Online Harmonizing Gradient Descent for Imbalanced Data Streams One-Pass Classification

Han Zhou , **Hongpeng Yin*** , **Xuanhong Deng** and **Yuyu Huang**
The School of Automation, Chongqing University, Chongqing, China, 400044.
zhouhan1515@foxmail.com, yinhongpeng@cqu.edu.cn. dengxh2022@163.com, hyy5476@163.com.

Abstract

Many real-world streaming data are sequentially collected over time and with skew-distributed classes. In this situation, online learning models may tend to favor samples from majority classes, making the wrong decisions for those from minority classes. Previous methods try to balance the instance number of different classes or assign asymmetric cost values. They usually require data-buffers to store streaming data or pre-defined cost parameters. This study alternatively shows that the imbalance of instances can be implied by the imbalance of gradients. Then, we propose the Online Harmonizing Gradient Descent (OHGD) for one-pass online classification. By harmonizing the gradient magnitude occurred by different classes, the method avoids the bias of the proposed method in favor of the majority class. Specifically, OHGD requires no data-buffer, extra parameters, or prior knowledge. It also handles imbalanced data streams the same way that it would handle balanced data streams, which facilitates its easy implementation. On top of a few common and mild assumptions, the theoretical analysis proves that OHGD enjoys a satisfying sub-linear regret bound. Extensive experimental results demonstrate the high efficiency and effectiveness in handling imbalanced data streams.

1 Introduction

Many real-world applications require learning models to quickly react to and learn from rapidly growing data, such as ad placement [Varnali, 2021], social media analysis [Zhou *et al.*, 2020], intrusion detection [Sovilj *et al.*, 2020] and industrial fault diagnosis [Zhou *et al.*, 2022]. Mining data streams thus attracts the attention of intensive research, and a family of techniques called online learning has emerged. Online learning is advantageous for its efficiency and effectiveness, which can make a decision after observing an instance and then further update based on the revealed true label. It provides an opportunity to solve many practical applications where data arrives sequentially while decisions need to be made immediately. Many online learning algorithms have been actively proposed in the literature, including Online Gradient Descent (OGD) [Zinkevich, 2003], Passive Aggressive (PA) [Crammer *et al.*, 2006] Confidence Weighted Classifier [Dredze *et al.*, 2008] etc.

Despite extensive online learning algorithms, most methods are inappropriate to deal with skew-distributed classes because they favor samples from majority classes and, accordingly make wrong decisions for those from minority classes. This is unacceptable when misclassifying the minority class samples may lead to severe problems, such as fault diagnosis, disease diagnosis, etc. In this situation, combining both issues of online learning and class imbalance learning, new challenges and research topics arise.

Various approaches have been proposed to deal with the class imbalance problem. The two most prominent directions include resampling and reweighting. Resampling ensures an equal number of instances within a data chunk by reducing the number of instances from majority classes (under-sampling), generating new instances of minority classes (over-sampling) or both [Wang and Pineau, 2016], [Lu *et al.*, 2019], [Bernardo *et al.*, 2020], [Klikowski and Woźniak, 2022]. Reweighting amends the importance of classes by assigning either uneven costs [Wang *et al.*, 2013a], [Zhao *et al.*, 2018], [Zhang *et al.*, 2019], [Loezer *et al.*, 2020] or imbalance ratio [Yu *et al.*, 2018]. Combination of resampling and reweighting approaches is also among the most popular methods to solve this problem [Wang and Pineau, 2016], [Zyblewski *et al.*, 2021]. While being intuitive and working reasonably well, an important thing is that these methods inevitably involve heuristic data buffer design, sampling rate selection or empirical cost determination, limiting their easy implementation to practical applications.

Different from the underlying assumptions of previous works on instance number inequality or cost asymmetry, this paper assumes that the class imbalance can be implied by the imbalance of the gradient. We analyze the validity of this assumption experimentally in the overall online learning stage. Further, the online harmonizing gradient descent (OHGD) is proposed, which attempts to balance the magnitude of gradients that occur by different classes. With ensuring the gradient balance, it avoids the bias of the proposed method in favor of majority classes and achieves balanced online learning. Specifically, OHGD requires no data-buffer, extra parameters, nor prior knowledge. It also handles imbalanced data streams the same way that it would handle balanced data streams, which facilitates its easy implementation. We analyze the theoretical regret bound made by the OHGD and extensively examine its empirical performance on several benchmarks, compared with state-of-the-art online imbalance learning algorithms. We elaborate on our contributions below.

- *Motivation.* We assume that the class imbalance can be implied by the imbalance of gradients and analyze the validity of this assumption experimentally.
- *Method.* We propose an easy-to-implement online harmonizing gradient descent (OHGD) method for imbalanced data stream one-pass classification.
- *Theoretical guarantee.* We prove that OHGD enjoys a satisfying sub-linear regret bound on top of a few common and mild assumptions.
- *Experimental results.* Extensive experimental results demonstrate the high efficiency and effectiveness of OHGD[1].

The rest of this paper is organized as follows. Section 2 briefs related works on class imbalance and online learning. Section 3 formulates the problem, describes the proposed methods and analyzes the theoretical bounds. Extensive experimental results and discussion are given in Section 4. Finally, Section 5 concludes this paper.

2 Related Work

This section provides a brief overview of related works on online learning and class imbalance learning.

2.1 Online Learning

In the setting of online learning, data are sequentially received and models need to commit to an immediate decision at each round. Then the decision model will suffer a loss and the loss is used to update model parameters based on some criterion. Many online learning problems can be formulated as an Online Convex Optimization task [Hoi *et al.*, 2021]. In the following, we introduce its basics and one of the most popular algorithms, Online Gradient Descent.

