

## Article

# RegCGAN: Resampling with Regularized CGAN for Imbalanced Big Data Problem

Liwen Xu \*  and Ximeng Wang

College of Science, North China University of Technology, Beijing 100144, China;  
2020312060117@mail.ncut.edu.cn

\* Correspondence: xulw@ncut.edu.cn

## Abstract

We consider the imbalanced data problem involving a new class of resampling-based models for classification. These models are variants of the conditional generative adversarial networks. An entropy regularization approach (RegCGAN) is employed to implement the corresponding imbalanced data learning. Its basic framework is introduced. Theoretical and simulation-based analyses are performed to demonstrate the existence and uniqueness of RegCGAN's equilibrium point, and RegCGAN has excellent minority class prediction ability. We apply the results to two synthetically constructed and a real imbalanced dataset.

**Keywords:** resampling; imbalanced data; conditional generative adversarial network; entropy regularization; classification

**MSC:** 62H30; 62G09

## 1. Introduction

Extensive array of practical tasks, from credit card fraud detection to bioinformatics, face the persistent challenge of imbalanced class distributions. Standard learning algorithms often perform poorly on imbalanced large-scale datasets, as they tend to prioritize majority class accuracy to minimize overall training error. This results in a noticeable decline in accuracy, most significantly for the minority class. In real-world applications, misclassifying minority class samples often incurs greater costs than errors related to majority instances. Thus, imbalance-handling techniques focus on improving minority class accuracy to boost model performance in skewed datasets. Various approaches to handling class imbalance have been identified in the literature [1], where artificial data can be directly generated by applying generative models that learn the distribution of the minority class. Generative adversarial networks (GANs) represent a novel class of generative models that leverage neural networks to learn data distributions [2], particularly in synthesizing photorealistic images [3–6] and learning meaningful data representations [7–9], which has led to their widespread adoption across various domains [9–13].

The GAN architecture is modeled as a minimax game involving two players, where the generator  $G$  transforms random noise  $z$  into fake data, and the discriminator  $D$  distinguishes between real and synthetic inputs. The generator attempts to produce outputs that the discriminator cannot distinguish from real data, and this competition is formally defined as follows:

$$\min_G \max_D U(G, D) = E_{x \sim f(x)} [\ln D(x)] + E_{z \sim f_z(z)} [\ln(1 - D(G(z)))]. \quad (1)$$



Academic Editor: Abbe Mowshowitz

Received: 13 May 2025

Revised: 13 June 2025

Accepted: 19 June 2025

Published: 21 June 2025

**Citation:** Xu, L.; Wang, X. RegCGAN: Resampling with Regularized CGAN for Imbalanced Big Data Problem.

*Axioms* **2025**, *14*, 485. <https://doi.org/10.3390/axioms14070485>

**Copyright:** © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Here,  $f(x)$  refers to the true data distribution, and  $f_z(z)$  corresponds to the prior over the noise variable  $z$ . From a theoretical perspective, this adversarial setup between the generator and discriminator is adequate for enabling unsupervised learning. Under the assumption of nonparametric models, once equilibrium is achieved, the distribution generated by  $G$  aligns with that of the real data.

Although unsupervised GANs show great potential, the learned representations often underperform in downstream tasks such as classification and disentangled conditional generation, which are highly relevant in practice [14–20]. Unsupervised deep generative models typically yield fewer discriminative representations than those learned by supervised DNNs, leading to suboptimal predictive performance [21]. In addition, learning disentangled representations that correspond to interpretable physical factors and generating data samples conditioned on these factors remains a difficult challenge in unsupervised settings and largely relies on model inductive biases [22].

One intuitive approach to overcome these limitations is to introduce label information into the GAN framework [23–28]. Shrivastava et al. [29] trained a conditional GAN on unlabeled data to generate alternative versions of given real images. Conditional GANs (cGANs) extend the original GAN framework by incorporating external information into the generator’s input during training. Douzas and Bacao [30] applied cGANs to binary-class imbalanced datasets, where the conditioning information was provided by the class labels.

In general, GANs have demonstrated effectiveness not only in generating realistic samples but also in semi-supervised learning (SSL). Within the same two-player game framework, CatGAN [24] extends the original GAN by incorporating a classification-discriminative network and a novel objective function. However, existing GAN-based approaches for SSL face two main challenges: (1) The generator and discriminator (which also serves as a classifier) often fail to reach an optimal balance simultaneously. (2) The generator lacks the ability to control the semantic content of the generated samples.

According to [31], these challenges stem from the inherent limitations of a two-player architecture, where the discriminator is required to perform two conflicting roles—distinguishing fake samples and predicting class labels. Regarding challenge (2), disentangling meaningful factors (such as object labels) from latent representations with limited supervision remains a fundamental issue in SSL.

To tackle these issues, ref. [31] introduced a versatile game-theoretic model for joint classification and conditional generation tasks. Their model introduces three networks and explicitly considers a real data-label joint distribution and two conditional distributions output by the networks. This multi-network setup allows for better control over both classification and generation tasks.

It is important to note, however, that such SSL-oriented network architectures are not directly applicable to our problem, as they typically assume a balanced class distribution across categories. This paper targets the analysis and processing of imbalanced big data. We aim to develop novel resampling-based inference methods and theoretical frameworks grounded in deep learning, thereby offering new statistical support for both the advancement of big data technologies and their practical applications across industries.

In Section 2, we first describe a regularized CGAN model, as well as the generative resampling approach. Moreover, we provide the theoretical analysis of RegCGAN under nonparametric assumptions. Section 3 reviews the existing literature relevant to RegCGAN. In Section 4, we describe the applications of the RegCGAN to a real SVHN dataset and the synthetically imbalanced variants of well-refined datasets, CIFAR-10 and CIFAR-100. Finally, Section 5 concludes the paper by summarizing the key findings and outlining possible avenues for future work.

