Handling Imbalanced dataset

Handling imbalanced datasets is crucial in machine learning because class imbalance can lead to biased models that favor the majority class, resulting in poor performance on the minority class. **Up-sampling** and **down-sampling** are two widely used techniques to address this issue by balancing the class distribution in the training dataset. Here's a detailed explanation of both approaches:

△ Up-Sampling (Oversampling)

➡ Increasing the size of the minority class

Up-sampling involves increasing the number of instances in the minority class to match or approach the size of the majority class. This can be achieved by:

1. Random Oversampling

- Duplicate existing minority class samples or generate new ones randomly.
- Example:
    - If the dataset has 90 samples of class A and 10 samples of class B, you randomly replicate the samples from class B until both classes have equal samples.

2. SMOTE (Synthetic Minority Over-sampling Technique)

- Generates synthetic data points rather than just duplicating existing ones.
- SMOTE works by:
    - Identifying the k-nearest neighbors of a sample from the minority class.
    - Creating new synthetic samples along the line between the sample and its neighbors.

Example:

- If there are two samples at (2, 3) and (4, 5), SMOTE can create a new point at (3, 4).

3. ADASYN (Adaptive Synthetic Sampling)

- An extension of SMOTE that generates more synthetic data in regions where the minority class is under-represented.

- It creates more samples in areas of higher class imbalance, focusing on improving decision boundaries.

☑ Advantages of Up-Sampling:

✔ Preserves information in the minority class.

✔ SMOTE and ADASYN introduce variability, helping the model generalize better.

❌ **Disadvantages of Up-Sampling:**

❌ Can lead to **overfitting** because the same minority samples (or highly similar synthetic ones) are repeated.

❌ Computationally expensive for large datasets.

▽ Down-Sampling (Undersampling)

➡ Reducing the size of the majority class

Down-sampling involves reducing the number of instances in the majority class to match the size of the minority class. This can be achieved by:

1. Random Undersampling

- Randomly remove samples from the majority class until both classes are balanced.
- Example:
    - If there are 100 samples in class A and 20 in class B, randomly remove 80 samples from class A to match class B's size.

2. Tomek Links

- A Tomek Link is a pair of samples from opposite classes that are very close to each other.
- Removing the sample from the majority class helps to clean up the class boundary and reduce overlap.

3. Edited Nearest Neighbors (ENN)

- Removes samples from the majority class that are misclassified by their k-nearest neighbors.

- This refines the decision boundary and reduces noise.

☑ Advantages of Down-Sampling:

✔ Reduces computational complexity.
✔ Helps simplify the model by reducing data size.

❌ **Disadvantages of Down-Sampling:**

❌ Leads to **loss of information** because valuable data from the majority class is discarded.
❌ Risk of underfitting if too many samples are removed.

💡 When to Use Each Technique

Technique	When to Use	Pros	Cons
Up-Sampling	When the minority class is very small, and you want to retain information	Preserves minority class information	Risk of overfitting
Down-Sampling	When the majority class is very large, and computation is expensive	Reduces dataset size, speeds up training	Loss of information, underfitting risk
SMOTE/ADASYN	When synthetic data is acceptable and you want to improve decision boundaries	More realistic minority samples	Computationally expensive
Tomek Links/ENN	When you want to clean up noisy class boundaries	Reduces overlap, improves separation	Loss of some majority class information

---

☑ Best Practices

- **SMOTE** is effective when the dataset is small and overfitting is not a concern.

- **Random oversampling** is simple but should be combined with regularization to avoid overfitting.
- **Down-sampling** works well with large datasets but risks losing valuable information.
- Combining **SMOTE with Tomek Links** or **SMOTE with ENN** is often used to create a cleaner decision boundary.
