FAIR CLASS-INCREMENTAL LEARNING USING SAMPLE WEIGHTING

Anonymous authors
Paper under double-blind review

ABSTRACT

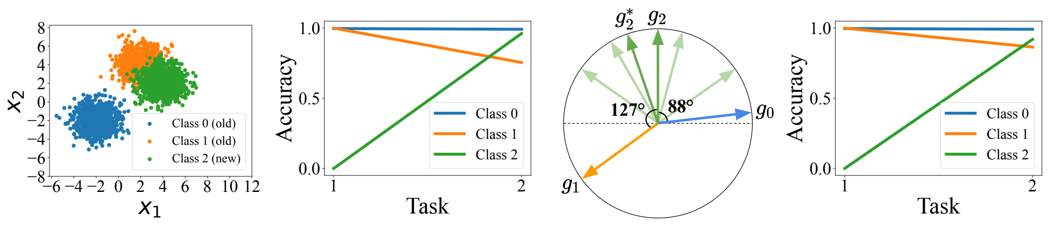
Model fairness is becoming important in class-incremental learning for Trustworthy AI. While accuracy has been a central focus in class-incremental learning, fairness has been relatively understudied. However, naïvely using all the samples of the current task for training results in *unfair catastrophic forgetting* for certain sensitive groups including classes. We theoretically analyze that forgetting occurs if the average gradient vector of the current task data is in an "opposite direction" compared to the average gradient vector of a sensitive group, which means their inner products are negative. We then propose a *fair class-incremental learning* framework that adjusts the training weights of current task samples to change the direction of the average gradient vector and thus reduce the forgetting of underperforming groups and achieve fairness. For various group fairness measures, we formulate optimization problems to minimize the overall losses of sensitive groups while minimizing the disparities among them. We also show the problems can be solved with linear programming and propose an efficient Fairness-aware Sample Weighting (FSW) algorithm. Experiments show that FSW achieves better accuracy-fairness tradeoff results than state-of-the-art approaches on real datasets.

1 INTRODUCTION

Trustworthy AI is becoming critical in various continual learning applications including autonomous vehicles, personalized recommendations, healthcare monitoring, and more (Liu et al., 2021; Kaur et al., 2023). In particular, it is important to improve model fairness along with accuracy when developing models incrementally in dynamic environments. Unfair model predictions have the potential to undermine the trust and safety in human-related automated systems, especially as observed frequently in the context of continual learning. There are largely three continual learning scenarios (van de Ven & Tolias, 2019): task-incremental, domain-incremental, and class-incremental learning where the task, domain, or class may change over time, respectively. In this paper, we focus on class-incremental learning, where the objective is to incrementally learn new classes as they appear.

The main challenge of class-incremental learning is to learn new classes of data, while not forgetting previously-learned classes (Belouadah et al., 2021; Lange et al., 2022). If we simply fine-tune the model on the new classes, the model will gradually forget about the previously-learned classes. This phenomenon called catastrophic forgetting (McCloskey & Cohen, 1989; Kirkpatrick et al., 2016) may easily occur in real-world scenarios where the model needs to continuously learn new classes. We cannot stop learning new classes to avoid this forgetting either. Instead, we need to have a balance between learning new information and retaining previously-learned knowledge, which is called the stability-plasticity dilemma (Abraham & Robins, 2005; Mermillod et al., 2013; Kim & Han, 2023).

In this paper, we solve the problem of *fair class-incremental learning* where the goal is to satisfy various notions of fairness among sensitive groups including classes in addition to classifying accurately. In some scenarios, the class itself can be considered a sensitive attribute, especially in classification tasks where a model produces biased predictions toward a specific group of classes (Truong et al., 2023). In continual learning, unfair forgetting may occur if the current task data has similar characteristics to previous data, but belongs to different sensitive groups including classes, which negatively affects the performance on the previous data during training. Despite the importance of the problem, the existing research (Chowdhury & Chaturvedi, 2023; Truong et al., 2023) is still nascent and has limitations in terms of technique or scope (see Sec. 2). In comparison, we support fairness more generally in class-incremental learning by satisfying various notions of group fairness for sensitive groups including classes.



(a) (b) (c) (d)

Figure 1: (a) A synthetic dataset for class-incremental learning. (b) After training on Classes 0 and 1, training on Class 2 results in unfair forgetting for Class 1 only. (c) The reason is that the average gradient vector of Class 2, g_2, is more than 90° apart from Class 1's g_1, which means the model is being trained in an opposite direction. Our method adjusts g_2 to g_2^* through sample weighting to be closer to g_1, but not too far from the original g_2. (d) As a result, the unfair forgetting is mitigated while minimally sacrificing accuracy for Class 2.

We demonstrate how unfair forgetting can occur on a synthetic dataset with two attributes (x_1, x_2), and one true label y as shown in Fig. 1a. We sample data for each class from three different normal distributions: $(x_1, x_2)|y = 0 \sim \mathcal{N}([-2; -2], [1; 1])$, $(x_1, x_2)|y = 1 \sim \mathcal{N}([2; 4], [1; 1])$, and $(x_1, x_2)|y = 2 \sim \mathcal{N}([4; 2], [1; 1])$. Note that each data distribution can also be defined as a sensitive group with a sensitive attribute z. To simulate class-incremental learning, we introduce data for Class 0 (blue) and Class 1 (orange) in Task 1, followed by Class 2 (green) data in Task 2, where Class 2's data is similar to Class 1's data. We observe that this setting frequently occurs in real datasets, where different classes of data exhibit similar features or characteristics, as shown in Sec. B.1. We assume a data replay setting where only a small amount of previous data from Classes 0 and 1 are stored and utilized together when training on Class 2 data. After training the model for Task 1, we observe how the model accuracies on the three classes change when training for Task 2 in Fig. 1b. As the accuracy on Class 2 improves, there is a catastrophic forgetting of Class 1 only, which leads to unfairness.

To analytically understand the unfair forgetting, we project the average gradient vector for each class data on a 2-dimensional space in Fig. 1c. Here g_0, g_1, and g_2 represent the average gradient vectors of the samples of Classes 0, 1, and 2, respectively. We observe that g_2 is 127° apart from g_1, but 88° from g_0, which means that the inner products $\langle g_2, g_1 \rangle$ and $\langle g_2, g_0 \rangle$ are negative and close to 0, respectively. In Sec. 3.1, we theoretically show that a negative inner product between average gradient vectors of current and previous data results in higher loss for the previous data as the model is being updated in an opposite direction and identify a sufficient condition for unfair forgetting. As a result, Class 1's accuracy decreases, while Class 0's accuracy remains stable.

Our solution to mitigate unfair forgetting is to adjust the average gradient vector of the current task data by weighting its samples. The light-green vectors in Fig. 1c are the gradient vectors of individual samples from Class 2, and by weighting them we can adjust g_2 to g_2^* to make the inner product with g_1 less negative. At the same time, we do not want g_2^* to be too different from g_2 and lose accuracy. In Sec. 3.2, we formalize this idea using the weighted average gradient vector of the current task data. We then optimize the sample weights such that unfair forgetting and accuracy reduction over sensitive groups including classes are both minimized. We show this optimization can be solved with linear programming and propose our efficient Fairness-aware Sample Weighting (FSW) algorithm. Fig. 1d shows how using FSW mitigates the unfair forgetting between Classes 0 and 1 without harming Class 2's accuracy much. Our framework supports the group fairness measures equal error rate (Venkatasubramanian, 2019), equalized odds (Hardt et al., 2016), and demographic parity (Feldman et al., 2015) and can be potentially extended to other measures.

In our experiments, we show that FSW achieves better fairness and competitive accuracy compared to state-of-the-art baselines on various image, text, and tabular datasets. The benefits come from assigning different training weights to the current task samples with accuracy and fairness in mind.

Summary of Contributions: (1) We theoretically analyze how unfair catastrophic forgetting can occur in class-incremental learning; (2) We formulate optimization problems for mitigating the unfairness for various group fairness measures and propose an efficient fairness-aware sample weighting algorithm, FSW; (3) We demonstrate how FSW outperforms state-of-the-art baselines in terms of fairness with comparable accuracy on various datasets.

2 Related Work

Class-incremental learning is a challenging type of continual learning where a model continuously learns new tasks, each composed of new disjoint classes, and the goal is to minimize catastrophic forgetting (Mai et al., 2022; Masana et al., 2023). Data replay techniques (Lopez-Paz & Ranzato, 2017; Chaudhry et al., 2019b) store a small portion of previous data in a buffer to utilize for training and is widely used with other techniques including knowledge distillation, model rectification, and dynamic networks (see more details in Sec. C). Simple buffer sample selection methods such as random or herding-based approaches (Rebuffi et al., 2017) are also commonly used as well. There are also more advanced gradient-based sample selection techniques like GSS (Aljundi et al., 2019) and OCS (Yoon et al., 2022) that manage buffer data to have samples with diverse and representative gradient vectors. All these works do not consider fairness and simply assume that the entire incoming data is used for model training, which may result in unfair forgetting as we show in our experiments.

Model fairness research mitigates bias by ensuring that a model's performance is equitable across different sensitive groups, thereby preventing discrimination based on race, gender, age, or other sensitive attributes (Mehrabi et al., 2022). Existing model fairness techniques can be categorized as pre-processing, in-processing, and post-processing (see more details in Sec. C). In addition, there are other techniques that assign adaptive weights for samples to improve fairness (Chai & Wang, 2022; Jung et al., 2023). However, most of these techniques assume that the training data is given all at once, which may not be realistic. There are techniques for fairness-aware active learning (Anahideh et al., 2022; Pang et al., 2024; Tae et al., 2024), in which the training data evolves with the acquisition of samples. However, these techniques store all labeled data and use them for training, which is impractical in continual learning settings.

A recent study addresses model fairness in class-incremental learning where there is a risk of disproportionately forgetting previously-learned sensitive groups including classes, leading to unfairness across different groups. A recent study (He, 2024) addresses the dual imbalance problem involving both inter-task and intra-task imbalance by reweighting gradients. However, the bias is not only caused by the data imbalance, but also by the inherent or acquired characteristics of data (Mehrabi et al., 2021; Angwin et al., 2022). CLAD (Xu et al., 2024) first discovers imbalanced forgetting between learned classes caused by conflicts in representation and proposes a class-aware disentanglement technique to improve accuracy. Among the fairness-aware techniques, FaIRL (Chowdhury & Chaturvedi, 2023) supports group fairness measures like demographic parity for class-incremental tasks, but proposes a representation learning method that does not directly optimize the given fairness measure and thus has limitations in improving fairness as we show in experiments. FairCL (Truong et al., 2023) also addresses fairness in a continual learning setup, but only focuses on resolving the imbalanced class distribution based on the number of pixels of each class in an image for semantic segmentation tasks. In comparison, we support fairness more generally in class-incremental learning by satisfying multiple notions of group fairness for sensitive groups including classes.

3 Framework

In this section, we first theoretically analyze unfair forgetting using gradient vectors of sensitive groups and the current task data. Next, we propose sample weighting to mitigate unfairness by adjusting the average gradient vector of the current task data and provide an efficient algorithm. We use the following notations for class-incremental learning and fairness.

Notations In class-incremental learning, a model incrementally learns new current task data along with previous buffer data using data replay. Suppose we train a model to incrementally learn L tasks $\{T_1, T_2, \ldots, T_L\}$ over time, and there are N classes in each task as $C^{T_l} = \{C_1^{T_l}, C_2^{T_l}, \ldots, C_N^{T_l}\}$ with no overlapping classes between different tasks (i.e., $C^{T_{l_1}} \cap C^{T_{l_2}} = \emptyset$ if $l_1 \neq l_2$). After learning the l^{th} task T_l, we would like the model to remember all $(l-1) \cdot N$ previous task classes and an additional N current task classes. We assume the buffer has a fixed size of M samples. For L tasks, we allocate $m = M/L$ samples of buffer data per task. If each task consists of N classes, then we allocate $m/N = M/(L \cdot N)$ samples of buffer data per class (Chaudhry et al., 2019a; Mirzadeh et al., 2020; Chaudhry et al., 2021). Each task $T_l = \{d_i = (X_i, y_i)\}_{i=1}^{k}$ is composed of feature-label pairs where a feature $X_i \in \mathbb{R}^d$ and a true label $y_i \in \mathbb{R}^c$. We also use $\mathcal{M}_l = \{d_j = (X_j, y_j)\}_{j=1}^{m}$ to represent the buffer data for each previous l^{th} task T_l. We assume the buffer data per task is small, i.e., $m \ll k$ (Chaudhry et al., 2019b).

When defining fairness for class-incremental learning, we utilize sensitive groups including classes. According to the fairness literature, sensitive groups are divided by sensitive attributes like gender and race. For example, if the sensitive attribute is gender, the sensitive groups can be Male and Female. The classes of class-incremental learning can also be viewed as sensitive groups where the sensitive attribute is the class. Since we would like to support any sensitive group in a class-incremental setting, we use the following unifying notations: (1) if the sensitive groups are classes, then they form the set $G_y = \{(X, \mathrm{y}) \in \mathcal{D} : \mathrm{y} = y, y \in \mathbb{Y}\}$ where $\mathcal{D}$ is a dataset, y is a class attribute, and $\mathbb{Y}$ is the set of classes; (2) if we are using sensitive attributes in addition to classes, we can further divide the classes into the set $G_{y,z} = \{(X, \mathrm{y}, \mathrm{z}) \in \mathcal{D} : \mathrm{y} = y, \mathrm{z} = z, y \in \mathbb{Y}, z \in \mathbb{Z}\}$ where z is a sensitive attribute, and $\mathbb{Z}$ is the set of sensitive attribute values.

3.1 Unfair Forgetting

Catastrophic forgetting occurs when a model adapts to a new task and exhibits a drastic decrease in performance on previously-learned tasks (Parisi et al., 2019). We take inspiration from GEM (Lopez-Paz & Ranzato, 2017), which theoretically analyzes catastrophic forgetting by utilizing the angle between gradient vectors of data. If the inner products of gradient vectors for previous tasks and the current task are negative (i.e., $90^\circ <$ angle $\leq 180^\circ$), the loss of previous tasks increases after learning the current task. Catastrophic forgetting thus occurs when the gradient vectors of different tasks point in opposite directions. Intuitively, the opposite gradient vectors update the model parameters in conflicting directions, leading to forgetting while learning.

Using the notion of catastrophic forgetting, we propose theoretical results for unfair forgetting:

Lemma 1. *Denote G as a sensitive group of data composed of features X and true labels y. Also, denote f_θ^{l-1} as a previous model and f_θ as the updated model after training on the current task T_l. Let ℓ be any differentiable standard loss function (e.g., cross-entropy loss), and η be a learning rate. Then, the loss of the sensitive group of data after training with a current task sample $d_i \in T_l$ is approximated as follows:*

$$\tilde{\ell}(f_\theta, G) = \ell(f_\theta^{l-1}, G) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i), \tag{1}$$

where $\tilde{\ell}(f_\theta, G)$ is the approximated average loss between model predictions $f_\theta(X)$ and true labels y, whereas $\ell(f_\theta^{l-1}, G)$ is the exact average loss, $\nabla_\theta \ell(f_\theta^{l-1}, G)$ is the average gradient vector for the samples in the group G, and $\nabla_\theta \ell(f_\theta^{l-1}, d_i)$ is the gradient vector for a sample d_i, each with respect to the previous model f_θ^{l-1}.

The proof is in Sec. A.1. We employ first-order Taylor series approximation for the proof, which is widely used in the continual learning literature, by assuming that the loss function is locally linear in small optimization steps and considering the first-order term as the cause of catastrophic forgetting (Lopez-Paz & Ranzato, 2017; Aljundi et al., 2019; Lee et al., 2019). We empirically find that the approximation error is large when a new task begins because new samples with unseen classes are introduced. However, the error gradually becomes quite small as the number of epochs increases while training a model for the task, as shown in Sec. B.2.

To define fairness in class-incremental learning with the approximated loss, we adopt the definition of approximate fairness that considers a model to be fair if it has approximately the same loss on the positive class, independent of the group membership (Donini et al., 2018). In this paper, we use the cross-entropy loss for training and compute fairness measures based on the disparity between approximated cross-entropy losses, which are derived from Lemma 1 using gradients. The following proposition shows how using the cross-entropy loss can effectively approximate common group fairness metrics such as equalized odds and demographic parity (see Sec. A.2 for more details).

Proposition 1. *(From Roh et al. (2021; 2023); Shen et al. (2022)) Using cross-entropy loss to measure fairness is empirically verified to provide reasonable proxies for common group fairness metrics.*

Using Lemma 1 and Proposition 1, the following theorem suggests a sufficient condition for unfair forgetting. Intuitively, if a training sample's gradient is in an opposite direction to the average gradient of an underperforming group, but not for an overperforming group, the training causes more unfairness between the two groups.

Theorem 1. *Let ℓ be the cross-entropy loss and we denote G_1 and G_2 as the overperforming and underperforming sensitive groups of data, and d_i as a training sample that satisfy the following conditions: $\ell(f_\theta^{l-1}, G_1) < \ell(f_\theta^{l-1}, G_2)$ while $\nabla_\theta \ell(f_\theta^{l-1}, G_1)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i) > 0$ and $\nabla_\theta \ell(f_\theta^{l-1}, G_2)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i) < 0$. Then $|\tilde{\ell}(f_\theta, G_1) - \tilde{\ell}(f_\theta, G_2)| > |\ell(f_\theta^{l-1}, G_1) - \ell(f_\theta^{l-1}, G_2)|$.*

The proof is in Sec. A.1. The result shows that the disparity of loss between the two groups could become larger after training on the current task sample, which leads to worse fairness. This theorem can be extended to when we have a set of current task samples $T_l = \{d_i = (X_i, y_i)\}_{i=1}^k$ where we can replace $\nabla_\theta \ell(f_\theta^{l-1}, d_i)$ with $\frac{1}{|T_l|}\sum_{d_i \in T_l} \nabla_\theta \ell(f_\theta^{l-1}, d_i)$. If the average gradient vector of the current task data satisfies the derived sufficient condition, training with all of the current task samples using equal weights could thus result in unfair catastrophic forgetting.