## 2. Methodology

In this section, we build upon the main model by combining the conditional generative adversarial network (CGAN) framework by incorporating a cross-entropy loss term, resulting in a classification-oriented network called RegCGAN (regularized CGAN). The model addresses the shortcomings of traditional generative methods on imbalanced data by directly optimizing classification performance.

### 2.1. Resampling Model for Imbalanced Data

RegCGAN is capable of fully and effectively utilizing all available information in imbalanced big data while addressing both theoretical and practical challenges. Specifically, we investigate the convergence properties of the generator and classifier in this newly proposed resampling-based model, as well as the robustness of the training algorithm and its applicability in real-world scenarios.

The architecture of the proposed model is as follows: RegCGAN is composed of three core modules: (a) A classifier  $C$  serves as the primary objective of the model. It approximates the conditional distribution  $f_c(y|x) \approx f(y|x)$ , aiming to optimize classification performance. (b) A generator  $G$  conditioned on class labels approximates the conditional distribution  $f_g(y|x) \approx f(y|x)$  from the complementary direction. By generating auxiliary samples, it helps mitigate the class imbalance problem. (c) A discriminator  $D$  determines whether a data pair  $(x, y)$  comes from the real joint distribution  $f(x, y)$  or is generated synthetically.

RegCGAN aims to attain distributional alignment by guiding the generator and classifier to produce the distribution that matches the real data distribution. This alignment allows the model to effectively overcome the limitations posed by imbalanced datasets.

More specifically, we train the generator with the dual guidance of both the classifier—driven by prediction accuracy—and the discriminator. This approach ensures that the generated samples are not only realistic but also beneficial for the classification task. Under the model architecture illustrated in Figure 1, we consider the following utility function:

$$\begin{aligned} \min_{C, G} \max_D U(C, G, D) &= E_{(x, y) \sim f(x, y)} [\ln D(x, y)] + E_{(z, y) \sim f_Z(z) f(y)} [\ln(1 - D(G(y, z), y))] \\ &\quad + E_{(x, y) \sim f(x, y)} [-\ln f_c(y|x)] + \alpha E_{(z, y) \sim f_Z(z) f(y)} [-\ln f_c(y|G(y, z))] \\ &= E_{(x, y) \sim f(x, y)} [\ln D(x, y)] + E_{(x, y) \sim f_g(x, y)} [\ln(1 - D(x, y))] \\ &\quad + E_{(x, y) \sim f(x, y)} [-\ln f_c(y|x)] + \alpha E_{(x, y) \sim f_g(x, y)} [-\ln f_c(y|x)] \end{aligned} \quad (2)$$

where the sum of the first two terms corresponds to the utility function of the standard CGAN, denoted as follows:

$$V(G, D) = E_{(x, y) \sim f(x, y)} [\ln D(x, y)] + E_{(x, y) \sim f_g(x, y)} [\ln(1 - D(x, y))] \quad (3)$$

Among the latter two terms, the first is denoted as  $L_C = E_{(x, y) \sim f(x, y)} [-\ln f_c(y|x)]$ , representing the cross-entropy loss of the classifier  $C$  on real data. The second is denoted as  $L_G = E_{(x, y) \sim f_g(x, y)} [-\ln f_c(y|x)]$ , which corresponds to the cross-entropy loss of the classifier  $C$  on the generated data. We fixed the entropy regularization coefficient  $\alpha \in (0, 1)$  at  $1/2$  to focus on the balanced setting throughout this study. In the study,  $\alpha$  was fixed at  $1/2$  based on both empirical observations and theoretical considerations. Specifically, our theoretical analysis established that for any  $\alpha \in (0, 1)$ , the training process of RegCGAN was guaranteed to converge. This provides a general foundation for choosing  $\alpha$  within this interval without compromising theoretical soundness.

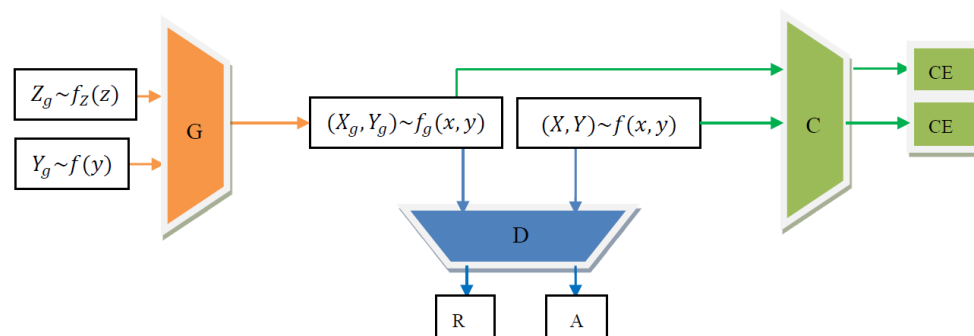


**Figure 1.** The architecture of the RegCGAN (regularized CGAN) model. Here,  $C$ ,  $G$ , and  $D$  denote the classifier, generator, and discriminator, respectively. The labels “R” and “A” represent rejection and acceptance decisions made by the discriminator. The notation “CE” refers to the cross-entropy loss, which guides the classifier to optimize classification accuracy.

With respect to the model depicted in Figure 1 and the utility function defined in Equation (2), this section focuses on the following key aspects:

- The equilibrium point in RegCGAN is demonstrated to exist and be unique;
- The minority class prediction capability of RegCGAN: specifically, we study whether the generator can accurately produce resampled data that follow the minority class distribution, and whether the classifier—trained using the utility function in Equation (2)—can, with the aid of the generator, effectively improve the prediction accuracy for the minority class while making full use of the information contained in the entire dataset.

## 2.2. Theoretical Analysis

In this subsection, a rigorous theoretical investigation of RegCGAN is carried out under nonparametric assumptions in this subsection. Analogous to the original GAN framework, we present the following lemma regarding the behavior of RegCGAN.