Let $\mathbf{x}_t \in \Re^d$ denotes the t-th sample whose true label is denoted as $y_t \in \{-1, +1\}$ in an infinite data stream $\mathcal{D}$. Considering a classifier in a convex set $\mathbf{w}_t \in \mathcal{W}$ at the t-round, it makes a prediction on sample $\mathbf{x}_t$ as $\hat{y}_t = \mathbf{w}_t \mathbf{x}_t$. The performance is usually measured by a convex loss function $\mathcal{L}_t(\mathbf{w}_t | \mathbf{x}_t, y_t)$, such as hinge loss or logistic loss, which are surrogate losses for 0-1 error. The overall goal is to minimize the regret between the cumulative mistake of $\mathbf{w}_t$ and that of one best fixed decision $\mathbf{w}^*$:

$$R(T) = \sum_{t=1} \mathcal{L}_t(\mathbf{w}_t) - \sum_{t=1} \mathcal{L}_t(\mathbf{w}^*). \tag{1}$$

One of the most well-known online learning algorithms would be the Online Gradient Descent, an online version of Stochastic Gradient Descent in convex optimization. OGD takes a step from the current model toward the steepest descent direction of the current loss function based on the loss on an instance $\mathbf{x}_t$. This gives $\mathbf{w}_{t+1} = \mathbf{w}_t - \lambda \nabla \mathcal{L}_t(\mathbf{w}_t)$, where $\mathbf{w}_t \in \mathcal{W}$ is the model in the t-th round and λ is the learning rate. In [Zinkevich, 2003], OGD achieves sublinear regret $\mathcal{O}(\sqrt{T})$ for any data stream $\mathcal{D}$ on convex loss functions with bounded gradients.

2.2 Learning with Imbalanced Data

Many approaches in the literature have been proposed to deal with the class imbalance problem in the setting of batch learning. The approaches can be roughly categorized into three categories: data-level [Chawla *et al.*, 2002], [Razavi-Far *et al.*, 2021], algorithm-level [Sun *et al.*, 2007], [Zong *et al.*, 2013], [Zhang *et al.*, 2018], [Wang *et al.*, 2021] as well as hybrid methods [Galar *et al.*, 2011]. However, learning imbalanced data streams is more challenging. Unlike batch learning, which assumes all training data is available in memory, online learning assumes data is observed sequentially and rapidly updated. Many efforts design data buffers to store continuous data. Once a full buffer is formed, they treat each of them as an imbalance dataset and many conventional data preprocessing methods can be adopted to modify the distribution, such as online SMOTE [Wang and Pineau, 2016], [Lu *et al.*, 2019], [Bernardo *et al.*, 2020], [Klikowski and Woźniak, 2022]. An even stricter scenario is that learning paradigms need to respond immediately to an instance and discard it, i.e. one-pass classification. In this situation, re-weighting or cost-sensitive strategy [Wang *et al.*, 2013a], [Zhao *et al.*, 2018], [Zhang *et al.*, 2019], [Loezer *et al.*, 2020] could be much more feasible since it can deal with instances one by one, requiring no data buffer. Some attempts at ensemble learning also achieve convincing performance, which re-samples instances sequentially [Wang and Pineau, 2016], [Wang *et al.*, 2013b], [Wang *et al.*, 2014]. Unfortunately, these works inevitably involve heuristic weighting parameter design, prior cost determination or sampling rate selection, which limits their generalization on many practical applications. With improper weighting parameters, such a strategy may also cause a reverse bias towards the minority class.

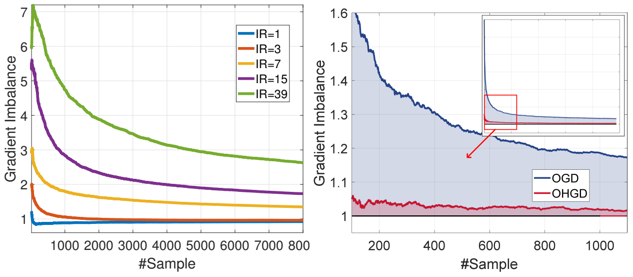
3 Methodology

This section presents the core idea of gradient reweighting for imbalanced online learning. We first start with the problem formulation of online learning on imbalanced data streams, and then propose a gradient reweighting strategy to solve the gradient asymmetry between the majority class and the minority one. We also provide the theoretical analysis on the soundness of the proposed strategy.

3.1 Problem Formulation

The Eq.1 indicates that each sample in the data stream $\mathcal{D}$ contributes equally to the regret of the model. This symmetrical treatment may result in bias toward samples in majority classes (labeled as -1) which incur larger accumulative loss values than those of samples from minority classes (labeled as +1). Many works attempt to resample datasets or assign larger weights to minority samples and thus maintain a more

[1] More information about the codes and supplementary can be found in https://github.com/Kan9594/OHGD.



(a) (b)

Figure 1: The gradient imbalance (a) when applying OGD on the *ijcnn1* with varied imbalance ratio. (b) when applying OGD and OHGD on the *ijcnn1*.

balanced distribution. Instead, this paper claims that the imbalance of examples can be implied by the imbalance of gradient and tries to sustain the balance by keeping the harmonizing gradient magnitude accumulated by diverse classes.

In many online learning methods, it is common to update model parameters by iteratively taking a gradient step. In its fundamental form, parameters at t-th iteration are forced to move towards the steepest descent direction of the loss at $\mathbf{w}_t$:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \lambda_t \nabla \mathcal{L}_t(\mathbf{w}_t|\mathbf{x}_t, y_t), \tag{2}$$

where λ_t is the learning rate. We then define the gradient imbalance in the following.

Definition 1. *Denoting the gradient norm of the t-th sample as $G_t = \|\nabla \mathcal{L}_t(\mathbf{w}_t|\mathbf{x}_t, y_t)\|_2$, the accumulative gradient norm of negative samples and positive samples before t-th round is $G^n = \sum_{i\in\mathbb{I}_{(y_i=-1)}}^{t} \|\nabla \mathcal{L}_i(\mathbf{w}_i|\mathbf{x}_i, y_i)\|_2$ and $G^p = \sum_{i\in\mathbb{I}_{(y_i=+1)}}^{t} \|\nabla \mathcal{L}_i(\mathbf{w}_i|\mathbf{x}_i, y_i)\|_2$, respectively. In this situation, the gradient imbalance (GI) can be measured by:*

$$GI = \frac{G_n}{G_p} = \frac{\sum_{i\in\mathbb{I}_{(y_i=-1)}}^{t} \|\nabla \mathcal{L}_i(\mathbf{w}_i|\mathbf{x}_i, y_i)\|_2}{\sum_{i\in\mathbb{I}_{(y_i=+1)}}^{t} \|\nabla \mathcal{L}_i(\mathbf{w}_i|\mathbf{x}_i, y_i)\|_2}. \tag{3}$$

Clearly, a larger GI value denotes a more heavily imbalanced data stream. $\mathbb{I}_{(y_i=-1)}$ denotes the dataset where the label of the i-th sample is negative. In the following, we use $\mathbb{I}_{(-)}$ for simplicity.

