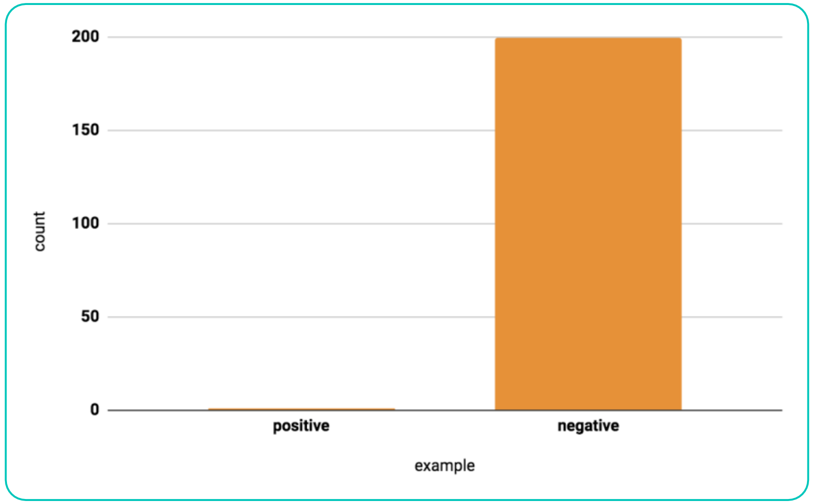
Addressing Class Imbalance in Machine Learning

Sara Iris Garcia

What is Class Imbalance?

- Data is not equally distributed.
- Most Machine Learning classifiers biased towards the majority class.
- Common problem in fraud detection, medical diagnosis, oil spill detection, etc.



Source: Google developers

Change the performance metric

- Accuracy is misleading in imbalance datasets.

High recall	High precision	Model is able to recognize the class
Low recall	High precision	Model can't recognize the class well but is highly trustable when it does
High recall	Low precision	Model can recognize the class well but does some misclassification
Low recall	Low precision	Model cant recognize the class well

- Try other metrics such as F1 measure and AUROC.

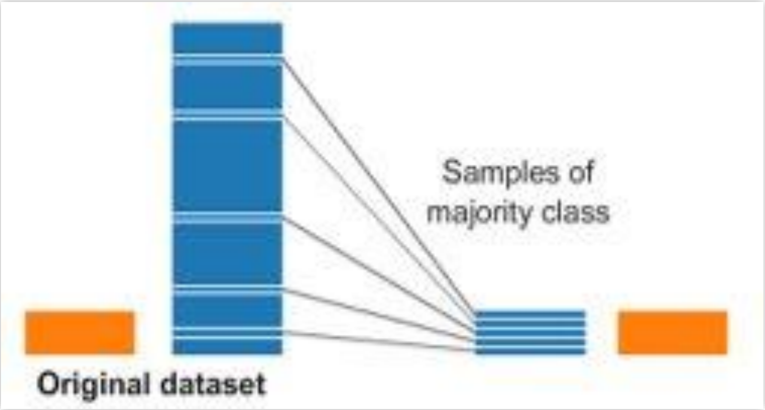
Methods for handling class imbalance

- Data-level techniques
  - Undersampling
  - Oversampling
  - Hybrid techniques
- Model-level techniques
  - Weight balancing
  - Ensemble Learning
- Hybrid techniques

Undersampling techniques

Undersampling consists in removing observations from the majority class until the class distribution is balanced.

- Random Undersampling
- Tomek Links
- Cluster Centroids



Source: Towards Data Science

Advantages

- Reduction of memory and computation needed since the data size decreases.

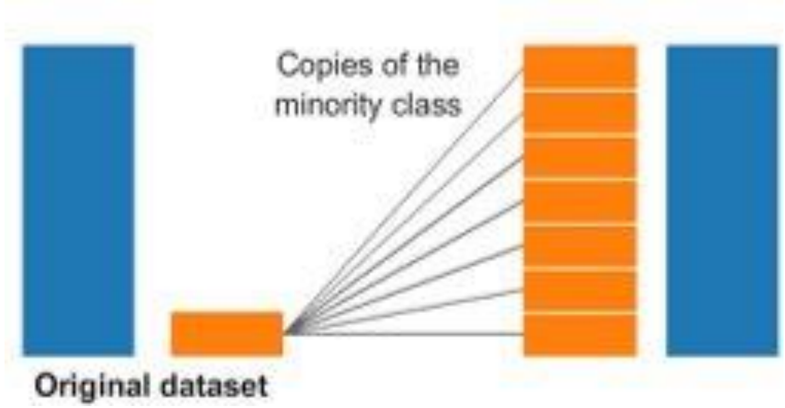
Disadvantages

- Risk of removing useful information.
- The chosen observations may not be an accurate representation of the population.

