Proceedings of CONF-CDS 2026 Symposium: Data-Centric AI Security: Securing Models, Learning Agents, and Autonomous Systems
DOI: 10.54254/2755-2721/2026.CH33069

Handling Class Imbalance in Machine Learning: A Review

Keyu Li

Vincenness University, Indiana, USA
2923053010@qq.com

Abstract. Many machine learning applications are faced with the challenge of class imbalance. Most traditional machine learning techniques only consider maximizing overall accuracy which will lead to the model being biased towards the majority class. As a result, the model will not be able to identify the minority class even if it was able to obtain high overall accuracy which is an issue when these samples are very rare in research or in applications where they are needed. This review summarizes three commonly used approaches to address this problem. It also clarifies the application logic of evaluation metrics tailored for imbalanced scenarios, including precision, recall, F1 score, and precision-recall curves. This review summarizes current progress in addressing class imbalance and highlights potential directions for future research.

Keywords: Class Imbalance, Imbalanced Learning, Sampling Methods

1. Introduction

In machine learning, it is not uncommon for sample sizes to differ across categories. An example of this phenomenon can be found in the medical record data, where there are plenty of healthy individuals whose data can be easily collected; however, there are usually only a few patient samples available. For this reason, models will typically predict the majority class by neglecting those patients that actually deserve care. The algorithms use imbalanced training sets; therefore, they are most likely to predict the predominant class [1]. An example of the problems created by this practice is that the algorithm results in an overall accuracy that is high but fails to identify the less frequently occurring class.

The issue has been addressed by several researchers with multiple methods, including: (1) methods using data-level modifications (i.e., modification of the numbers of occurrences in a data set) to address imbalance; (2) methods modifying algorithms (i.e., including change in loss function) to address the issue; and (3) ensemble methods to improve classification accuracy by using multiple classifiers in combination to create a combined classifier for improved performance.

None of the techniques above will solve every problem and the choice of which to use will again depend on the problem being solved. In practice, it often turns out that the methods seem to work better or worse depending on the size of dataset, the degree of imbalance in the classes and how the feature space is structured. Choosing a sensible approach often requires empirical comparisons rather than relying on theoretical advantages. Some methods will work better on small datasets while others will work better when the dataset is very large and very imbalanced.This review

DOI: 10.54254/2755-2721/2026.CH33069

explores several standard approaches and explains the advantages and disadvantages of each. The purpose of this paper is not to discuss all details, but to help readers understand the available methods and when they can be applied.

2. Literature review

2.1. The definition of class imbalance

In practical machine learning data, classes frequently have very different numbers of examples. This is a class imbalance. In binary classification – where there are just two classes – the less frequent class tends to be rare events (such as fraud or disease), and the more frequent class the normal ones. In multi class problems, some groups will have a lot of examples, and others very few. The amount of imbalance is usually expressed using the "imbalance ratio", which is typically around the number of examples in the more frequent class, divided by the number of examples in the less frequent class. There's no strong rule for what counts as "very imbalanced", but ratios of 100:1 or more are not unusual.

The problem has been known for many years. Initial studies investigated how imbalance can affect classifier performance, and how it might be measured. Though the definition of imbalance is straightforward, the effects on learning algorithms are often subtle and don't become obvious until models are applied to real data.

2.2. Class imbalance is a problem

Class imbalance creates problems, as most standard algorithms aim to improve overall accuracy without accounting for highly imbalanced class distributions. Indeed, in a balanced dataset, accuracy is a good performance metric. However, if one class is much larger than the other, a model can still achieve high accuracy by predicting that class. For example, imagine a dataset with 95% negative examples and 5% positive examples; a model that always guesses "negative" would achieve 95% accuracy but would fail in unusual situations. Research has demonstrated that conventional measures can be deceptive in these situations [1,2]. To better assess performance, alternative measures such as precision, recall, and F1-score are often used. These measures concentrate more on the less frequent class and better indicate whether the model has identified important features.