3.2 Sample Weighting for Unfairness Mitigation

To mitigate unfairness, we propose sample weighting as a way to suppress samples that negatively impact fairness and promote samples that help. Finding the weights is not trivial as there can be many sensitive groups, and even a single sample may improve the fairness of a pair of groups, but worsen the fairness for another pair of groups. Given training weights $\mathbf{w}_l \in [0,1]^{|T_l|}$ for the samples in the current task data, the approximated loss of a group G after training is now:

$$\tilde{\ell}(f_\theta, G) = \ell(f_\theta^{l-1}, G) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G)^\top \left(\frac{1}{|T_l|} \sum_{d_i \in T_l} \mathbf{w}_l^i \nabla_\theta \ell(f_\theta^{l-1}, d_i) \right), \tag{2}$$

where $\mathbf{w}_l^i$ is a training weight for the current task sample d_i. We then formulate an optimization problem to find the weights such that both loss and unfairness are minimized. Here we define $\mathbb{Y}$ as the set of all classes and $\mathbb{Y}_c$ as the set of classes in the current task. We represent accuracy as the average loss over the current task data and minimize the cost function $L_{acc} = \tilde{\ell}(f_\theta, G_{\mathbb{Y}_c}) = \frac{1}{|\mathbb{Y}_c|}\sum_{y \in \mathbb{Y}_c} \tilde{\ell}(f_\theta, G_y) = \frac{1}{|\mathbb{Y}_c||\mathbb{Z}|}\sum_{y \in \mathbb{Y}_c, z \in \mathbb{Z}} \tilde{\ell}(f_\theta, G_{y,z})$. For fairness, the cost function L_{fair} depends on the group fairness measure as we explain below. We then minimize $L_{fair} + \lambda L_{acc}$ where λ is a hyperparameter that balances fairness and accuracy.

Equal Error Rate (EER) This measure (Venkatasubramanian, 2019) is defined as $\Pr(\hat{\mathrm{y}} \neq y_1 | \mathrm{y} = y_1) = \Pr(\hat{\mathrm{y}} \neq y_2 | \mathrm{y} = y_2)$ for $y_1, y_2 \in \mathbb{Y}$, where $\hat{\mathrm{y}}$ is the predicted class and y is the true class. We define the cost function for EER as the average absolute difference between the loss of a class and the average loss of all classes, following the definition of group fairness metrics: $L_{EER} = \frac{1}{|\mathbb{Y}|}\sum_{y \in \mathbb{Y}} |\tilde{\ell}(f_\theta, G_y) - \tilde{\ell}(f_\theta, G_{\mathbb{Y}})|$. The entire optimization problem is:

$$\min_{\mathbf{w}_l} \frac{1}{|\mathbb{Y}|} \sum_{y \in \mathbb{Y}} |\tilde{\ell}(f_\theta, G_y) - \tilde{\ell}(f_\theta, G_{\mathbb{Y}})| + \lambda \frac{1}{|\mathbb{Y}_c|} \sum_{y \in \mathbb{Y}_c} \tilde{\ell}(f_\theta, G_y), \tag{3}$$

$$\text{where } \tilde{\ell}(f_\theta, G_y) = \ell(f_\theta^{l-1}, G_y) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G_y)^\top \left(\frac{1}{|T_l|} \sum_{d_i \in T_l} \mathbf{w}_l^i \nabla_\theta \ell(f_\theta^{l-1}, d_i) \right).$$

Equalized Odds (EO) This measure (Hardt et al., 2016) is satisfied when sensitive groups have the same accuracy, i.e., $\ell(f_\theta, G_{y,z_1}) = \ell(f_\theta, G_{y,z_2})$ for $y \in \mathbb{Y}$ and $z_1, z_2 \in \mathbb{Z}$. We design the cost function for EO as $L_{EO} = \frac{1}{|\mathbb{Y}||\mathbb{Z}|}\sum_{y \in \mathbb{Y}, z \in \mathbb{Z}} |\tilde{\ell}(f_\theta, G_{y,z}) - \tilde{\ell}(f_\theta, G_y)|$ to compute the EO disparity, and the entire optimization problem is:

$$\min_{\mathbf{w}_l} \frac{1}{|\mathbb{Y}||\mathbb{Z}|} \sum_{y \in \mathbb{Y}, z \in \mathbb{Z}} |\tilde{\ell}(f_\theta, G_{y,z}) - \tilde{\ell}(f_\theta, G_y)| + \lambda \frac{1}{|\mathbb{Y}_c||\mathbb{Z}|} \sum_{y \in \mathbb{Y}_c, z \in \mathbb{Z}} \tilde{\ell}(f_\theta, G_{y,z}), \tag{4}$$

$$\text{where } \tilde{\ell}(f_\theta, G_{y,z}) = \ell(f_\theta^{l-1}, G_{y,z}) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G_{y,z})^\top \left(\frac{1}{|T_l|} \sum_{d_i \in T_l} \mathbf{w}_l^i \nabla_\theta \ell(f_\theta^{l-1}, d_i) \right).$$

Demographic Parity (DP) This measure (Feldman et al., 2015) is satisfied by minimizing the difference in positive prediction rates between sensitive groups. Here, we extend the notion of demographic parity to the multi-class setting (Alabdulmohsin et al., 2022; Denis et al., 2023), i.e., $\Pr(\hat{\mathrm{y}} = y | \mathrm{z} = z_1) = \Pr(\hat{\mathrm{y}} = y | \mathrm{z} = z_2)$ for $y \in \mathbb{Y}$ and $z_1, z_2 \in \mathbb{Z}$. In the binary setting of $\mathbb{Y} = \mathbb{Z} = \{0, 1\}$, a sufficient condition for demographic parity is suggested using the loss multiplied

Algorithm 1 Fair Class-Incremental Learning

Input: Current task data T_l, previous buffer data $\mathcal{M} = \{\mathcal{M}_1, \dots, \mathcal{M}_{l-1}\}$, previous model f_θ^{l-1}, loss function ℓ, learning rate η, hyperparameters $\{\alpha, \lambda, \tau\}$, fairness measure F

1: **for** each epoch **do**
2: $\quad \mathbf{w}_l^* = \boldsymbol{FSW}(T_l, \mathcal{M}, f_\theta^{l-1}, \ell, \alpha, \lambda, F)$
3: $\quad g_{curr} = \frac{1}{|T_l|} \sum_{d_i \in T_l} \mathbf{w}_l^{*i} \nabla_\theta \ell(f_\theta^{l-1}, d_i)$
4: $\quad g_{prev} = \nabla_\theta \ell(f_\theta^{l-1}, \mathcal{M})$
5: $\quad \theta^l = \theta^{l-1} - \eta(g_{curr} + \tau g_{prev})$
6: $\mathcal{M}_l$ = *Buffer Sample Selection*(T_l)
7: $\mathcal{M} = \mathcal{M} \cup \mathcal{M}_l$

Algorithm 2 Fairness-aware Sample Weighting (FSW)

Input: Current task data $T_l = \{d_1, \dots, d_k\}$, previous buffer data $\mathcal{M} = \cup_{y \in \mathbb{Y} - \mathbb{Y}_c, z \in \mathbb{Z}} G_{y,z}$, previous model f_θ^{l-1}, loss function ℓ, hyperparameters $\{\alpha, \lambda\}$, fairness measure F

Output: Optimal training weights $\mathbf{w}_l^*$ for current task data

1: $\ell_G = [\ell(f_\theta^{l-1}, G_{1,1}), \dots, \ell(f_\theta^{l-1}, G_{|\mathbb{Y}|,|\mathbb{Z}|})]$
2: $g_G = [\nabla_\theta \ell(f_\theta^{l-1}, G_{1,1}), \dots, \nabla_\theta \ell(f_\theta^{l-1}, G_{|\mathbb{Y}|,|\mathbb{Z}|})]$
3: $g_d = [\nabla_\theta \ell(f_\theta^{l-1}, d_1), \dots, \nabla_\theta \ell(f_\theta^{l-1}, d_k)]$
4: **switch** F **do**
5: $\quad$ **case** EER: $\mathbf{w}_l^* \leftarrow$ Solve Eq. 3
6: $\quad$ **case** EO: $\mathbf{w}_l^* \leftarrow$ Solve Eq. 4
7: $\quad$ **case** DP: $\mathbf{w}_l^* \leftarrow$ Solve Eq. 5
8: **return** $\mathbf{w}_l^*$

by the ratios of sensitive groups (Roh et al., 2021). By extending the setting to multi-class, we derive a sufficient condition for demographic parity as follows: $\frac{m_{y,z_1}}{m_{*,z_1}} \ell(f_\theta, G_{y,z_1}) = \frac{m_{y,z_2}}{m_{*,z_2}} \ell(f_\theta, G_{y,z_2})$ where $m_{y,z} := |\{i : \mathrm{y}_i = y, \mathrm{z}_i = z\}|$ and $m_{*,z} := |\{i : \mathrm{z}_i = z\}|$. The proof is in Sec. A.3. Let us define $\ell'(f_\theta, G_{y,z}) = \frac{m_{y,z}}{m_{*,z}} \ell(f_\theta, G_{y,z})$ and $\ell'(f_\theta, G_y) = \frac{1}{|\mathbb{Z}|} \sum_{n=1}^{|\mathbb{Z}|} \frac{m_{y,z_n}}{m_{*,z_n}} \ell(f_\theta, G_{y,z_n})$. We then define the cost function for DP using the sufficient condition as $L_{DP} = \frac{1}{|\mathbb{Y}||\mathbb{Z}|} \sum_{y \in \mathbb{Y}, z \in \mathbb{Z}} |\tilde{\ell}'(f_\theta, G_{y,z}) - \tilde{\ell}'(f_\theta, G_y)|$. The entire optimization problem is:

$$\min_{\mathbf{w}_l} \frac{1}{|\mathbb{Y}||\mathbb{Z}|} \sum_{y \in \mathbb{Y}, z \in \mathbb{Z}} |\tilde{\ell}'(f_\theta, G_{y,z}) - \tilde{\ell}'(f_\theta, G_y)| + \lambda \frac{1}{|\mathbb{Y}_c||\mathbb{Z}|} \sum_{y \in \mathbb{Y}_c, z \in \mathbb{Z}} \tilde{\ell}(f_\theta, G_{y,z}), \tag{5}$$

$$\text{where } \tilde{\ell}(f_\theta, G_{y,z}) = \ell(f_\theta^{l-1}, G_{y,z}) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G_{y,z})^\top \left(\frac{1}{|T_l|} \sum_{d_i \in T_l} \mathbf{w}_l^i \nabla_\theta \ell(f_\theta^{l-1}, d_i) \right).$$

To find the optimal sample weights for the current task data considering both model accuracy and fairness, we first transform the defined optimization problems of Eq. 3, 4, and 5 into the form of linear programming (LP) problems.

Theorem 2. *The fairness-aware optimization problems (Eq. 3, 4, and 5) can be transformed into the form of linear programming (LP) problems.*

The loss of each group can be approximated as a linear function, as described in Lemma 1. This implies that the optimization problems, consisting of the loss of each group, can likewise be transformed into LP problems. A comprehensive proof of this assertion can be found in Sec. A.4. We then solve the LP problems using linear optimization solvers (e.g., CPLEX (Cplex, 2009)).

3.3 Algorithm

We describe the overall process of fair class-incremental learning in Alg. 1. For the recently arrived current task data, we first perform fairness-aware sample weighting (FSW) to assign training weights that can help learn new knowledge of the current task while retaining accurate and fair memories of previous tasks (Step 2). Next, we train the model using the current task data with its corresponding weights and stored buffer data of previous tasks (Steps 3–5), where η is a learning rate, and τ is a hyperparameter to balance between them during training. The sample weights are computed once at the beginning of each epoch, and they are applied to all batches for computational efficiency (Killamsetty et al., 2021b;a). This procedure is repeated until the model converges (Steps 1–5). Before moving on to the next task, we employ buffer sample selection to store a small data subset for the current task (Steps 6–7). Buffer sample selection can also be done with consideration for fairness, but our experimental observations indicate that selecting representative and diverse samples for the buffer, as previous studies have shown, results in better accuracy and also fairness performance. We thus employ a simple random sampling technique for the buffer sample selection in our framework.

Alg. 2 shows the fairness-aware sample weighting (FSW) algorithm for the current task data. We first divide both the previous buffer data and the current task data into groups based on each class and sensitive attribute. Next, we compute the average loss and gradient vectors for each group

Table 1: Experimental settings for the five datasets. If the class is used as the sensitive attribute, the number of classes is the same as the number of sensitive groups.

Dataset	Size	#Features	#Classes	#Tasks	#Sensitive groups
MNIST	60K	28×28	10	5	10
FMNIST	60K	28×28	10	5	10
Biased MNIST	60K	3×28×28	10	5	2
DRUG	1.3K	12	6	3	2
BiasBios	253K	128×768	25	5	2

(Steps 1–2), and individual gradient vectors for the current task data (Step 3). To compute gradient vectors, we use the last layer approximation, which only considers the gradients of the model's last layer, that is efficient and known to be reasonable (Katharopoulos & Fleuret, 2018; Ash et al., 2020; Mirzasoleiman et al., 2020). We then solve linear programming to find the optimal sample weights for a user-defined target fairness measure such as EER (Step 5), EO (Step 6), and DP (Step 7). We use CPLEX as a linear optimization solver that employs an efficient simplex-based algorithm. Since the gradient norm of the current task data is significantly larger than that of the buffer data, we utilize normalized gradients to update the loss of each group and replace the learning rate parameter η with a hyperparameter α in the equations. Finally, we return the weights for the current task samples to be used during training (Step 8).

Training with FSW theoretically guarantees model convergence under the assumptions that the training loss is Lipschitz continuous and strongly convex, and that a proper learning rate is used (Killamsetty et al., 2021a; Chai & Wang, 2022; Lu et al., 2020). The computational complexity of FSW is quadratic to the number of current task samples, as CPLEX generally has quadratic complexity with respect to the number of variables when solving LP problems (Bixby, 2002). However, our empirical results show that for about ten thousand current task samples, the time to solve an LP problem is a few seconds, which leads to a few minutes of overall runtime for MNIST datasets (see Sec. B.3 for details). Since we focus on continual offline training of large batches or separate tasks, rather than online learning, the overhead is manageable enough to deploy updated models in real-world applications. If the task size becomes too large, clustering similar samples and assigning weights to the clusters, rather than samples, could be a solution to reduce the computational overhead.

4 Experiments

We implement FSW using Python and PyTorch. All evaluations are performed on separate test sets and repeated with five random seeds. We write the average and standard deviation of performance results and run experiments on Intel Xeon Silver 4114 CPUs and NVIDIA TITAN RTX GPUs.

Metrics We evaluate all methods using accuracy and fairness metrics as in the fair continual learning literature (Chowdhury & Chaturvedi, 2023; Truong et al., 2023).

- **Average Accuracy**: We denote $A_l = \frac{1}{l}\sum_{t=1}^{l} a_{l,t}$ as the accuracy at the l^{th} task, where $a_{l,t}$ is the accuracy of the t^{th} task after learning the l^{th} task. We measure accuracy for each task and then take the average across all tasks to produce the final average accuracy, denoted as $\overline{A_l} = \frac{1}{L}\sum_{l=1}^{L} A_l$ where L represents the total number of tasks.
- **Average Fairness**: We measure fairness for each task and then take the average across all tasks to produce the final average fairness. We use one of three measures for per-task fairness: (1) Equal Error Rate (EER) disparity, which computes the average difference in test error rates among classes: $\frac{1}{|\mathbb{Y}|}\sum_{y\in\mathbb{Y}} |\Pr(\hat{\mathrm{y}} \neq y|\mathrm{y}=y) - \Pr(\hat{\mathrm{y}} \neq \mathrm{y})|$; (2) Equalized Odds (EO) disparity, which computes the average difference in accuracy among sensitive groups for all classes: $\frac{1}{|\mathbb{Y}||\mathbb{Z}|}\sum_{y\in\mathbb{Y},z\in\mathbb{Z}} |\Pr(\hat{\mathrm{y}} = y|\mathrm{y}=y, \mathrm{z}=z) - \Pr(\hat{\mathrm{y}} = y|\mathrm{y}=y)|$; and (3) Demographic Parity (DP) disparity, which computes the average difference in class prediction ratios among sensitive groups for all classes: $\frac{1}{|\mathbb{Y}||\mathbb{Z}|}\sum_{y\in\mathbb{Y},z\in\mathbb{Z}} |\Pr(\hat{\mathrm{y}} = y|\mathrm{z}=z) - \Pr(\hat{\mathrm{y}} = y)|$. For all measures, low disparity is desirable.

Datasets We use a total of five datasets as shown in Table 1. We first utilize commonly used benchmarks for continual image classification tasks, which include MNIST and Fashion-MNIST (FMNIST). Here we regard the class as the sensitive attribute and evaluate fairness with EER disparity.

We also use multi-class fairness benchmark datasets that have sensitive attributes (Xu et al., 2020; Putzel & Lee, 2022; Churamani et al., 2023; Denis et al., 2023): Biased MNIST, Drug Consumption (DRUG), and BiasBios. We consider background color as the sensitive attribute for Biased MNIST, and gender for DRUG and BiasBios, respectively. We use EO disparity and DP disparity to evaluate fairness on these datasets. We also consider using standard benchmark datasets in the fairness field, but they are unsuitable for class-incremental learning experiments because either there are only two classes, or it is difficult to compute fairness (see Sec. B.4 for more details). For datasets with a total of $\mathcal{C}$ classes, we divide the datasets into L sequences of tasks where each task consists of $\mathcal{C}/L$ classes, and assume that task boundaries are available (van de Ven & Tolias, 2019).

Models Following the experimental setups of Chaudhry et al. (2019a); Mirzadeh et al. (2020), we use a two-layer MLP with each 256 neurons for the MNIST, FMNIST, Biased MNIST, and DRUG datasets. For BiasBios, we use a pre-trained BERT language model (Devlin et al., 2019; Xian et al., 2023). We employ single-head evaluation where a final layer of the model is shared for all the tasks (Farquhar & Gal, 2018; Chaudhry et al., 2018). For training, we use an SGD optimizer with momentum 0.9 for all the experiments. We set appropriate learning rates and epochs for each dataset, with detailed experimental settings provided in Sec. B.5.