Based on the above definition, Fig.1 shows the gradient imbalance when applying OGD to the *ijcnn1* dataset. We varied the imbalance ratio (IR) from 1 to 39. As we can see, the positive gradient norms are overwhelmed by the negative gradient norms at the very beginning of the learning procedure. This implies that the OGD performs a greater number of gradient descent actions guided by negative samples, resulting in overfitting to the majority class. Moreover, a larger imbalance ratio would result in a higher gradient imbalance. In contrast, when the dataset is in a balanced distribution (the blue line), the gradient imbalance fluctuates around 1.

In this situation, this paper attempts to harmonize the gradient imbalance during the learning procedure, corresponding to a weighted gradient descent step:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \alpha_t \lambda_t \nabla \mathcal{L}_t(\mathbf{w}_t|\mathbf{x}_t, y_t). \tag{4}$$

Such that

$$GI = \frac{\sum_{i\in\mathbb{I}_{(-)}}^{t} \alpha_i \lambda_i \|\nabla \mathcal{L}_i(\mathbf{w}_i|\mathbf{x}_i, y_i)\|_2}{\sum_{i\in\mathbb{I}_{(+)}}^{t} \alpha_i \lambda_i \|\nabla \mathcal{L}_i(\mathbf{w}_i|\mathbf{x}_i, y_i)\|_2} \rightarrow 1. \tag{5}$$

Compared with the conventional OGD in Fig.1b, this strategy (Online Harmonizing Gradient Descent, OHGD) encourages the gradient ratio to fluctuate around 1 (red line). This means that each class contributes equally to the learning model.

3.2 Online Harmonized Gradient Descent

We first define the *response* of a learning model to t-th instance as:

$$R_t(\mathbf{x}_t, y_t) = \lambda_t \alpha_t \|\nabla \mathcal{L}_t(\mathbf{w}_t|\mathbf{x}_t, y_t)\|_2 . \tag{6}$$

It reflects the expected gradient descent magnitude when a model learns the instance $\mathbf{x}_t$. Thus, we reshape the gradient to down-weight/up-weight easy/difficult instances, putting more focus on the hard, misclassified instances.

Then the weight α_t for the gradient at this iteration can be formulated by:

$$\alpha_t = 2\frac{\rho_t[\sum_{i\in\mathbb{I}_{(-)}} R_i(\mathbf{x}_i, y_i)]\mathbb{I}_{(+)} + [\sum_{i\in\mathbb{I}_{(+)}} R_i(\mathbf{x}_i, y_i)]\mathbb{I}_{(-)}}{\sum_{i\in\mathbb{I}_{(-)}} R_i(\mathbf{x}_i, y_i) + \sum_{i\in\mathbb{I}_{(+)}} R_i(\mathbf{x}_i, y_i)} \mathcal{L}_t(\mathbf{w}_t|\mathbf{x}_t, y_t), \tag{7}$$

where the parameter $\rho_t = \frac{N_n^t}{N_p^t}$ is the imbalance ratio of the negative to positive instance number at the t-th iteration. The reason behind adding the loss value factor is the different contribution of each sample. Applying the weights to Eq.4, the harmonized gradients become:

$$\mathbf{w}_{t+1} = \begin{cases} \mathbf{w}_t - \lambda_t \rho_t \frac{\sum_{i\in\mathbb{I}_{(-)}} R_i(\mathbf{x}_i, y_i)}{\sum_{i=1}^{t} R_i(\mathbf{x}_i, y_i)} \mathcal{L}_t(\mathbf{w}_t|\mathbf{x}_t, y_t) \nabla \mathcal{L}_t(\mathbf{w}_t|\mathbf{x}_t, y_t), & \text{if } y_t = +1; \\ \mathbf{w}_t - \lambda_t \frac{\sum_{i\in\mathbb{I}_{(+)}} R_i(\mathbf{x}_i, y_i)}{\sum_{i=1}^{t} R_i(\mathbf{x}_i, y_i)} \mathcal{L}_t(\mathbf{w}_t|\mathbf{x}_t, y_t) \nabla \mathcal{L}_t(\mathbf{w}_t|\mathbf{x}_t, y_t), & \text{if } y_t = -1. \end{cases} \tag{8}$$

Finally, we summarize the pseudo-code of the proposed OHGD in Algorithm.1. This paper adopts the widely used hinge loss. It is easy to observe that the time complexity of OHGD is $\mathcal{O}(Td)$, which scales linearly with the number T of instances in the data stream $\mathcal{D}$ as well as the dimensionality d of instances.

Remarks. In some real-world online cases, the imbalance ratio can shift over time. To dynamical capture the imbalance ratio, we can use exponential smoothing $\rho_t = \eta\rho_{t-1} + (1 - \lambda)\mathbb{I}_{(-)}$, where $\eta \in (0, 1)$ is a pre-defined time decay factor. By doing so, the older data affect the class ratio less with time, so that the model can follow the imbalance change quickly.