Oversampling

Oversampling consists in replicating observations from the minority class to balance the representation of the class.

- Random Oversampling
- SMOTE
- ADASYN



Source: Towards Data Science

Advantages

- No information loss.

Disadvantages

- Increases the probability of overfitting since it replicates observations from the minority class.
- The size of the data augments, meaning more running time.

Some rule of thumb

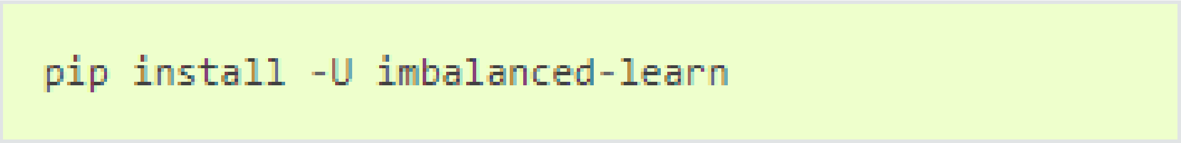
Consider applying Undersampling when the dataset is large.

Consider applying Oversampling when the dataset is small.

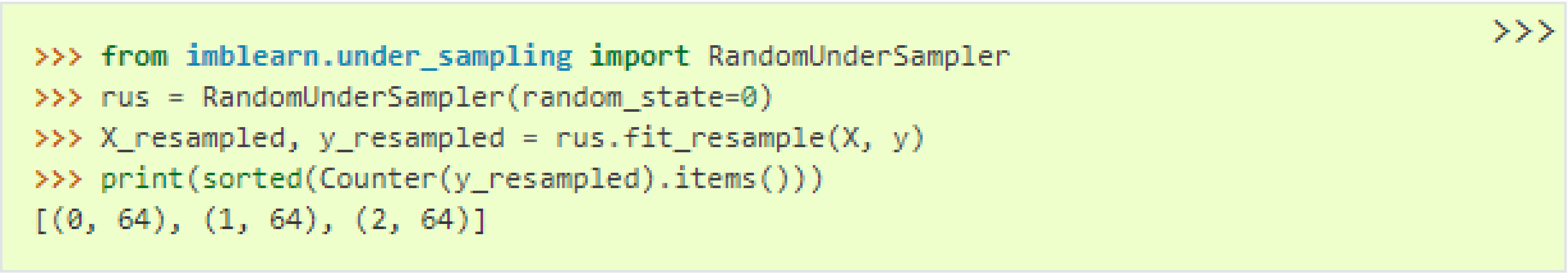
Consider testing random and non-random sampling.

Always resample only on the training data.

Implementation in Python



```
pip install -U imbalanced-learn
```



```
>>> from imblearn.under_sampling import RandomUnderSampler
>>> rus = RandomUnderSampler(random_state=0)
>>> X_resampled, y_resampled = rus.fit_resample(X, y)
>>> print(sorted(Counter(y_resampled).items()))
[(0, 64), (1, 64), (2, 64)]
```

