# Data Analysis

Data Imbalance and Text Encodings

**CentraleDigitalLab@Nice**

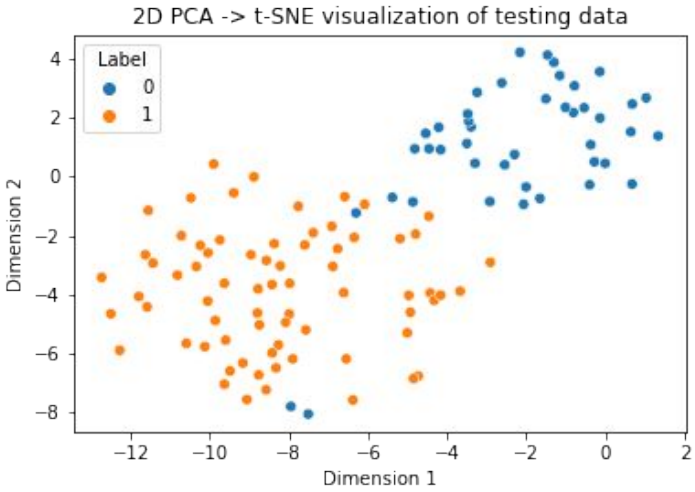
# Pipeline + PCA + T-SNE

It is a common practice of using PCA followed by **t-SNE for error analysis** to check how good our features are.

Visualizing the test data in this way helps to **check the separability of the classes** in the transformed space.

Misclassified observations could inform further steps in the analysis or model tuning.

Decision Boundary Plot with TSNE



# Demo with notebook

## 07__pca_with_tsne.ipynb

# Imbalanced datasets

Data Imbalance refers to an **unequal distribution of classes** within a dataset.

In a binary classification problem, it's where **one class significantly outnumbers the other**. (Normally the positive class).

Example:

In fraud detection, fraudulent transactions (positive class) are far less common than non-fraudulent ones (negative class).

# Imbalanced datasets

Consequences of Data Imbalance:

- **Model Bias:** Models trained on imbalanced data tend to be biased towards the majority class. They might **simply predict the majority class for all inputs** to minimize the error.
- **Poor Generalization:** The model might perform poorly on unseen data.
- **Misleading Accuracy:** High accuracy score might be misleading as it could be merely reflecting the underlying class distribution.
- **Reduced Sensitivity:** The model may have reduced sensitivity (true positive rate) for the minority class.

# Data Augmentation

Generation of **new samples from the existing data** to enhance the size and variability of the dataset.

Methods:

- For image data: rotation, flipping, scaling, cropping, and color variations.
- For text data: synonym replacement, back-translation, and sentence shuffling.

# Data Augmentation

Generation of **new samples from the existing data** to enhance the size and variability of the dataset.

Benefits:

- Enhances the model's ability to generalize.
- Helpful in scenarios where **collecting more data is not feasible**.

# Oversampling

Increasing the number of minority class instances to balance the class distribution.

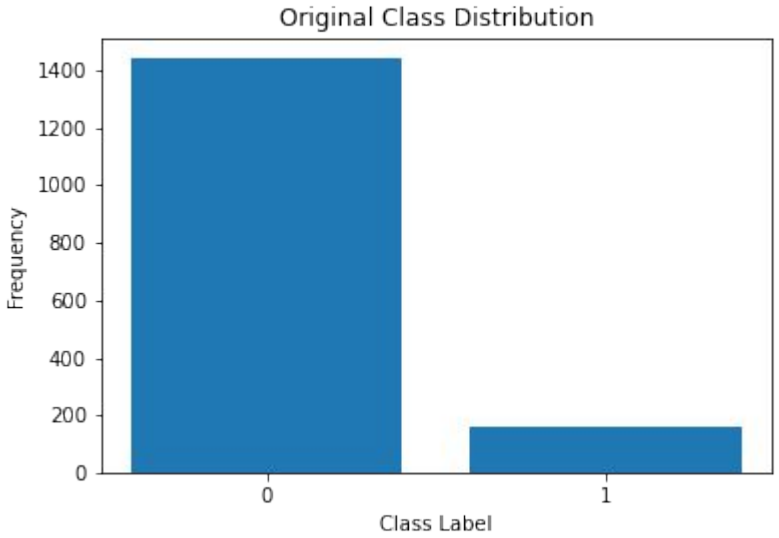
Algorithm: Random Over-sampler (ROS): **Randomly duplicates instances** of the minority class or generates synthetic examples.

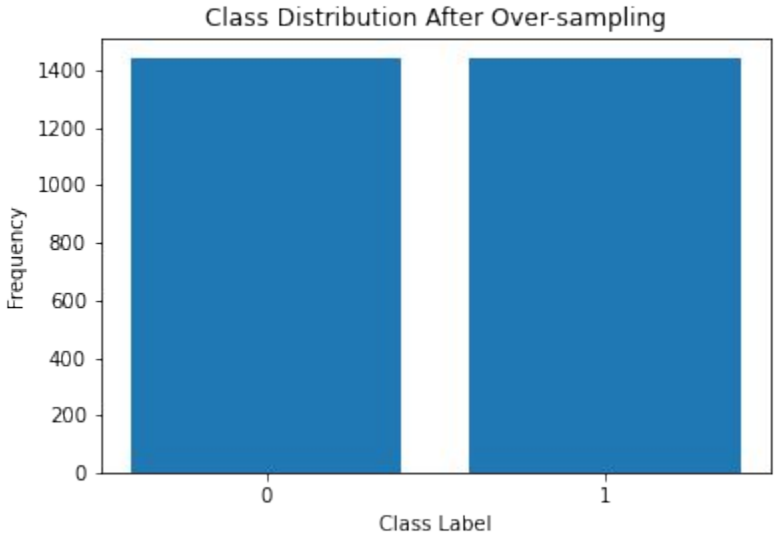
Benefits: No loss of data.