**Lemma 1 ([2]).** Given a fixed generator  $G$ , the optimal solution for the discriminator  $D$  under the objective  $V(G, D)$  is

$$D_G^*(x, y) = \frac{f(x, y)}{f(x, y) + f_g(x, y)}. \quad (4)$$

**Proof.** With  $G$  fixed, the objective function  $V(G, D)$  is reformulated as

$$\begin{aligned} V(G, D) &= \iint f(x, y) \ln D(x, y) dy dx + \iint f_g(x, y) \ln(1 - D(x, y)) dy dx \\ &= \iint f(x, y) \ln D(x, y) dy dx + f_g(x, y) \log(1 - D(x, y)) dy dx. \end{aligned}$$

Under the nonparametric setting, that is, for any given  $(x, y)$ , the discriminator  $D(x, y)$  is allowed to take arbitrary values. To maximize the integrand in the objective function  $V(G, D)$

$$h(D) = f(x, y) \ln D(x, y) + f_g(x, y) \log(1 - D(x, y))$$

for any fixed  $(x, y)$ , it suffices that  $D(x, y)$  satisfies the following equation:

$$\frac{dh(D)}{dD} = 0$$

It is straightforward to verify that the solution to this equation is  $D(x, y) = \frac{f(x, y)}{p(x, y) + f_g(x, y)}$ . The second derivative  $h''(D) = -\frac{f(x, y)}{D^2} - \frac{f_g(x, y)}{(1-D)^2}$  is negative, which proves that it is a maximum.  $\square$

**Lemma 2 ([2]).**  $V(G, D)$  reaches its global minimum precisely when the generated distribution  $f_g(x, y)$  matches the real distribution  $f(x, y)$ .

**Proof.** Given  $D_G^*(x, y)$ , we can obtain from Lemma 1 that

$$\begin{aligned} \max_D V(G, D) &= V(G, D^*) \\ &= \iint f(x, y) \ln \frac{f(x, y)}{f(x, y) + f_g(x, y)} dy dx + \iint f_g(x, y) \log \frac{f_g(x, y)}{f(x, y) + f_g(x, y)} dy dx. \end{aligned}$$

Noting that the  $V(G, D^*)$  can be rewritten as

$$V(G, D^*) = -\ln 4 + 2\mathbb{D}_{JS}(f(x, y) || f_g(x, y)), \quad (5)$$

where  $\mathbb{D}_{JS}$  is the Jensen–Shannon divergence, which is always non-negative, and the unique optimum is achieved if and only if  $f(x, y) = f_g(x, y)$ .  $\square$

**Theorem 1.** The equilibrium of  $U(C, G, D)$  is achieved if and only if the joint distributions satisfy  $f(x, y) = f_g(x, y) = f_c(x, y)$ .

**Proof.** Based on the defined Equations (2) and (3),

$$U(C, G, D) = V(G, D) + L_C + \alpha L_G$$

where

$$L_C = E_{(x, y) \sim f(x, y)} [-\ln f_c(y|x)] = \mathbb{D}_{KL}(f(x, y) || f_c(x, y)) + \mathbb{H}_f(y|x).$$

where  $\mathbb{D}_{KL}$  denotes the KL divergence, and  $\mathbb{H}$  is the conditional entropy. It is important to note that the second term in the above expression is independent of  $C, G, D$ , and hence the objective  $L_C$  reaches its minimum when the KL divergence  $\mathbb{D}_{KL}(f(x, y) || f_c(x, y))$  is minimized, attaining zero only if  $f(x, y)$  matches  $f_c(x, y)$ . Similarly,  $L_G$  reaches its minimum if and only if  $f_g(x, y) = f_c(x, y)$ .  $\square$

Noting that  $L_C + \alpha L_G$  depends only on the pair  $(C, G)$ , and that

$$\max_D U(C, G, D) = \max_D V(G, D) + L_C + \alpha L_G$$

it follows from Lemma 2 and the non-negativity of  $L_C + \alpha L_G$  that the global minimum of the objective is attained if and only if  $f(x, y) = f_g(x, y) = f_c(x, y)$ . This completes the proof.

Theorem 1 demonstrates that, under the proposed model framework, RegCGAN can theoretically guarantee the optimality of the resulting classifier  $C$ .

### 2.3. Optimization

RegCGAN decouples the hypothesis spaces of the discriminator and classifier, allowing for the integration of recent innovations from supervised learning and GAN research, such as advanced architectures and loss designs. Algorithm 1 demonstrates the overall training pipeline.

**Algorithm 1** Minibatch Minibatch SGD training of RegCGAN.

- 1: **for all** numbers of training iterations **do**
- 2:   Generate a minibatch of  $m_g$  synthetic samples  $(x_g, y_g) \sim f_g(x, y)$  and  $m_d$  real labeled samples  $(x, y) \sim f(x, y)$ ;
- 3:   Update  $D$  by performing stochastic gradient ascent as defined in Equation (2):

$$\nabla_{\theta_d} \left[ \frac{1}{m_d} \sum_{(x_d, y_d)} \ln D(x_d, y_d) + \frac{1}{m_g} \sum_{(x_g, y_g)} \ln(1 - D(x_g, y_g)) \right]$$

- 4:   Update  $C$  performing stochastic gradient ascent as defined in Equation (2):

$$\nabla_{\theta_c} \left[ -\frac{1}{m_d} \sum_{(x_d, y_d)} \ln f_c(y_d | x_d) - \alpha \frac{1}{m_g} \sum_{(x_g, y_g)} \ln f_c(y_g | x_g) \right].$$

- 5:   Update  $G$  performing stochastic gradient ascent as defined in Equation (2):

$$\nabla_{\theta_g} \left[ \frac{1}{m_g} \sum_{(x_g, y_g)} \ln(1 - D(x_g, y_g)) \right]$$

- 6: **end for**

**3. Related Work**

Learning from minority classes is often crucial, as they may correspond to rare but significant events [32], or because collecting such data is expensive and challenging [33]. Most machine learning algorithms are designed to optimize predictive accuracy and generalization performance. However, this inductive bias can become problematic when dealing with imbalanced datasets [34].

