc, class_names=dtc.classes_,feature_names=feature_cols, fontsize=12, filled=True)
plt.show()
```

20. Tính Entropy có trọng số (WE) và Information Gain khi phân tách Root thành Left – Right node

---------- ĐÁNH GIÁ MÔ HÌNH

21. Sinh viên tiến hành đánh giá dựa trên các độ đo
    a. Accuracy
    b. Confusion matrix
22. In ra bảng báo cáo các giá trị đánh giá theo từng nhóm thuốc phân lớp

```
from sklearn.metrics import classification_report
target_names = ['drugA', 'drugB', 'drugC', 'drugX', 'drugY']
print(classification_report(y_test, y_test_pred, target_names=target_names))
```

---------- XÂY DỰNG ỨNG DỤNG TRÍ TUỆ NHÂN TẠO

23. Xây dựng chương trình đề xuất hỗ trợ cấp thuốc cho bệnh nhân dựa trên các chỉ số Age, Blood_Pressure, Cholesterol, Sodium_to_Potassium và Sex được nhập từ bàn phím. Hãy in ra kết quả loại thuốc được cấp. Ví dụ: Age=32, Blood_Pressure='HIGH', Cholesterol='NORMAL', Sodium_to_Potassium=13, Sex='Female'

---------- TỐI ƯU VÀ CẢI TIẾN MÔ HÌNH CÓ ĐIỀU KIỆN

24. Sinh viên điều chỉnh các siêu tham số trong mô hình như sau: max_depth chạy từ 2 đến 10 và max_leaf_nodes chạy từ 2 đến 10. Sau đó, vẽ biểu đồ thể hiện sự thay đổi của độ đo accuracy. Từ đó đưa đến kết luận với siêu tham số điều chỉnh nào thì mô hình tốt nhất.
25. Sinh viên tìm hiểu giải thuật C4.5 (sử dụng Gain Ratio) và CART (sử dụng Gini Impurity) sau đó cài đặt với các mô hình đó, rồi so sánh với giải thuật ID3 dựa trên độ đo đánh giá accuracy.
26. Giải sử drugX là một loại thuốc đặc trị có tác dụng rất mạnh không tốt cho bệnh nhân (hạn chế sử dụng) tức là bệnh rất nghiêm trọng mới cần sử dụng. Hãy đánh giá các giá trị bên dưới trên nhóm phân lớp drugX và cho biết ta cần tối ưu giá trị nào khi cải thiện mô hình có điều kiện là hạn chế cấp thuốc drugX?
    a. Accuracy
    b. Confusion matrix
    c. [tn, fp, fn, tp]
    d. [TPR, FNR, FPR, TNR]
    e. [precision, recal, F1]
    f. Đồ thị AUC & ROC