3. Methods

3.1. Data-level methods

When faced with imbalanced datasets, many people's first instinct is to adjust the dataset itself. The reasoning is straightforward: if one class has extremely few samples, why not simply increase the number of these minority samples? Though this idea is simple, it lies at the core of oversampling.

The most common way to address this is SMOTE [3]. Rather than simply duplicating minority samples, SMOTE interpolates to generate new samples. Geometrically you can think of SMOTE as expanding the minority class in feature space so that classifiers that draw decision boundaries can do so with lesser bias. It has been demonstrated to drastically improve recall for the minority class on benchmark datasets [3]. However, this method has drawbacks. When minority samples are sparsely distributed or overlap heavily with the majority, SMOTE may generate synthetic samples within the overlapping region. In such cases, oversampling can even blur the class boundary further.

DOI: 10.54254/2755-2721/2026.CH33069

To address this drawback, improvements aimed to improve the sampling process. Borderline-SMOTE only considers samples that fall near the decision boundary to make additional samples, as these points are the area of most interest for classification [4]. ADASYN makes additional samples in the area where learning is harder still [5]. This quality of the samples with respect to the class boundary makes this method more appealing theoretically than simple uniform oversampling, although in practice the result is heavily dataset dependent, and thus also is the value of the sample distribution in the feature space.

Of course, oversampling is not free from controversy itself. The review paper states that synthetically generated data is likely to distort the original data distribution and may even introduce some synthetic artifacts [6]. This goes against the implicit assumption that matching the class distribution as closely as possible is always good and leads to better generalisation. It also raises a fundamental question: Are people truly addressing the imbalance, or merely altering the dataset distribution without resolving model bias?

In contrast, undersampling alleviates imbalance by removing the samples from the majority class. Early methods, such as Tomek links, remove most highly similar samples from the minority class, thereby sharpening decision boundaries and reducing class overlap [1]. Later approaches, such as NearMiss, assume that the majority of samples near minority regions carry greater informational value, as they help define decision boundaries and indicate areas prone to misclassification by models [7]. Undersampling remains one of the most fundamental data-level approaches, as it directly adjusts class distributions without introducing artificial samples.

In summary, data-level methods are attractive for their simplicity and flexibility, enabling application before training virtually any classifier.

3.2. Algorithm-level methods

Algorithmic approaches do not delete or add samples; instead, they adjust the learning objective to focus more on minority-class samples.

One of the oldest implementations of this idea is cost-sensitive learning. Misclassifications should not all be treated the same since in imbalanced learning tasks, the true cost of misclassifying a minority class sample is much higher than misclassifying a majority class sample. By imposing greater penalties on minority class samples, the model emerges misclassifying them less, even if accuracy overall may decrease [8]. This approach alters the optimization objective without modifying the dataset itself.

In practice, cost-sensitive learning can be implemented in various ways. Many classifiers, such as logistic regression and support vector machines, allow direct specification of class weights, thereby increasing their contribution to the loss function. Rezvani et al provide a comprehensive review and empirical evaluation of SVM-based approaches for class-imbalanced learning, including weighted SVM variants [9].

This idea also carries over to deep learning. Researchers are increasingly using weights per category or designing different losses that focus better on examples that are hard to learn or minority examples over cross-entropy. One such example is the focal loss, which modifies cross-entropy loss and down-weights easy majority samples during training and focuses more heavily on hard minority examples [10].

Still, setting the weights or loss design is often not straightforward. Because of this, some studies have turned to ensemble methods.

DOI: 10.54254/2755-2721/2026.CH33069

3.3. Ensemble methods

As its name suggests, ensemble methods combine multiple models rather than relying on a single one. The core idea is that individual models may suffer from bias or instability, so combining multiple models aims to produce more reliable predictions, especially in scenarios with class imbalance.