Baselines In the continual learning literature (Aljundi et al., 2019; Yoon et al., 2022), it is natural for all the baselines to be continual learning methods. In particular, we consider *FaIRL* (Chowdhury & Chaturvedi, 2023) to be the first fairness paper for continual learning. We thus compare our algorithm with the following baselines categorized into four types:

- **Naïve methods**: *Joint Training* assumes access to all the data of previous classes for training and thus has an upper-bound performance; *Fine Tuning* trains a model using only new classes of data without access to previous data and thus has a lower-bound performance.
- **State-of-the-art methods**: *iCaRL* (Rebuffi et al., 2017) performs herding-based buffer selection and representation learning using additional knowledge distillation loss; *WA* (Zhao et al., 2020) is a model rectification method designed to correct the bias in the last fully-connected layer of the model. *WA* uses weight aligning techniques to align the norms of the weight vectors over classes; *CLAD* (Xu et al., 2024) is a representation learning method that disentangles the representation interference between old and new classes.
- **Sample selection methods**: *GSS* (Aljundi et al., 2019) and *OCS* (Yoon et al., 2022) are gradient-based sample selection methods. *GSS* selects a buffer with diverse gradients of samples; *OCS* uses gradient-based similarity, diversity, and affinity scores to rank and select samples for both current and buffer data.
- **Fairness-aware methods**: *FaIRL* (Chowdhury & Chaturvedi, 2023) performs fair representation learning by controlling the rate-distortion function of representations. *FairCL* (Truong et al., 2023) addresses fairness in semantic segmentation tasks arising from the imbalanced class distribution of pixels, but we consider this problem to be unrelated from ours to add the method as a baseline.

Hyperparameters For our buffer storage, we evenly divide the buffer by the sensitive groups including classes. We store 32 samples per sensitive group for all experiments. For the hyperparameters α, λ, and τ used in our algorithms, we perform cross-validation with a sequential grid search to find their optimal values one by one while freezing the other parameters. See Sec. B.5 for more details.

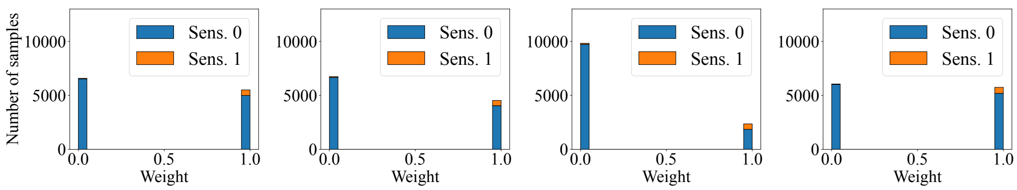
4.1 Accuracy and Fairness Results

We compare FSW against other baselines on the five datasets with respect to accuracy and corresponding fairness metrics as shown in Table 2. The results for DP disparity and BiasBios dataset are similar and shown in Sec. B.7. We mark the best and second-best results with bold and underline, respectively, excluding the upper-bound results of *Joint Training* and the lower-bound results of *Fine Tuning*. For any method, we store a fixed number of samples per task in a buffer, which may not be identical to its original setup, but necessary for a fair comparison. The detailed accuracy-fairness tradeoff and sequential performance results are shown in Sec. B.6 and Sec. B.7, respectively.

Overall, FSW achieves better accuracy-fairness tradeoff results compared to the baselines for all the datasets. For the DRUG dataset, although FSW does not achieve the best performance in either

Table 2: Accuracy and fairness results on the four datasets with respect to (1) EER disparity, where class is considered the sensitive attribute for the MNIST and FMNIST datasets, and (2) EO disparity, where background color and gender are the sensitive attributes for the Biased MNIST and DRUG datasets, respectively. We compare FSW with four types of baselines: naïve (*Joint Training* and *Fine Tuning*), state-of-the-art (*iCaRL*, *WA*, and *CLAD*), sample selection (*GSS* and *OCS*), and fairness-aware (*FaIRL*) methods.

Methods	MNIST		FMNIST		Biased MNIST		DRUG	
	Acc.	EER Disp.	Acc.	EER Disp.	Acc.	EO Disp.	Acc.	EO Disp.
Joint Training	.970±.004	.014±.006	.895±.010	.035±.004	.945±.002	.053±.002	.441±.015	.179±.052
Fine Tuning	.453±.000	.323±.000	.450±.000	.324±.000	.448±.001	.010±.003	.357±.009	.125±.034
iCaRL	**.934±.004**	.037±.003	**.862±.002**	.053±.003	.818±.011	.347±.025	**.458±.014**	.216±.056
WA	.911±.007	.052±.006	.809±.005	.088±.003	.447±.001	**.018±.002**	.358±.009	.112±.038
CLAD	.835±.015	.099±.015	.775±.018	.115±.019	.872±.001	.195±.020	.410±.026	.114±.043
GSS	.886±.007	.080±.009	.730±.013	.150±.011	.819±.009	.313±.021	.433±.011	.177±.045
OCS	.901±.003	.061±.004	.785±.012	.092±.007	.833±.012	.303±.024	.429±.007	.169±.026
FaIRL	.458±.008	.306±.004	.455±.005	.316±.001	.759±.008	.408±.018	.318±.006	**.015±.009**
FSW	.924±.003	**.032±.004**	.825±.006	**.037±.007**	**.909±.003**	.060±.004	.429±.020	.138±.037



(a) Task 2. (b) Task 3. (c) Task 4. (d) Task 5.

Figure 2: Distribution of sample weights for EO in sequential tasks of the Biased MNIST dataset.

accuracy or fairness, FSW shows the best fairness results among the baselines with similar accuracies (e.g., *CLAD*, *GSS*, and *OCS*) and thus has the best accuracy-fairness tradeoff. We observe that FSW sometimes improves model accuracy while enhancing the performance of underperforming groups for fairness. The state-of-the-art method, *iCaRL*, generally achieves high accuracy with low EER disparity results. However, since *iCaRL* uses a nearest-mean-of-exemplars approach for its classification model, the predictions are significantly affected by sensitive attribute values, resulting in high disparities for EO. Although *WA* also performs well, the method sometimes increases the model weights for the current task classes, which leads to more forgetting of previous tasks and unstable results. The closest work to FSW is *CLAD*, which disentangles the representations of new classes and a fixed proportion of conflicting old classes to mitigate imbalanced forgetting across classes. However, the proportion of conflicts may vary by task in practice, limiting *CLAD*'s ability to achieve group fairness. The two sample selection methods *GSS* and *OCS* perform worse. While storing diverse and representative samples in the buffer, these methods sometimes result in an imbalance in the number of buffer samples across sensitive groups. The fairness-aware method *FaIRL* leverages an adversarial debiasing framework combined with a rate-distortion function, but the method loses significant accuracy because training the feature encoder and discriminator together is unstable. In comparison, FSW explicitly utilizes approximated loss and fairness measures to adjust the training weights for the current task samples, which leads to much better model accuracy and fairness.

4.2 Sample Weighting Analysis

We next analyze how our FSW algorithm weights the current task samples at each task using the Biased MNIST dataset results shown in Fig. 2. The results for the other datasets are similar and shown in Sec. B.8. As the acquired sample weights may change with epochs during training, we show the average weight distribution of sensitive groups over all epochs. Note that the acquired sample weights are close to 0 or 1 in practice, but they are not strictly binary (0 or 1). Since FSW is not applied to the first task, where the model is trained with only the current task data, we present results starting from the second task. Unlike naïve methods that use all the current task data with

Table 3: Ablation study on the MNIST, FMNIST, Biased MNIST, and DRUG datasets with respect to EER and EO disparity when FSW is used or not.

Methods	MNIST		FMNIST		Biased MNIST		DRUG	
	Acc.	EER Disp.	Acc.	EER Disp.	Acc.	EO Disp.	Acc.	EO Disp.
W/o FSW	.921±.004	.040±.005	**.836±.006**	.048±.005	**.911±.003**	.063±.002	.423±.013	.162±.034
FSW	**.924±.003**	**.032±.004**	.825±.006	**.037±.007**	.909±.003	**.060±.003**	**.429±.020**	**.138±.037**

Table 4: Accuracy and fairness results when combining fair post-processing technique (ϵ-fair) with continual learning methods (*iCaRL*, *CLAD*, and FSW) with respect to DP disparity.

Methods	Biased MNIST		DRUG		BiasBios	
	Acc.	DP Disp.	Acc.	DP Disp.	Acc.	DP Disp.
iCaRL	.818±.011	.012±.001	.458±.014	.098±.020	**.828±.002**	.022±.000
CLAD	.872±.011	.013±.001	.410±.026	.069±.019	.785±.004	.022±.001
FSW	**.889±.006**	.007±.002	.405±.013	.043±.004	.797±.003	.022±.000
iCaRL – ϵ-fair	.805±.014	.007±.002	**.460±.015**	.035±.013	**.828±.001**	**.016±.000**
CLAD – ϵ-fair	.868±.015	.006±.002	.411±.023	.030±.010	.759±.049	.017±.000
FSW – ϵ-fair	.883±.007	**.005±.001**	.403±.010	**.020±.004**	.796±.003	**.016±.000**

equal training weights, FSW usually adjusts different training weights between sensitive groups as shown in Fig. 2. For the Biased MNIST dataset, FSW assigns higher weights on average to the underperforming group (Sensitive group 1 in Fig. 2) compared to the overperforming group (Sensitive group 0 in Fig. 2). We also observe that FSW assigns a weight of zero to a considerable number of samples, indicating that relatively less data is used for training. This weighting approach provides an additional advantage in enabling efficient model training while retaining accuracy and fairness.

4.3 Ablation Study

To show the effectiveness of FSW on accuracy and fairness, we perform an ablation study comparing the performance of using FSW versus using all the current task samples for training with equal weights. Table 3 shows the results for the four datasets, while the results for DP disparity and the BiasBios dataset are similar and shown in Sec. B.9. As a result, applying sample weighting to the current task data is necessary to improve fairness while maintaining comparable accuracy.

4.4 Integrating FSW with a Fair Post-processing Method

In this section, we emphasize the extensibility of FSW by showing how it can be combined with a post-processing method to further improve fairness. We integrate FSW and other existing continual learning methods (*iCaRL*, *CLAD*, and *OCS*) with the state-of-the-art fair post-processing technique in multi-class tasks, ϵ-fair (Denis et al., 2023), as shown in Table 4 and Table 11 in Sec. B.10. Since ϵ-fair only supports DP, we only show DP results using the Biased MNIST, DRUG, and BiasBios datasets. Overall, combining the fair post-processing technique can further improve fairness without degrading accuracy much. In addition, FSW still shows a better accuracy-fairness tradeoff with the combination of the fair post-processing technique, compared to existing continual learning methods.

5 Conclusion

We proposed FSW, a fairness-aware sample weighting algorithm for fair class-incremental learning. Unlike conventional class-incremental learning, we showed how training with all the current task data using equal weights may result in unfair catastrophic forgetting. We theoretically showed that the average gradient vector of the current task data should not solely be in the opposite direction of the average gradient vector of a sensitive group to avoid unfair forgetting. We then proposed FSW as a solution to adjust the average gradient vector of the current task data such that unfairness is mitigated without harming accuracy much. FSW supports various group fairness measures and is efficient as it solves the optimization by converting it into a linear program. In our experiments, FSW outperformed other baselines in terms of fairness while having comparable accuracy across various datasets with different domains.

Ethics Statement We anticipate our research will have a positive societal impact by improving fairness in continual learning. However, improving fairness may result in a decrease in accuracy, although we try to minimize the tradeoff. In addition, choosing the right fairness measure can be challenging depending on the application and social context.

Reproducibility Statement To ensure the reproducibility of our work, we provide detailed explanations of all the theoretical and experimental results throughout the appendix and supplementary material. For the theoretical results, we include complete proofs of all our theorems in the appendix. For the experimental results, we present a thorough description of the datasets used, as well as the experimental settings of model architectures and hyperparameters, in the appendix. In addition, we submit the source code necessary for reproducing our experimental results as a part of the supplementary material.

References

Wickliffe C. Abraham and Anthony Robins. Memory retention – the synaptic stability versus plasticity dilemma. *Trends in Neurosciences*, 28(2):73–78, 2005.

Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna M. Wallach. A reductions approach to fair classification. In *ICML*, volume 80, pp. 60–69, 2018.

Ibrahim M. Alabdulmohsin, Jessica Schrouff, and Sanmi Koyejo. A reduction to binary approach for debiasing multiclass datasets. In *NeurIPS*, 2022.

Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. In *NeurIPS*, pp. 11816–11825, 2019.

Hadis Anahideh, Abolfazl Asudeh, and Saravanan Thirumuruganathan. Fair active learning. *Expert Systems with Applications*, 199:116981, 2022.

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias: There's software used across the country to predict future criminals. and it's biased against blacks, 2016.

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. In *Ethics of data and analytics*, pp. 254–264. 2022.

Mohammad Asghari, Amir M. Fathollahi-Fard, S. M. J. Mirzapour Al-e hashem, and Maxim A. Dulebenets. Transformation and linearization techniques in optimization: A state-of-the-art survey. *Mathematics*, 10(2), 2022.

Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *ICLR*, 2020.

Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *ICML*, volume 119, pp. 528–539, 2020.

Eden Belouadah, Adrian Popescu, and Ioannis Kanellos. A comprehensive study of class incremental learning algorithms for visual tasks. *Neural Networks*, 135:38–54, 2021.

Robert E. Bixby. Solving real-world linear programs: A decade and more of progress. *Oper. Res.*, 50 (1):3–15, 2002.

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. In *NeurIPS*, 2020.

Flávio P. Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R. Varshney. Optimized pre-processing for discrimination prevention. In *NIPS*, pp. 3992–4001, 2017.

Junyi Chai and Xiaoqian Wang. Fairness with adaptive weights. In *International Conference on Machine Learning*, pp. 2853–2866. PMLR, 2022.

Arslan Chaudhry, Puneet Kumar Dokania, Thalaiyasingam Ajanthan, and Philip H. S. Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *ECCV*, volume 11215, pp. 556–572, 2018.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with A-GEM. In *ICLR*, 2019a.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet Kumar Dokania, Philip H. S. Torr, and Marc'Aurelio Ranzato. Continual learning with tiny episodic memories. *CoRR*, abs/1902.10486, 2019b.

Arslan Chaudhry, Albert Gordo, Puneet K. Dokania, Philip H. S. Torr, and David Lopez-Paz. Using hindsight to anchor past knowledge in continual learning. In *AAAI*, pp. 6993–7001, 2021.

Somnath Basu Roy Chowdhury and Snigdha Chaturvedi. Sustaining fairness via incremental learning. In *AAAI*, pp. 6797–6805, 2023.

Nikhil Churamani, Ozgur Kara, and Hatice Gunes. Domain-incremental continual learning for mitigating bias in facial expression and action unit recognition. *IEEE Trans. Affect. Comput.*, 14 (4):3191–3206, 2023.

Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Leveraging labeled and unlabeled data for consistent fair binary classification. In *NeurIPS*, pp. 12739–12750, 2019.

inversion Jeffrey Sorensen Lucas Dixon Lucy Vasserman nithum cjadams, Daniel Borkan. Jigsaw unintended bias in toxicity classification, 2019.

Andrew Cotter, Heinrich Jiang, and Karthik Sridharan. Two-player games for efficient non-convex constrained optimization. In *ALT*, volume 98, pp. 300–332, 2019.

IBM ILOG Cplex. V12. 1: User's manual for cplex. *International Business Machines Corporation*, 46(53):157, 2009.

Maria De-Arteaga, Alexey Romanov, Hanna M. Wallach, Jennifer T. Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Cem Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *FAT*, pp. 120–128, 2019.

Christophe Denis, Romuald Elie, Mohamed Hebiri, and François Hu. Fairness guarantee in multi-class classification, 2023.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pp. 4171–4186, 2019.

Michele Donini, Luca Oneto, Shai Ben-David, John Shawe-Taylor, and Massimiliano Pontil. Empirical risk minimization under fairness constraints. In *NeurIPS*, pp. 2796–2806, 2018.

Sebastian Farquhar and Yarin Gal. Towards robust evaluations of continual learning. *CoRR*, abs/1805.09733, 2018.

E. Fehrman, A. K. Muhammad, E. M. Mirkes, V. Egan, and A. N. Gorban. The five factor model of personality and evaluation of drug consumption risk, 2017.

Michael Feldman, Sorelle A. Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *KDD*, pp. 259–268, 2015.

Robert O.. Ferguson and Lauren F.. Sargent. *Linear Programming: Fundamentals and Applications*. McGraw-Hill, 1958.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *NIPS*, pp. 3315–3323, 2016.

Jiangpeng He. Gradient reweighting: Towards imbalanced class-incremental learning. In *CVPR*, pp. 16668–16677, 2024.

Heinrich Jiang and Ofir Nachum. Identifying and correcting label bias in machine learning. In *AISTATS*, volume 108, pp. 702–712, 2020.

Sangwon Jung, Taeeon Park, Sanghyuk Chun, and Taesup Moon. Re-weighting based group fairness regularization via classwise robust optimization. *arXiv preprint arXiv:2303.00442*, 2023.

Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowl. Inf. Syst.*, 33(1):1–33, 2011.

Angelos Katharopoulos and François Fleuret. Not all samples are created equal: Deep learning with importance sampling. In *ICML*, volume 80, pp. 2530–2539, 2018.

Davinder Kaur, Suleyman Uslu, Kaley J. Rittichier, and Arjan Durresi. Trustworthy artificial intelligence: A review. *ACM Comput. Surv.*, 55(2):39:1–39:38, 2023.

Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *International conference on machine learning*, pp. 2564–2572. PMLR, 2018.

KrishnaTeja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, Abir De, and Rishabh K. Iyer. GRAD-MATCH: gradient matching based data subset selection for efficient deep model training. In *ICML*, volume 139, pp. 5464–5474, 2021a.

KrishnaTeja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, and Rishabh K. Iyer. GLISTER: generalization based data subset selection for efficient and robust learning. In *AAAI*, pp. 8110–8118, 2021b.

Dongwan Kim and Bohyung Han. On the stability-plasticity dilemma of class-incremental learning. In *CVPR*, pp. 20196–20204, 2023.

James Kirkpatrick, Razvan Pascanu, Neil C. Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *CoRR*, abs/1612.00796, 2016.

Ron Kohavi. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *KDD*, pp. 202–207, 1996.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory G. Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(7):3366–3385, 2022.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

Jaehoon Lee, Lechao Xiao, Samuel S. Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. In *NeurIPS*, pp. 8570–8581, 2019.

