Approaches :- → applied on training data set

① Data Level ② Algorithm

- up Sampling
- Down Sampling
- SMOTE (Synthetic Minority Oversampling Technique)
  - ↳ ADASYN
  - ↳ SMOTE + ENN

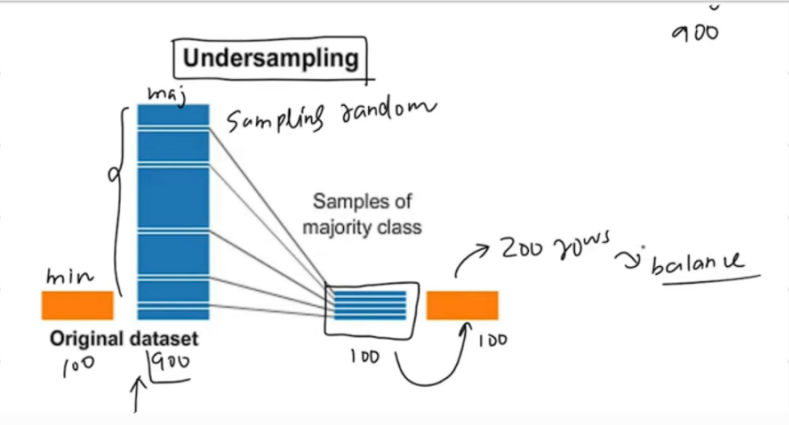
imblearn ↑ Lib

Problem with Imbalanced data :-

1) Bios → Majority class / Minority class → biased decision tree

2) Metrics are → accuracy
Not reliable

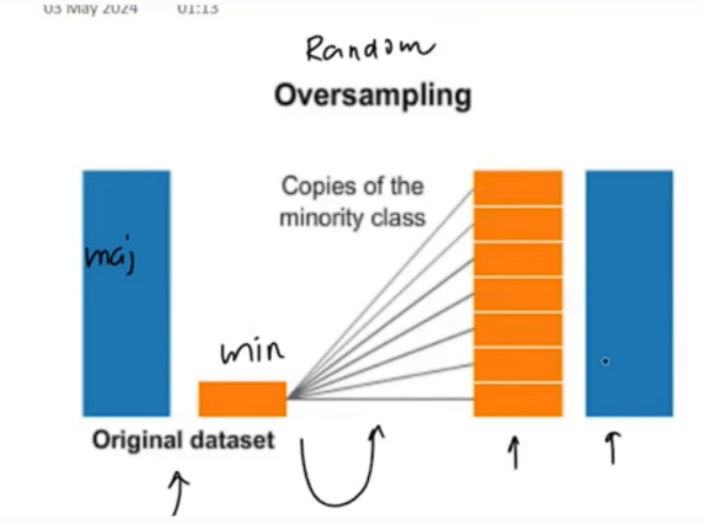
Under Sampling :-



Adv :-
① Reduction in bias
② Faster training

Disadv :-
① Info loss leading to underfitting
② Sampling Bias

Over Sampling :-



Maj Min

900 100 → balance

900 → 800 → duplication

duplicate

Adv :-

1. Reduce Bias

Disadv

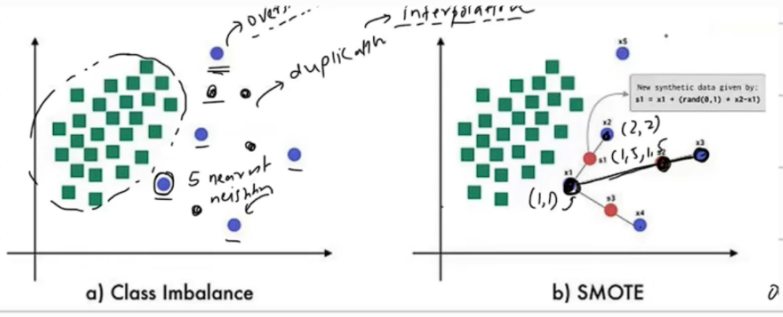
1. Inc Size
2. Duplication of data may cause Overfitting

3 SMOTE :-

Min Class → upsample

But not by duplications

We generate new data points



5 KNN

- Train a KNN on minority class observations - find each observation's 5 closest neighbours

To create the new synthetic data:

- Select examples from the minority class at random
- Select a neighbour of each example at random (for the interpolation)
- Extract a random number between 0 and 1
- Calculate the new examples as = original sample - factor * (original sample - neighbour)
- The final dataset consists of the original dataset + the newly created examples

→ 0.5

sample − factor × [sample − neighbor]