Source: Documentation

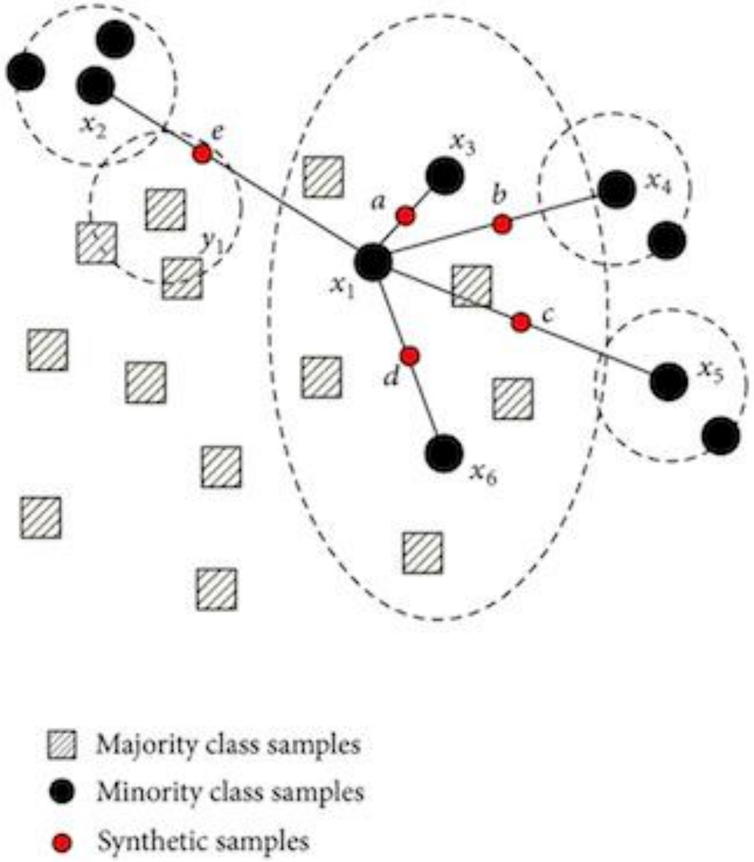
Hybrid techniques

Synthetic samples generation combining undersampling on the majority class and oversampling on the minority class

- SMOTE
- ADASYN
- SCUT

SMOTE
Synthetic Minority Over-sampling Technique

SMOTE consists in undersampling the majority class and oversampling the minority class by creating "synthetic" instances. These new instances are generated by calculating the distance between an observation and its nearest neighbors. The observations are randomly chosen depending upon the amount of oversampling required.



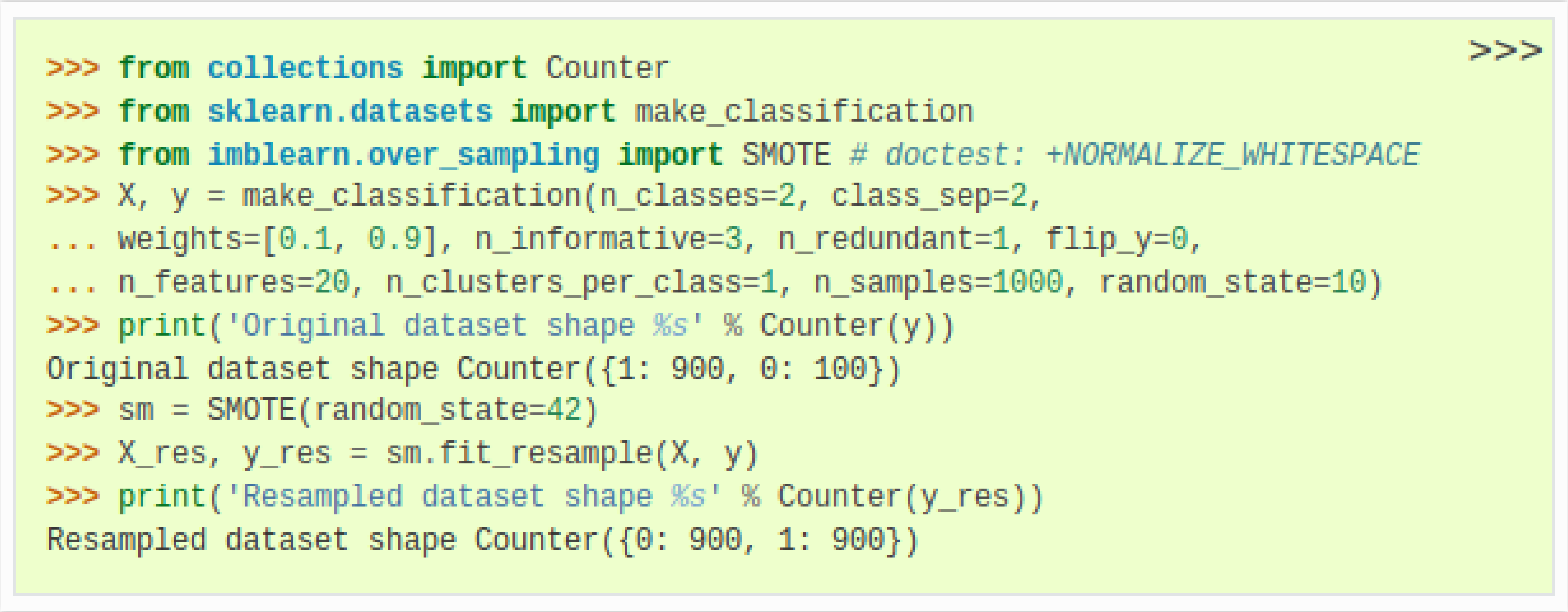
Advantages

- Reduction of the probability of overfitting, since new samples are generated.

Disadvantages

- The synthetic samples may overlap observations of the majority class.

