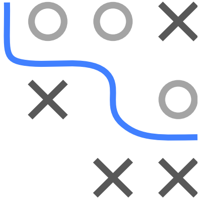
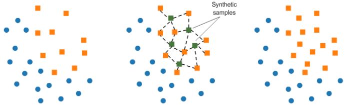
Advanced Machine Learning



Imbalanced Learning: Sampling Methods Part 1

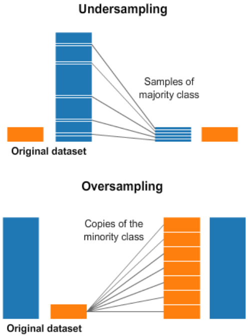


Learning goals

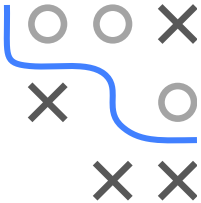
- Know the idea of sampling methods for coping with imbalanced data
- Understand the different sampling techniques

SAMPLING METHODS: OVERVIEW

- Balance training data distribution to perform better on minority classes.
- Independent of classifier $\leadsto$ very flexible and general.
- Three groups:
  - Undersampling — Removing instances of majority class(es).
  - Oversampling — Adding/Creating new instances of minority class(es). (Slower, but usually works better.)
  - Hybrid — Combining both methods.

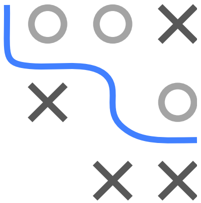


RANDOM UNDERSAMPLING/OVERSAMPLING



- Random oversampling (ROS):
  - Randomly **replicate minority** instances.
  - Prone to overfitting due to multiple tied instances.
- Random undersampling (RUS):
  - Randomly **eliminate majority** instances.
  - Might remove informative instances and destroy important concepts in data.
- Better: Introduce heuristics in removal process (RUS) and do not create exact copies (ROS).

UNDERSAMPLING: TOMEK LINKS

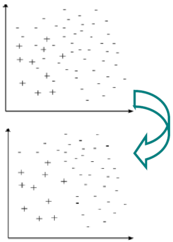


- Remove “noisy borderline” examples (very close observations of different classes) of majority class(es).
- Let $E^{(i)} = (\mathbf{x}^{(i)}, y^{(i)})$ and $E^{(j)} = (\mathbf{x}^{(j)}, y^{(j)})$ be two data points in $\mathcal{D}$.

- A pair $(E^{(i)}, E^{(j)})$ is called *Tomek link* iff there is no other data point $E^{(k)} = (\mathbf{x}^{(k)}, y^{(k)})$ such that

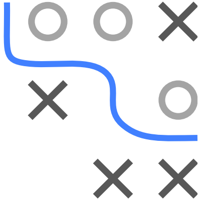
  $$d(\mathbf{x}^{(i)}, \mathbf{x}^{(k)}) < d(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \text{ or}$$
  $$d(\mathbf{x}^{(j)}, \mathbf{x}^{(k)}) < d(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \text{ holds,}$$

  where d is some distance on $\mathcal{X}$.
- $y^{(i)} \neq y^{(j)} \rightsquigarrow$ noisy borderline examples.
- Remove majority instance in each data pair in a Tomek link where $y^{(i)} \neq y^{(j)}$.



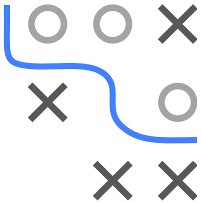
Franciso Herrera (2013), Imbalanced Classification: Common Approaches and Open Problems (URL).

UNDERSAMPLING: CONDENSED NEAREST NEIGHBOR (CNN)



- Remove majority instances far away from decision boundary.
- Construct a **consistent** subset $\tilde{\mathcal{D}}$ of $\mathcal{D}$.
- A subset $\tilde{\mathcal{D}}$ of $\mathcal{D}$ is called consistent if using a 1-NN classifier on $\tilde{\mathcal{D}}$ classifies each instance in $\mathcal{D}$ correctly.
- Create a consistent subset:
  1. Initialize $\tilde{\mathcal{D}}$ by selecting **all minority** instances and randomly picking **one majority** instance.
  2. Classify each instance in $\mathcal{D}$ with 1-NN classifier based on $\tilde{\mathcal{D}}$.
  3. Remove all misclassified instances from $\mathcal{D}$.

UNDERSAMPLING: OTHER APPROACHES



- Neighborhood cleaning rule (NCL):
  1. Find 3 nearest neighbors for each $(\mathbf{x}^{(i)}, y^{(i)})$ in $\mathcal{D}$.
  2. If $y^{(i)}$ is majority class *and* 3-NN classifies it as minority $\rightsquigarrow$ Remove $(\mathbf{x}^{(i)}, y^{(i)})$ from $\mathcal{D}$.
  3. If $y^{(i)}$ is minority class *and* 3-NN classifies it as majority $\rightsquigarrow$ Remove 3 nearest neighbors from $\mathcal{D}$.
- One-sided selection (OSS): Tomek link + CNN
- CNN + Tomek link: to reduce computation of finding Tomek links $\rightsquigarrow$ first use CNN and then remove the Tomek links.
- Clustering approaches: Class Purity Maximization (CPM) and Undersampling based on Clustering (SBC).