# Classification of class-imbalanced data


**Tozammel Hossain**

Data Science & Analytics
University of Missouri

# Class Imbalance Problem?

- The total number of a class of data is far less than the total number of other classes of data.

- This problem is extremely common in practice and can be observed in various domains including
  - Outlier/intrusion detection
  - fraud detection
  - anomaly detection
  - medical diagnosis
  - churn prediction
  - spam detection

# Why is it a problem?

- Most machine learning algorithms work best when the number of instances of each classes are roughly equal.
- When the number of instances of one class far exceeds the other, the classifier tends to be more biased towards the majority class

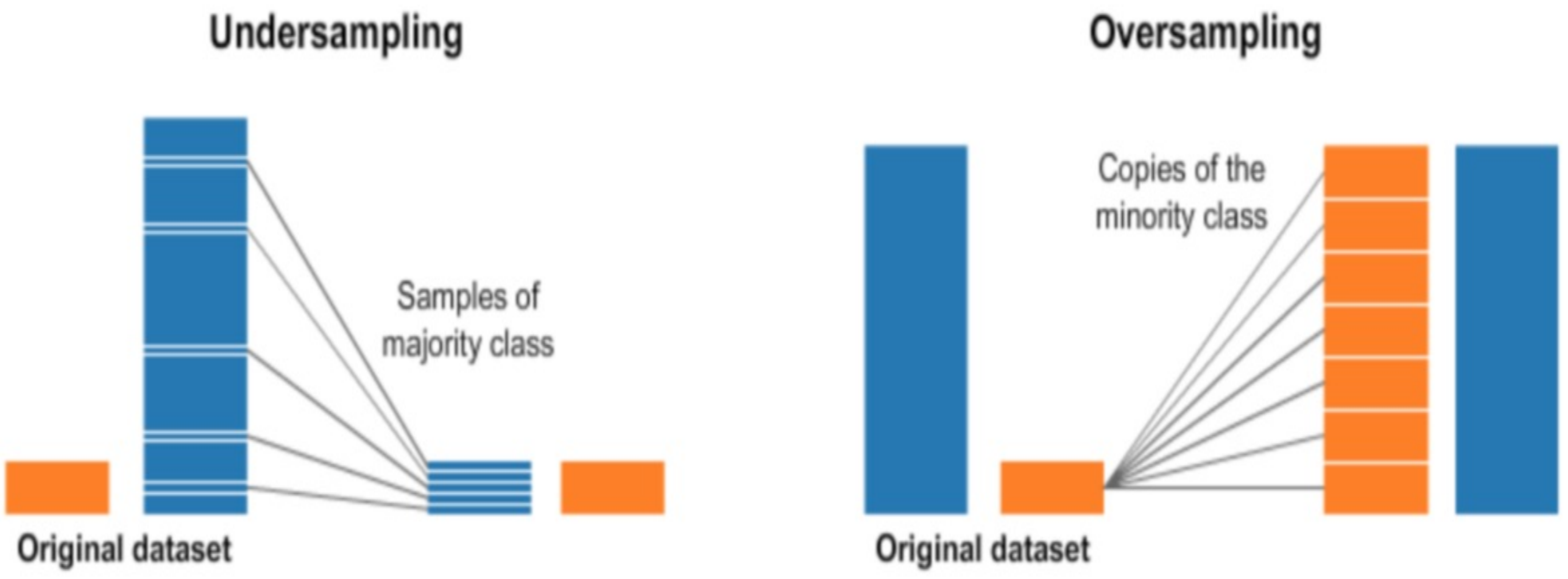
# Remedy

- **Change the performance metrics**
  - Accuracy is a terrible measure when a class-imbalance problem exists
- **Sample data to make the class distribution roughly equal**
  - Over sampling
  - Under sampling
  - SMOTE

