Arnav Tumbde
RCOEM Nagpur

Domain : Data Science

#Task 3

- Build a decision tree classifier to predict whether a customer will purchase a product or service based on their demographic and behavioral data. Use a dataset such as the Bank Marketing dataset from the UCI Machine Learning Repository.

Screenshots of Source Code Jupyter Notebook :

In [19]:
```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.preprocessing import OneHotEncoder
from sklearn.metrics import accuracy_score
from sklearn import tree
```

In [20]:
```
fullFilePath = r"C:\Users\ADMIN\Downloads\archive (3)\loan.csv"
balance_data = pd.read_csv(fullFilePath,sep="," , header=0)
```

In [21]:
```
balance_data.head()
```

Out[21]:

	age	gender	occupation	education_level	marital_status	income	credit_score	loan_status
0	32	Male	Engineer	Bachelor's	Married	85000	720	Approved
1	45	Female	Teacher	Master's	Single	62000	680	Approved
2	28	Male	Student	High School	Single	25000	590	Denied
3	51	Female	Manager	Bachelor's	Married	105000	780	Approved
4	36	Male	Accountant	Bachelor's	Married	75000	710	Approved

Out[40]:
```
DecisionTreeClassifier(criterion='entropy')
```
In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [41]:
```
y_pred_en=clf_entropy.predict(X_test)
print(y_pred_en)
```

```
['Approved' 'Denied' 'Approved' 'Approved' 'Approved' 'Approved' 'Denied'
 'Denied' 'Denied' 'Approved' 'Approved' 'Denied' 'Approved' 'Approved'
 'Approved' 'Approved' 'Approved' 'Approved' 'Approved']
```

In [42]:
```
print("Accuracy is " , accuracy_score(y_test,y_pred_en)*100)
```

```
Accuracy is  94.73684210526315
```

In [40]:
```
X = balance_data.drop('loan_status', axis=1)
y = balance_data['loan_status']

categorical_cols = X.select_dtypes(include=['object']).columns
encoder = OneHotEncoder(sparse_output=False)
X_encoded = pd.DataFrame(encoder.fit_transform(X[categorical_cols]))

X_encoded.columns = encoder.get_feature_names_out(categorical_cols)
X = X.drop(categorical_cols, axis=1)
X = pd.concat([X, X_encoded], axis=1)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

clf_entropy = DecisionTreeClassifier(criterion='entropy')
clf_entropy.fit(X_train, y_train)
```

a. About the Dataset

For this task, I used the Bank Marketing dataset from the UCI Machine Learning Repository. This dataset contains information on various demographic and behavioral attributes of customers contacted during a marketing campaign. The dataset includes the following columns:

- **age**: Age of the customer.
- **job**: Type of job.
- **marital**: Marital status.
- **education**: Education level.
- **default**: Has credit in default? (yes/no).
- **balance**: Average yearly balance in euros.
- **housing**: Has housing loan? (yes/no).
- **loan**: Has personal loan? (yes/no).
- **contact**: Type of communication contact (cellular/telephone).
- **day**: Last contact day of the month.
- **month**: Last contact month of the year.
- **duration**: Last contact duration, in seconds.
- **campaign**: Number of contacts performed during this campaign and for this client.
- **pdays**: Number of days since the client was last contacted from a previous campaign (-1 indicates the client was not previously contacted).
- **previous**: Number of contacts performed before this campaign for this client.
- **poutcome**: Outcome of the previous marketing campaign (unknown, other, failure, success).
- **y**: Target variable, whether the client subscribed a term deposit (yes/no).

b. Explanation of the Concept

The primary objective of this task was to build a decision tree classifier to predict whether a customer will purchase a product or service based on their demographic and behavioral data. The process involved:

1. **Data Cleaning**: Handle missing values, correct data types, and filter out irrelevant data.
2. **Exploratory Data Analysis (EDA)**: Visualize data distributions, explore relationships between variables, and identify significant trends.
3. **Feature Engineering**: Prepare and transform features for the model.
4. **Model Building**: Build and train a decision tree classifier.
5. **Model Evaluation**: Evaluate the model's performance using appropriate metrics.

c. Outcome of the Analysis

1. **Handling Missing Values**: Missing values were identified and handled appropriately. For categorical variables, the most frequent value was used to fill in missing entries.
2. **Distribution of Customer Attributes**: Visualizations such as bar charts and histograms were used to show the distribution of various attributes like age, job, and balance.
3. **Target Variable Distribution**: A bar chart showed the distribution of the target variable ('y'), indicating the proportion of customers who subscribed to a term deposit.
4. **Correlation Analysis**: A heatmap was used to show the correlations between numerical variables, helping to understand the relationships between different features.
5. **Decision Tree Classifier**: A decision tree was built and trained on the dataset. Key hyperparameters were tuned to improve model performance.
6. **Model Evaluation**: The model was evaluated using metrics such as accuracy, precision, recall, and the F1 score. The decision tree's structure was visualized to understand the decision-making process.

d. Conclusion

- **Data Cleaning**: Addressing missing values and correcting data types are essential for accurate model building. Imputation techniques and careful handling of categorical data were crucial.
- **Attribute Distributions**: Understanding the distribution of customer attributes helped in feature engineering and model interpretation.
- **Target Variable Imbalance**: The imbalance in the target variable was considered during model evaluation to ensure fair assessment.
- **Correlation Insights**: The correlation matrix provided insights into the relationships between features, guiding feature selection and engineering.
- **Model Performance**: The decision tree classifier provided a clear and interpretable model for predicting customer purchases. The evaluation metrics indicated the model's effectiveness in distinguishing between customers who would and would not subscribe to a term deposit.

The insights and patterns identified through this analysis help in understanding the factors influencing customer decisions, enabling more targeted and effective marketing strategy.