Firstly, when model training is driven by maximizing overall accuracy, it tends to favor the majority class, as it dominates the dataset. Secondly, decision rules targeting the minority (positive) class are usually highly specific with limited coverage, making them more likely to be rejected in favor of broader rules that classify the majority (negative) class. In practice, differentiating between noise and minority class samples is challenging, often causing classifiers to ignore or incorrectly label minority instances. Numerous strategies have been proposed to address the challenge of class imbalance, applicable to both standard learning models and ensemble frameworks [35–37]. These methods are generally divided into three main categories:

- **Supervised Learning:** The majority of existing research addresses class imbalance within the supervised learning framework. These approaches are generally grouped into three categories: (a) Sampling-based data-level methods mitigate class imbalance by modifying the training dataset. These are considered external strategies [38–42]. (b) Algorithm-level strategies adapt or redesign learning algorithms to emphasize the minority class, making them internal methods that integrate imbalance awareness directly into the learning process [43–45]. (c) Cost-sensitive methods assign higher penalties to errors involving minority class instances. These techniques can be applied at the data level, algorithm level, or both, with the goal of reducing high-cost misclassifications [46–50]. However, none of these approaches can be directly applied to unsupervised scenarios, as they inherently depend on class labels or prediction outputs tied to labels. Recent studies have also revealed that the feature extractor (backbone) and the classifier can be trained separately [51,52], inspiring new strategies such as imbalance-aware pre-training and later fine-tuning for target tasks.

- **Self-Supervised Learning:** The exploration of self-supervised learning on naturally imbalanced datasets was initiated by [53]. Their findings suggest that pre-training with self-supervised tasks, such as image rotation [54] or contrastive learning methods like MoCo [55], consistently improves performance over direct end-to-end training. This implies a regularizing effect that leads to a more balanced feature representation. Subsequent studies [56,57] further validate the advantage of contrastive learning in mitigating data imbalance while also pointing out that it does not completely resolve the issue.
- **Active Learning:** In conventional machine learning, active learning has been widely studied as an effective approach to data-efficient sampling, with techniques including information-theoretic approaches [58], ensemble-based strategies [59,60], and uncertainty sampling [61,62]. A recent study [63] addressed the challenge of imbalanced seed data in active learning by introducing a model-aware K-center sampling strategy, a unified sampling framework designed to enhance learning efficiency in such settings.

One key benefit of data-level approaches is their flexibility, as they can be applied regardless of the specific classifier being used. Additionally, data-level techniques allow for preprocessing datasets in advance, enabling the reuse of the same processed data across multiple classifiers, thereby eliminating the need for repeated preparation. Various data rebalancing strategies can be employed during preprocessing, which generally fall into three categories: undersampling, oversampling, and hybrid approaches. We also observe that recent concurrent studies [64–66] propose domain-specific techniques, including the use of energy functionals and finite element method (FEM) frameworks for defect detection, as well as fuzzy divergence-based methods for quantifying classifier uncertainty and confidence in decision-making. Incorporating such techniques into RegCGAN may, in principle, lead to improved performance, and we consider this a promising direction for future exploration.

## 4. Application

Open-world data typically exhibit a long-tail distribution, which exacerbates class imbalance in supervised and semi-supervised learning. This work addresses this issue through RegCGAN, a unified resampling approach that improves classification performance by separating the generator from the classifier. In this section, we are interested in uncovering differences in the classification accuracy between the RegCGAN-based resampling model and the original conventional convolutional neural networks without resampling.

### 4.1. Setting

On the one hand, this study creates artificially imbalanced versions of standard benchmark datasets such as CIFAR-10 and CIFAR-100. In exactly the same way as [67], we consider two types of imbalance: (a) long-tailed distribution (see Figure 2); (b) step distribution (see Figure 3).

On the other hand, as an application, we consider Street View House Numbers (SVHN) which consists of images extracted from Google Street View [68].

GAN training is often time-consuming, as it involves repeated adversarial updates between the generator and discriminator to reach convergence. To accelerate training speed, researchers often explore approaches such as employing more efficient optimization algorithms and reducing network complexity. In the simulated experiments described in this section, we adopted a pre-trained model strategy to enhance training efficiency. Specifically, for the RegCGAN model on the CIFAR10 dataset, we conducted approximately 30,000 iterations, requiring about 75 h of training on an NVIDIA P100 GPU with Adam

optimizer, which was applied to all three networks. All networks utilized ReLU activation functions, with batch size configured as 100. The learning rates for the classifier, generator, and discriminator were set to 0.003, 0.0001, and 0.0001, respectively, while their network architectures employed CNN and ResNet frameworks. Figure 4 illustrates the training learning curve of RegCGAN.