3.3 Theoretical Justification

In this section, we provide the theoretical analysis that gives the sublinear regret bound achieved by OHGD. To ease our discussion, we denote $\mathcal{M}$ as the prediction error indexes set: $\mathcal{M} = \{t|\mathcal{L}(\mathbf{w}_t) > 0\}$. Similarly, we denote M_p^t and M_n^t as

Algorithm 1 The Proposed Online Harmonized Gradient Descent Algorithms.

```
Input: learning rate $\lambda_t$;
Initialize $\alpha_0 = 1$
for $t = 1, \cdots, T$ do
    Receive an instance: $\mathbf{x}_t \in \Re^d$;
    Predict $\widehat{y}_t$;
    Receive the true label $y_t$;
    Calculate the weight $\alpha_t$ using Eq.7;
    Incur the loss $\mathcal{L}_t(\mathbf{w}_t|\mathbf{x}_t, y_t)$
    if $\mathcal{L}_t > 0$ then
        Update the model using Eq.8.
    end if
end for
Output: $\mathbf{w}_{T+1}$;
```

the number of misclassified positive and negative instances before t-th round, respectively. $M^T = |M| = M_p^T + M_n^T$ is the total number of misclassified instances in data stream $\mathcal{D}$. We begin the analysis with the following assumptions.

Assumption 1. *Let $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_T, y_T)$ be the data stream $\mathcal{D}$, where $y_i \in \{-1, +1\}$ and $\|\mathbf{x}_i\| \le 1$ for all t. The imbalance ratio is stationary $\rho = \frac{N_n^T}{N_p^T} \ge 1$.*

Assumption 2. *The decision domain $\mathcal{W}$ contains the origin $\mathbf{0}$, and its diameter is bounded by D, i.e.,*

$$\max_{\mathbf{w_1}, \mathbf{w_2} \in \mathcal{W}} \|\mathbf{w_1} - \mathbf{w_2}\|_2 \le D. \tag{9}$$

Corollary 1. *Under these assumptions, for the hinge loss $\mathcal{L}(\mathbf{w}_i) = \max(0, 1 - y_i\mathbf{w}_i\mathbf{x}_i)$, we have $0 \le \mathcal{L}(\mathbf{w}_i) \le (D+1)$.*

We also provide the following lemma that would facilitate the theoretical analysis.

Lemma 1. *The sum of the re-weighting parameter α_t has an upper bound:*

$$\sum_{t=1}^{T} \alpha_t \le 2(D+1)(\rho M_p^T + M_n^T). \tag{10}$$

The proof details of Lemma 1 can be found in the supplementary.

Theorem 1. *By adaptively setting $\lambda_t = \sqrt{\frac{D+1}{\rho M_n^t + M_p^t}}$, for any $\mathbf{w}^* \in \Re^d$, the following regret bound holds for the proposed OHGD on the data stream $\mathcal{D}$:*

$$R(T) \le \frac{(D+1)^{\frac{3}{2}}\sqrt{\rho}}{2}\left(\frac{2\epsilon+1}{\epsilon}\sqrt{T} - 1\right), \tag{11}$$

where the ϵ denotes the minimum of α_t.

Proof. Due to the convexity of the loss function, the following inequality holds for any $\mathbf{w}$:

$$R(T) = \mathcal{L}_t(\mathbf{w}_t) - \mathcal{L}_t(\mathbf{w}^*) \le \nabla\mathcal{L}_t(\mathbf{w}_t)(\mathbf{w}_t - \mathbf{w}^*). \tag{12}$$

Since we defined the harmonized gradient descent updating $\mathbf{w}_{t+1} = \mathbf{w}_t - \alpha_t\lambda_t\nabla\mathcal{L}_t(\mathbf{w}_t)$ in Eq.5, then we have:

$$\begin{aligned}\|\mathbf{w}_{t+1} - \mathbf{w}^*\|^2 = \|\mathbf{w}_t - \mathbf{w}^*\|^2 + \alpha_t^2\lambda_t^2 \|\nabla\mathcal{L}_t(\mathbf{w}_t)\|^2 \\ - 2\alpha_t\lambda_t\nabla\mathcal{L}_t(\mathbf{w}_t)(\mathbf{w}_t - \mathbf{w}^*).\end{aligned} \tag{13}$$

Accordingly,

$$\begin{aligned}&\nabla\mathcal{L}_t^\top(\mathbf{w}_t)(\mathbf{w}_t - \mathbf{w}^*) = \\ &\frac{\|\mathbf{w}_t - \mathbf{w}^*\|^2 - \|\mathbf{w}_{t+1} - \mathbf{w}^*\|^2}{2\alpha_t\lambda_t} + \frac{\alpha_t\lambda_t}{2}\|\nabla\mathcal{L}_t\|^2.\end{aligned} \tag{14}$$

By summing over T and using Eq.12, we can get:

$$\begin{aligned}&R(T) \le \sum_{t=1}^{T} \nabla\mathcal{L}_t^\top(\mathbf{w}_t)(\mathbf{w}_t - \mathbf{w}^*) \le \\ &\sum_{t=1}^{T}\left\{\frac{\|\mathbf{w}_t - \mathbf{w}^*\|^2 - \|\mathbf{w}_{t+1} - \mathbf{w}^*\|^2}{2\alpha_t\lambda_t} + \frac{\alpha_t\lambda_t}{2}\|\nabla\mathcal{L}_t\|^2\right\}.\end{aligned} \tag{15}$$

Let ϵ denotes the minimum of α_t. Since $\|\mathbf{w}_t\|^2 < D$ and $\|\nabla\mathcal{L}_t\| \le 1$, we have:

$$\begin{aligned}R(T) &\le \frac{\|\mathbf{w}_1 - \mathbf{w}^*\|^2}{2\epsilon\lambda_T} + \sum_{t=1}^{T}\frac{\alpha_t\lambda_t}{2}\|\nabla\mathcal{L}_t\|^2 \\ &\le \frac{D^2}{2\epsilon\lambda_T} + \sum_{t=1}^{T}\frac{\alpha_t\lambda_t}{2}.\end{aligned} \tag{16}$$

Individually analyzing the two terms:

$$\frac{D^2}{2\epsilon\lambda_T} \le \frac{(D+1)^{\frac{3}{2}}\sqrt{\rho M_n^T + M_p^T}}{2\epsilon} \le \frac{(D+1)^{\frac{3}{2}}\sqrt{\rho}\sqrt{T}}{2\epsilon}. \tag{17}$$

For the second term:

$$\sum_{t=1}^{T}\frac{\alpha_t\lambda_t}{2} \le \frac{(D+1)^{\frac{3}{2}}\sqrt{\rho}}{2}(2\sqrt{T} - 1). \tag{18}$$

Combining Eq.17 and Eq.18,we get the regret bound:

$$R(T) \le \frac{(D+1)^{\frac{3}{2}}\sqrt{\rho}}{2}\left(\frac{2\epsilon+1}{\epsilon}\sqrt{T} - 1\right). \tag{19}$$

In this setting, OHGD achieves an $O(\sqrt{T})$ regret bound. This completes the proof.