Early works include EasyEnsemble and BalanceCascade [11]. EasyEnsemble samples random subsets from the majority class and pairs these subsets with all samples from the minority class. This process generates several balanced training sets. A classifier is trained on each subset and their outputs are combined. Although it does not solve the problem perfectly, it alleviates the model's tendency to blindly predict the majority class. BalanceCascade proposes an improvement on this idea. After classifiers have been learned, BalanceCascade discards majority class samples that have been correctly classified so that the resulting classifiers focus more on the remaining hard-to-classify samples.

Beyond these classical approaches, recent years have seen an increase in ensemble methods that combine sampling with boosting algorithms. For instance, CUSBoost first clusters the majority samples, performs undersampling within each cluster, and then applies a boosting algorithm to the processed data [12]. In contrast, SMOTEBoost integrates the SMOTE oversampling technique into the boosting procedure. At each boosting iteration, SMOTE generates synthetic minority class samples, increasing their representation in the training distribution and forcing the model to pay greater attention to these cases [13]. These approaches strongly demonstrate that sampling and ensemble learning need not be separate steps but can be integrated into a unified training process.

Recent surveys have also looked at the interplay between ensemble models and data augmentation [14]. Adding perturbed implementations of minority data samples help stabilize the model and improve recognition of the minority however the these designs also increase model complexity and generally have some associated increase in tuning requirements.

In summary, ensemble approaches serve as a practical alternative when single-model methods encounter limitations. They function by distributing the learning task across multiple models; this helps mitigate the tendency for models to favor majority classes and provides minority classes with more opportunities to be learned.

4. Evaluation and metrics

As discussed earlier, overall accuracy is not a reliable metric. In extremely imbalanced scenarios, classifiers that predict only the majority class can achieve high accuracy while completely failing to recognize minority-class samples, which are often the primary targets [1]. Therefore, alternative evaluation metrics are commonly introduced. Precision and Recall serve as foundational evaluation metrics: Precision quantifies the proportion of correctly predicted positive samples, while Recall quantifies the proportion of actual positive samples correctly identified. Furthermore, because of the trade-off between these two metrics, researchers use the F1-score. Defined as the harmonic mean of precision and recall, it provides a single metric that balances both.

Curves can be utilized to evaluate performance as well. The ROC curve and AUC are commonly applied in balanced environments. Conversely, the Precision-Recall curve and its AUC are generally better suited for imbalanced datasets since they inherently emphasize detecting the minority class of samples [15]. As such, PR curves will show how well the model can detect positive examples (the samples that are not listed as true negatives), and will not be as impacted by the overwhelming number of true negatives present in imbalanced datasets.

DOI: 10.54254/2755-2721/2026.CH33069

Another approach is cost-sensitive evaluation (note: this refers to the evaluation-level concept, distinct from the training-level cost-sensitive learning mentioned earlier). This method weights errors based on their actual impact, thereby more accurately reflecting the model's performance across different classes. In practical applications, researchers typically combine multiple metrics, such as precision, recall, and the F1 score, to provide a more comprehensive evaluation of model performance.

It is worth noting that real-world scenarios rarely involve isolated imbalanced data problems. Factors such as limited labels, noisy observations, and data distribution drift often coexist, making it harder to predict model performance across domains. Consequently, researchers typically combine multiple imbalance mitigation strategies and report diverse evaluation metrics to achieve more robust and reliable results. This approach reflects a growing pragmatic trend in recent research on imbalanced learning.

5. Conclusion

This review outlines three core approaches to solving class imbalance in machine learning—put simply, solutions fall into three directions: data-level, algorithm-level, and ensemble learning. Data-level methods balance data by adjusting sample distribution; they're simple and easy to use, but they either generate unrealistic synthetic samples or lose useful data. Algorithm-level methods directly modify the model's learning logic by assigning higher weights to minority classes, but there's no universal standard for setting these weights. Ensemble methods rely on multiple models working together, which offers better stability but costs more in computation. Clearly, no single method works for all scenarios; the right approach depends on the specific data situation.

Agreement maintains that overall accuracy isn't reliable by itself. To effectively assess the performance of models, we require metrics related to minority classes, such as precision, recall, and PR curve. Therefore, current research is reporting multiple metrics, rather than just using accuracy, to find loopholes.