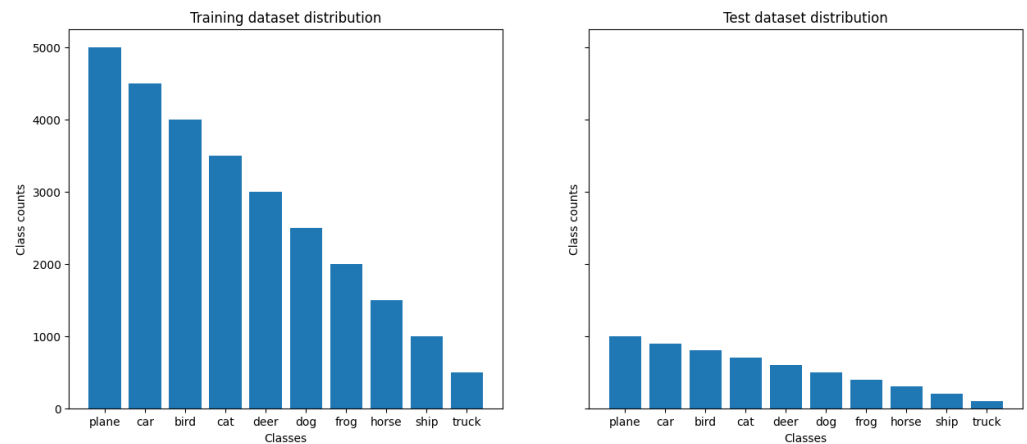


**Figure 2.** Long-tailed distributions of training and testing examples on imbalanced variants of CIFAR-10.

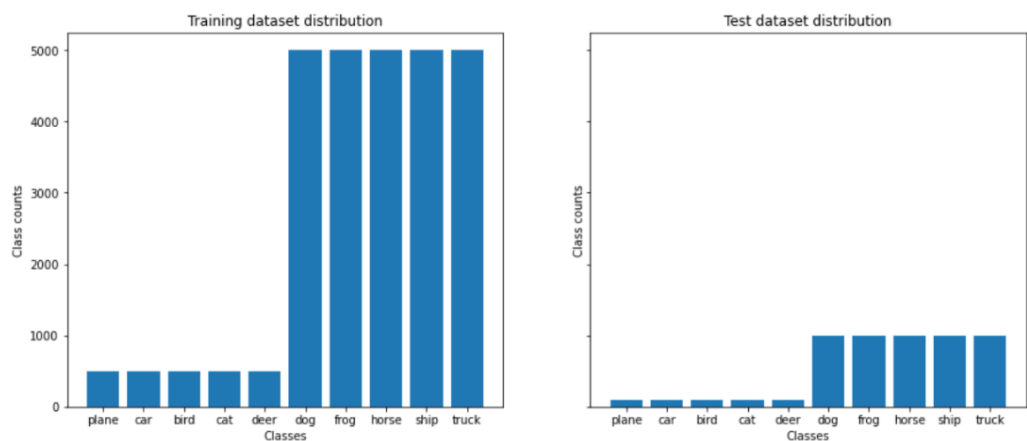


**Figure 3.** Step distributions of training and testing examples on imbalanced variants of CIFAR-10.

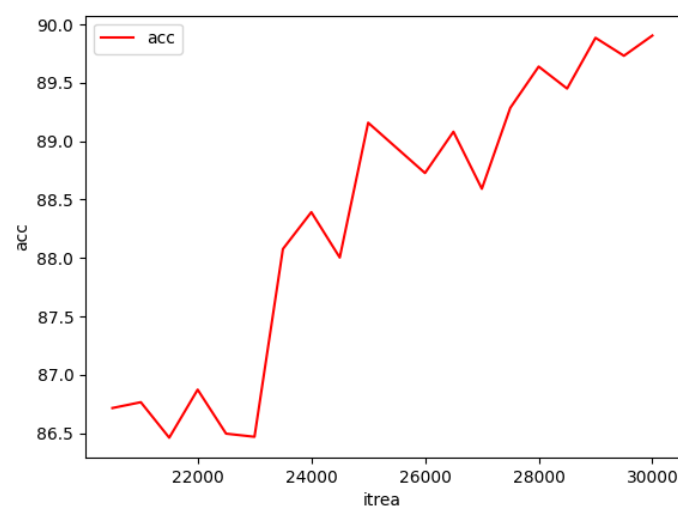


**Figure 4.** Learning curve of RegCGAN.

In our empirical experiments on the SVHN dataset, the experimental configuration for RegCGAN training remained fundamentally consistent with the simulated experiments, with the notable exception of extending the iteration count to 40,000 for this particular dataset. This study used balanced accuracy to assess image classification performance [69].

#### 4.2. Experimental Results

This section introduces a CNN baseline for validating the performance of RegCGAN. A comparative simulation study was conducted employing RegCGAN alongside the baseline CNN model on both the original CIFAR10 dataset and its imbalanced variant. The model efficacy was evaluated through prediction accuracy rates, with the comparative results presented in the table below.

As indicated in Table 1, the CNN performs poorly regardless of whether the dataset is balanced or imbalanced. Unlike baseline models, RegCGAN enhances prediction accuracy by using generative resampling of minority samples during training, which helps the classifier better capture their feature representations.

**Table 1.** Prediction accuracy (%) for CIFAR-10.

| Model   | Balanced Data | Long-Tailed Data | Step Data |
|---------|---------------|------------------|-----------|
| CNN     | 71.75         | 69.80            | 71.13     |
| RegCGAN | 80.93         | 81.60            | 86.38     |

Compared to the CIFAR-10 dataset, the most significant difference in the CIFAR-100 dataset lies in its larger number of categories—100 in total. Similar to the experimental results on CIFAR-10, the classifier incorporating RegCGAN achieves substantially higher prediction accuracy than the baseline CNN model under both balanced and imbalanced data settings. Specifically, the prediction accuracies of the baseline CNN model are 40.08% and 41.26% for the balanced and imbalanced datasets, respectively; in contrast, the classifier enhanced with RegCGAN achieves prediction accuracies of 49.35% and 56.09%, respectively.

On the other hand, Figure 5 shows the CIFAR-100 images generated by the RegCGAN model: the left panel corresponds to iteration 3000, while the right panel represents outputs at iteration 30,000. As training progressed, the quality of generated images improved significantly, becoming increasingly similar to real images.