Haochen Liu, Yiqi Wang, Wenqi Fan, Xiaorui Liu, Yaxin Li, Shaili Jain, Anil K. Jain, and Jiliang Tang. Trustworthy AI: A computational perspective. *CoRR*, abs/2107.06641, 2021.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, pp. 3730–3738, 2015.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *NIPS*, pp. 6467–6476, 2017.

Songtao Lu, Ioannis C. Tsaknakis, Mingyi Hong, and Yongxin Chen. Hybrid block successive approximation for one-sided non-convex min-max problems: Algorithms and applications. *IEEE Trans. Signal Process.*, 68:3676–3691, 2020.

Zheda Mai, Ruiwen Li, Jihwan Jeong, David Quispe, Hyunwoo Kim, and Scott Sanner. Online continual learning in image classification: An empirical survey. *Neurocomputing*, 469:28–51, 2022.

Marc Masana, Xialei Liu, Bartlomiej Twardowski, Mikel Menta, Andrew D. Bagdanov, and Joost van de Weijer. Class-incremental learning: Survey and performance evaluation on image classification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(5):5513–5533, 2023.

Bruce A McCarl and Thomas H Spreen. Applied mathematical programming using algebraic systems. 1997. *Internet site: http://agrinet. tamu. edu/faculty/mccarl/regbook. htm (Accessed January 2000)*, 2021.

Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. volume 24 of *Psychology of Learning and Motivation*, pp. 109–165. 1989.

Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)*, 54(6):1–35, 2021.

Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Comput. Surv.*, 54(6):115:1–115:35, 2022.

Martial Mermillod, Aurélia Bugaiska, and Patrick BONIN. The stability-plasticity dilemma: investigating the continuum from catastrophic forgetting to age-limited learning effects. *Frontiers in Psychology*, 4, 2013.

Seyed-Iman Mirzadeh, Mehrdad Farajtabar, Razvan Pascanu, and Hassan Ghasemzadeh. Understanding the role of training regimes in continual learning. In *NeurIPS*, 2020.

Baharan Mirzasoleiman, Jeff A. Bilmes, and Jure Leskovec. Coresets for data-efficient training of machine learning models. In *ICML*, volume 119, pp. 6950–6960, 2020.

Jinlong Pang, Jialu Wang, Zhaowei Zhu, Yuanshun Yao, Chen Qian, and Yang Liu. Fairness without harm: An influence-guided active sampling approach. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL `https://openreview.net/forum?id=YYJojVBCcd`.

German Ignacio Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019.

Geoff Pleiss, Manish Raghavan, Felix Wu, Jon M. Kleinberg, and Kilian Q. Weinberger. On fairness and calibration. In *NIPS*, pp. 5680–5689, 2017.

Preston Putzel and Scott Lee. Blackbox post-processing for multiclass fairness. *arXiv preprint arXiv:2201.04461*, 2022.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. icarl: Incremental classifier and representation learning. In *CVPR*, pp. 5533–5542, 2017.

Yuji Roh, Kangwook Lee, Steven Whang, and Changho Suh. Fr-train: A mutual information-based approach to fair and robust training. In *ICML*, volume 119, pp. 8147–8157, 2020.

Yuji Roh, Kangwook Lee, Steven Euijong Whang, and Changho Suh. Fairbatch: Batch selection for model fairness. In *ICLR*, 2021.

Yuji Roh, Weili Nie, De-An Huang, Steven Euijong Whang, Arash Vahdat, and Anima Anandkumar. Dr-fairness: Dynamic data ratio adjustment for fair training on real and generated data. *Trans. Mach. Learn. Res.*, 2023.

Aili Shen, Xudong Han, Trevor Cohn, Timothy Baldwin, and Lea Frermann. Optimising equal opportunity fairness in model training. In *NAACL*, pp. 4073–4084, 2022.

Ki Hyun Tae, Hantian Zhang, Jaeyoung Park, Kexin Rong, and Steven Euijong Whang. Falcon: Fair active learning using multi-armed bandits. *Proceedings of the VLDB Endowment*, 17(5):952–965, 2024.

Thanh-Dat Truong, Hoang-Quan Nguyen, Bhiksha Raj, and Khoa Luu. Fairness continual learning approach to semantic scene understanding in open-world environments. In *NeurIPS*, 2023.

Gido M. van de Ven and Andreas S. Tolias. Three scenarios for continual learning. *CoRR*, abs/1904.07734, 2019.

Suresh Venkatasubramanian. Algorithmic fairness: Measures, methods and representations. In *PODS*, pp. 481, 2019.

Fu-Yun Wang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. FOSTER: feature boosting and compression for class-incremental learning. In *ECCV*, volume 13685, pp. 398–414, 2022.

Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *CVPR*, pp. 374–382, 2019.

Ruicheng Xian, Lang Yin, and Han Zhao. Fair and optimal classification via post-processing. In *ICML*, volume 202, pp. 37977–38012, 2023.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017.

Shixiong Xu, Gaofeng Meng, Xing Nie, Bolin Ni, Bin Fan, and Shiming Xiang. Defying imbalanced forgetting in class incremental learning. In *AAAI*, volume 38, pp. 16211–16219, 2024.

Tian Xu, Jennifer White, Sinan Kalkan, and Hatice Gunes. Investigating bias and fairness in facial expression recognition. In *ECCV*, volume 12540, pp. 506–523, 2020.

Shipeng Yan, Jiangwei Xie, and Xuming He. DER: dynamically expandable representation for class incremental learning. In *CVPR*, pp. 3014–3023, 2021.

Jaehong Yoon, Divyam Madaan, Eunho Yang, and Sung Ju Hwang. Online coreset selection for rehearsal-based continual learning. In *ICLR*, 2022.

Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *AIES*, pp. 335–340, 2018.

Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. Maintaining discrimination and fairness in class incremental learning. In *CVPR*, pp. 13205–13214, 2020.

Da-Wei Zhou, Qi-Wei Wang, Zhi-Hong Qi, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Deep class-incremental learning: A survey. *CoRR*, abs/2302.03648, 2023a.

Da-Wei Zhou, Qi-Wei Wang, Han-Jia Ye, and De-Chuan Zhan. A model or 603 exemplars: Towards memory-efficient class-incremental learning. In *ICLR*, 2023b.

A APPENDIX – THEORY

A.1 THEORETICAL ANALYSIS OF UNFAIRNESS IN CLASS-INCREMENTAL LEARNING

Continuing from Sec. 3.1, we prove the lemma on the updated loss of a group of data after learning the current task data.

Lemma. *Denote G as a sensitive group of data composed of features X and true labels y. Also, denote f_θ^{l-1} as a previous model and f_θ as the updated model after training on the current task T_l. Let ℓ be any differentiable standard loss function (e.g., cross-entropy loss), and η be a learning rate. Then, the loss of the sensitive group of data after training with a current task sample $d_i \in T_l$ is approximated as follows:*

$$\tilde{\ell}(f_\theta, G) = \ell(f_\theta^{l-1}, G) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i),$$

where $\tilde{\ell}(f_\theta, G)$ is the approximated average loss between model predictions $f_\theta(X)$ and true labels y, whereas $\ell(f_\theta^{l-1}, G)$ is the exact average loss, $\nabla_\theta \ell(f_\theta^{l-1}, G)$ is the average gradient vector for the samples in the group G, and $\nabla_\theta \ell(f_\theta^{l-1}, d_i)$ is the gradient vector for a sample d_i, each with respect to the previous model f_θ^{l-1}.

Proof. We update the model using gradient descent with the current task sample $d_i \in T_l$ and learning rate η as follows:

$$\theta = \theta^{l-1} - \eta \nabla_\theta \ell(f_\theta^{l-1}, d_i).$$

Using the Taylor series approximation,

$$\begin{aligned}
\tilde{\ell}(f_\theta, G) &= \ell(f_\theta^{l-1}, G) + \nabla_\theta \ell(f_\theta^{l-1}, G)^\top (\theta - \theta^{l-1}) \\
&= \ell(f_\theta^{l-1}, G) + \nabla_\theta \ell(f_\theta^{l-1}, G)^\top (-\eta \nabla_\theta \ell(f_\theta^{l-1}, d_i)) \\
&= \ell(f_\theta^{l-1}, G) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i).
\end{aligned}$$

If we update the model using all the current task data T_l, the equation is formulated as $\tilde{\ell}(f_\theta, G) = \ell(f_\theta^{l-1}, G) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G)^\top \nabla_\theta \ell(f_\theta^{l-1}, T_l)$. Therefore, if the average gradient vectors of the sensitive group and the current task data have opposite directions, i.e., $\nabla_\theta \ell(f_\theta^{l-1}, G)^\top \nabla_\theta \ell(f_\theta^{l-1}, T_l) < 0$, learning the current task data increases the loss of the sensitive group data and finally leads to catastrophic forgetting. □

We next derive a sufficient condition for unfair forgetting.

Theorem. *Let ℓ be the cross-entropy loss and we denote G_1 and G_2 as the overperforming and underperforming groups of data, and d_i as a training sample that satisfy the following conditions: $\ell(f_\theta^{l-1}, G_1) < \ell(f_\theta^{l-1}, G_2)$ while $\nabla_\theta \ell(f_\theta^{l-1}, G_1)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i) > 0$ and $\nabla_\theta \ell(f_\theta^{l-1}, G_2)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i) < 0$. Then $|\tilde{\ell}(f_\theta, G_1) - \tilde{\ell}(f_\theta, G_2)| > |\ell(f_\theta^{l-1}, G_1) - \ell(f_\theta^{l-1}, G_2)|$.*

Proof. Using the derived equation in the lemma above $\tilde{\ell}(f_\theta, G) = \ell(f_\theta^{l-1}, G) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i)$, we compute the disparity of losses between the two groups G_1 and G_2 after the model update as follows:

$$\begin{aligned}
|\tilde{\ell}(f_\theta, G_1) - \tilde{\ell}(f_\theta, G_2)| &= |(\ell(f_\theta^{l-1}, G_1) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G_1)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i)) - \\
&\qquad (\ell(f_\theta^{l-1}, G_2) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G_2)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i))| \\
&= |(\ell(f_\theta^{l-1}, G_1) - \ell(f_\theta^{l-1}, G_2)) - \\
&\qquad \eta(\nabla_\theta \ell(f_\theta^{l-1}, G_1)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i) - \nabla_\theta \ell(f_\theta^{l-1}, G_2)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i))|.
\end{aligned}$$

Since $\ell(f_\theta^{l-1}, G_1) < \ell(f_\theta^{l-1}, G_2)$, it leads to $\ell(f_\theta^{l-1}, G_1) - \ell(f_\theta^{l-1}, G_2) < 0$. Next, the two assumptions of $\nabla_\theta \ell(f_\theta^{l-1}, G_1)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i) > 0$ and $\nabla_\theta \ell(f_\theta^{l-1}, G_2)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i) < 0$ make $-\eta(\nabla_\theta \ell(f_\theta^{l-1}, G_1)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i) - \nabla_\theta \ell(f_\theta^{l-1}, G_2)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i)) < 0$. Since the two terms in the absolute value equation are both negative,

$$
\begin{aligned}
|\tilde{\ell}(f_\theta, G_1) - \tilde{\ell}(f_\theta, G_2)| = |\ell(f_\theta^{l-1}, G_1) - \ell(f_\theta^{l-1}, G_2)| + \\
|-\eta(\nabla_\theta \ell(f_\theta^{l-1}, G_1)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i) - \nabla_\theta \ell(f_\theta^{l-1}, G_2)^\top \nabla_\theta \ell(f_\theta^{l-1}, d_i))| \\
> |\ell(f_\theta^{l-1}, G_1) - \ell(f_\theta^{l-1}, G_2)|.
\end{aligned}
$$

We finally have $|\tilde{\ell}(f_\theta, G_1) - \tilde{\ell}(f_\theta, G_2)| > |\ell(f_\theta^{l-1}, G_1) - \ell(f_\theta^{l-1}, G_2)|$, which implies that fairness deteriorates after training on the current task data. $\square$

A.2 From Cross-Entropy Loss to Group Fairness Metrics

Continuing from Sec. 3.1, we explain how we can approximate the group fairness metrics using cross-entropy loss. Existing works (Shen et al., 2022; Roh et al., 2021; 2023) empirically verified that using other functions like cross-entropy loss can provide reasonable proxies for common group fairness metrics such as equalized odds (EO) and demographic parity (DP). In addition, we theoretically describe how minimizing the cost function for EO using cross-entropy loss (i.e., $L_{EO} = \frac{1}{|\mathbb{Y}||\mathbb{Z}|}\sum_{y\in\mathbb{Y},z\in\mathbb{Z}} |\ell(f_\theta, G_{y,z}) - \ell(f_\theta, G_y)|$ where ℓ is a cross-entropy loss) leads to ensuring EO. Shen et al. (2022) theoretically and empirically showed that using cross-entropy loss instead of the 0-1 loss (i.e., $\mathbf{1}(\mathrm{y} \neq \hat{\mathrm{y}})$ where $\mathbf{1}(\cdot)$ is an indicator function, which is equivalent to the probability of correct prediction) can still capture EO in binary classification. We now prove how applying the cross-entropy loss for EO can be extended to multi-class classification as follows:

Let $m_{y,z}$ be the size of a sensitive group (i.e., $m_{y,z} := |\{i : \mathrm{y}_i = y, \mathrm{z}_i = z\}|$) and $\mathbb{Y}$ be a set of all classes. Let $\begin{pmatrix} \vdots \\ \mathrm{y}_i^j \\ \vdots \end{pmatrix}$ be the one-hot encoding vector of y_i. Similarly, $\hat{\mathrm{y}}_i$ is a predicted label and $\begin{pmatrix} \vdots \\ \hat{\mathrm{y}}_i^j \\ \vdots \end{pmatrix}$ denotes a probability distribution for each label of the sample i. Then, the cross-entropy loss for a sensitive group $G_{y,z}$ can be transformed as follows:

$$
\begin{aligned}
\ell(f_\theta, G_{y,z}) &= -\frac{1}{m_{y,z}} \sum_{i=1}^{m_{y,z}} \left(\sum_{j=1}^{|\mathbb{Y}|} \mathrm{y}_i^j \cdot \log(\hat{\mathrm{y}}_i^j) \right) \\
&= -\frac{1}{m_{y,z}} \sum_{i=1}^{m_{y,z}} \log(\hat{\mathrm{y}}_i^y).
\end{aligned}
$$

Since $\hat{\mathrm{y}}_i^y$ is equivalent to $p(\hat{\mathrm{y}}_i = y)$ and we are measuring a loss for the sensitive group ($\mathrm{y} = y$, $\mathrm{z} = z$), $\ell(f_\theta, G_{y,z}) = -\frac{1}{m_{y,z}} \sum_i \log(p(\hat{\mathrm{y}}_i))$ is an unbiased estimator of $-\log p(\hat{\mathrm{y}}|\mathrm{y} = y, \mathrm{z} = z)$. Likewise, $\ell(f_\theta, G_y)$ is an unbiased estimator of $-\log p(\hat{\mathrm{y}}|\mathrm{y} = y)$ and our cost function becomes equivalent to $\left|\log \frac{p(\hat{\mathrm{y}}|\mathrm{y}=y)}{p(\hat{\mathrm{y}}|\mathrm{y}=y,\mathrm{z}=z)}\right|$. Since $\frac{p(\hat{\mathrm{y}}|\mathrm{y}=y)}{p(\hat{\mathrm{y}}|\mathrm{y}=y,\mathrm{z}=z)} = 1$ for all y, z implies $\hat{Y} \perp\!\!\!\perp Z \,|\, Y$, we conclude that minimizing the cost function for EO can satisfy the equalized odds.

We next perform experiments to evaluate how well the cost function for EO approximates EO disparity (i.e., $\frac{1}{|\mathbb{Y}||\mathbb{Z}|}\sum_{y\in\mathbb{Y},z\in\mathbb{Z}} |\Pr(\hat{\mathrm{y}} = y|\mathrm{y} = y, \mathrm{z} = z) - \Pr(\hat{\mathrm{y}} = y|\mathrm{y} = y)|$) on the Biased MNIST dataset as shown in Fig. 3. Although the scales of the two metrics are different, the simultaneous movement of these two trends suggests that our cost function is effective in promoting equalized odds satisfaction.

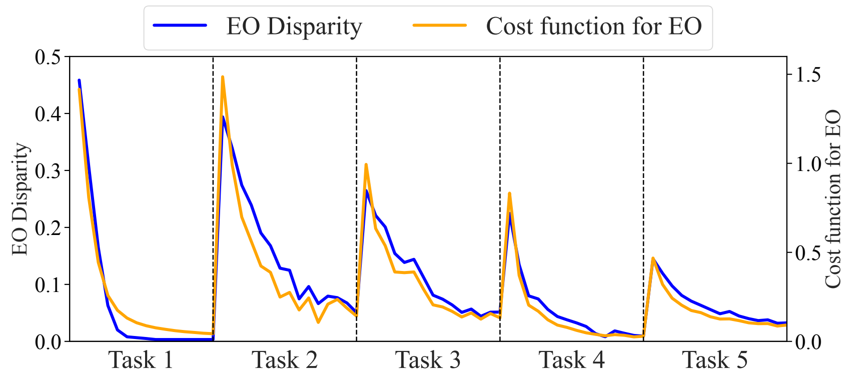


Figure 3: Comparison of EO disparity and cost function for EO during training on the Biased MNIST dataset. We train a model for 15 epochs at each task.

A.3 Derivation of a Sufficient Condition for Demographic Parity in the Multi-Class Setting

Continuing from Sec. 3.2, we derive a sufficient condition for satisfying demographic parity in the multi-class setting.