Although there has been a lot of progress in this field, there are still significant challenges that need to be addressed. The first challenge relates to 'scalability' since some sampling or ensemble algorithms are very costly in terms of time and money to train models when data sets are very large or a model needs to process data in real-time from one source. Also, deep learning is very complicated by itself to begin with, and almost every way that the current research has to address class bias is usually done as a stand-alone component of a class imbalance; hence they are not well integrated into the overall architecture of the neural network.

The second direction worth studying is "dynamically changing class imbalance". For example, in practical applications such as fraud detection or surveillance systems, the number of samples per class can change over time. Methods that assume the class imbalance ratio remains constant will degrade over time. So we need an "adaptive" strategy that can automatically adjust the sampling method or weight settings while the model is running.

The third point is "interpretability"—in other words, the ability to explain why the model makes a particular prediction. When we adjust the model to focus more on minority-class samples, it's often harder to explain its prediction logic. Future research will likely stop fixating on single techniques and instead move toward "unified frameworks" that integrate data processing, model optimization, and evaluation metrics. As more applications need to target these rare events, class imbalance learning will remain a key focus in machine learning, not a niche field.

DOI: 10.54254/2755-2721/2026.CH33069

References

[1] He, H., & Garcia, E. A. (2009). Learning from imbalanced data. IEEE Transactions on knowledge and data engineering, 21(9), 1263-1284.

[2] Davis, J., & Goadrich, M. (2006, June). The relationship between Precision-Recall and ROC curves. In Proceedings of the 23rd International Conference on Machine learning (pp. 233-240).

[3] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321-357.

[4] Han, H., Wang, W. Y., & Mao, B. H. (2005, August). Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In International Conference on Intelligent Computing (pp. 878-887). Berlin, Heidelberg: Springer Berlin Heidelberg.

[5] He, H., Bai, Y., Garcia, E. A., & Li, S. (2008, June). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In 2008, IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence) (pp. 1322-1328). Ieee.

[6] Tarawneh, A. S., Hassanat, A. B., Altarawneh, G. A., & Almuhaimeed, A. (2022). Stop oversampling for class imbalance learning: A review. IEEE Access, 10, 47643-47660

[7] Mani, I., & Zhang, I. (2003, August). KNN approach to unbalanced data distributions: a case study involving information extraction. In Proceedings of workshop on learning from imbalanced datasets (Vol. 126, No. 1, pp. 1-7). United States: ICML.

[8] Elkan, C. (2001, August). The foundations of cost-sensitive learning. In International Joint Conference on Artificial Intelligence (Vol. 17, No. 1, pp. 973-978). Lawrence Erlbaum Associates Ltd.

[9] Rezvani, S., Pourpanah, F., Lim, C. P., & Wu, Q. J. (2024). Methods for class-imbalanced learning with support vector machines: a review and an empirical evaluation: S. Rezvani et al. Soft Computing, 28(20), 11873-11894.

[10] Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision (pp. 2980-2988).

[11] Liu, X. Y., Wu, J., & Zhou, Z. H. (2008). Exploratory undersampling for class-imbalance learning. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 39(2), 539-550.

[12] Rayhan, F., Ahmed, S., Mahbub, A., Jani, R., Shatabda, S., & Farid, D. M. (2017, December). Cusboost: Cluster-based under-sampling with boosting for imbalanced classification. In 2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS) (pp. 1-5). IEEE.

[13] Chawla, N. V., Lazarevic, A., Hall, L. O., & Bowyer, K. W. (2003, September). SMOTEBoost: Improving prediction of the minority class in boosting. In European Conference on Principles of Data Mining and Knowledge Discovery (pp. 107-119). Berlin, Heidelberg: Springer Berlin Heidelberg.

[14] Khan, A. A., Chaudhari, O., & Chandra, R. (2024). A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation. Expert Systems with Applications, 244, 122778.

[15] Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PloS one, 10(3), e0118432.