□

Remarks. The proposed OHGD enjoys an $O(\sqrt{T})$ performance by setting an adaptive learning rate. It can be influenced by the weighting parameter α. To avoid $\alpha_t = 0$, one can initialize the response $R_0 = 1$ and set $\rho_0 = N_n/N_p = 1/1$ at the beginning. In terms of the implementation, we can force α_t to be in $[10^{-2}, 10^2]$ in order to achieve the satisfied sub-linear regret bound.

Regarding easy implementation, we provide another choice for selecting the learning rate, and the algorithm performs well.

Theorem 2. *By dynamically setting $\lambda_t = 1/\sqrt{t}$, for any $\mathbf{w}^* \in \Re^d$, the following regret bound holds for the proposed OHGD on the data stream $\mathcal{D}$:*

$$R(T) \le \left(\frac{D^2}{2\epsilon} + \rho D + \rho\right)\sqrt{T} - \frac{\rho D + \rho}{2}, \tag{20}$$

where the ϵ denotes the minimum of α_t.

Dataset	#Instances	#Fea	IR	Dataset	#Instances	#Fea	IR
mushroom	8124	21	1.07	**segment0**	2308	19	6.02
splice	3175	60	1.08	**yeast3**	1484	8	8.1
svmguide1	7089	1	1.29	**pageblocks0**	5472	10	8.79
spambase	4601	57	1.53	**ijcnn1**	49990	22	9.3
magic	19020	10	1.84	**vowel0**	988	13	9.98
rna	59535	8	2	**led7digit**	443	7	10.97
yeast1	1484	8	2.45	**shuttle04**	1829	9	13.87
a8a	32561	123	3.15	**ecoli4**	336	7	15.8
a9a	48842	123	3.17	**yeast4**	1484	8	28.1
svmguide3	1243	21	3.2	**w8a**	64700	300	32.47
vehicle0	846	18	3.25	**yeast5**	1484	8	32.73
musk2	6598	166	5.48	**yeast6**	1484	8	41.4

Table 1: Specifications of the Datasets Used in the Experiments.

Proof. Recall Eq.16:

$$R(T) \leq \frac{\|\mathbf{w}_1 - \mathbf{w}^*\|^2 - \|\mathbf{w}_{t+1} - \mathbf{w}^*\|^2}{2\epsilon\lambda_T} + \sum_{t=1}^{T} \frac{\alpha_t \lambda_t}{2} \|\nabla \mathcal{L}_t\|^2$$
$$\leq (\frac{D^2}{2\epsilon} + \rho D + \rho)\sqrt{T} - \frac{\rho D + \rho}{2}. \tag{21}$$

In this setting, OHGD achieves an $O(\sqrt{T})$ regret bound. This completes the proof.

□

Remarks. Similar to the Theorem.1, the proposed OHGD enjoys a satisfied$O(\sqrt{T})$ regret when we set $\lambda_t = 1/\sqrt{t}$. Setting the learning rate λ_t as a constant cannot promise a sub-linear regret but it can also achieve satisfied classification performance as some previous works discussed [Wang *et al.*, 2013a]. More detailed derivation for Theorems 1-2 can be found in our *supplementary*.

Note that although the imbalance ratio ρ is assumed to be constant in these theorems, it is easy to infer that the regret is also sub-linear even if ρ_t changes, as long as ρ_t has an upper bound.

4 Experimental Results

In this section, we evaluate the performance of the proposed OHGD for imbalanced data streams, using public benchmark datasets. The implementations of this work can be found in https://github.com/Kan9594/OHGD.

4.1 Settings

Datasets. Twenty-four datasets from the UCI repository and KEEL with different imbalance ratios were selected as the test rigs for performance evaluation. Table.1 summarizes the details of the selected datasets, including the number of instances (#Instance), the dimensionality of features (#Fea) as well as the imbalance ratio ($IR = \#Majority/\#Minority$). More details can be found in the UCI [2] and KEEL [3] websites.

The Tested Methods. We compared our OHGD algorithms with various online learning algorithms for imbalanced data streams, including cost-sensitive online learning [Wang *et al.*, 2013a] (CSOGD), cost-sensitive online ensemble learning [Wang and Pineau, 2016] [Du *et al.*, 2021] (onlineAdaC2, onlineCSB2, onlineUnderOverBagging, onlineRUSBoost, onlineEffectiveBagging) and resampling based online ensemble learning [Wang *et al.*, 2013b], [Wang *et al.*, 2014] (onlineUnderBagging, onlineOverBagging, onlineWeightedUnderBagging, onlineWeightedOverBagging). As the purpose of the experiments is to make fair comparisons among different online algorithms, we chose OGD as the base learners of ensemble learning. The number of base learners M was set as 10. The learning rate η_t was set as $1/\sqrt{t}$. Other parameters were set as the original work suggested. Note that our method does not require any additional parameter settings for imbalance learning. The *supplementary* summarizes the characteristics of different methods.

Although our method focuses on one-pass data stream learning, we still compared it with some chunk based methods, including HDWE [Grzyb *et al.*, 2021] and KNORAE2 [Zyblewski *et al.*, 2021]. The number of base learners M was set as 10 and other parameters were set as the original works suggested.

Metrics. As the data distributions are highly skewed, we adopted AUC value, G-MEANS and F1 Score to evaluate the performance of each algorithm, instead of accuracy. Each algorithm on each dataset was conducted 10 times, and we reported its mean and standard deviation.