# Sampling methods

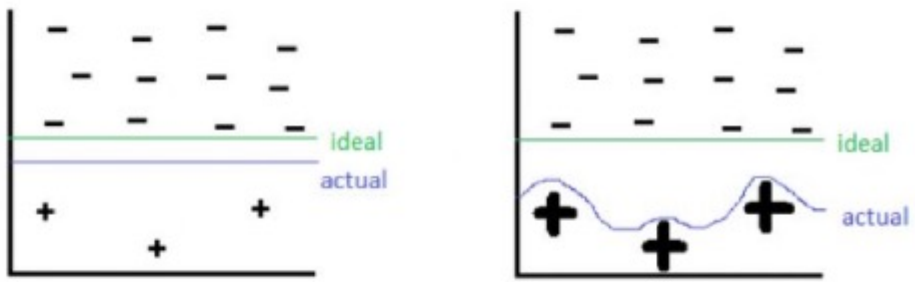
- Over-sampling
  - Add more of the minority class instances (duplicates)
  - It will inflate the dataset
  - Useful when we don’t have much data
- Under-sampling
  - Remove some of the majority class instances
  - Useful when we have tons of data

# Over-sampling vs Under-sampling



# Oversampling

- By oversampling, just duplicating the minority classes could lead the classifier to overfitting to a few examples, which can be illustrated below:



# Oversampling

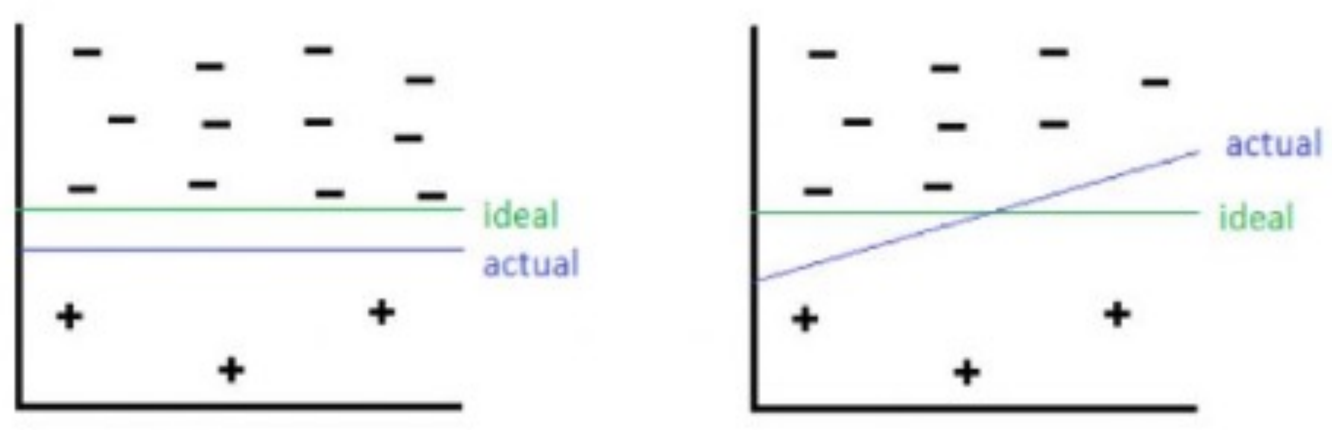
- Left hand side is before oversampling and on the right hand side is oversampling has been applied
- On the right side, the thick positive signs indicate there are multiple repeated copies of that data instance
- The machine learning algorithm observe these minority instances many times and thus designs to overfit to these examples specifically, resulting in a blue line boundary as above

# Types of Oversampling

- **Random Oversampling**
  - Randomly samples the minority classes and simply duplicates the sampled observations
  - Reduce the variance of the dataset.
- **SMOTE**
  - Synthetic Minority Over-sampling Technique
  - Generates new observations by interpolating between observations in the original dataset
- **ADASYN**
  - Adaptive Synthetic is an algorithm that generates synthetic data
  - Do not copy the same minority data
  - Generating more data for “harder to learn” examples

# Under-sampling

- By under-sampling, we could risk removing some of the majority class instances which is more representative, thus discarding useful information. This can be illustrated as follows:



# Under-sampling

- Here the green line is the ideal decision boundary we would like to have, and blue is the actual result. On the left side is the result of just applying a general machine learning algorithm without using under-sampling. On the right, we under-sampled the negative class but removed some informative negative class, and caused the blue decision boundary to be slanted, causing some negative class to be classified as positive class wrongly.

# Types of Under-sampling

- **Random under-sampling**: Randomly remove samples from the majority class, with or without replacement.

- **Cluster**: Cluster centroids is a method that replaces cluster of samples by the cluster centroid of a K-means algorithm, where the number of clusters is set by the level of under-sampling.

# Rebalancing the dataset