**How will the dataset be balanced?**

The target column, "OS event", was found to be imbalanced, meaning one class had more records than the other. To correct this, SMOTE (Synthetic Minority Over-sampling Technique) was applied only to the training dataset after splitting. This method generates synthetic examples for the minority class, making both classes more even in the training set. Balancing only the training set helps the model learn from both classes equally without causing data leakage.

**How will the dataset be split?**

The dataset was first split into three parts. First, 70% of the data was set aside for training, and the remaining 30% was temporarily separated. That temporary set was then split into 20% for validation, to fine-tune the model and check for overfitting, and 10% for testing, to evaluate model performance on unseen data. Stratified splitting was used to maintain class proportions across all parts of the data. A random state was set to ensure reproducibility.

# Import necessary libraries

from sklearn.model\_selection import train\_test\_split

from imblearn.over\_sampling import SMOTE

import pandas as pd

import os

# Load the prepared dataset

prepared\_data = pd.read\_csv('/home/ec2-user/SageMaker/data/final\_prepared\_data.csv')

# Separate features (X) and target (y)

X = prepared\_data.drop("OS event", axis=1) # All columns except OS event

y = prepared\_data["OS event"] # OS event is the target (0 = alive, 1 = death event)

# Split into train (70%) and temp (30%) sets first

X\_train, X\_temp, y\_train, y\_temp = train\_test\_split(

X, y,

test\_size=0.3, # 30% will be split further into validation and test

stratify=y, # preserve the proportion of classes

random\_state=42

)

# Apply SMOTE to the training set only

smote = SMOTE(random\_state=42) # reproducibility

X\_train\_resampled, y\_train\_resampled = smote.fit\_resample(X\_train, y\_train)

# Split the temp set into validation (20%) and test (10%)

X\_val, X\_test, y\_val, y\_test = train\_test\_split(

X\_temp, y\_temp,

test\_size=0.33, # 0.33 \* 30% = ~10% test set

stratify=y\_temp, # maintain class distribution

random\_state=42

)