**INVESTMENT DECISION RECOMMENDATION PROJECT**

-KANISHKA KANNAN

**Data Extraction from MySQL Database**

Utilized the **MySQL connector** to establish a connection with the database.

Employed SQL query to extract relevant data from the MySQL database.

Transformed the extracted data into a Pandas DataFrame for further processing and analysis.

**Data Preprocessing:**

**Cleaning the Data**:

* Removing **Duplicates**: The function drops duplicate rows from the DataFrame df using the drop\_duplicates method. This helps ensure that each observation in the dataset is unique.
* Handling **Numerical Missing Values**: For numerical columns (columns with data types int or float), missing values are filled using forward fill (ffill). This means that missing values are replaced with the last observed non-null value in the column. This approach is suitable for sequential data where missing values are expected to have similar values to the previous observations.
* Handling **Categorical Missing Values**: For categorical columns (columns with data type object), missing values are filled with the mode of the respective column. The mode is the most frequently occurring value in the column and is used to replace missing categorical values. This ensures that missing categorical values are replaced with valid category labels.

**Label Encoding**:

* Unique values in the column are identified.
* A mapping dictionary is created to map each unique value to an integer encoding.
* The column is then encoded using the mapping dictionary, and the encoded values are stored in a new column.

**Convert to Numeric**:

* **Input Cleaning**:The function first removes any non-numeric characters and commas from the input string using a regular expression.
* **Handling "Above" Values**:If the string contains the word "Above", indicating a minimum income threshold, the function extracts the numeric value following "Above" and returns it as the income value.
* **Handling Ranges**:If the string represents an income range (e.g., "US$ 2736 to US$ 8205"), the function splits the string into its individual components.

If the range consists of two values, the function calculates the mean value of the range and returns it as the income value.

If the range consists of only one value, that value is returned as the income value.

* **Error Handling:**The function includes error handling to deal with cases where non-numeric characters are present or when the input format is invalid. In such cases, the function returns None.

**Outlier Detection**:

* Utilized boxplots for visual representation of the outliers.
* Interquartile range (IQR) to detect and remove the outliers
* Z-score methods to detect and handle outliers in the dataset.

**Feature Importance**:

* **Data Preparation and Exploration:**Non-numeric columns are converted to numeric using label encoding to facilitate correlation analysis.

Numeric columns relevant to demographic, employment, behavioral characteristics, and investment outcomes are selected.

* **Correlation Analysis:**A correlation matrix is calculated to quantify the relationships between different factors and investment outcomes.

The correlation matrix is visualized using a heatmap to identify patterns and correlations between variables.

* **Factors Contributing to Investment Success:**The correlation coefficients between each factor and the investment outcomes are examined.

The factors contributing most significantly to investment success are identified based on their correlation with investment outcomes.

The correlation values are summarized, and the sum of correlations within each group is calculated.

* **Visualization of Summed Correlations:**The summed correlations for demographic, employment, and behavioral characteristics are plotted using a bar chart.

**Model Selection:**

**Decision Tree**: Employed a decision tree classifier after performing **ridge regularization** to mitigate overfitting.

**Random Forest**: Utilized random forest classifier with **hyperparameter tuning** using techniques like

* **GridSearchCV and**
* **RandomizedSearchCV**

to optimize model performance.

**Neural Networks**: Implemented neural network classifiers with different optimizers like

* **Adam**
* **SGD**

and ensemble techniques like

* **AdaBoost,**
* **Gradient Boosting**.

**Evaluation:**

**Confusion Matrix**: Calculated confusion matrices to understand the performance of each model in classifying investment decisions.

**Precision, Recall, F1-score**: Evaluated the precision, recall, and F1-score metrics to measure the performance of the models in predicting investment outcomes.

**Accuracy**: Assessed the overall accuracy of the models in making correct predictions.

Saves the best model in a **pickle file** to use it further for user prediction