4.2 Parameter Sensitivity

We first show that the proposed algorithm achieves consistently satisfying classification performance without any additional parameter tuning. Fig.2 illustrates the performance variation of algorithms under different parameter settings. Due to the length limitation, we only provide the performance on the dataset ijcnn1 in terms of AUC. More results can be found in the *supplementary*. Since no hyper-parameters are required in OHGD, its performance curves are horizontal lines. From these figures, we can observe that the performance of competitors depends heavily on the selection of hyper-parameters. Although they prevail over the proposed algorithm in some cases, it is quite tricky to determine precise parameters for different practical situations. On the contrary, OHGD is elegantly formulated without many hyper-parameters to tune, allowing for easy implementation in practical applications.

4.3 Performance Comparison

Applying all algorithms for classifying all datasets in terms of AUC, G-mean and F1 score, we provide the performance comparison of different methods. Here, we present the results of the *a8a* dataset, and additional results can be found in the *Supplementary*. We calculate and summarize the average ranks over all three performance metrics in Fig.3. As we can observe, the proposed OHGD is ranked as the best method on average over all datasets since it outperforms other competitors in terms of each performance measure

[2] http://archive.ics.uci.edu/ml/index.php

[3] https://sci2s.ugr.es/keel/datasets.php

a8a	OHGD	CSOGD				AdaC2	CSB2	UOB	RUSB			OOB	OUB	WOOB	WOUB	OEB
		Cost$_I$	Cost$_{II}$	Sum$_I$	Sum$_{II}$				1	2	3					
AUC	0.816 ±0.003	0.791 ±0.002	0.737 ±0.002	0.782 ±0.002	0.805 ±0.002	0.802 ±0.002	0.814 ±0.002	0.812 ±0.003	0.814 ±0.005	0.805 ±0.001	0.815 ±0.002	0.814 ±0.002	0.798 ±0.003	0.808 ±0.002	0.81 ±0.003	0.807 ±0.004
GMEANS	0.814 ±0.002	0.801 ±0.001	0.700 ±0.003	0.786 ±0.002	0.804 ±0.002	0.789 ±0.001	0.813 ±0.002	0.808 ±0.002	0.777 ±0.025	0.786 ±0.003	0.811 ±0.004	0.809 ±0.002	0.782 ±0.003	0.810 ±0.001	0.800 ±0.003	0.804 ±0.002
F1	0.668 ±0.002	0.654 ±0.001	0.547 ±0.003	0.669 ±0.002	0.656 ±0.003	0.625 ±0.002	0.667 ±0.003	0.656 ±0.003	0.615 ±0.028	0.622 ±0.003	0.667 ±0.007	0.657 ±0.002	0.619 ±0.004	0.675 ±0.001	0.641 ±0.004	0.648 ±0.004

Table 2: The results of different methods on dataset a8a.

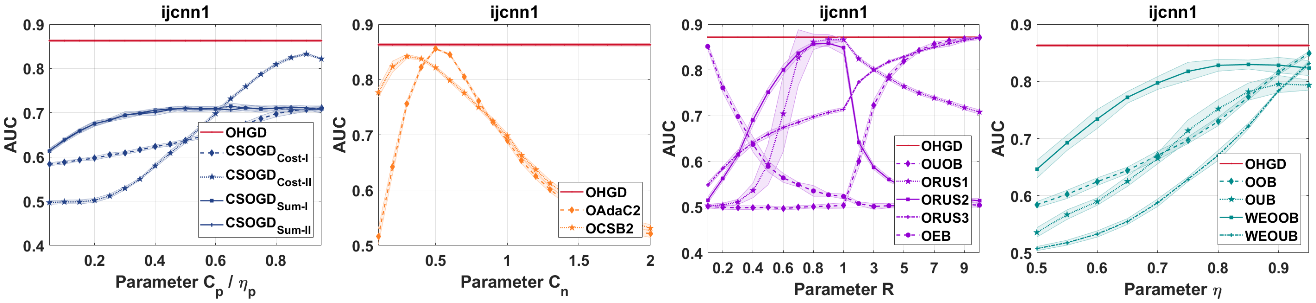
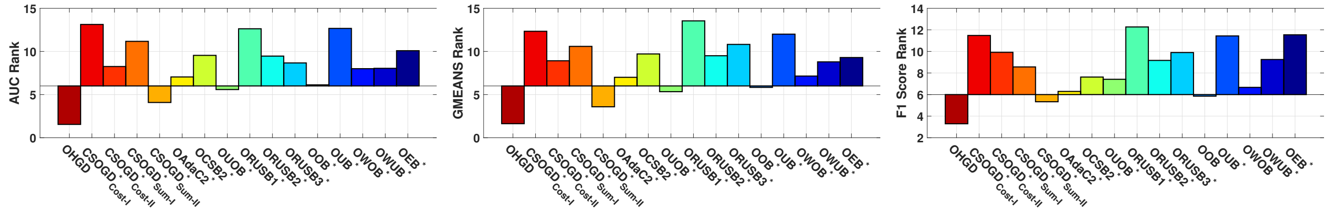
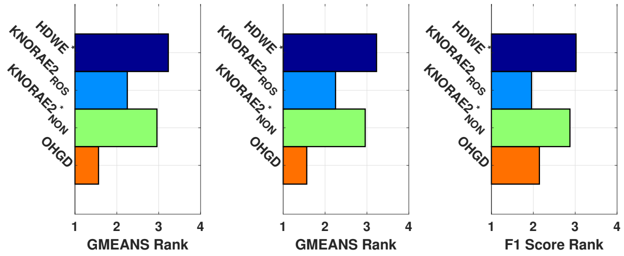


Figure 2: Performance evaluation with varying parameters on dataset ijcnn1, in terms of AUC.



(a) AUC (b) Gmeans (c) F1 Score

Figure 3: Overall ranks compared with one-pass learning methods, in terms of (a) AUC, (b) Gmeans, (c) F1 Score. Rank 1 indicates the method with the highest performance. The star denotes that the proposed method is significantly different with the corresponding methods by the Wilcoxon signed-rank test at $\alpha = 0.05$.



(a) AUC (b) Gmeans (c) F1 Score

Figure 4: Overall ranks compared with chunk-based methods, in terms of (a) AUC, (b) Gmeans, (c) F1 Score. Rank 1 indicates the method with the highest performance. The star denotes that the proposed method is significantly different with the corresponding methods by the Wilcoxon signed-rank test at $\alpha = 0.05$.