Proposition. *In the multi-class setting,* $\frac{m_{y,z_1}}{m_{*,z_1}}\ell(f_\theta, G_{y,z_1}) = \frac{m_{y,z_2}}{m_{*,z_2}}\ell(f_\theta, G_{y,z_2})$ *where* $m_{y,z} := |\{i : y_i = y, z_i = z\}|$ *and* $m_{*,z} := |\{i : z_i = z\}|$ *for* $y \in \mathbb{Y}$ *and* $z_1, z_2 \in \mathbb{Z}$ *can serve as a sufficient condition for demographic parity.*

Proof. In the multi-class setting, we can extend the definition of demographic parity as $\Pr(\hat{\mathrm{y}} = y|\mathrm{z} = z_1) = \Pr(\hat{\mathrm{y}} = y|\mathrm{z} = z_2)$ for $y \in \mathbb{Y}$ and $z_1, z_2 \in \mathbb{Z}$. The term $\Pr(\hat{\mathrm{y}} = y|\mathrm{z} = z)$ can be decomposed as follows: $\Pr(\hat{\mathrm{y}} = y|\mathrm{z} = z) = \Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y|\mathrm{z} = z) + \sum_{y_n \neq y} \Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y_n|\mathrm{z} = z)$. Without loss of generality, we set $z_1 = 0$ and $z_2 = 1$. Then the definition of demographic parity in the multi-class setting now becomes

$$\Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y|\mathrm{z} = 0) + \sum_{y_n \neq y} \Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y_n|\mathrm{z} = 0)$$
$$= \Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y|\mathrm{z} = 1) + \sum_{y_n \neq y} \Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y_n|\mathrm{z} = 1).$$

The term $\Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y|\mathrm{z} = 0)$ can be represented with the 0-1 loss as follows:

$$\begin{aligned}\Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y|\mathrm{z} = 0) &= \frac{\Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y, \mathrm{z} = 0)}{\Pr(\mathrm{z} = 0)} \\ &= \frac{\Pr(\hat{\mathrm{y}} = y|\mathrm{y} = y, \mathrm{z} = 0)\Pr(\mathrm{y} = y, \mathrm{z} = 0)}{\Pr(\mathrm{z} = 0)} \\ &= \frac{1}{m_{*,0}} \sum_{i: y_i = y, z_i = 0} (1 - \mathbb{1}(y_i \neq \hat{y}_i))\end{aligned}$$

Similarly, $\Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y_n|\mathrm{z} = 0)$ for $y_n \neq y$ can be transformed as follows:

$$\begin{aligned}\Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y_n|\mathrm{z} = 0) &= \frac{\Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y_n, \mathrm{z} = 0)}{\Pr(\mathrm{z} = 0)} \\ &= \frac{\Pr(\hat{\mathrm{y}} = y|\mathrm{y} = y_n, \mathrm{z} = 0)\Pr(\mathrm{y} = y_n, \mathrm{z} = 0)}{\Pr(\mathrm{z} = 0)} \\ &= \frac{1}{m_{*,0}} \sum_{j: y_j = y_n, z_j = 0} \mathbb{1}(y_j \neq \hat{y}_j)\end{aligned}$$

By applying the same technique to $\Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y | \mathrm{z} = 1)$ and $\Pr(\hat{\mathrm{y}} = y, \mathrm{y} = y_n | \mathrm{z} = 1)$, we have the 0-1 loss-based definition of demographic parity:

$$\frac{1}{m_{*,0}} \sum_{i:y_i=y,z_i=0} (1 - \mathbb{1}(y_i \neq \hat{y}_i)) + \sum_{i:y_i\neq y} \frac{1}{m_{*,0}} \sum_{j:y_j=y_i,z_j=0} \mathbb{1}(y_j \neq \hat{y}_j)$$
$$= \frac{1}{m_{*,1}} \sum_{i:y_i=y,z_i=1} (1 - \mathbb{1}(y_i \neq \hat{y}_i)) + \sum_{i:y_i\neq y} \frac{1}{m_{*,1}} \sum_{j:y_j=y_i,z_j=1} \mathbb{1}(y_j \neq \hat{y}_j).$$

Since the 0-1 loss is not differentiable, it is not suitable to approximate the updated loss using gradients as in Eq. 1. We thus approximate the 0-1 loss to a standard loss function ℓ (e.g., cross-entropy loss),

$$\frac{1}{m_{*,0}} \sum_{i:y_i=y,z_i=0} -\ell(f_\theta, d_i) + \sum_{i:y_i\neq y} \frac{1}{m_{*,0}} \sum_{j:y_j=y_i,z_j=0} \ell(f_\theta, d_j)$$
$$= \frac{1}{m_{*,1}} \sum_{i:y_i=y,z_i=1} -\ell(f_\theta, d_i) + \sum_{i:y_i\neq y} \frac{1}{m_{*,1}} \sum_{j:y_j=y_i,z_j=1} \ell(f_\theta, d_j),$$

where $\ell(f_\theta, d_j)$ is the loss between the model prediction $f_\theta(d_j)$ and the true label y_j. By replacing $\sum_{i:y_i=y,z_i=z} \ell(f_\theta, d_i) = m_{y,z}\ell(f_\theta, G_{y,z})$,

$$\frac{m_{y,0}}{m_{*,0}}(-\ell(f_\theta, G_{y,0})) + \sum_{i:y_i\neq y} \frac{m_{y_i,0}}{m_{*,0}} \ell(f_\theta, G_{y_i,0}) = \frac{m_{y,1}}{m_{*,1}}(-\ell(f_\theta, G_{y,1})) + \sum_{i:y_i\neq y} \frac{m_{y_i,1}}{m_{*,1}} \ell(f_\theta, G_{y_i,1}).$$

To satisfy the constraint for all $y \in \mathbb{Y}$, the corresponding terms on the left-hand side and the right-hand side of the equation should be equal, i.e., $\frac{m_{y,0}}{m_{*,0}}\ell(f_\theta, G_{y,0}) = \frac{m_{y,1}}{m_{*,1}}\ell(f_\theta, G_{y,1})$. In general, we derive a sufficient condition for demographic parity as $\frac{m_{y,z_1}}{m_{*,z_1}}\ell(f_\theta, G_{y,z_1}) = \frac{m_{y,z_2}}{m_{*,z_2}}\ell(f_\theta, G_{y,z_2})$. □

A.4 LP Formulation of our Fairness-aware Optimization Problems

Continuing from Sec. 3.2, we prove that minimizing the sum of absolute values with linear terms can be transformed into a linear programming form.

Lemma. *The following optimization problem can be reformulated into a linear programming form. Note that in the following equation, y and z refer to arbitrary variables, not to the label or sensitive attribute, respectively.*

$$\min_{\mathbf{x}} \sum_{i=1}^{n} |y_i| + z_i$$
$$\begin{aligned} \textit{s.t.} \quad & y_i = a_i - \mathbf{b}_i^\top \mathbf{x}, \quad z_i = c_i - \mathbf{d}_i^\top \mathbf{x} \\ & a_i, c_i, y_i, z_i \in \mathbb{R}, \quad \mathbf{b}_i, \mathbf{d}_i \in \mathbb{R}^{m\times 1} \quad \forall i \in \{1, \ldots, n\} \\ & \mathbf{x} \in [0,1]^{m\times 1}. \end{aligned}$$

Proof. The transformation for minimizing the sum of absolute values was introduced in Ferguson & Sargent (1958); McCarl & Spreen (2021); Asghari et al. (2022). Note that considering the additional affine term does not affect the flow of the proof. We first substitute y_i for $y_i^+ - y_i^-$ where both y_i^+ and y_i^- are nonnegative. Then, the optimization problem becomes

$$\min_{\mathbf{x}} \sum_{i=1}^{n} |y_i^+ - y_i^-| + z_i$$
$$\begin{aligned} \text{s.t.} \quad & y_i^+ - y_i^- = a_i - \mathbf{b}_i^\top \mathbf{x}, \quad z_i = c_i - \mathbf{d}_i^\top \mathbf{x}, \quad y_i^+ - y_i^- = y_i \\ & y_i^+, y_i^- \in \mathbb{R}^+, \quad a_i, c_i, y_i, z_i \in \mathbb{R}, \quad \mathbf{b}_i, \mathbf{d}_i \in \mathbb{R}^{m\times 1} \quad \forall i \in \{1, \ldots, n\} \\ & \mathbf{x} \in [0,1]^{m\times 1}. \end{aligned}$$

This problem is still nonlinear. However, the absolute value terms can be simplified when either y_i^+ or y_i^- equals to zero (i.e., $y_i^+ y_i^- = 0$), as the consequent absolute value reduces to zero plus the other term. Then, the absolute value term can be written as the sum of two variables,

$$|y_i^+ - y_i^-| = |y_i^+| + |y_i^-| = y_i^+ + y_i^- \quad \text{if} \quad y_i^+ y_i^- = 0.$$

By using the assumption, the formulation becomes

$$\min_{\mathbf{x}} \sum_{i=1}^{n} y_i^+ + y_i^- + z_i$$
$$\text{s.t.} \quad y_i^+ - y_i^- = a_i - \mathbf{b}_i^\top \mathbf{x}, \quad z_i = c_i - \mathbf{d}_i^\top \mathbf{x}, \quad y_i^+ - y_i^- = y_i, \quad \underline{y_i^+ y_i^- = 0}$$
$$y_i^+, y_i^- \in \mathbb{R}^+, \quad a_i, c_i, y_i, z_i \in \mathbb{R}, \quad \mathbf{b}_i, \mathbf{d}_i \in \mathbb{R}^{m\times 1} \quad \forall i \in \{1, \ldots, n\}$$
$$\mathbf{x} \in [0,1]^{m\times 1}.$$

with the underlined condition added. However, this condition can be dropped. Assume there exist y_i^+ and y_i^-, which do not satisfy $y_i^+ y_i^- = 0$. When $y_i^+ \geq y_i^- > 0$, there exists a better solution ($y_i^+ - y_i^-$, 0) instead of (y_i^+, y_i^-), which satisfies all the conditions, but has a smaller objective function value $y_i^+ - y_i^- + 0 + z_i < y_i^+ + y_i^- + z_i$. For the case of $y_i^- > y_i^+ > 0$, a solution (0, $y_i^- - y_i^+$) works as the same manner. Thus, the minimization automatically leads to $y_i^+ y_i^- = 0$, and the underlined nonlinear constraint becomes unnecessary. Consequently, the final formulation becomes the linear problem as follows:

$$\min_{\mathbf{x}} \sum_{i=1}^{n} y_i^+ + y_i^- + z_i$$
$$\text{s.t.} \quad y_i^+ - y_i^- = a_i - \mathbf{b}_i^\top \mathbf{x}, \quad z_i = c_i - \mathbf{d}_i^\top \mathbf{x}, \quad y_i^+ - y_i^- = y_i$$
$$y_i^+, y_i^- \in \mathbb{R}^+, \quad a_i, c_i, y_i, z_i \in \mathbb{R}, \quad \mathbf{b}_i, \mathbf{d}_i \in \mathbb{R}^{m\times 1} \quad \forall i \in \{1, \ldots, n\}$$
$$\mathbf{x} \in [0,1]^{m\times 1}.$$

□

By applying this lemma, we next prove the transformation of the defined fairness-aware optimization problems in Eq. 3, 4, and 5 to the form of linear programming.

Theorem. *The fairness-aware optimization problems (Eq. 3, 4, and 5) can be transformed into the form of linear programming (LP) problems.*

Proof. For every update of the model, the corresponding loss of each group can be approximated linearly in the same way as in Sec. A.1: $\tilde{\ell}(f_\theta, G) = \ell(f_\theta^{l-1}, G) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G)^\top \nabla_\theta \ell(f_\theta^{l-1}, T_l)$. With a technique of sample weighting for the current task data, $\nabla_\theta \ell(f_\theta^{l-1}, T_l)$ can be changed as $\frac{1}{|T_l|}\sum_{d_i \in T_l} \mathbf{w}_l^i \nabla_\theta \ell(f_\theta^{l-1}, d_i)$ where $\mathbf{w}_l^i$ represents a training weight for the current task sample d_i. Thus, $\tilde{\ell}(f_\theta, G)$ can be rewritten as follows:

$$\begin{aligned}
\tilde{\ell}(f_\theta, G) &= \ell(f_\theta^{l-1}, G) - \eta \nabla_\theta \ell(f_\theta^{l-1}, G)^\top \left(\frac{1}{|T_l|} \sum_{d_i \in T_l} \mathbf{w}_l^i \nabla_\theta \ell(f_\theta^{l-1}, d_i) \right) \\
&= \ell(f_\theta^{l-1}, G) - \frac{\eta}{|T_l|} \nabla_\theta \ell(f_\theta^{l-1}, G)^\top \begin{bmatrix} \cdots & \nabla_\theta \ell(f_\theta^{l-1}, d_i) & \cdots \end{bmatrix} \begin{bmatrix} \vdots \\ \mathbf{w}_l^i \\ \vdots \end{bmatrix} \\
&= a_G - \mathbf{b}_G^\top \mathbf{w},
\end{aligned}$$

where $a_G := \ell(f_\theta^{l-1}, G)$ and $\mathbf{b}_G := \frac{\eta}{|T_l|} \begin{bmatrix} \cdots & \nabla_\theta \ell(f_\theta^{l-1}, d_i) & \cdots \end{bmatrix}^\top \nabla_\theta \ell(f_\theta^{l-1}, G)$ are a constant and a vector with constants, respectively, and $\mathbf{w} := \begin{bmatrix} \vdots \\ w_l^i \\ \vdots \end{bmatrix}$ is a variable where $w_l^i \in [0, 1]$.

Case 1. *If target fairness measure is EER ($L_{fair} = L_{EER}$),*

$$
\begin{aligned}
L_{EER} + \lambda L_{acc} &= \frac{1}{|\mathbb{Y}|}\sum_{y\in\mathbb{Y}} |\tilde{\ell}(f_\theta, G_y) - \tilde{\ell}(f_\theta, G_{\mathbb{Y}})| + \lambda \frac{1}{|\mathbb{Y}_c|}\sum_{y\in\mathbb{Y}_c} \tilde{\ell}(f_\theta, G_y) \\
&= \frac{1}{|\mathbb{Y}|}\sum_{y\in\mathbb{Y}} |(a_{G_y} - a_{G_{\mathbb{Y}}}) - (\mathbf{b}_{G_y} - \mathbf{b}_{G_{\mathbb{Y}}})^\top \mathbf{w}| + \lambda \frac{1}{|\mathbb{Y}_c|}\sum_{y\in\mathbb{Y}_c} (a_{G_y} - \mathbf{b}_{G_y}^\top \mathbf{w}).
\end{aligned}
$$

Case 2. *If target fairness measure is EO ($L_{fair} = L_{EO}$),*

$$
\begin{aligned}
L_{EO} + \lambda L_{acc} &= \frac{1}{|\mathbb{Y}||\mathbb{Z}|}\sum_{y\in\mathbb{Y}, z\in\mathbb{Z}} |\tilde{\ell}(f_\theta, G_{y,z}) - \tilde{\ell}(f_\theta, G_y)| + \lambda \frac{1}{|\mathbb{Y}_c||\mathbb{Z}|}\sum_{y\in\mathbb{Y}_c, z\in\mathbb{Z}} \tilde{\ell}(f_\theta, G_{y,z}) \\
&= \frac{1}{|\mathbb{Y}||\mathbb{Z}|}\sum_{y\in\mathbb{Y}, z\in\mathbb{Z}} |(a_{G_{y,z}} - a_{G_y}) - (\mathbf{b}_{G_{y,z}} - \mathbf{b}_{G_y})^\top \mathbf{w}| + \\
&\quad \lambda \frac{1}{|\mathbb{Y}_c||\mathbb{Z}|}\sum_{y\in\mathbb{Y}_c, z\in\mathbb{Z}} (a_{G_{y,z}} - \mathbf{b}_{G_{y,z}}^\top \mathbf{w}).
\end{aligned}
$$

Case 3. *If target fairness measure is DP ($L_{fair} = L_{DP}$),*

$$
\begin{aligned}
L_{DP} + \lambda L_{acc} &= \frac{1}{|\mathbb{Y}||\mathbb{Z}|}\sum_{y\in\mathbb{Y}, z\in\mathbb{Z}} |\tilde{\ell}'(f_\theta, G_{y,z}) - \tilde{\ell}'(f_\theta, G_y)| + \lambda \frac{1}{|\mathbb{Y}_c||\mathbb{Z}|}\sum_{y\in\mathbb{Y}_c, z\in\mathbb{Z}} \tilde{\ell}(f_\theta, G_{y,z}) \\
&= \frac{1}{|\mathbb{Y}||\mathbb{Z}|}\sum_{y\in\mathbb{Y}, z\in\mathbb{Z}} |(a'_{G_{y,z}} - a'_{G_y}) - (\mathbf{b}'_{G_{y,z}} - \mathbf{b}'_{G_y})^\top \mathbf{w}| + \\
&\quad \lambda \frac{1}{|\mathbb{Y}_c||\mathbb{Z}|}\sum_{y\in\mathbb{Y}_c, z\in\mathbb{Z}} (a_{G_{y,z}} - \mathbf{b}_{G_{y,z}}^\top \mathbf{w}),
\end{aligned}
$$

where $a'_{G_{y,z}} := \frac{m_{y,z}}{m_{*,z}} a_{G_{y,z}}$, $a'_{G_y} := \sum_{z\in\mathbb{Z}} \frac{m_{y,z}}{m_{*,z}} a_{G_{y,z}}$, $\mathbf{b}'_{G_{y,z}} := \frac{m_{y,z}}{m_{*,z}} \mathbf{b}_{G_{y,z}}$, $\mathbf{b}'_{G_y} := \sum_{z\in\mathbb{Z}} \frac{m_{y,z}}{m_{*,z}} \mathbf{b}_{G_{y,z}}$.

Since a_G and $\mathbf{b}_G$ are composed of constant values, each equation above can be reformulated to a linear programming form by applying the above lemma. ☐

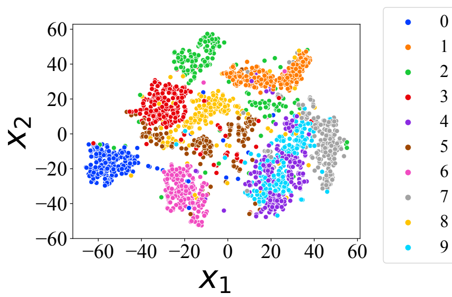
B Appendix – Experiments

B.1 T-SNE Results for Real Datasets

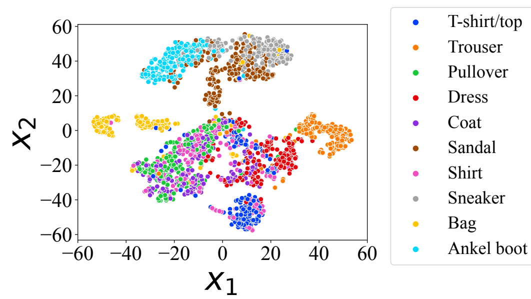
Continuing from Sec. 1, we provide t-SNE results for real datasets to show that data overlapping between different classes also occurs in real scenarios, similar to the synthetic dataset results depicted in Fig. 1a. Using t-SNE, we project the high-dimensional data of the MNIST, FMNIST, Biased MNIST, and DRUG datasets into a lower-dimensional 2D space with x_1 and x_2, as shown in Fig. 4. Since BiasBios is a text dataset that requires pre-trained embeddings to represent the data, we do not include the t-SNE results for it. In the MNIST dataset, the images with labels of 3 (red), 5 (brown), and 8 (yellow) exhibit similar characteristics and overlap, but belong to different classes. As another example, in the FMNIST dataset, the images of the classes 'Sandal' (brown), 'Sneaker' (gray), and 'Ankel boot' (sky-blue) also have similar characteristics and overlap.

