# **In Case of Classification**

1) first we have to check the output variable how many unique class value present in it :- either we get binary or multi class classifier.

2)**Below Scenario is applicable Only for Binary class classifier**!!!

* Step1) counting the frequency binary class or we can visualize binary class using countplot graph(it is frequency graph generally suitable of object type column ke liye)
* Step2) Checking binary class classifier is balanced or imbalanced in nature
* Step3) below formula se we can get minority class percentage .

# Calculate minority percentage

class\_counts = data.Exited.value\_counts()

minority\_class\_count = class\_counts.min()

minority\_percent = (minority\_class\_count / data.shape[0]) \* 100

print(f"Minority Class Percentage: {minority\_percent:.2f}%")

Minority Class % Status

> 40% Balanced

30–40% Slight Imbalance

10–30% Imbalanced

< 10% Highly Imbalanced

* If output variable or target variable are imbalanced in nature there few technique (such as :-SMOTE synthetic minority oversampling technique OR class weights) we can used to make minority class OVERSAMPLED to majority at time of TRAINING THE MODEL only.
* If we are splitting the dataset into train test split mei at time we passed parameter stratify=y (what does this stratify do :- stratify make ensure at time of splitting dataset both class ka distribution same rahe always do)

**How Smote Work?**

* SMOTE ko sirf training data pe apply karte hain, taki model ko balanced data mile aur wo minority class ko seekh sake.
* Validation set ko original state mein rakhte hain(means validation or testing data pe smote apply nahi karte hai) taaki aap model ki real-world performance ko accurately measure kar sakein.

FORMULA TO CALCULATE CLASS WEIGHTS:-

Class Weight=Total Samples/Number of Classes × Samples in Class

* Class weights are used during training, where the model is assigned more weight to the samples of the minority class to give them greater importance.
* In this method, the majority class is given less weight, and the minority class is given more weight so that the model avoids misclassifying the minority class.
* This means, if you have imbalanced data, class weights help the model focus more on the minority class.

✅ Best Practices:

Scenario Recommendation

Imbalance mild ho (e.g. 60:40) Use class\_weight only

Imbalance zyada ho (e.g. 90:10) Use SMOTE or other oversampling

Bahut zyada imbalance + small dataset Use SMOTE carefully, maybe try class\_weight only

Large data + imbalance Try SMOTE first, then check if class\_weight zarurat hai ya nahi