Implementation in Python



```
>>> from collections import Counter
>>> from sklearn.datasets import make_classification
>>> from imblearn.over_sampling import SMOTE # doctest: +NORMALIZE_WHITESPACE
>>> X, y = make_classification(n_classes=2, class_sep=2,
... weights=[0.1, 0.9], n_informative=3, n_redundant=1, flip_y=0,
... n_features=20, n_clusters_per_class=1, n_samples=1000, random_state=10)
>>> print('Original dataset shape %s' % Counter(y))
Original dataset shape Counter({1: 900, 0: 100})
>>> sm = SMOTE(random_state=42)
>>> X_res, y_res = sm.fit_resample(X, y)
>>> print('Resampled dataset shape %s' % Counter(y_res))
Resampled dataset shape Counter({0: 900, 1: 900})
```

Source: Documentation

Model-based techniques

- Class weighting
- Ensemble of classifiers
    - Bootstrap aggregation or Bagging
    - Boosting

Class Weighting

Adding weights to the observations of the minority class into the cost function.

Focal Loss method down-weights the observations that are correctly classified.

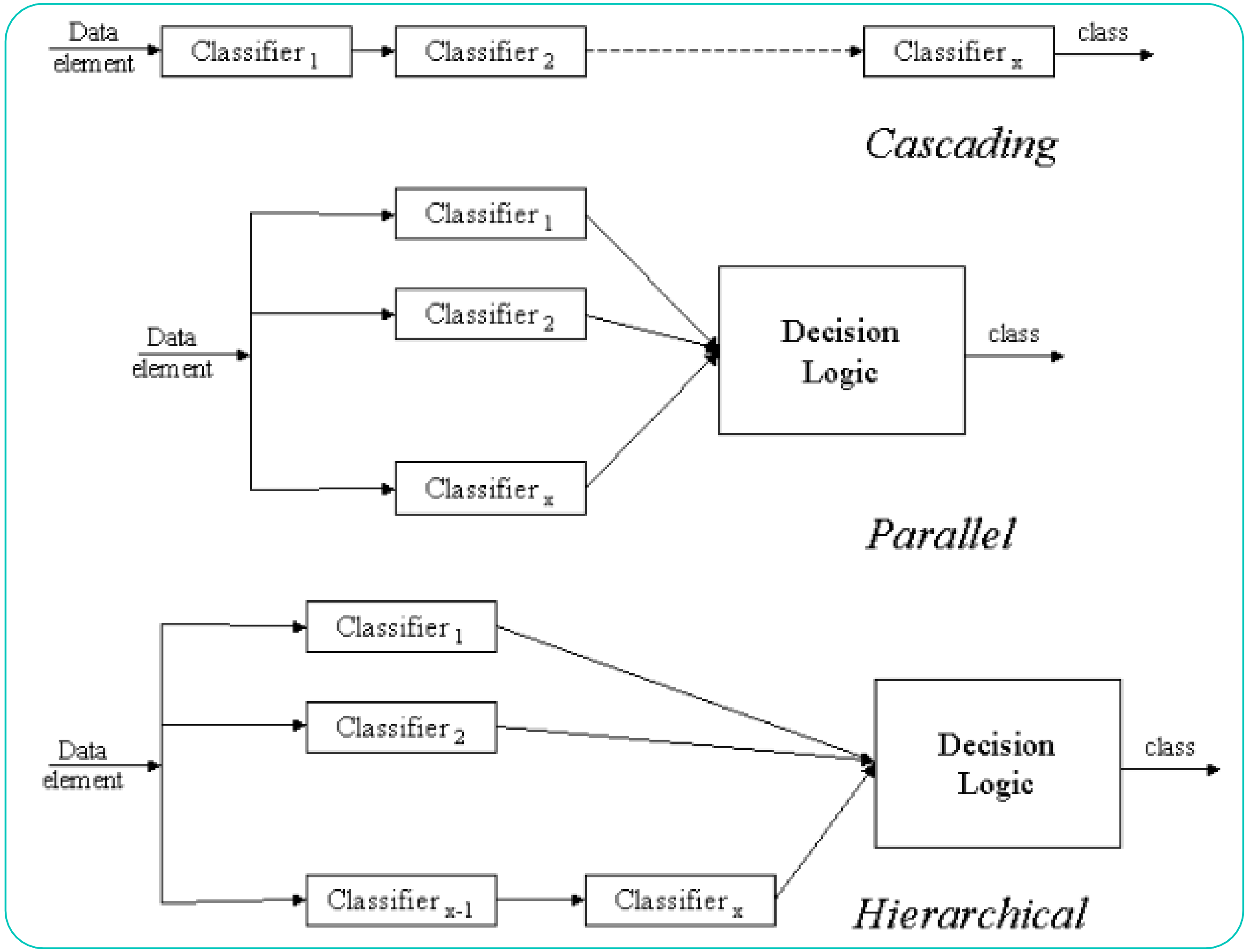
Read the paper

Ensemble of classifiers

When more classifiers are properly combined, the performance of the ensemble system will never be worse than the best performing classifier.