B.2 Approximation Error of Taylor Series

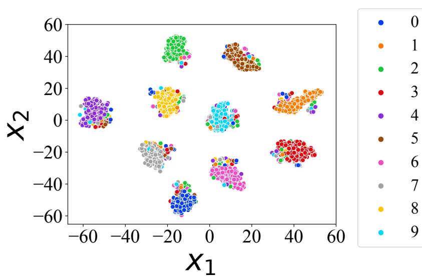
Continuing from Sec. 3.1, we provide empirical approximation errors between true losses and approximated losses derived from first-order Taylor series on the MNIST and Biased MNIST datasets as shown in Fig. 5. For each task, we train the model for 5 epochs and 15 epochs on the MNIST and Biased MNIST datasets, respectively. The approximation error is large when a new task begins because new samples with unseen classes are introduced. However, the error gradually decreases as the number of epochs increases while training a model for the task.



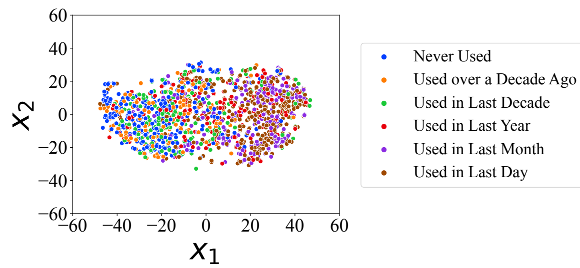
(a) MNIST.



(b) FMNIST.

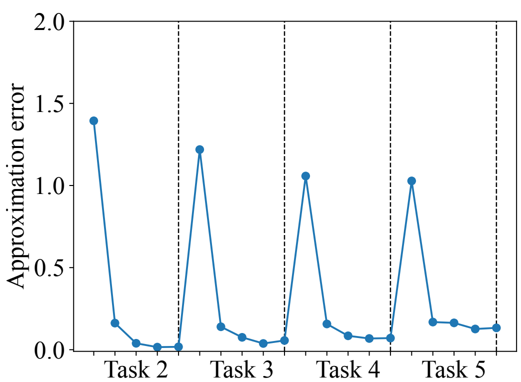


(c) Biased MNIST.

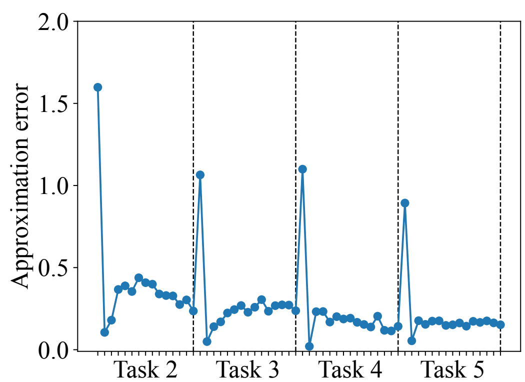


(d) DRUG.

Figure 4: t-SNE results for the MNIST, FMNIST, Biased MNIST, and DRUG datasets.



(a) MNIST.



(b) Biased MNIST.

Figure 5: Absolute errors between true losses and approximated losses derived from first-order Taylor series while training a model.

B.3 Computational Complexity and Runtime Results of FSW

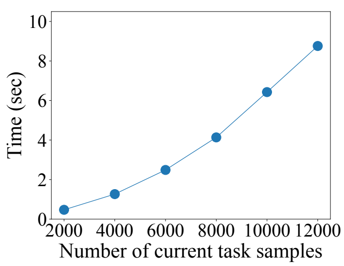


Figure 6: Runtime results of solving a single LP problem in FSW using CPLEX for the MNIST.

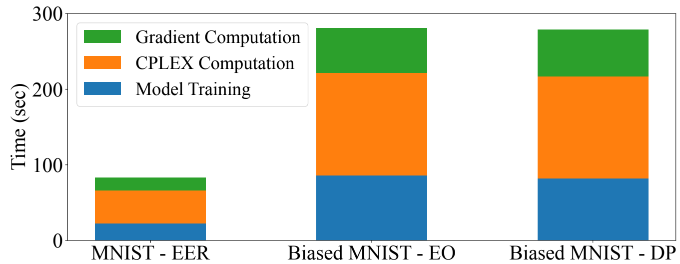


Figure 7: Overall runtime results of our framework for all tasks in three experimental settings: MNIST–EER, Biased MNIST–EO, and Biased MNIST–DP.

Continuing from Sec. 3.3, we provide computational complexity and overall runtime results of FSW using the MNIST and Biased MNIST datasets as shown in Fig. 6 and Fig. 7. Our empirical results show that for about ten thousand current-task samples, the time to solve an LP problem is a few seconds for the MNIST dataset as shown in Fig. 6. By applying the log-log regression model to the results in Fig. 6, the computational complexity of solving LP at each epoch is $\mathcal{O}(|T_l|^{1.642})$ where $|T_l|$ denotes the number of current task samples. We note that this complexity can be quadratic in the worst case. If the task size becomes too large, we believe that clustering similar samples and assigning weights to the clusters, rather than samples, could be a solution to reduce the computational overhead. In Fig. 7, we compute the overall runtime of FSW divided into three components: Gradient Computation, CPLEX Computation, and Model Training.

B.4 Dataset Descriptions

Continuing from Sec. 4, we provide more details of the two datasets using the class as the sensitive attribute and the three datasets with separate sensitive attributes. We also consider using standard benchmark datasets in the fairness field, but they are unsuitable for class-incremental learning experiments because either there are only two classes (e.g., COMPAS (Angwin et al., 2016), AdultCensus (Kohavi, 1996), and Jigsaw (cjadams, 2019)), or it is difficult to compute fairness (e.g., for CelebA (Liu et al., 2015), each person is a class).

- **MNIST** (LeCun et al., 1998): The MNIST dataset is a standard benchmark for evaluating the performance of machine learning models, especially in image classification tasks. The dataset is a collection of grayscale images of handwritten digits ranging from 0 to 9, each measuring 28 pixels in width and 28 pixels in height. The dataset consists of 60,000 training images and 10,000 test images. We configure a class-incremental learning setup, where a total of 10 classes are evenly distributed across 5 tasks, with 2 classes per task. We assume the class itself is the sensitive attribute.
- **Fashion-MNIST (FMNIST)** (Xiao et al., 2017): The Fashion-MNIST dataset is a specialized variant of the original MNIST dataset, designed for the classification of various clothing items into 10 distinct classes. The classes include ‘T-shirt/top’, ‘Trouser’, ‘Pullover’, ‘Dress’, ‘Coat’, ‘Sandal’, ‘Shirt’, ‘Sneaker’, ‘Bag’, and ‘Ankle boot’. The dataset consists of grayscale images with dimensions of 28 pixels by 28 pixels including 60,000 training images and 10,000 test images. We configure a class-incremental learning setup, where a total of 10 classes are evenly distributed across 5 tasks, with 2 classes per task. We assume the class itself is the sensitive attribute.
- **Biased MNIST** (Bahng et al., 2020): The Biased MNIST dataset is a modified version of the MNIST dataset that introduces bias by incorporating background colors highly correlated with the digits. We select 10 distinct background colors and assign one to each digit from 0 to 9. For the training images, each digit is assigned the selected background color with a probability of 0.95, or one of the other colors at random with a probability of 0.05. For the test images, the background color of each digit is assigned from the selected color or other random colors with equal probability of 0.5. The dataset consists of 60,000 training images and 10,000 test images. We configure a class-incremental learning setup, where a total of 10 classes are evenly distributed across 5 tasks,

with 2 classes per task. We set the background color as the sensitive attribute and consider two sensitive groups: the origin color and other random colors for each digit.

- **Drug Consumption (DRUG)** (Fehrman et al., 2017): The Drug Consumption dataset contains information about the usage of various drugs by individuals and correlates it with different demographic and personality traits. The dataset includes records for 1,885 respondents, each with 12 attributes including NEO-FFI-R, BIS-11, ImpSS, level of education, age, gender, country of residence, and ethnicity. We split the dataset into the ratio of 70/30 for training and testing. All input attributes are originally categorical, but we quantify them as real values for training. Participants were questioned about their use of 18 drugs, and our task is to predict cannabis usage. The label variable contains six classes: 'Never Used', 'Used over a Decade Ago', 'Used in Last Decade', 'Used in Last Year', 'Used in Last Month', and 'Used in Last Day'. We configure a class-incremental learning setup, where a total of 6 classes are distributed across 3 tasks, with 2 classes per task. We set gender as the sensitive attribute and consider two sensitive groups: male and female.
- **BiasBios** (De-Arteaga et al., 2019): The BiasBios dataset is a benchmark designed to explore and evaluate bias in natural language processing models, particularly in the context of profession classification from bios. The dataset consists of short textual biographies collected from online sources, labeled with one of the 28 profession classes, such as 'professor', 'nurse', or 'software engineer'. The dataset includes gender annotations, which makes it suitable for studying biases related to gender. The dataset contains approximately 350k biographies where 253k are for training and 97k for testing. We configure a class-incremental learning setup using the 25 most-frequent professions, where a total of 25 classes are distributed across 5 tasks, with 5 classes per task. As the number of samples for each class varies significantly, we arrange the classes in descending order based on their size (Chowdhury & Chaturvedi, 2023). We set gender as the sensitive attribute and consider two sensitive groups: male and female.

B.5 More Details on Experimental Settings

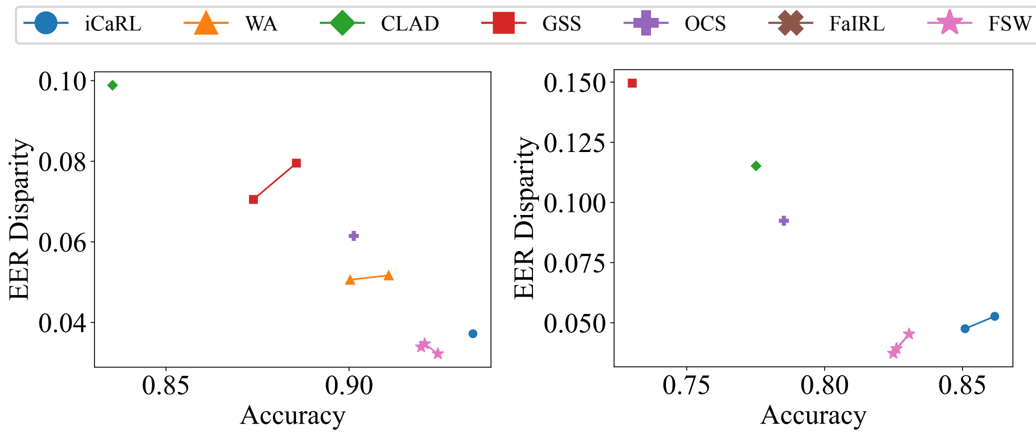
Continuing from Sec. 4, we provide more details on experimental settings. We use a batch size of 64 for all the experiments. We set the initial learning rate and the total epochs for each dataset. For the MNIST, FMNIST, and DRUG datasets, we train both our model and baselines with initial learning rates of [0.001, 0.01, 0.1], for 5, 5, and 25 epochs, respectively. For Biased MNIST, we use learning rates of [0.001, 0.01, 0.1] for 15 epochs. For the BiasBios dataset, we use learning rates of [0.00002, 0.0001, 0.001] for 10 epochs and set the maximum token length to 128. For hyperparameters, we perform cross-validation with a grid search for $\alpha \in \{0.0005, 0.001, 0.002, 0.01\}$, $\lambda \in \{0.1, 0.5, 1\}$, and $\tau \in \{1, 2, 5, 10\}$. To solve the fairness-aware optimization problems and find optimal sample weights, we use CPLEX, a high-performance optimization solver developed by IBM that specializes in solving linear programming (LP) problems.

B.6 Tradeoff Results between Accuracy and Fairness

Continuing from Sec. 4.1, we evaluate the tradeoff between accuracy and fairness of FSW with other baselines as shown in Fig. 8–Fig. 11 on the following pages. FSW in the figures represents the result for different values of λ, a hyperparameter that balances fairness and accuracy. Since other baselines do not have a balancing parameter, we select Pareto-optimal points from all search spaces, where a Pareto-optimal point is defined as a point for which there does not exist another point with both higher accuracy and lower fairness disparity. The figures show FSW positioned in the lower right corner of the graph, indicating better accuracy-fairness tradeoff results compared to other baseline methods.

B.7 More Results on Accuracy and Fairness

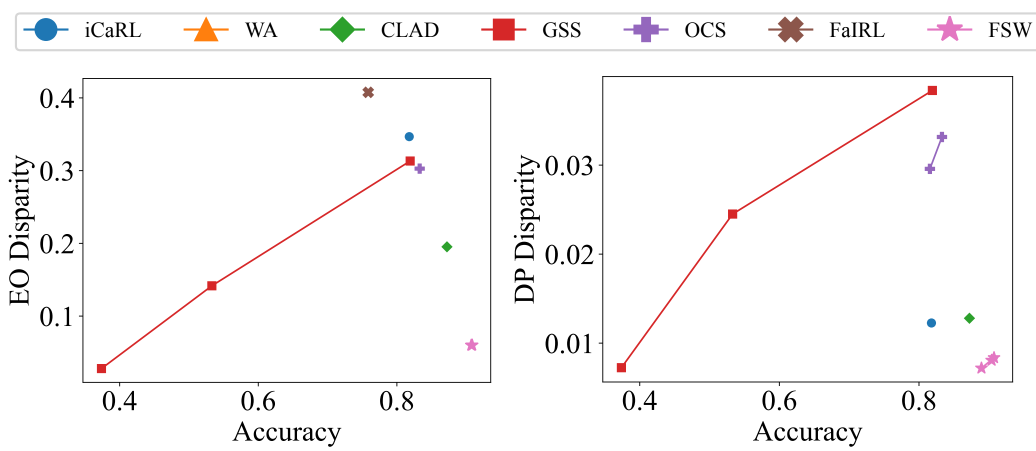
Continuing from Sec. 4.1, we compare FSW with other baselines with respect to EER, EO, and DP disparity as shown in Tables 5, 6, and 7, respectively, on page 27. In addition, we present the sequential performance results for each task as shown in Fig. 12–Fig. 19, starting on page 28. Due to the excessive time required to run *OCS* on BiasBios, we are not able to measure the results.



(a) MNIST (EER).

(b) FMNIST (EER).

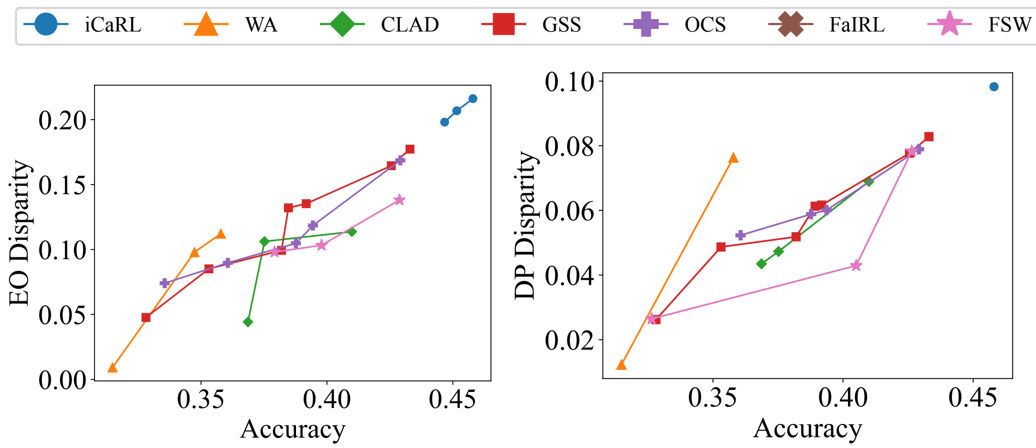
Figure 8: Tradeoff results between accuracy and fairness (EER) on the MNIST and FMNIST datasets.



(a) Biased MNIST (EO).

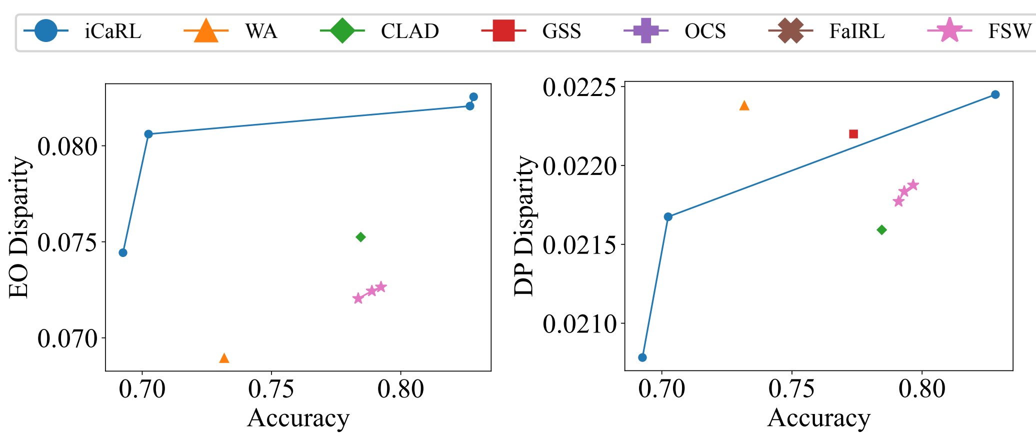
(b) Biased MNIST (DP).