$[1,1] - 0.5\,[(1,1) - (2,2)]$

$(1,1) - 0.5\,[(-1,-1)]$

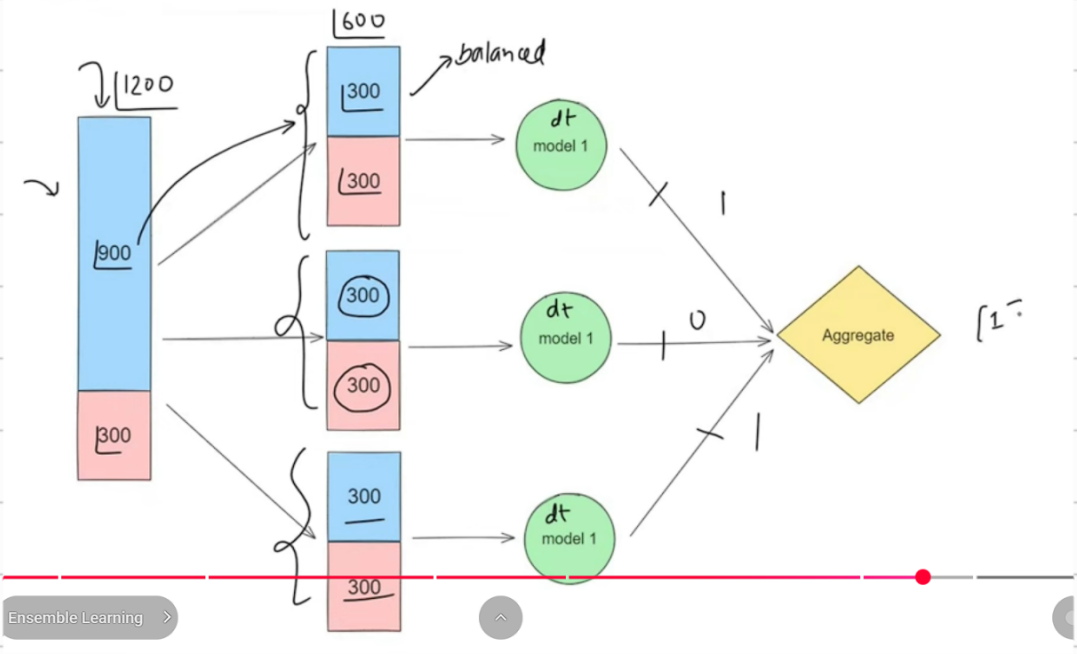
$(1,1) - (-0.5,-0.5)$

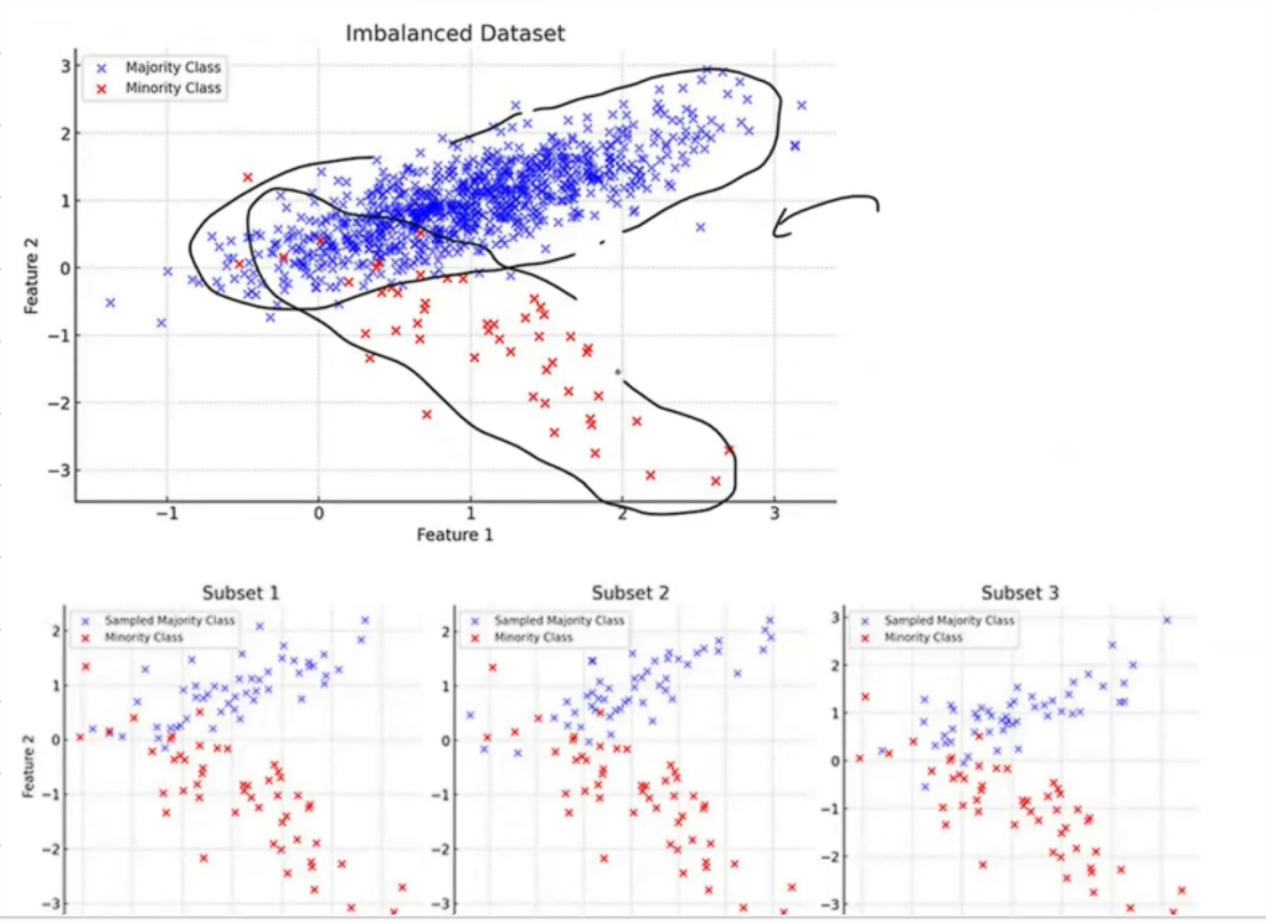
$= 2\,[1.5, 1.5]$

Disadvantages:-

1. Do not Handle Categorical data well
2. Computational Complexity
3. Dependency on Choice of Neighbors
4. Sensitive to Outliers
5. Balanced Achieved May not Reflect True Nature

Ensemble Methods :-





(5) Cost Sensitive Learning:-

Changes in Learning process:-

(1) Class Weights:-

1 / 0 weight →

900 100

[2 1 0] •

Error cone for Class 0 it will we weighted 10 times than original cost → Model will take Seriously.

(6) Use Custom Loss Function :-

Algo which give this feature : GB → XGBoost, LightGBM