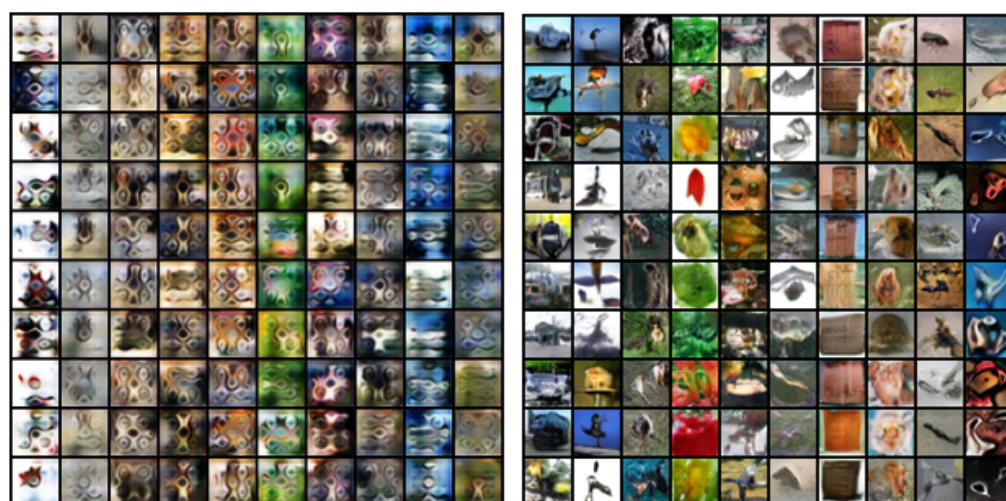


**Figure 5.** Progression of generated samples at different training stages of RegCGAN.

### 4.3. Empirical Results

SVHN consists of images sourced from Google Street View [68] and exhibits an inherently long-tailed distribution. In this subsection, RegCGAN is applied to the real-world imbalanced dataset SVHN. The results show that the proposed model achieves a prediction accuracy of 92.54%, outperforming both the standard convolutional neural network and the residual network. This further demonstrates the effectiveness of the model in addressing imbalanced data problems. Detailed results are presented in Table 2.

**Table 2.** Prediction accuracy (%) for SVHN.

| Model    | CNN   | Resnet34 | RegCGAN |
|----------|-------|----------|---------|
| accuracy | 89.30 | 90.68    | 92.54   |

## 5. Conclusions and Discussion

A generative resampling method was introduced in this study to address the widespread problem of data imbalance prevalent in real-world applications. Leveraging the conditional GAN structure, we proposed RegCGAN to improve accuracy in classification tasks. A theoretical investigation was carried out to verify that RegCGAN uses a unique and well-defined optimal classifier. The results on both synthetic and real imbalanced datasets show that the generator accurately models the minority class distribution during resampling. The classifier trained with the proposed utility function, aided by the generator, significantly improves the prediction accuracy for minority classes while effectively leveraging information from the entire dataset.

In many real-world scenarios, labeled data are scarce or entirely unavailable [63,70], and the data distribution in the open world is highly diverse, often characterized by long-tail patterns [71,72]. As a result, semi-supervised and unsupervised learning methods under imbalanced conditions emerge as important directions for future research. To illustrate how our framework could be extended, we take semi-supervised learning as an example and propose a simple idea for its potential adaptation.

We adopt a minor assumption in semi-supervised learning that the marginal input distribution is easy to sample from the marginal distribution  $f(x)$ . In this scenario, after a sample  $x$  is drawn from  $f(x)$ , the discriminator  $C$  produces a fake label  $y$  given  $x$  following the conditional distribution  $f_c(y|x)$ . As a result, the fake input-label pair is a sample from the joint distribution  $f_c(x, y) = f(x)f_c(y|x)$ . For the semi-supervised learning task, in view of Equation (2), we consider the following utility function:

$$\begin{aligned}
 \min_{C, G} \max_D \tilde{U}(C, G, D) &= U(C, G, D) + E_{(x, y) \sim f_c(x, y)} [\ln(1 - D(x, y))] \\
 &= E_{(x, y) \sim f(x, y)} [\ln D(x, y)] + E_{(x, y) \sim f_g(x, y)} [\ln(1 - D(x, y))] \\
 &\quad + E_{(x, y) \sim f(x, y)} [-\ln f_c(y|x)] + \alpha E_{(x, y) \sim f_g(x, y)} [-\ln f_c(y|x)] \\
 &\quad + E_{(x, y) \sim f_c(x, y)} [\ln(1 - D(x, y))].
 \end{aligned}$$

The remaining procedure follows a similar implementation to that of the RegCGAN algorithm. In fact, a valuable direction for future work would be to provide theoretical foundations and empirical analyses analogous to those conducted for RegCGAN.

**Author Contributions:** Conceptualization, L.X. and X.W.; methodology, L.X. and X.W.; software, X.W.; validation, L.X. and X.W.; formal analysis, L.X.; investigation, L.X. and X.W.; resources, X.W.; data curation, X.W.; writing—original draft preparation, L.X. and X.W.; writing—review and editing, L.X. and X.W.; visualization, X.W.; supervision, L.X.; project administration, L.X.; funding acquisition, L.X. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Social Science Foundation of China under Grant 20BTJ046.