In order to provide a certain reassurance and validate the results and analysis, we employ a non-parametric statistical test, the Wilcoxon signed-rank test, to examine whether the proposed method is significantly better than competitors. The results are shown with stars on the algorithm names in Fig.3. Particularly, the star over the name indicates that the performance of OHGD significantly differs from the corresponding methods at the significance level $\alpha = 0.05$. As we can see, in most cases, OHGD significantly outperforms other competitors in terms of AUC, F-measure and G-mean. Similar observations also can be found in the comparisons with the chunk-based methods (Figure.(4)).

4.4 Average Accumulative Loss Analysis

We proceed to analyze the theoretical performance of OHGD in this sub-section. Particularly, we present the *Average Accumulative Loss* to show the regret trends:

$$AvgLoss_t = \frac{\sum_{i=1}^{t} \mathcal{L}_t}{t} \tag{22}$$

Figure.5 illustrates the $AvgLoss$ trends of OHGD on the datasets a8a and ijcnn. We observe that all the curves decrease rapidly and generally converge to a constant, which shows agreement with our $O(\sqrt{T})$ regret convergence expectations in the Theorem.2.

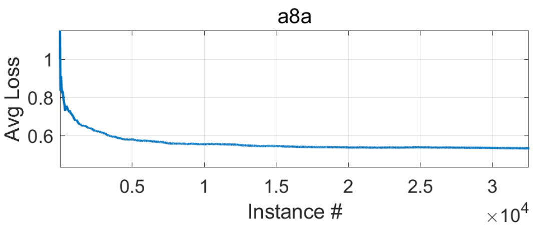


Figure 5: The average accumulative loss curves of the proposed method, on datasets a8a.

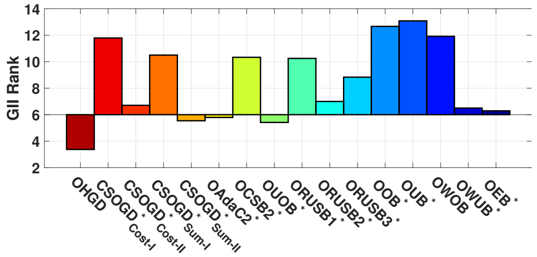


Figure 6: Overall ranks compared with one-pass learning methods, in terms of GII. Rank 1 indicates the method with the highest performance. The star denotes that the proposed method is significantly different with the corresponding methods by the Wilcoxon signed-rank test at $\alpha = 0.05$.

4.5 Rationality of Gradient Imbalance

We also verify the rationality of our motivation in this paper, i.e. gradient imbalance. Recalling the defined gradient imbalance in Eq.3, to further measure it over the learning procedure, we modify it as GI Index (GII):

$$GII = \sum_{t=1}^{T} (GI_t - 1)^2 \tag{23}$$

A smaller GII clearly indicates a more harmonizing gradient ratio over the online classification procedure. We report the rank of the GII obtained by one-pass algorithms in the Figure.6, as we did in the previous sections. Not surprisingly, the rank in terms of the GII generally agrees with those in terms of classification metrics, indicating that ensuring the gradient ratio can indeed improve learning performance under imbalanced data streams.

4.6 Time Efficiency

Table.3 provides the overall time cost of each algorithm on the 24 datasets. Generally speaking, the algorithms can be divided into two groups: one that includes a single classifier and the other that includes multiple classifiers. Not surprisingly, ensemble learning algorithms require more time for learning. This relates to two aspects. 1) Ensemble learning algorithms are required to update more classifiers when training a received instance. 2) Ensemble learning algorithms adopt a resampling strategy for imbalanced data streams and they may update an instance several times if this instance needs to be over-sampled. Compared with the ensemble learning methods, the re-weighting strategy (OHGD, CSOGD) is much more efficient because it only updates weighting parameters at each round. The OHGD costs slightly more computational time because it is required to calculate the gradient norm. However, it is still generally efficient.

Methods	**OHGD**	**CSOGD**$_{Cost_I}$	**CSOGD**$_{Cost_{II}}$	**CSOGD**$_{Sum_I}$
Time	1.22	0.87	0.91	0.86
Methods	**CSOGD**$_{Sum_{II}}$	**OAdaC2**	**OCSB2**	**OUOB**
Time	0.96	20.83	63.71	18.65
Methods	**ORUSB1**	**ORUSB2**	**ORUSB3**	**OOB**
Time	26.58	25.97	25.46	17.76
Methods	**OUB**	**WOOB**	**WOUB**	**OEB**
Time	20.51	16.88	21.00	50+

Table 3: The overall running time (seconds) of different methods on 24 datasets.

5 Conclusion

Different from the previous assumptions on instance number imbalance or cost imbalance, this paper experimentally shows that the imbalance of instances can be implied by the imbalance of gradient. Then, we designed a weighted gradient descent step to harmonize the gradient imbalance and proposed the Online Harmonizing Gradient Descent, a solution to the class imbalance problem in the setting of online learning. The OHGD achieves satisfied $\mathcal{O}(\sqrt{T})$ regret bound with the time horizon. The proposed method was examined in 24 public datasets and compared with several SOTA methods. The encouraging experimental results show the competitive performance on online imbalance learning. Future work could extend the current model to multiclass problems.

References

[Bernardo *et al.*, 2020] Alessio Bernardo, Heitor Murilo Gomes, Jacob Montiel, Bernhard Pfahringer, Albert Bifet, and Emanuele Della Valle. C-smote: Continuous synthetic minority oversampling for evolving data streams. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 483–492. IEEE, 2020.

[Chawla *et al.*, 2002] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[Crammer *et al.*, 2006] Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. Online passive-aggressive algorithms. *J. Mach. Learn. Res.*, 7:551–585, dec 2006.

[Dredze *et al.*, 2008] Mark Dredze, Koby Crammer, and Fernando Pereira. Confidence-weighted linear classification. In *Proceedings of the 25th international conference on Machine learning*, pages 264–271, 2008.

