**Loading the Data**: The stroke dataset is loaded from a CSV file using pandas.

**Cleaning**: Unnecessary columns like id and Site are removed, and messy column names are cleaned up.

**Handling Missing Values**:

For number-type columns, missing values are filled with the middle value (median).

For category-type columns, missing values are filled with the most common value (mode).

**Converting Categories to Numbers**: All the categorical (non-numerical) data is turned into numbers so the models can use them.

**Splitting the Data**: The dataset is divided into two parts — one for training the model and the other for testing it. The split keeps the class distribution balanced.

### 2. Fixing Class Imbalance

**SMOTE**: Since stroke cases are rare, the dataset is unbalanced. SMOTE-NC is used to:

Add more examples of the minority class (stroke cases).

Keep both number and category features balanced.

### 3. Training and Tuning Models

#### ****XGBoost Model****

**Tuning with Optuna**: Optuna is used to automatically try different settings for the XGBoost model to find the best ones.

**Training**: The model is trained using the best settings found.

#### ****Random Forest Model****

**Tuning with GridSearchCV**: A method that checks many combinations of settings to find the best ones for the Random Forest model.

**Training**: The best version of the Random Forest model is trained.

### 4. Evaluating the Models

**Classification Report**: Shows how well the model predicts each class — including accuracy, precision, recall, and F1-score.

**ROC AUC Score**: Measures how well the model can tell the difference between stroke and non-stroke cases.

**Confusion Matrix**: A simple table that shows how many predictions were correct or incorrect.

### 5. Understanding the Model’s Decisions

**SHAP (SHapley Additive exPlanations)**: A tool that explains which features were most important in making predictions.

A plot is used to show how each feature affects the model's decisions.

### 6. Saving the Trained Models

**XGBoost**: The final trained model is saved using save\_model.

**Random Forest**: The model is saved using pickle.dump so it can be used later.