**Data Availability Statement:** All the experimental data related to this paper can be requested from the author Ximeng Wang via email 2020312060117@mail.ncut.edu.cn.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

|         |                                                        |
|---------|--------------------------------------------------------|
| GAN     | Generative adversarial network                         |
| CGAN    | Conditional generative adversarial network             |
| RegCGAN | Regularized conditional generative adversarial network |
| SSL     | Semi-supervised learning                               |
| FEM     | Finite element method                                  |

## References

1. Fernández, A.; López, V.; Galar, M.; Del Jesus, M.J.; Herrera, F. Analysing the classification of imbalanced data-sets with multiple classes: Binarization techniques and ad-hoc approaches. *Knowl. Based Syst.* **2013**, *42*, 97–110. [[CrossRef](#)]
2. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 1–9.
3. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv* **2015**, arXiv:1511.06434.
4. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein generative adversarial networks. In Proceedings of the International Conference on Machine Learning, PMLR, Sydney, Australia, 6–11 August 2017; pp. 214–223.
5. Karras, T.; Laine, S.; Aila, T. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4401–4410.
6. Brock, A.; Donahue, J.; Simonyan, K. Large scale GAN training for high fidelity natural image synthesis. *arXiv* **2018**, arXiv:1809.11096.
7. Donahue, J.; Krähenbühl, P.; Darrell, T. Adversarial feature learning. *arXiv* **2016**, arXiv:1605.09782.
8. Dumoulin, V.; Belghazi, I.; Poole, B.; Mastropietro, O.; Lamb, A.; Arjovsky, M.; Courville, A. Adversarially learned inference. *arXiv* **2016**, arXiv:1606.00704.
9. Donahue, J.; Simonyan, K. Large scale adversarial representation learning. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 1–11.
10. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1125–1134.
11. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2223–2232.
12. Zhang, H.; Xu, T.; Li, H.; Zhang, S.; Wang, X.; Huang, X.; Metaxas, D.N. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5907–5915.
13. Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.
14. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1–9. [[CrossRef](#)]
15. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
16. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
17. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
18. Yang, J.; Reed, S.E.; Yang, M.H.; Lee, H. Weakly-supervised disentangling with recurrent transformations for 3d view synthesis. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 1099–1107.

19. Chen, X.; Duan, Y.; Houthoofd, R.; Schulman, J.; Sutskever, I.; Abbeel, P. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 2180–2188.
20. Li, C.; Welling, M.; Zhu, J.; Zhang, B. Graphical generative adversarial networks. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 1–12.
21. Li, C.; Zhu, J.; Shi, T.; Zhang, B. Max-margin deep generative models. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 1837–1845.
22. Locatello, F.; Bauer, S.; Lucic, M.; Raetsch, G.; Gelly, S.; Schölkopf, B.; Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled representations. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 4114–4124.
23. Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv* **2014**, arXiv:1411.1784.
24. Springenberg, J.T. Unsupervised and semi-supervised learning with categorical generative adversarial networks. *arXiv* **2015**, arXiv:1511.06390.
25. Odena, A. Semi-supervised learning with generative adversarial networks. *arXiv* **2016**, arXiv:1606.01583.
26. Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X. Improved techniques for training gans. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 2234–2242.
27. Dai, Z.; Yang, Z.; Yang, F.; Cohen, W.W.; Salakhutdinov, R.R. Good semi-supervised learning that requires a bad gan. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 6513–6523.
28. Zhang, X.; Wang, Z.; Liu, D.; Ling, Q. Dada: Deep adversarial data augmentation for extremely low data regime classification. In Proceedings of the ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 2807–2811.
29. Shrivastava, A.; Pfister, T.; Tuzel, O.; Susskind, J.; Wang, W.; Webb, R. Learning from simulated and unsupervised images through adversarial training. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
30. Douzas, G.; Bacao, F. Effective data generation for imbalanced learning using conditional generative adversarial networks. *Expert Syst. Appl.* **2018**, *91*, 464–471. [[CrossRef](#)]
31. Li, C.; Xu, T.; Zhu, J.; Zhang, B. Triple generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 4091–4101.
32. Weiss, G.M. Mining with rarity: A unifying framework. *Acm Sigkdd Explor. Newsl.* **2004**, *6*, 7–19. [[CrossRef](#)]
33. Weiss, G.M.; Tian, Y. Maximizing classifier utility when there are data acquisition and modeling costs. *Data Min. Knowl. Discov.* **2008**, *17*, 253–282. [[CrossRef](#)]
34. Sun, Y.; Wong, A.K.; Kamel, M.S. CLASSIFICATION OF IMBALANCED DATA: A REVIEW. *Int. J. Pattern Recognit. Artif. Intell.* **2009**, *23*, 687–719. [[CrossRef](#)]
35. Soda, P. A multi-objective optimisation approach for class imbalance learning. *Pattern Recognit.* **2011**, *44*, 1801–1810. [[CrossRef](#)]
36. Khoshgoftaar, T.M.; Hulse, J.V.; Napolitano, A. Comparing Boosting and Bagging Techniques With Noisy and Imbalanced Data. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2011**, *41*, 552–568. [[CrossRef](#)]
37. Galar, M. A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **2012**, *42*, 463–484. [[CrossRef](#)]
38. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic Minority Over-sampling Technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [[CrossRef](#)]
39. Fernandez, A.; Garcia, S.; Jesus, M.J.D.; Herrera, F. A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets. *Fuzzy Sets Syst.* **2008**, *159*, 2378–2398. [[CrossRef](#)]
40. Garcia, S.; Derrac, J.; Triguero, I.; Carmona, C.J.; Herrera, F. Evolutionary-based selection of generalized instances for imbalanced classification. *Knowl. Based Syst.* **2012**, *25*, 3–12. [[CrossRef](#)]
41. Tahir, M.A.; Kittler, J.; Yan, F. Inverse random under sampling for class imbalance problem and its application to multi-label classification. *Pattern Recognit.* **2012**, *45*, 3738–3750. [[CrossRef](#)]
42. Tang, Y.; Zhang, Y.Q.; Chawla, N.V.; Krasser, S. SVMs Modeling for Highly Imbalanced Classification. *IEEE Trans. Cybern.* **2009**, *39*, 281–288. [[CrossRef](#)]
43. Barandela, R.; Sánchez, J.S.; Garcia, V.; Rangel, E. Strategies for learning in class imbalance problems. *Pattern Recognit.* **2003**, *36*, 849–851. [[CrossRef](#)]
44. Cieslak, D.A.; Hoens, T.R.; Chawla, N.V.; Kegelmeyer, W.P. Hellinger distance decision trees are robust and skew-insensitive. *Data Min. Knowl. Discov.* **2012**, *24*, 136–158. [[CrossRef](#)]
45. García-Pedrajas, N.; Pérez-Rodríguez, J.; García-Pedrajas, M.; Ortiz-Boyer, D.; Fyfe, C. Class imbalance methods for translation initiation site recognition in DNA sequences. *Knowl. Based Syst.* **2010**, *25*, 22–34. [[CrossRef](#)]
46. Domingos, P. MetaCost: A General Method for Making Classifiers Cost-Sensitive. In Proceedings of the Acm Sigkdd International Conference on Knowledge Discovery & Data Mining, San Diego, CA, USA, 15–18 August 1999.
47. Sun, Y.; Kamel, M.S.; Wong, A.K.C.; Wang, Y. Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognit.* **2007**, *40*, 3358–3378. [[CrossRef](#)]