Drawbacks: **May lead to overfitting** if synthetic examples do not sufficiently represent the true data distribution.

# Oversampling

Increasing the number of minority class instances to balance the class distribution.



# Undersampling

Reducing the number of majority class instances to balance the class distribution.

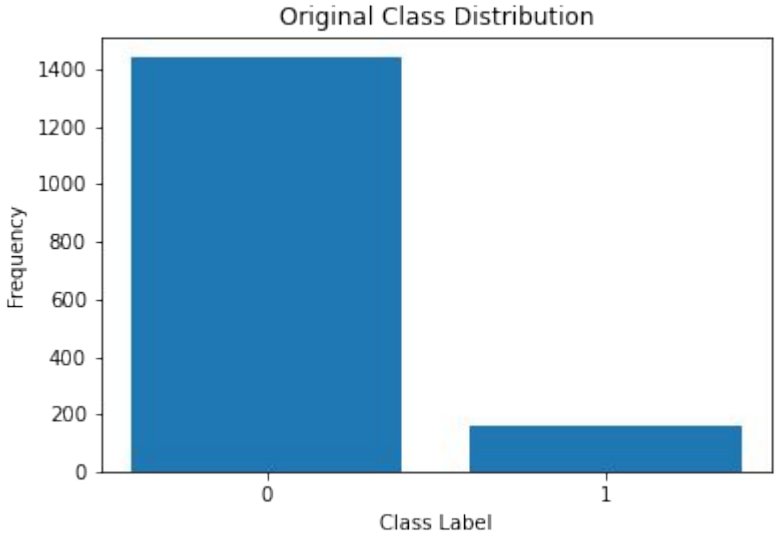
Algorithm: Random Under-sampler (RUS): **Randomly removes instances** of the majority class until a desired balance is achieved.

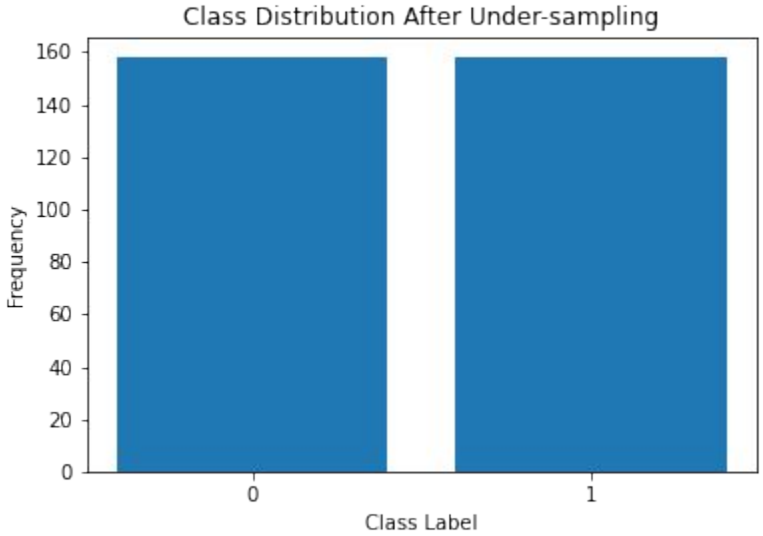
Benefits: Quick and easy to implement.

Drawbacks: **Loss of potentially important data** from the majority class.

# Undersampling

Reducing the number of majority class instances to balance the class distribution.



## Demo with notebook

**08__imbalanced_datasets.ipynb**

# Text Encoding

The process of **converting text data into numerical data** that can be used by machine learning algorithms.

Machine learning models understand numbers, not text. Text encoding bridges this gap, **enabling models to analyze and learn from text data**.

Encoding extracts features from text, capturing the information necessary for the model to make predictions or understand patterns.

# Bag of Words

Bag of Words (BoW) is a text encoding technique that represents text data as a **"bag" of individual words**.

- Tokenization: Split text into words (tokens).
- Vocabulary Building: Create a vocabulary of unique words.
- Encoding: **Count the occurrence of each word** in the text and represent the text as a vector of these counts.

# Bag of Words

Bag of Words (BoW) is a text encoding technique that represents text data as a **"bag" of individual words**.

**Advantages:**

- Simple and easy to understand.
- Effective for text classification tasks.

**Drawbacks:**

- Ignores the order of words and context.
- Can lead to a high-dimensional sparse matrix.

# TF-IDF

TF-IDF is an encoding technique that **weighs the importance of words** in a document relative to a collection of documents.

- **Term Frequency (TF)**: Count the occurrence of each word in a document.
- **Inverse Document Frequency (IDF)**: Logarithmically scale the inverse fraction of the documents that contain the word.
- **Encoding**: Multiply TF by IDF to obtain the TF-IDF weight for each word in a document.

# TF-IDF

TF-IDF is an encoding technique that **weighs the importance of words** in a document relative to a collection of documents.

**Advantages:**

- Reduces the weight of common words.
- Highlights important words unique to a document.

**Drawbacks:**

- More complex than BoW.
- Still disregards the order of words.

# Demo with notebook

**09__text_encoding.ipynb**