[Du *et al.*, 2021] Hongle Du, Yan Zhang, Ke Gang, Lin Zhang, and Yeh-Cheng Chen. Online ensemble learning algorithm for imbalanced data stream. *Applied Soft Computing*, 107:107378, 2021.

[Galar *et al.*, 2011] Mikel Galar, Alberto Fernandez, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera. A review on ensembles for the class imbalance problem:

bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4):463–484, 2011.

[Grzyb *et al.*, 2021] Joanna Grzyb, Jakub Klikowski, and Michał Woźniak. Hellinger distance weighted ensemble for imbalanced data stream classification. *Journal of Computational Science*, 51:101314, 2021.

[Hoi *et al.*, 2021] Steven CH Hoi, Doyen Sahoo, Jing Lu, and Peilin Zhao. Online learning: A comprehensive survey. *Neurocomputing*, 459:249–289, 2021.

[Klikowski and Woźniak, 2022] Jakub Klikowski and Michał Woźniak. Deterministic sampling classifier with weighted bagging for drifted imbalanced data stream classification. *Applied Soft Computing*, page 108855, 2022.

[Loezer *et al.*, 2020] Lucas Loezer, Fabrício Enembreck, Jean Paul Barddal, and Alceu de Souza Britto Jr. Cost-sensitive learning for imbalanced data streams. In *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, pages 498–504, 2020.

[Lu *et al.*, 2019] Yang Lu, Yiu-Ming Cheung, and Yuan Yan Tang. Adaptive chunk-based dynamic weighted majority for imbalanced data streams with concept drift. *IEEE Transactions on Neural Networks and Learning Systems*, 31(8):2764–2778, 2019.

[Razavi-Far *et al.*, 2021] Roozbeh Razavi-Far, Maryam Farajzadeh-Zanajni, Boyu Wang, Mehrdad Saif, and Shiladitya Chakrabarti. Imputation-based ensemble techniques for class imbalance learning. *IEEE Transactions on Knowledge and Data Engineering*, 33(5):1988–2001, 2021.

[Sovilj *et al.*, 2020] Dušan Sovilj, Paul Budnarain, Scott Sanner, Geoff Salmon, and Mohan Rao. A comparative evaluation of unsupervised deep architectures for intrusion detection in sequential data streams. *Expert Systems with Applications*, 159:113577, 2020.

[Sun *et al.*, 2007] Yanmin Sun, Mohamed S Kamel, Andrew KC Wong, and Yang Wang. Cost-sensitive boosting for classification of imbalanced data. *Pattern recognition*, 40(12):3358–3378, 2007.

[Varnali, 2021] Kaan Varnali. Online behavioral advertising: An integrative review. *Journal of Marketing Communications*, 27(1):93–114, 2021.

[Wang and Pineau, 2016] Boyu Wang and Joelle Pineau. Online bagging and boosting for imbalanced data streams. *IEEE Transactions on Knowledge and Data Engineering*, 28(12):3353–3366, 2016.

[Wang *et al.*, 2013a] Jialei Wang, Peilin Zhao, and Steven CH Hoi. Cost-sensitive online classification. *IEEE Transactions on Knowledge and Data Engineering*, 26(10):2425–2438, 2013.

[Wang *et al.*, 2013b] Shuo Wang, Leandro L Minku, and Xin Yao. A learning framework for online class imbalance learning. In *2013 IEEE Symposium on Computational Intelligence and Ensemble Learning (CIEL)*, pages 36–45. IEEE, 2013.

[Wang *et al.*, 2014] Shuo Wang, Leandro L Minku, and Xin Yao. Resampling-based ensemble methods for online class imbalance learning. *IEEE Transactions on Knowledge and Data Engineering*, 27(5):1356–1368, 2014.

[Wang *et al.*, 2021] Nan Wang, Ruozhou Liang, Xibin Zhao, and Yue Gao. Cost-sensitive hypergraph learning with f-measure optimization. *IEEE Transactions on Cybernetics*, 2021.

[Yu *et al.*, 2018] Hualong Yu, Xibei Yang, Shang Zheng, and Changyin Sun. Active learning from imbalanced data: A solution of online weighted extreme learning machine. *IEEE transactions on neural networks and learning systems*, 30(4):1088–1103, 2018.

[Zhang *et al.*, 2018] Chong Zhang, Kay Chen Tan, Haizhou Li, and Geok Soon Hong. A cost-sensitive deep belief network for imbalanced classification. *IEEE transactions on neural networks and learning systems*, 30(1):109–122, 2018.

[Zhang *et al.*, 2019] Yifan Zhang, Peilin Zhao, Shuaicheng Niu, Qingyao Wu, Jiezhang Cao, Junzhou Huang, and Mingkui Tan. Online adaptive asymmetric active learning with limited budgets. *IEEE Transactions on Knowledge and Data Engineering*, 33(6):2680–2692, 2019.

[Zhao *et al.*, 2018] Peilin Zhao, Yifan Zhang, Min Wu, Steven CH Hoi, Mingkui Tan, and Junzhou Huang. Adaptive cost-sensitive online classification. *IEEE Transactions on Knowledge and Data Engineering*, 31(2):214–228, 2018.

[Zhou *et al.*, 2020] Han Zhou, Hongpeng Yin, Hengyi Zheng, and Yanxia Li. A survey on multi-modal social event detection. *Knowledge-Based Systems*, 195:105695, 2020.

[Zhou *et al.*, 2022] Han Zhou, Hongpeng Yin, Dandan Zhao, and Li Cai. Incremental learning and conditional drift adaptation for non-stationary industrial process fault diagnosis. *IEEE Transactions on Industrial Informatics*, 2022.

[Zinkevich, 2003] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 928–936, 2003.

[Zong *et al.*, 2013] Weiwei Zong, Guang-Bin Huang, and Yiqiang Chen. Weighted extreme learning machine for imbalance learning. *Neurocomputing*, 101:229–242, 2013.

[Zyblewski *et al.*, 2021] Paweł Zyblewski, Robert Sabourin, and Michał Woźniak. Preprocessed dynamic classifier ensemble selection for highly imbalanced drifted data streams. *Information Fusion*, 66:138–154, 2021.