48. Ting, K.M. An instance-weighting method to induce cost-sensitive trees. *IEEE Trans. Knowl. Data Eng.* **2002**, *14*, 659–665. [\[CrossRef\]](#)
49. Zhao, H. Instance weighting versus threshold adjusting for cost-sensitive classification. *Knowl. Inf. Syst.* **2008**, *15*, 321–334. [\[CrossRef\]](#)
50. Zhou, Z.H.; Liu, X.Y. ON MULTI-CLASS COST-SENSITIVE LEARNING. *Comput. Intell.* **2010**, *26*, P.232–257. [\[CrossRef\]](#)
51. Kang, B.; Xie, S.; Rohrbach, M.; Yan, Z.; Gordo, A.; Feng, J.; Kalantidis, Y. Decoupling Representation and Classifier for Long-Tailed Recognition. *arXiv* **2019**, arXiv:1910.09217.
52. Zhang, J.; Liu, L.; Wang, P.; Shen, C. To Balance or Not to Balance: A Simple-yet-Effective Approach for Learning with Long-Tailed Distributions. *arXiv* **2019**, arXiv:1912.04486.
53. Yang, Y.; Xu, Z. Rethinking the Value of Labels for Improving Class-Imbalanced Learning. *arXiv* **2020**, arXiv:2006.07529.
54. Gidaris, S.; Singh, P.; Komodakis, N. Unsupervised Representation Learning by Predicting Image Rotations. *arXiv* **2018**, arXiv:1803.07728.
55. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum Contrast for Unsupervised Visual Representation Learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9729–9738.
56. Kang, B.; Li, Y.; Yuan, Z.; Feng, J. FEATURE SPACES FOR REPRESENTATION LEARNING. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 4 May 2021.
57. Jiang, Z.; Chen, T.; Mortazavi, B.; Wang, Z. Self-Damaging Contrastive Learning. In Proceedings of the International Conference on Machine Learning, Shenzhen, China, 26 February–1 March 2021.
58. MacKay, D.J. Information-based objective functions for active data selection. *Neural Comput.* **1992**, *4*, 590–604. [\[CrossRef\]](#)
59. McCallumzy, A.K.; Nigamy, K. Employing em and pool-based active learning for text classification. In Proceedings of the International Conference on Machine Learning, Madison, WI, USA, 24–27 July 1998; pp. 359–367.
60. Yoav Freund, H.; Sebastian Seung, E.S.; Tishby, N. Selective sampling using the query by committee algorithm. *Mach. Learn.* **1997**, *28*, 133–168. [\[CrossRef\]](#)
61. Tong, S.; Koller, D. Support vector machine active learning with applications to text classification. *J. Mach. Learn. Res.* **2001**, *2*, 45–66.
62. Li, X.; Guo, Y. Adaptive active learning for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 859–866.
63. Jiang, Z.; Chen, T.; Chen, T.; Wang, Z. Improving contrastive learning on imbalanced data via open-world sampling. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 5997–6009.
64. Versaci, M.; Laganà, F.; Morabito, F.C.; Palumbo, A.; Angiulli, G. Adaptation of an Eddy Current Model for Characterizing Subsurface Defects in CFRP Plates Using FEM Analysis Based on Energy Functional. *Mathematics* **2024**, *12*, 2854. [\[CrossRef\]](#)
65. Versaci, M.; Angiulli, G.; Crucitti, P.; De Carlo, D.; Laganà, F.; Pellicanò, D.; Palumbo, A. A fuzzy similarity-based approach to classify numerically simulated and experimentally detected carbon fiber-reinforced polymer plate defects. *Sensors* **2022**, *22*, 4232. [\[CrossRef\]](#)
66. Versaci, M.; Angiulli, G.; La Foresta, F.; Laganà, F.; Palumbo, A. Intuitionistic fuzzy divergence for evaluating the mechanical stress state of steel plates subject to bi-axial loads. *Integr. Comput. Aided Eng.* **2024**, *31*, 363–379. [\[CrossRef\]](#)
67. Cao, K.; Wei, C.; Gaidon, A.; Arechiga, N.; Ma, T. Learning imbalanced datasets with label-distribution-aware margin loss. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 1567–1578.
68. Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; Ng, A.Y. Reading digits in natural images with unsupervised feature learning. In Proceedings of the NIPS Workshop on Deep Learning and Unsupervised Feature Learning, Granada, Spain, 16 December 2011; Volume 2011, p. 4.
69. Brodersen, K.H.; Ong, C.S.; Stephan, K.E.; Buhmann, J.M. The balanced accuracy and its posterior distribution. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 3121–3124.
70. Kingma, D.P.; Mohamed, S.; Jimenez Rezende, D.; Welling, M. Semi-supervised learning with deep generative models. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 1–9.
71. Zhu, X.; Anguelov, D.; Ramanan, D. Capturing long-tail distributions of object subcategories. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 915–922.
72. Feldman, V. Does learning require memorization? A short tale about a long tail. In Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, Chicago, IL, USA, 22–26 June 2020; pp. 954–959.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.