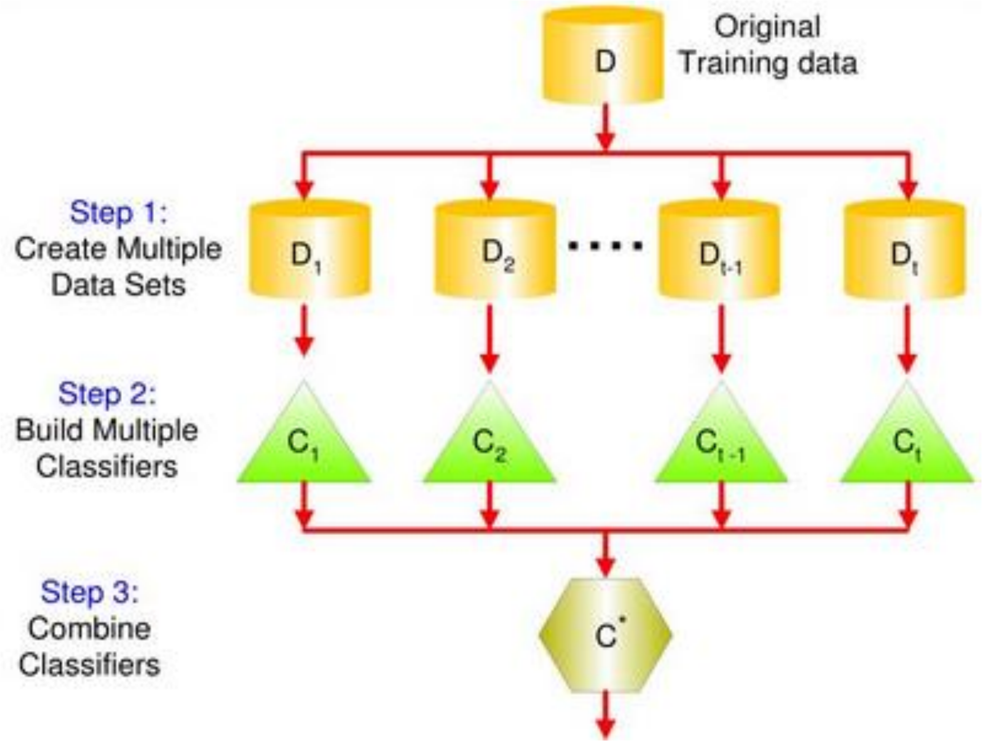
- Cascading
- Parallel
- Hierarchical

Topologies of the ensemble of classifiers



Boosting aggregation or Bagging

- Classifiers are given equal weights
- The predicted class is chosen by majority voting


D
Original Training data
Step 1: Create Multiple Data Sets
D_1
D_2
....
D_{t-1}
D_t
Step 2: Build Multiple Classifiers
C_1
C_2
C_{t-1}
C_t
Step 3: Combine Classifiers
C^*

Source: Analytics Vidhya

Advantages

- Useful when data is limited.
- Work better when the base classifiers are unstable.
- Less probability of overfitting by reducing the variance in the data.

Disadvantages

- Not suitable when the computation resources is limited.
- Loss of interpretability of the model.

Boosting

Weights are assigned to the models according on their performance.

- AdaBoost
- Gradient Boosting

Hybrid techniques

A common strategy consists in combine data sampling with ensemble of models.

- EasyEnsemble
- Balance Cascading
- SMOTEBoost

Final words

- Check your confusion matrix
- Take into consideration your computation resources
- Small data: consider Oversampling methods like Stratified sampling, SMOTE
- Large data: consider Undersampling methods like NearMiss or TomekLinks
- High variance: consider Bagging
- Biased model: consider Boosting

Questions?

Thanks!

ขอบคุณ

✉ sarairis.garcia@gmail.com

@montjoile

Further Reading

1. SMOTE: Synthetic Minority Over-sampling Technique
2. Focal Loss for Dense Object Detection
3. Survey on deep learning with class imbalance
4. The Right Way to Oversample in Predictive Modeling
5. SCUT: Multi-class imbalanced data classification using SMOTE and cluster-based undersampling

References

- DataCamp Diving Deep with Imbalanced Data
- TowardsDataScience Handling Imbalanced Datasets in Deep Learning
- Machine Learning Mastery 8 Tactics to Combat Imbalanced Classes in Your Machine Learning Dataset
- Analytics Vidhya How to handle Imbalanced Classification Problems in machine learning