Figure 9: Tradeoff results between accuracy and fairness (EO and DP) on the Biased MNIST dataset.



(a) DRUG (EO). (b) DRUG (DP).

Figure 10: Tradeoff results between accuracy and fairness (EO and DP) on the DRUG dataset.



(a) BiasBios (EO). (b) BiasBios (DP).

Figure 11: Tradeoff results between accuracy and fairness (EO and DP) on the BiasBios dataset.

Table 5: Accuracy and fairness results on the MNIST and FMNIST datasets with respect to EER disparity, where the class is the sensitive attribute. We compare FSW with four types of baselines: naïve (*Joint Training* and *Fine Tuning*), state-of-the-art (*iCaRL*, *WA*, and *CLAD*), sample selection (*GSS* and *OCS*), and fairness-aware (*FaIRL*) methods. We mark the best and second best results with **bold** and underline, respectively.

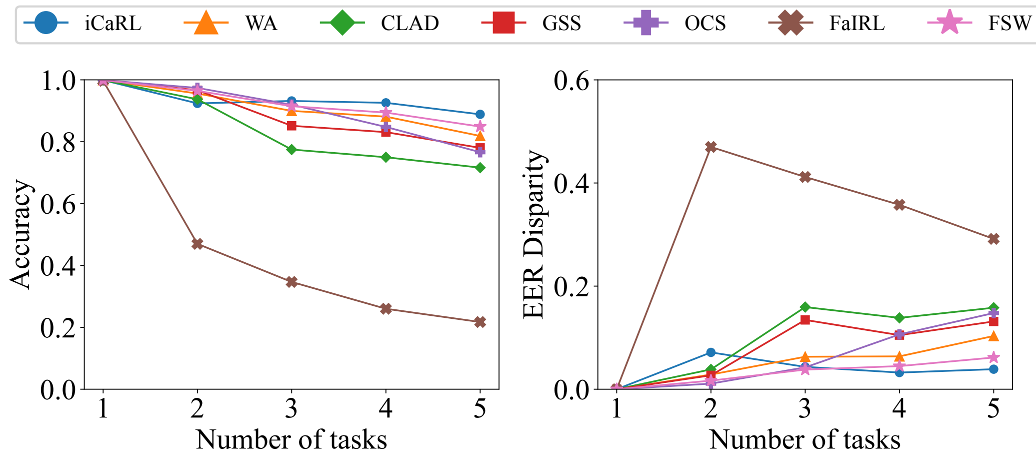
Methods	MNIST		FMNIST	
	Acc.	EER Disp.	Acc.	EER Disp.
Joint Training	.970±.004	.014±.006	.895±.010	.035±.004
Fine Tuning	.453±.000	.323±.000	.450±.000	.324±.000
iCaRL	**.934±.004**	.037±.003	**.862±.002**	.053±.003
WA	.911±.007	.052±.006	.809±.005	.088±.003
CLAD	.835±.015	.099±.015	.775±.018	.115±.019
GSS	.886±.007	.080±.009	.730±.013	.150±.011
OCS	.901±.003	.061±.004	.785±.012	.092±.007
FaIRL	.458±.008	.306±.004	.455±.005	.316±.001
FSW	.924±.003	**.032±.004**	.825±.006	**.037±.007**

Table 6: Accuracy and fairness results on the Biased MNIST, DRUG, and BiasBios datasets with respect to EO disparity, where background color is the sensitive attribute for Biased MNIST, and gender for DRUG and BiasBios, respectively. Due to the excessive time required to run *OCS* on BiasBios, we are not able to measure the results and mark them as '–'. The other settings are same as in Table 5.

Methods	Biased MNIST		DRUG		BiasBios	
	Acc.	EO Disp.	Acc.	EO Disp.	Acc.	EO Disp.
Joint Training	.945±.002	.053±.002	.441±.015	.179±.052	.823±.003	.075±.001
Fine Tuning	.448±.001	.010±.003	.357±.009	.125±.034	.425±.006	.029±.002
iCaRL	.818±.011	.347±.025	**.458±.014**	.216±.056	**.828±.002**	.083±.003
WA	.447±.001	**.018±.002**	.358±.009	.112±.038	.732±.008	.069±.002
CLAD	.872±.011	.195±.020	.410±.026	.114±.043	.785±.004	.075±.001
GSS	.819±.009	.313±.021	.433±.011	.177±.045	.774±.007	.086±.005
OCS	.833±.012	.303±.024	.429±.007	.169±.026	–	–
FaIRL	.759±.008	.408±.018	.318±.006	**.015±.009**	.332±.009	**.039±.003**
FSW	**.909±.003**	.060±.004	.429±.020	.138±.037	.792±.005	.073±.003

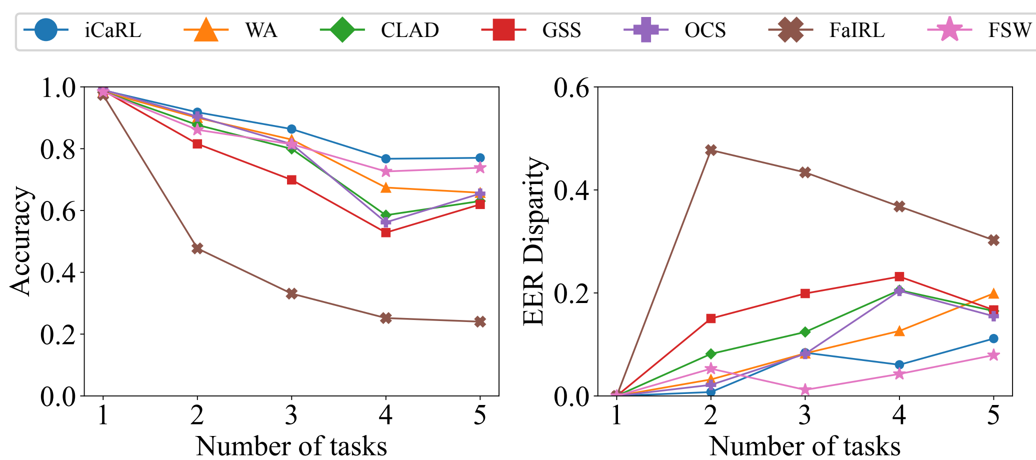
Table 7: Accuracy and fairness results on the Biased MNIST, DRUG, and BiasBios datasets with respect to DP disparity. The other settings are the same as in Table 6.

Methods	Biased MNIST		DRUG		BiasBios	
	Acc.	DP Disp.	Acc.	DP Disp.	Acc.	DP Disp.
Joint Training	.945±.002	.005±.001	.441±.015	.091±.020	.823±.003	.021±.000
Fine Tuning	.448±.001	.016±.008	.357±.009	.102±.013	.425±.006	.028±.001
iCaRL	.818±.011	.012±.001	**.458±.014**	.098±.020	**.828±.002**	**.022±.000**
WA	.447±.001	.016±.004	.358±.009	.076±.019	.732±.008	**.022±.001**
CLAD	.872±.011	.013±.001	.410±.026	.069±.019	.785±.004	**.022±.000**
GSS	.819±.009	.038±.005	.433±.011	.083±.018	.774±.007	**.022±.001**
OCS	.816±.012	.030±.003	.429±.007	.079±.020	–	–
FaIRL	.759±.008	.033±.001	.318±.006	**.015±.007**	.332±.009	.026±.002
FSW	**.889±.006**	**.007±.002**	.405±.013	.043±.004	.797±.003	**.022±.000**



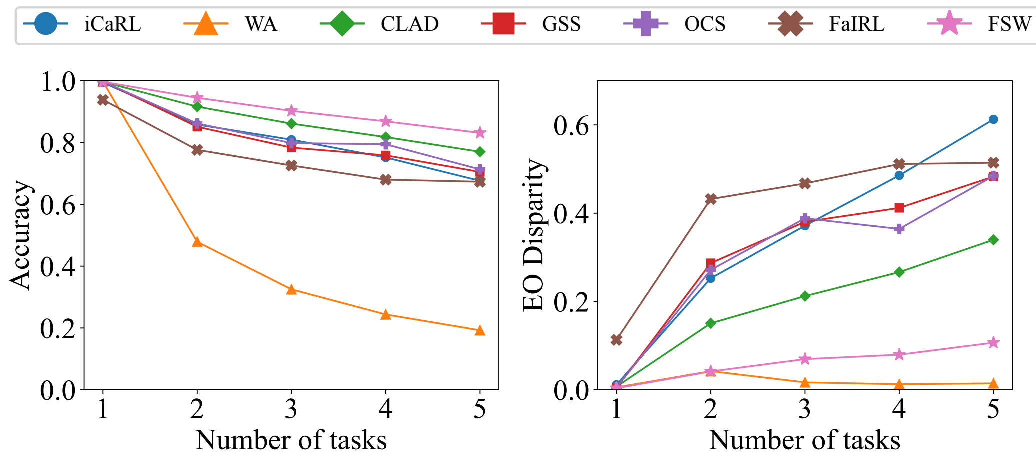
(a) Accuracy. (b) EER Disparity.

Figure 12: Sequential accuracy and fairness (EER) results on the MNIST dataset.



(a) Accuracy. (b) EER Disparity.

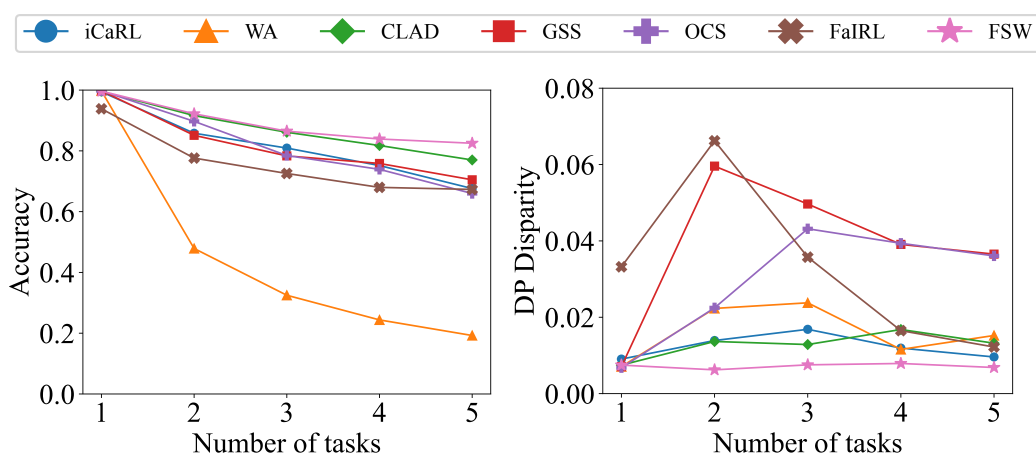
Figure 13: Sequential accuracy and fairness (EER) results on the FMNIST dataset.



(a) Accuracy.

(b) EO Disparity.

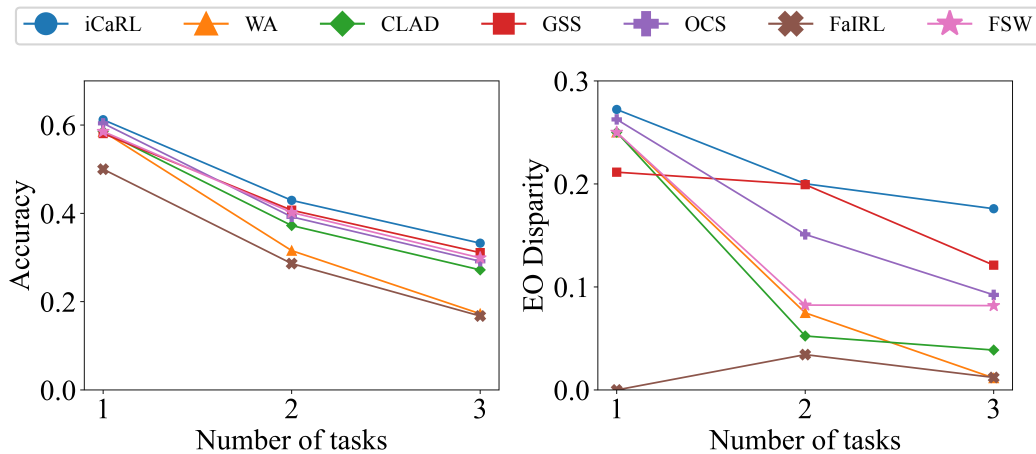
Figure 14: Sequential accuracy and fairness (EO) results on the Biased MNIST dataset.



(a) Accuracy.

(b) DP Disparity.

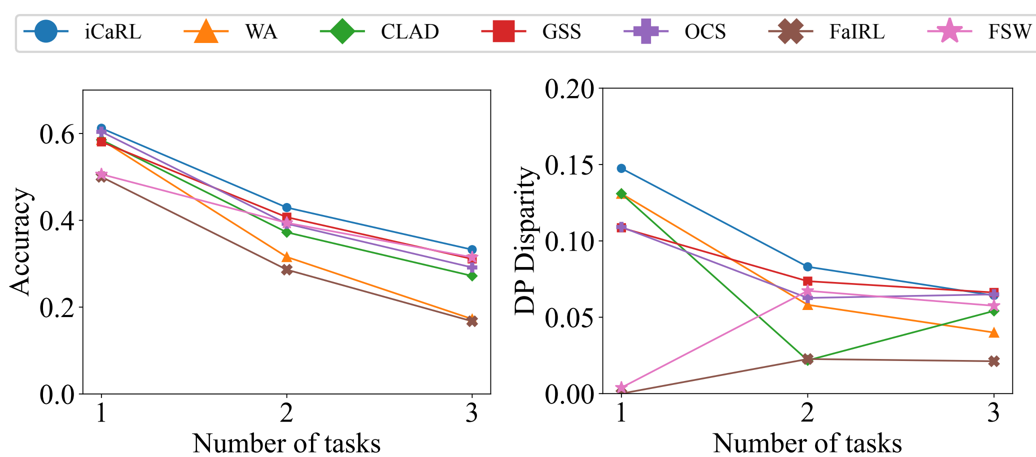
Figure 15: Sequential accuracy and fairness (DP) results on the Biased MNIST dataset.



(a) Accuracy.

(b) EO Disparity.

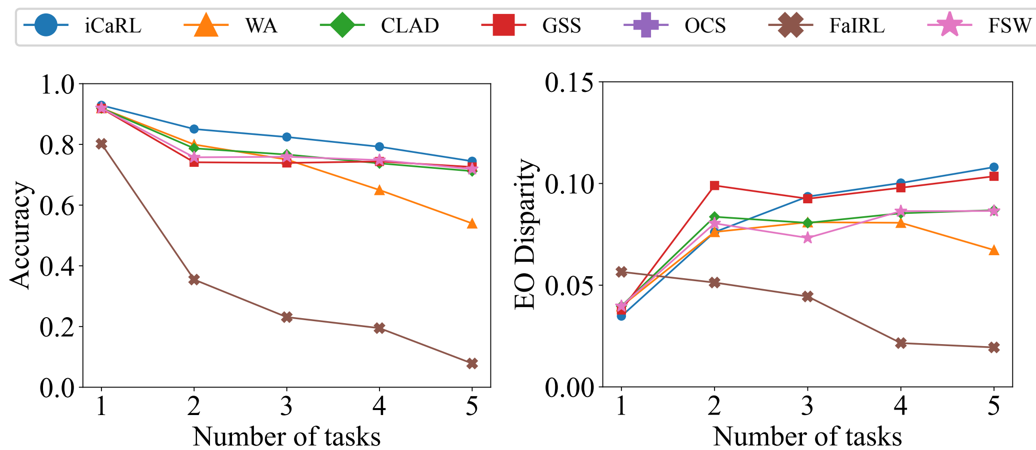
Figure 16: Sequential accuracy and fairness (EO) results on the DRUG dataset.



(a) Accuracy.

(b) DP Disparity.

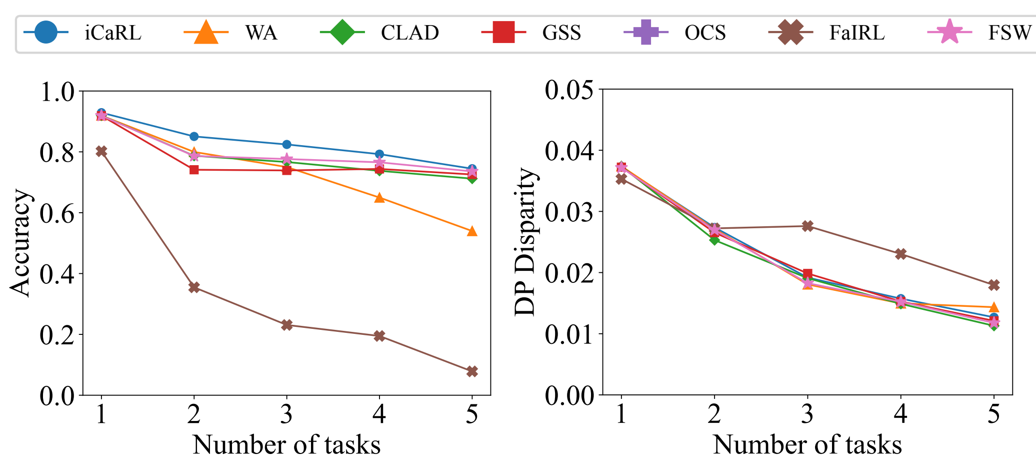
Figure 17: Sequential accuracy and fairness (DP) results on the DRUG dataset.



(a) Accuracy.

(b) EO Disparity.

Figure 18: Sequential accuracy and fairness (EO) results on the BiasBios dataset.



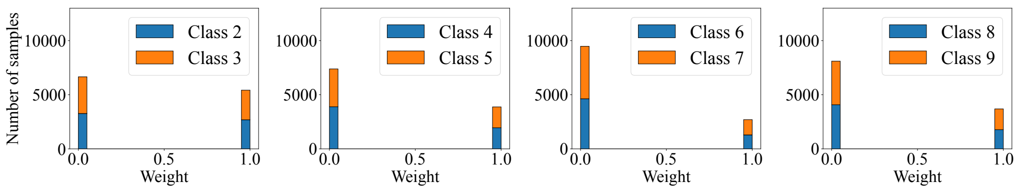
(a) Accuracy.

(b) DP Disparity.

Figure 19: Sequential accuracy and fairness (DP) results on the BiasBios dataset.

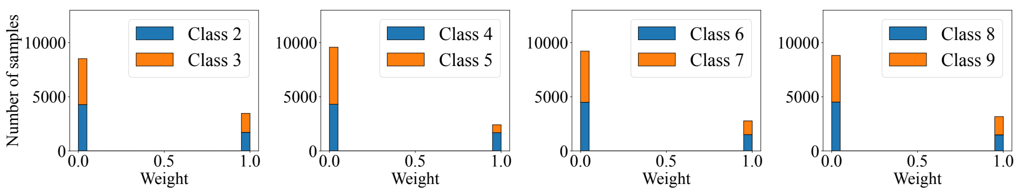
B.8 More Results on Sample Weighting Analysis

Continuing from Sec. 4.2, we show more results from the sample weighting analysis for all sequential tasks of each dataset, as shown in the figures below (Fig. 20–Fig. 27). We compute the number of samples for weights in sensitive groups including classes. For each task, we show the average weight distribution over all epochs, as sample weights may change during each epoch of training. Since FSW is not applied to the first task, where the model is trained with only the current task data, we present the results starting from the second task.



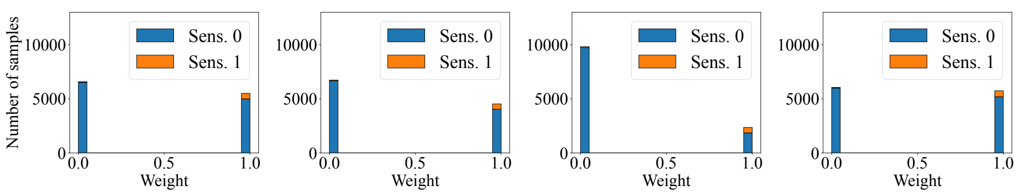
(a) Task 2. (b) Task 3. (c) Task 4. (d) Task 5.

Figure 20: Distribution of sample weights for EER in sequential tasks of the MNIST dataset.



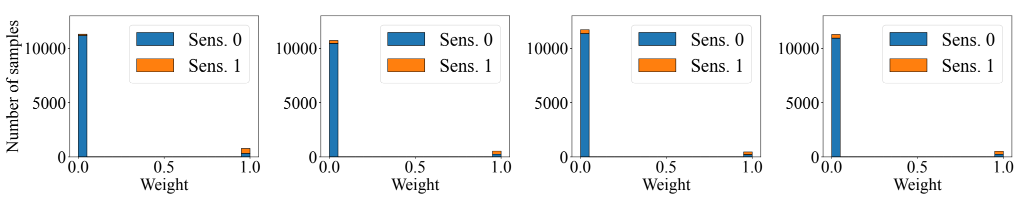
(a) Task 2. (b) Task 3. (c) Task 4. (d) Task 5.

Figure 21: Distribution of sample weights for EER in sequential tasks of the FMNIST dataset.



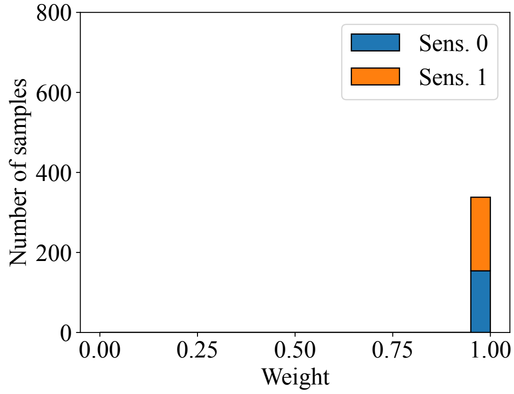
(a) Task 2. (b) Task 3. (c) Task 4. (d) Task 5.

Figure 22: Distribution of sample weights for EO in sequential tasks of the Biased MNIST dataset.

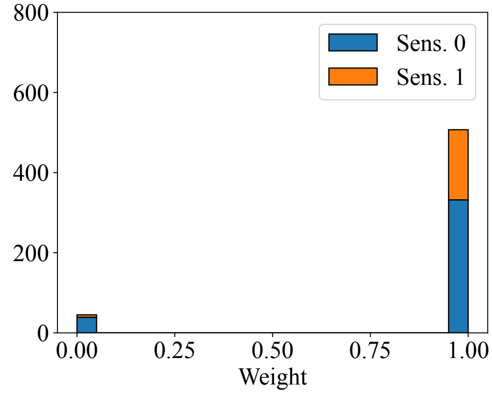


(a) Task 2. (b) Task 3. (c) Task 4. (d) Task 5.

Figure 23: Distribution of sample weights for DP in sequential tasks of the Biased MNIST dataset.

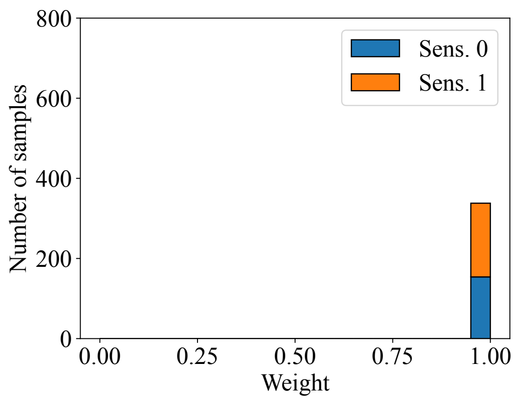


(a) Task 2.

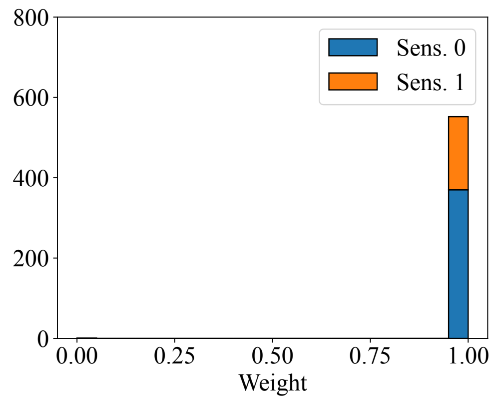


(b) Task 3.

Figure 24: Distribution of sample weights for EO in sequential tasks of the DRUG dataset.

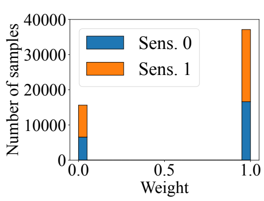


(a) Task 2.

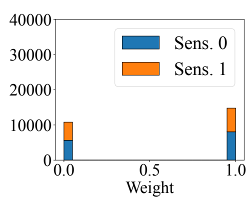


(b) Task 3.

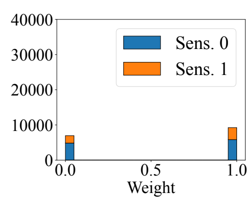
Figure 25: Distribution of sample weights for DP in sequential tasks of the DRUG dataset.



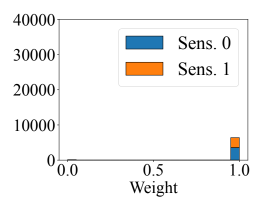
(a) Task 2.



(b) Task 3.

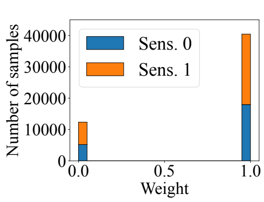


(c) Task 4.

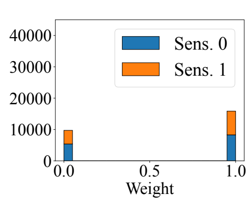


(d) Task 5.

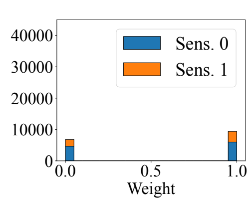
Figure 26: Distribution of sample weights for EO in sequential tasks of the BiasBios dataset.



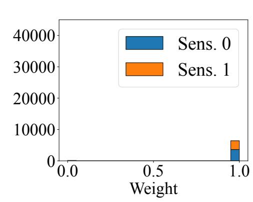
(a) Task 2.



(b) Task 3.



(c) Task 4.



(d) Task 5.

Figure 27: Distribution of sample weights for DP in sequential tasks of the BiasBios dataset.

B.9 More Results on Ablation Study

Continuing from Sec. 4.3, we present additional results of the ablation study to demonstrate the contribution of our proposed fairness-aware sample weighting (FSW) to the overall accuracy and fairness performance. The results are shown in Tables 8, 9, and 10.

Table 8: Accuracy and fairness results on the MNIST and FMNIST datasets with respect to EER disparity when FSW is used or not.

Methods	MNIST		FMNIST	
	Acc.	EER Disp.	Acc.	EER Disp.
W/o FSW	.921±.004	.040±.005	**.836±.006**	.048±.005
FSW	**.924±.003**	**.032±.004**	.825±.006	**.037±.007**

Table 9: Accuracy and fairness results on the Biased MNIST, DRUG, and BiasBios datasets with respect to EO disparity when FSW is used or not.

Methods	Biased MNIST		DRUG		BiasBios	
	Acc.	EO Disp.	Acc.	EO Disp.	Acc.	EO Disp.
W/o FSW	**.911±.003**	.063±.002	.423±.013	.162±.034	.790±.003	.076±.001
FSW	.909±.003	**.060±.004**	**.429±.020**	**.138±.037**	**.792±.005**	**.073±.003**

Table 10: Accuracy and fairness results on the Biased MNIST, DRUG, and BiasBios datasets with respect to DP disparity when FSW is used or not.

Methods	Biased MNIST		DRUG		BiasBios	
	Acc.	DP Disp.	Acc.	DP Disp.	Acc.	DP Disp.
W/o FSW	**.911±.003**	.009±.001	**.423±.013**	.080±.015	.790±.003	**.022±.000**
FSW	.889±.006	**.007±.002**	.405±.013	**.043±.004**	**.797±.003**	**.022±.000**

B.10 More Results on Integrating FSW with a Fair Post-processing Method

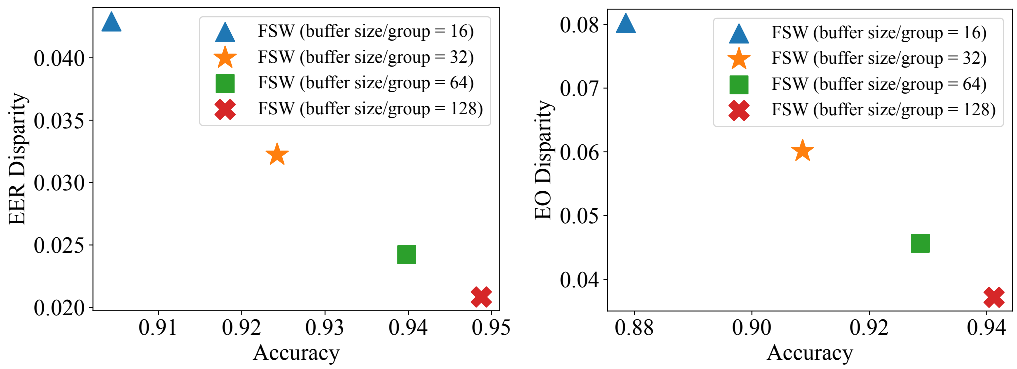
Continuing from Sec. 4.4, we provide additional results on integrating continual learning methods with fair post-processing, including *OCS* and *OCS* – ϵ-fair performances as shown in Table 11.

Table 11: Accuracy and fairness results when combining fair post-processing (ϵ-fair) with continual learning methods (*iCaRL*, *CLAD*, *OCS*, and FSW) with respect to DP disparity. Due to the excessive time required to run *OCS* on BiasBios, we are not able to measure the results and mark them as '–'.

Methods	Biased MNIST		DRUG		BiasBios	
	Acc.	DP Disp.	Acc.	DP Disp.	Acc.	DP Disp.
iCaRL	.818±.011	.012±.001	.458±.014	.098±.020	**.828±.002**	.022±.000
CLAD	.872±.011	.013±.001	.410±.026	.069±.019	.785±.004	.022±.001
OCS	.816±.012	.030±.003	.429±.007	.079±.020	–	–
FSW	**.889±.006**	.007±.002	.405±.013	.043±.004	.797±.003	.022±.000
iCaRL – ϵ-fair	.805±.014	.007±.002	**.460±.015**	.035±.013	**.828±.001**	**.016±.000**
CLAD – ϵ-fair	.868±.015	.006±.002	.411±.023	.030±.010	.759±.049	.017±.000
OCS – ϵ-fair	.825±.016	**.005±.001**	.431±.021	.033±.007	–	–
FSW – ϵ-fair	.883±.007	**.005±.001**	.403±.010	**.020±.004**	.796±.003	**.016±.000**

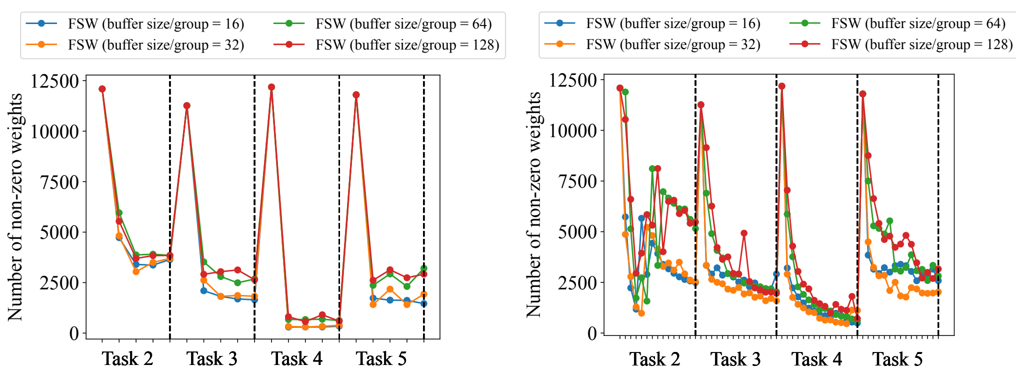
B.11 More Results of FSW when Varying the Buffer Size

We have additional experimental results of FSW on the MNIST and Biased MNIST datasets when varying the buffer size to 16, 32, 64, and 128 per sensitive group as shown in Fig. 28. As the buffer size increases, both accuracy and fairness performances improve. In addition, we compute the number of current task data assigned with non-zero weights (i.e., not close to zero) as shown in Fig. 29, and there is no clear relationship between buffer size and weights.



(a) MNIST. (b) Biased MNIST.

Figure 28: Accuracy and fairness results of FSW when varying the buffer size on the MNIST and Biased MNIST datasets.



(a) MNIST. (b) Biased MNIST.

Figure 29: Number of current task data assigned with non-zero weights (i.e., not close to zero) when varying the buffer size on the MNIST and Biased MNIST datasets.

C Appendix – More Related Work

Continuing from Sec. 2, we discuss more related work.

Class-incremental learning is a challenging type of continual learning where a model continuously learns new tasks, each composed of new disjoint classes, and the goal is to minimize catastrophic forgetting (Mai et al., 2022; Masana et al., 2023). Data replay techniques (Lopez-Paz & Ranzato, 2017; Rebuffi et al., 2017; Chaudhry et al., 2019b) store a small portion of previous data in a buffer to utilize for training and is widely used with other techniques (Zhou et al., 2023a) including knowledge distillation (Rebuffi et al., 2017; Buzzega et al., 2020), model rectification (Wu et al., 2019; Zhao et al., 2020), and dynamic networks (Yan et al., 2021; Wang et al., 2022; Zhou et al., 2023b). Simple buffer sample selection methods such as random or herding-based approaches (Rebuffi et al., 2017) are also commonly used as well. There are also more advanced gradient-based sample selection techniques like GSS (Aljundi et al., 2019) and OCS (Yoon et al., 2022) that manage buffer data to have samples with diverse and representative gradient vectors. All these works do not consider fairness and simply assume that the entire incoming data is used for model training, which may result in unfair forgetting as we show in our experiments.

Model fairness research mitigates bias by ensuring that a model's performance is equitable across different sensitive groups, thereby preventing discrimination based on race, gender, age, or other sensitive attributes (Mehrabi et al., 2022). Existing model fairness techniques can be categorized as pre-processing (Kamiran & Calders, 2011; Feldman et al., 2015; Calmon et al., 2017; Jiang & Nachum, 2020), in-processing (Agarwal et al., 2018; Zhang et al., 2018; Cotter et al., 2019; Roh et al., 2020), and post-processing (Hardt et al., 2016; Pleiss et al., 2017; Chzhen et al., 2019). In addition, there are other techniques that assign adaptive weights for samples to improve fairness (Chai & Wang, 2022; Jung et al., 2023). However, most of these techniques assume that the training data is given all at once, which may not be realistic. There are techniques for fairness-aware active learning (Anahideh et al., 2022; Pang et al., 2024; Tae et al., 2024), in which the training data evolves with the acquisition of samples. However, these techniques store all labeled data and use them for training, which is impractical in continual learning settings.

D Appendix – Future Work

D.1 Generalization to Multiple sensitive attributes

FSW can be extended to tasks involving multiple sensitive attributes by defining a sensitive group as a combination of sensitive attributes. For instance, recall the loss for EO in a single sensitive attribute is $\frac{1}{|\mathbb{Y}||\mathbb{Z}|}\sum_{y\in\mathbb{Y},z\in\mathbb{Z}}|\tilde{\ell}(f_\theta, G_{y,z}) - \tilde{\ell}(f_\theta, G_y)|$. This definition can be extended to the case of multiple sensitive attributes as $\frac{1}{|\mathbb{Y}||\mathbb{Z}_1||\mathbb{Z}_2|}\sum_{y\in\mathbb{Y},z_1\in\mathbb{Z}_1,z_2\in\mathbb{Z}_2}|\tilde{\ell}(f_\theta, G_{y,z_1,z_2}) - \tilde{\ell}(f_\theta, G_y)|$. The new definition for multiple sensitive attributes allows the overall optimization problem to optimize both sensitive attributes simultaneously. The design above can also help prevent 'fairness gerrymandering' (Kearns et al., 2018), a situation where fairness is superficially achieved across multiple groups, but specific individuals or subgroups within those groups are systematically disadvantaged. This is achieved by minimizing all combinations of subgroups, thereby disrupting the potential for unfair prediction based on certain attribute combinations. However, having multiple loss functions may increase the complexity of optimization, and a more advanced loss function may need to be designed for multiple sensitive attributes. We leave the extension of this work to multiple sensitive attributes in future work.